C sco Reader Comment Card

General Information

1 Years of networking experience ______ Years of experience with Cisco products ______

2 I have these network types: ☐ LAN ☐ Backbone ☐ WAN
☐ Other: ____________

3 I have these Cisco products: ☐ Switches ☐ Routers
☐ Other: Specify model(s) ____________

4 I perform these types of tasks: ☐ H/W Install and/or Maintenance ☐ S/W Config
☐ Network Management ☐ Other: ____________

5 I use these types of documentation: ☐ H/W Install ☐ H/W Config ☐ S/W Config
☐ Command Reference ☐ Quick Reference ☐ Release Notes ☐ Online Help
☐ Other: ____________

6 I access this information through: ____% Cisco Connection Online (CCO) ____% CD-ROM
____% Printed docs ____% Other: ____________

7 Which method do you prefer? ____________

8 I use the following three product features the most:

Document Information

Document Title: ATM Switch Router Software Configuration Guide

Part Number: 78-12281-01 S/W Release: 12.1(5a)EY

On a scale of 1–5 (5 being the best) please let us know how we rate in the following areas:

_____ The document was written at my technical level of understanding.

_____ The information was accurate.

_____ The document was complete.

_____ The information I wanted was easy to find.

_____ The information was well organized.

_____ The information I found was useful to my job.

Please comment on our lowest score(s):

Mailing Information

Company Name ____________ Date ____________

Contact Name ____________ Job Title ____________

Mailing Address ____________

City ____________ State/Province ____________ ZIP/Postal Code ____________

Country ____________ Phone (   ) ____________ Extension ____________

Fax (   ) ____________ E-mail ____________

Can we contact you further concerning our documentation? ☐ Yes ☐ No

You can also send us your comments by e-mail to **bug-doc@cisco.com**, or fax your comments to us at **(408) 527-8089**.

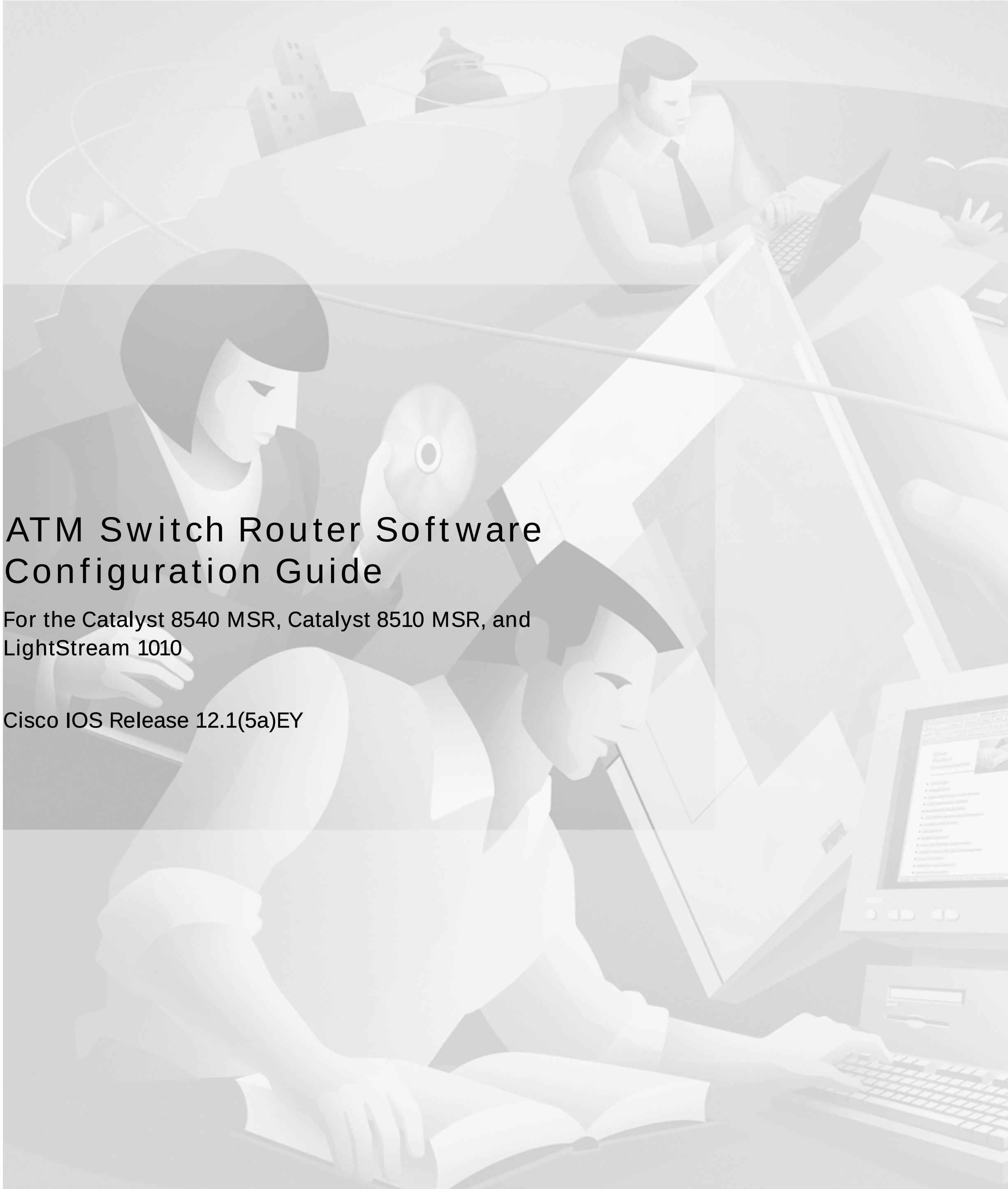
NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY MAIL

FIRST-CLASS MAIL PERMIT NO. 4631 SAN JOSE CA

POSTAGE WILL BE PAID BY ADDRESSEE

ATTN DOCUMENT RESOURCE CONNECTION
CISCO SYSTEMS INC
170 WEST TASMAN DRIVE
SAN JOSE CA 95134-9883



ATM Switch Router Software Configuration Guide

For the Catalyst 8540 MSR, Catalyst 8510 MSR, and LightStream 1010

Cisco IOS Release 12.1(5a)EY

Corporate Headquarters
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
http://www.cisco.com
Tel: 408 526-4000
800 553-NETS (6387)
Fax: 408 526-4100

Customer Order Number: DOC-7812281=
Text Part Number: 78-12281.01


CONTENTS

Preface

This preface describes the audience, organization, and conventions for the *ATM Switch Router Software Configuration Guide*, and provides information on how to obtain related documentation.

Audience

This publication is intended for experienced network administrators who are responsible for configuring and maintaining the Layer 3 enabled ATM switch router.

New and Changed Information

Feature	Description	Chapter or Section
CES Soft PVC Per Interface State	This feature allows the state of a CES interface to be configured to reflect the state of the physical interface.	Chapter 18, "Configuring Circuit Emulation Services"
Catalyst 8540 MSR Enhanced ATM Router Module	The Catalyst 8540 MSR has a new ATM router module that supports RFC 1577 SVCs and ACLs.	Chapter 21, "Configuring ATM Router Module Interfaces"

Organization

The major sections of this guide are as follows:

Chapter	Title	Description
Chapter 1	Product Overview	Provides an overview of the ATM switch router features and functions.
Chapter 2	Understanding the User Interface	Describes how to access the commands available in each command mode and explains the primary uses for each command mode.
Chapter 3	Initially Configuring the ATM Switch Router	Describes the initial configuration of the ATM switch router.
Chapter 4	Configuring System Management Functions	Describes the tasks to manage the general system features, such as access control and basic management of the ATM switch router.
Chapter 5	Configuring ATM Network Interfaces	Describes how to configure typical ATM network interfaces after autoconfiguration has established the default network connections.
Chapter 6	Configuring Virtual Connections	Describes how to configure virtual connections after autoconfiguration has determined the default virtual connections.
Chapter 7	Configuring Operation, Administration, and Maintenance	Describes the OAM fault management and performance management functions of the ATM switch router.
Chapter 8	Configuring Resource Management	Describes how to configure the management of switch, interface, and connection resources.
Chapter 9	Configuring ILMI	Describes the Integrated Local Management Interface (ILMI) protocol implementation and configuration.
Chapter 10	Configuring ATM Routing and PNNI	Describes how to configure the Interim Interswitch Signaling Protocol (IISP) and the Private Network-Network Interface (PNNI) protocol.
Chapter 11	Using Access Control	Describes how to configure and maintain access control lists.
Chapter 12	Configuring IP over ATM	Describes how to configure the Ethernet port for IP over ATM connections.
Chapter 13	Configuring LAN Emulation	Describes how to configure LAN emulation on the ATM switch router.
Chapter 14	Configuring ATM Accounting and ATM RMON	Describes the ATM accounting and ATM Remote Monitoring features and their configuration.
Chapter 15	Configuring Tag Switching	Describes how to configure tag switching on the ATM switch router.
Chapter 16	Configuring Signalling Features	Describes how to configure common and specialized signalling features.

Chapter	Title	Description
Chapter 17	Configuring Interfaces	Describes the steps required to configure the individual port adapter and interface module.
Chapter 18	Configuring Circuit Emulation Services	Describes the steps to configure the Circuit Emulation Services port adapter modules.
Chapter 19	Configuring Frame Relay to ATM Interworking Port Adapter Interfaces	Describes the steps to configure the Frame Relay to ATM interworking port adapter modules.
Chapter 20	Configuring IMA Port Adapter Interfaces	Describes the steps to configure inverse multiplexing over ATM port adapter interfaces.
Chapter 21	Configuring ATM Router Module Interfaces	Describes the steps to integrate Layer 3 routing and ATM switching with the ATM router module.
Chapter 22	Managing Configuration Files, System Images, and Functional Images	Includes procedures for updating and maintaining the ATM switch router software and configurations.
Appendix A	PNNI Migration Examples	Provides examples for migrating from a flat PNNI topology to a hierarchical topology.
Appendix B	Acronyms	Lists the acronyms used in this guide.

Related Documentation

This document provides detailed ATM software configuration examples; however, it does not provide complete ATM software command syntax descriptions or extensive background information on ATM features. For detailed ATM software command syntax information, refer to the *ATM Switch Router Command Reference* publication. For detailed background information on ATM features and functionality, refer to the *Guide to ATM Technology*.

You will also find useful information on the command-line interface (CLI) and basic ATM switch router management in the *Configuration Fundamentals Configuration Guide* and *Configuration Fundamentals Command Reference* publications.

The ATM switch router documentation set is primarily ATM-specific. You might be referred to the Cisco IOS documentation set for information about IP and router configuration and other non-ATM related features. For example, when configuring the IP address on the ATM switch processor, only basic configuration steps are provided. If you need additional overview or detailed IP configuration information, refer to the Cisco IOS documentation set.

The ATM switch router documents are separated into two groups:

- Basic documents are provided in the accessory kit with the hardware and are all the documentation you need for initial installation and configuration information.
- Advanced configuration documents are not provided in the accessory kit unless specifically ordered. They are available on Cisco.com and the Documentation CD-ROM and offer configuration information for more advanced applications of the ATM switch router.

The *ATM Switch Router Software Configuration Guide* is one of the advanced configuration documents and should only be used after you have completed the processes described in the basic document set. Refer to the following documents for detailed hardware installation, basic configuration information, and troubleshooting information:

- *Site Preparation and Safety Guide*
- *Quick Reference Catalyst 8540 CSR and MSR Hardware Information* (poster)
- *Quick Reference Catalyst 8510 and LightStream 1010 Hardware Information* (poster)
- *ATM and Layer 3 Port Adapter and Interface Module Installation Guide*
- *ATM Switch Router Quick Software Configuration Guide*
- *Layer 3 Switching Software Feature and Configuration Guide*
- *ATM Switch Router Command Reference*
- *Guide to ATM Technology*
- *ATM Switch Router Troubleshooting Guide*

Note The carrier modules are documented in the *ATM Port Adapter and Interface Module Installation Guide*.

Document Conventions

Unless otherwise noted, all information in this document is relevant to the Catalyst 8540 MSR, Catalyst 8510 MSR and LightStream 1010 ATM switch routers. Platform specific sections have the platform name appended to the title in parentheses. For example, the “Configuring Redundancy and Enhanced High System Availability (Catalyst 8540 MSR)” section on page 3-23 is only relevant to the Catalyst 8540 MSR ATM switch router.

This document uses the following conventions:

Convention	Description
boldface font	Commands and keywords are in **boldface**.
italic font	Arguments for which you supply values are in *italics*.
[ ]	Elements in square brackets are optional.
{x | y | z}	Alternative keywords are grouped in braces and separated by vertical bars.
[x | y | z]	Optional alternative keywords are grouped in brackets and separated by vertical bars.

Convention	Description
string	A nonquoted set of characters. Do not use quotation marks around the string or the string will include the quotation marks.
`screen` font	Terminal sessions and information the system displays are in `screen` font.
`boldface screen` font	Information you must enter is in **`boldface screen`** font.
italic screen font	Arguments for which you supply values are in *italic screen* font.
➔	`This pointer highlights an important line of text in an example.`
^	The symbol ^ represents the key labeled Control—for example, the key combination ^D in a screen display means hold down the Control key while you press the D key.
< >	Nonprinting characters, such as passwords are in angle brackets.

Notes use the following conventions:

Note Means *reader take note*. Notes contain helpful suggestions or references to material not covered in the publication.

Cautions use the following conventions:

Caution Means *reader be careful*. In this situation, you might do something that could result in equipment damage or loss of data.

Obtaining Documentation

The following sections provide sources for obtaining documentation from Cisco Systems.

World Wide Web

You can access the most current Cisco documentation on the World Wide Web at the following sites:

- http://www.cisco.com
- http://www-china.cisco.com
- http://www-europe.cisco.com

Documentation CD-ROM

Cisco documentation and additional literature are available in a CD-ROM package, which ships with your product. The Documentation CD-ROM is updated monthly and may be more current than printed documentation. The CD-ROM package is available as a single unit or as an annual subscription.

Ordering Documentation

Cisco documentation is available in the following ways:

- Registered Cisco Direct Customers can order Cisco Product documentation from the Networking Products MarketPlace:

  http://www.cisco.com/cgi-bin/order/order_root.pl
- Registered Cisco.com users can order the Documentation CD-ROM through the online Subscription Store:

  http://www.cisco.com/go/subscription
- Nonregistered CCO users can order documentation through a local account representative by calling Cisco corporate headquarters (California, USA) at 408 526-7208 or, in North America, by calling 800 553-NETS(6387).

Documentation Feedback

If you are reading Cisco product documentation on the World Wide Web, you can submit technical comments electronically. Click **Feedback** in the toolbar and select **Documentation**. After you complete the form, click **Submit** to send it to Cisco.

You can e-mail your comments to bug-doc@cisco.com.

To submit your comments by mail, for your convenience many documents contain a response card behind the front cover. Otherwise, you can mail your comments to the following address:

Cisco Systems, Inc.
Document Resource Connection
170 West Tasman Drive
San Jose, CA 95134-9883

We appreciate your comments.

Obtaining Technical Assistance

Cisco provides Cisco.com as a starting point for all technical assistance. Customers and partners can obtain documentation, troubleshooting tips, and sample configurations from online tools. For Cisco.com registered users, additional troubleshooting tools are available from the TAC website.

Cisco.com

Cisco.com is the foundation of a suite of interactive, networked services that provides immediate, open access to Cisco information and resources at anytime, from anywhere in the world. This highly integrated Internet application is a powerful, easy-to-use tool for doing business with Cisco.

Cisco.com provides a broad range of features and services to help customers and partners streamline business processes and improve productivity. Through Cisco.com, you can find information about Cisco and our networking solutions, services, and programs. In addition, you can resolve technical issues with online technical support, download and test software packages, and order Cisco learning materials and merchandise. Valuable online skill assessment, training, and certification programs are also available.

Customers and partners can self-register on Cisco.com to obtain additional personalized information and services. Registered users can order products, check on the status of an order, access technical support, and view benefits specific to their relationships with Cisco.

To access Cisco.com, go to the following website:

http://www.cisco.com

Technical Assistance Center

The Cisco TAC website is available to all customers who need technical assistance with a Cisco product or technology that is under warranty or covered by a maintenance contract.

Contacting TAC by Using the Cisco TAC Website

If you have a priority level 3 (P3) or priority level 4 (P4) problem, contact TAC by going to the TAC website:

http://www.cisco.com/tac

P3 and P4 level problems are defined as follows:

- P3—Your network performance is degraded. Network functionality is noticeably impaired, but most business operations continue.
- P4—You need information or assistance on Cisco product capabilities, product installation, or basic product configuration.

In each of the above cases, use the Cisco TAC website to quickly find answers to your questions.

To register for Cisco.com, go to the following website:

http://www.cisco.com/register/

If you cannot resolve your technical issue by using the TAC online resources, Cisco.com registered users can open a case online by using the TAC Case Open tool at the following website:

http://www.cisco.com/tac/caseopen

Contacting TAC by Telephone

If you have a priority level 1(P1) or priority level 2 (P2) problem, contact TAC by telephone and immediately open a case. To obtain a directory of toll-free numbers for your country, go to the following website:

http://www.cisco.com/warp/public/687/Directory/DirTAC.shtml

P1 and P2 level problems are defined as follows:

- P1—Your production network is down, causing a critical impact to business operations if service is not restored quickly. No workaround is available.
- P2—Your production network is severely degraded, affecting significant aspects of your business operations. No workaround is available.

CHAPTER 1

Product Overview

This chapter provides an introduction to the Catalyst 8540 MSR, Catalyst 8510 MSR, and LightStream 1010 ATM switch routers.

Note This chapter provides hardware and software information for the Catalyst 8540 MSR, Catalyst 8510 MSR, and LightStream 1010 ATM switch routers. For descriptions of software features, refer to the *Guide to ATM Technology*.

This chapter includes the following sections:

Layer 3 Enabled ATM Switch Router Hardware Overview

This section provides an overview of the hardware available for the Catalyst 8540 MSR, Catalyst 8510 MSR, and LightStream 1010 Layer 3 enabled ATM switch routers and includes the following sections:

- Layer 3 Enabled ATM Switch Router Hardware (Catalyst 8540 MSR)
- Layer 3 Enabled ATM Switch Router Hardware (Catalyst 8510 MSR and LightStream 1010)

Layer 3 Enabled ATM Switch Router Hardware (Catalyst 8540 MSR)

The Layer 3 enabled ATM switch router uses a 13-slot, modular chassis featuring dual, fault-tolerant, load-sharing AC or DC power supplies. Slots 4 and 8 are occupied by the dual, field-replaceable route processors, which perform central processing functions and provide redundancy. The route processors can also accommodate the network clock module, which features a stratum 3 oscillator and two building integrated timing supply (BITS) ports. Slots 5, 6, and 7 are occupied by either two or three switch processors, for a 20-Gbps non-EHSA or 20-Gbps EHSA switch fabric. The switch processors also accommodate the switch processor feature card.

The remaining slots hold either a full-width module, such as the new four-port OC-12 module, or the carrier module, which in turn accommodates one or two port adapters, such as the four-port OC-3 port adapters. Along with other available interfaces, the ATM switch router provides switched ATM connections to individual workstations, servers, LAN segments, or other ATM switches and routers using fiber-optic, unshielded twisted-pair (UTP), and coaxial cable.

Available Hardware Components (Catalyst 8540 MSR)

The Catalyst 8540 MSR features the following available hardware components:

- Optional switch feature card, supporting usage parameter control (UPC) and statistics
- Optional network clock module
- Full-width 1-port OC-48c single-mode intermediate reach *plus* 4-port OC-12 single-mode fiber interface modules
- Full-width 1-port OC-48c single-mode intermediate reach *plus* 4-port OC-12 multimode fiber interface modules
- Full-width 1-port OC-48c single-mode long reach *plus* 4-port OC-12 multimode fiber interface modules
- Full-width 2-port OC-48c single-mode intermediate reach interface modules
- Full-width 2-port OC-48c single-mode long reach interface modules
- Full-width 4-port OC-12 single-mode intermediate reach interface modules
- Full-width 4-port OC-12 multimode short reach interface modules
- Full-width 16-port OC-3 multimode short reach interface modules
- Full-width ATM router modules
- Full-width 2-port Fast Ethernet interface modules
- Full-width 8-port Gigabit Ethernet interface modules
- Full-width 16-port Fast Ethernet interface modules
- Full-width Enhanced 2-port Gigabit Ethernet interface modules
- Full-width 1-port POS OC-12c/STM-4 SMF-IR and 1-port Gigabit Ethernet interface modules
- Full-width 1-port POS OC-12c/STM-4 SMF-LR and 1-port Gigabit Ethernet interface modules
- Support for the following Catalyst 8510 MSR and LightStream 1010 ATM switch router port adapters via the carrier module:
  - 1-port OC-12 port adapters (multimode, single-mode, and single-mode long reach)
  - 4-port OC-3 port adapters (multimode, single-mode, single-mode long reach, mixed, and UTP)
  - 4-port DS3/E3 port adapters
  - 4-port channelized E1 Frame Relay port adapters
  - 1-port channelized DS3 Frame Relay port adapters
  - 4-port T1/E1 port adapters
  - 4-port T1/E1 circuit emulation service (CES) port adapters
  - 8-port T1/E1 inverse multiplexing over ATM (IMA) port adapters

Layer 3 Enabled ATM Switch Router Hardware (Catalyst 8510 MSR and LightStream 1010)

The Catalyst 8510 MSR and LightStream 1010 ATM switch routers both use a five-slot, modular chassis featuring the option of dual, fault-tolerant, load-sharing AC or DC power supplies. A single, field-replaceable ATM switch processor module supports both the 5-Gbps shared memory and the fully nonblocking switch fabric. The processor also supports the feature card and high performance reduced instruction set computing (RISC) processor (CPU) that provides the central intelligence for the device. The remaining slots support up to four hot-swappable carrier modules. Each carrier module can hold up to two hot-swappable port adapters for a maximum of eight port adapters per switch, supporting a wide variety of desktop, backbone, and wide-area interfaces.

The ATM switch provides switched ATM connections to individual workstations, servers, LAN segments, or other ATM switches and routers using fiber-optic, unshielded twisted-pair (UTP), and coaxial cable.

Note The ATM switch processor and port adapters can be installed in the Catalyst 5500 switch chassis. In the Catalyst 5500 switch chassis the processor must be installed in slot number 13 and the port adapters in slot numbers 9 though 12. The examples in this guide assume that the ATM switch router is in its own chassis, with the processor in slot number 2 and the port adapters in slot numbers 0, 1, 3, and 4.

Processor and Feature Card Models (Catalyst 8510 MSR and LightStream 1010)

The Catalyst 8510 MSR and LightStream 1010 ATM switch routers are equipped with one of the following combinations of processor and feature card:

- ASP-B with feature card per-class queuing (FC-PCQ) or feature card per-flow queuing (FC-PFQ)
- ASP-C with FC-PCQ or FC-PFQ
- Multiservice ATM switch route processor

ASP-B with FC-PCQ and ASP-C with FC-PCQ are functionally equivalent, offering the same features and performance. FC-PFQ, however, provides an enhanced feature set, including advanced traffic management. ASP-B and ASP-C, equipped with FC-PFQ, also provide identical functionality for ATM applications. However, ASP-C with FC-PFQ provides the additional capability for supporting both ATM and Layer 3 switching on the same platform. ASP-C with FC-PFQ and the multiservice ATM switch route processor, used in the Catalyst 8510 MSR, are identical.

FC-PCQ provides a subset of the ATM Forum traffic management features provided by FC-PFQ, as described in Table 1-1

.

Table 1-1 FC-PCQ and FC-PFQ Feature Comparison

Feature	FC-PCQ	FC-PFQ
Traffic classes	CBR[1], RT-VBR[2], NRT-VBR[3], ABR[4] (EFCI[5] and RR[6]), UBR[7]	CBR, RT-VBR, NRT-VBR, ABR (EFCI and RR), UBR
Output queuing	Four classes per port	Per-VC or per-VP
Output scheduling	Strict priority	Strict priority, rate scheduling, and WRR[8]
Intelligent early packet discard	Multiple fixed thresholds	Multiple, weighted, dynamic thresholds
Intelligent tail (partial) packet discard	Supported	Supported
Selective cell marking and discard	Multiple fixed thresholds	Multiple, weighted, dynamic thresholds

Table 1-1 FC-PCQ and FC-PFQ Feature Comparison

Feature	FC-PCQ	FC-PFQ
Shaping	Per-port (pacing)	Per-VC or per-VP (128 shaped VP tunnels)
Policing (UPC[9])	Dual mode, single leaky bucket	Dual leaky bucket
Frame mode VC-merge	–	Supported
Point-to-multipoint VC (multicast)	One leaf per output port, per point-to-multipoint	Multiple leaves per output port, per point-to-multipoint
Network clock switchover	Automatic upon failure	Programmable clock selection criteria
Nondisruptive snooping	Per-port transmit or receive	Per-VC, per-VP, or per-port

1. CBR = constant bit rate
2. RT-VBR = real time variable bit rate
3. NRT-VBR = non real time variable bit rate
4. ABR = available bit rate
5. EFCI = Explicit Forward Congestion Indication
6. RR = relative rate
7. UBR = unspecified bit rate
8. WRR = weighted round-robin
9. UPC = usage parameter control

The Catalyst 8510 MSR is equipped with the multiservice ATM switch route processor.

For additional information, refer to the *Processor Installation Guide*.

Available Physical Interfaces (Catalyst 8510 MSR and LightStream 1010)

The ATM switch router features the following available hardware components:

- The ATM switch router supports the following port adapters:
  - 4-port channelized E1 Frame Relay port adapters
  - 1-port channelized DS3 Frame Relay port adapters
  - 1-port OC-12 port adapters (multimode, single-mode, and single-mode long reach)
  - 4-port OC-3 port adapters (multimode, single-mode, single-mode long reach, mixed, and UTP)
  - 2-port DS3/E3 port adapters
  - 4-port DS3/E3 port adapters
  - 4-port T1/E1 port adapters
  - 4-port T1/E1 circuit emulation service (CES) port adapters
  - 25-Mbps port adapters
  - 8-port T1/E1 inverse multiplexing over ATM (IMA) port adapters
- Full-width ATM router modules
- Full-width 8-port Gigabit Ethernet interface modules
- Full-width 1-port Gigabit Ethernet interface modules

Summary of Software Features

The following sections provide a brief overview of the software features of the Layer 3 enabled ATM switch router, including the following features:

System Availability (Catalyst 8540 MSR)

The Catalyst 8540 MSR provides Enhanced High System Availability (EHSA) during hardware and software upgrades as well as fault resistance with the following features:

- Dual power supplies
- Dual route processors
- Switching fabric with optional spare switch processor
- Optional dual network clock modules

In the event one of the route processors becomes unavailable due to failure or for software upgrade, the secondary route processor takes over with zero boot time. To support switching fabric availability, an optional third switch processor, running in standby mode, takes over if one of the other switch processor cards fails. Finally, the optional network clock modules are able to retain clock configuration should one of the modules fail.

ATM Addressing and Plug-and-Play Operation

The ATM switch router provides the following self-configuring features:

- Preconfigured ATM address prefixes and MAC address, permitting small-scale ATM internetworks to be deployed prior to obtaining officially-allocated ATM addresses
- Automatic reassignment of addresses when reconfiguration is necessary
- Automatic recognition of port adapter types and ATM interface type using ILMI
- Automatic IP address configuration features, such as BOOTP
- Online-insertion-and-replacement (OIR) diagnostic tests

Connections

The ATM switch router supports connections with the following characteristics:

- Full 8-bit virtual path identifier (VPI) and 16-bit virtual channel identifier (VCI) with configurable boundaries.
- 12-bit VPI support available on ATM Network-Network Interface (NNI) interfaces on the Catalyst 8510 MSR and LightStream 1010
- Up to 256,000 total virtual connections on the Catalyst 8540 MSR and up to 64,000 total virtual connections on the Catalyst 8510 MSR and LightStream 1010
- VC and virtual path (VP) switching, VP tunneling, and VC merging
- The following virtual connection types:
  - Permanent virtual channel (PVC) connections
  - Permanent virtual path (PVP) connections
  - Soft permanent virtual channel (soft PVC) and soft permanent virtual path (soft PVP) connections with route optimization
  - Switched virtual channel (SVC) and switched virtual path (SVP) connections
  - Virtual path (VP) tunneling with traffic shaping and QoS guarantees for multiple service categories (hierarchical VP tunnels)
  - Point-to-point ATM connections
  - Point-to-multipoint ATM connections
- F4 and F5 Operation, Administration, and Maintenance (OAM) segment-loopback and end-to-end remote deflect identification (RDI) and alarm indication signal (AIS)
- OAM-based ping of IP or ATM address on the Catalyst 8510 MSR and LightStream 1010
- Frame Relay to ATM interworking features on the channelized E1 port adapter:
  - PVCs and soft-VCs with Network Interworking
  - PVCs and soft-VCs with Service Interworking
  - Support for various LMIs

Resource Management

Resource management provides support for the following features:

- Traffic categories:
  - Constant bit rate (CBR)
  - Real-time variable bit rate (VBR-RT)
  - Non-real time variable bit rate (VBR-NRT)
  - Available bit rate (ABR) + minimum cell rate (MCR)
  - Unspecified bit rate (UBR) + MCR

Note FC-PCQ-equipped systems only support MCR value 0 for ABR and UBR traffic categories.

- Quality of service (QoS) guarantees with traffic policing and intelligent packet discard
- Connection admission control (CAC)
- Congestion control and traffic pacing

Note Some newer port adapters do not support traffic pacing.

- ABR with explicit forward congestion indication (EFCI) and relative rate (RR) marking

Note Relative rate marking of ABR traffic is not supported on the Catalyst 8540 MSR.

Signalling and Routing

The following signalling and routing features are supported:

- User-Network Interface (UNI) 3.0, 3.1, and 4.0
- Integrated Local Management Interface 4.0
- ATM network service access point (NSAP) and E.164 addressing
- Interim Interswitch Signalling Protocol (IISP) routing protocol
- Single-level and full hierarchical Private Network-Network Interface (PNNI) routing protocol, including PNNI complex node support
- Closed user groups (CUGs) for ATM virtual private networks (VPNs)
- ATM signalling and ILMI access lists with support for time of day-based policies
- ATM anycast

ATM Internetworking Services (Catalyst 8540 MSR)

The following internetworking services are provided:

- LAN emulation configuration server (LECS), LAN emulation server (LES), and broadcast-and-unknown server (BUS) for Ethernet emulated LANs (ELANs)
- Cisco Simple Server Redundancy Protocol (SSRP) for LANE
- RFC 1577 classical IP over ATM and Address Resolution Protocol (ARP) server and client
- Tag switching for Open Shortest Path First (OSPF), Routing Information Protocol (RIP), and Enhanced Interior Gateway Routing Protocol (EIGRP) routing of IP packets
- ATM Circuit Emulation Service (CES) as defined by ATM Forum CES 1.0
- RFC 1483 multiprotocol encapsulation over ATM

ATM Internetworking Services (Catalyst 8510 MSR and LightStream 1010)

The following internetworking services are provided:

- LAN emulation configuration server (LECS), LAN emulation server (LES), and broadcast and unknown server (BUS) for Ethernet and Token Ring emulated LANs (ELANs)
- Cisco Simple Server Redundancy Protocol (SSRP) for LANE
- RFC 1577 classical IP over ATM and Address Resolution Protocol (ARP) server and client
- Tag switching for Open Shortest Path First (OSPF) routing of IP packets
- ATM Circuit Emulation Service (CES) as defined by ATM Forum CES 1.0
- RFC 1483 multiprotocol encapsulation over ATM

Network Clocking

Any interface on the ATM switch router can be synchronized to an internal source (system clock) or to an external source, such as another network. Synchronous residual time stamp (SRTS), and adaptive clocking modes are supported for CES.

With the optional network clock module on the Catalyst 8540 MSR, the ATM switch router can be synchronized to a BITS source or to the module's own stratum 3 clock.

Management and Monitoring

The following features provide support for managing the ATM switch router:

- Text-based command-line interface (CLI) for configuration and troubleshooting
- Simple Network Management Protocol (SNMP) agent provides dynamic status, statistics, and configuration information
- Configuration and system image files saved in NVRAM and Flash memory
- Boot from network or from Flash memory
- Upload and download system images using Trivial File Transfer Protocol (TFTP)
- Update hardware controller microcode independently of system image on channelized E1 port adapter
- In-band device network management using IP over ATM
- In-band device network management using LAN emulation client, RFC 1577 client, and RFC 1483 client
- Out-of-band device network management using Ethernet and console ports
- ATM forum and enterprise Management Information Bases (MIBs) including, but not limited to, the following features:
  - AToM MIB RFC1695
  - SVC MIB
  - ILMI MIB
  - PNNIv1.0 MIB
  - ATM Signaling and Diagnostic MIB

- ATM RMON MIB
- ATM Accounting MIB

- Port, VC, and VP snooping for monitoring and troubleshooting
- ATM accounting
  - Remote and local periodic collection of records
  - Accounting records for PVC/PVPs
  - 5-second peak interval transmit and receive cell counter for PVC/PVPs only
- Online diagnostics tests that run in the background and monitor system hardware status

Available Network Management Applications

The CiscoWorks 2000 family of network management software provides tools for managing your ATM switch router. CiscoWorks 2000 includes the following packages:

- CWSI Resource Manager Essentials—a suite of web-based network management tools that allow you to collect the monitoring, fault, and availability information needed to track devices.
- CWSI Campus—a suite of network management applications that allow you to configure, monitor, and manage a switched internetwork.

The functionality provided by the CWSI Campus suite of applications includes the following features:

- Automatically discover and display a map of your enterprise or campus network
- Display and configure emulated LANs
- Configure PNNI
- Obtain end-station user information
- Display and configure device information
- Monitor traffic

Layer 3 Features

With the ATM router module, the ATM switch router support the following Layer 3 features:

- Bridging
- Integrated routing and bridging (IRB)
- IP fragmentation support
- IP multicast routing
- IP and IPX load balancing
- Routing protocol MIB support
- ISL trunking for routing and bridging
- Standard and extended ACL support for IP
- Standard ACL support for IPX
- Packet over SONET (POS) RFC 1619 PPP support
- POS RFC 1662 PPP

CHAPTER 2

Understanding the User Interface

This chapter describes the ATM switch router user interface and provides instructions for using the command-line interface (CLI).

Note This chapter provides advanced configuration instructions for the Catalyst 8540 MSR, Catalyst 8510 MSR, and LightStream 1010 ATM switch routers. For complete descriptions of the commands mentioned in this chapter, refer to the *ATM Switch Router Command Reference* publication.

The following sections are included:

User Interface Overview

The user interface for the ATM switch router provides access to several different command modes, each with related commands. Users familiar with the Cisco IOS user interface will find the interfaces very similar. This chapter describes how to access and list the commands available in each command mode, and explains the primary uses for each command mode.

For security purposes, the user interface provides two levels of command access: *user* and *privileged*. The unprivileged user mode is called user EXEC mode; the privileged mode is called privileged EXEC mode, and requires a password.

Note Because all commands available in user EXEC mode are also available in privileged EXEC mode, user EXEC mode is referred to as EXEC mode in this guide.

From the privileged level, you can access global configuration mode; from global configuration mode you can access numerous submodes that allow you to configure specific, related features. Read-only memory (ROM) monitor mode accesses a basic system kernel to which the ATM switch router may default at startup if it does not find a valid system image, or if its configuration file is corrupted.

You can enter commands in uppercase, lowercase, or a mix of both. Only passwords are case sensitive. You can abbreviate commands and keywords to a minimum unique string of characters. For example, you can abbreviate the **show** command to **sh**. After entering the command line at the system prompt, press the **Return** key to execute the command.

Almost every configuration command has a **no** form. In general, use the **no** form to disable a feature or function. Use the command without the **no** keyword to reenable a disabled feature or enable a feature disabled by default.

Note Refer to the *ATM Switch Router Command Reference* publication for the complete syntax of commands specific to the ATM switch router and a description of the function of the **no** form of a command. Refer to the *Configuration Fundamentals Command Reference* publication for the complete syntax of other IOS commands.

Accessing Each Command Mode

This section describes how to access the command modes for the ATM switch router. Table 2-1 and Table 2-2 list the command modes, access to each mode, the prompt you see while in that mode, the main uses for each configuration mode, and the method to exit that mode. The prompts listed assume the default ATM switch router name "Switch." Table 2-1 and Table 2-2 might not include all of the possible ways to access or exit each command mode.

Table 2-1 Summary of Command Modes

Command Mode	Access Method	Prompt	Exit Method
EXEC (user)	Log in to the ATM switch router.	`Switch>`	Use the **logout** command.
Privileged EXEC	From user EXEC mode, use the **enable** EXEC command and enter your password.	`Switch#`	To return to user EXEC mode, use the **disable** command.
ROM monitor	From privileged EXEC mode, use the **reload** EXEC command. Press **Break** during the first 60 seconds while the system boots.	`>`	To exit to user EXEC mode, type **continue**.
Global configuration	From privileged EXEC mode, use the **configure** privileged EXEC command. Use the keyword **terminal** to enter commands from your terminal.	`Switch(config)#`	To exit to privileged EXEC mode, use the **exit** or **end** command or press **Ctrl-Z**.
Interface configuration	From global configuration mode, specify an interface with an **interface** command.	`Switch(config-if)#`	To exit to global configuration mode, use the **exit** command. To exit directly to privileged EXEC mode, use the **end** command or press **Ctrl-Z**.

Table 2-1 Summary of Command Modes (continued)

Command Mode	Access Method	Prompt	Exit Method
Subinterface configuration	From interface configuration mode, specify a subinterface with an **interface** command.	`Switch(config-subif)#`	To exit to global configuration mode, use the **exit** command. To exit directly to privileged EXEC mode, use the **end** command or press **Ctrl-Z**.
Line configuration	From global configuration mode, specify a line with a **line** command.	`Switch(config-line)#`	To exit to global configuration mode, use the **exit** command. To exit directly to privileged EXEC mode, use the **end** command or press **Ctrl-Z**.
Map-list configuration	From global configuration mode, define a map list with the **map-list** command.	`Switch(config-map-list)#`	To exit to global configuration mode, use the **exit** command. To enter map-class configuration mode, use the **map-class** command. To exit directly to privileged EXEC mode, use the **end** command or press **Ctrl-Z**.
Map-class configuration	From global configuration mode, configure a map class with the **map-class** command.	`Switch(config-map-class)#`	To exit to global configuration mode, use the **exit** command. To enter map-list configuration mode, use the **map-list** command. To exit directly to privileged EXEC mode, use the **end** command or press **Ctrl-Z**.
ATM router configuration	From global configuration mode, configure the PNNI routing protocol with the **atm router pnni** command.	`Switch(config-atm-router)#`	To exit to global configuration mode, use the **exit** command. To exit directly to privileged EXEC mode use the **end** command or press **Ctrl-Z**.
PNNI node configuration	From ATM router configuration mode, configure the PNNI routing node with the **node** command.	`Switch(config-pnni-node)#`	To exit to ATM router configuration mode, use the **exit** command. To exit directly to privileged EXEC mode, use the **end** command or press **Ctrl-Z**.

Table 2-1 Summary of Command Modes (continued)

Command Mode	Access Method	Prompt	Exit Method
PNNI explicit path configuration	From global configuration mode, enter the **atm pnni explicit-path** command.	`Switch(cfg-pnni-expl-path)#`	To exit to global configuration mode, use the **exit** command. To exit directly to privileged EXEC mode, use the **end** command or press **Ctrl-Z**.
ATM accounting file configuration	From global configuration mode, define an ATM accounting file with the **atm accounting file** command.	`Switch(config-acct-file)#`	To exit to global configuration mode, use the **exit** command. To exit directly to privileged EXEC mode, use the **end** command or press **Ctrl-Z**.
ATM accounting selection configuration	From global configuration mode, define an ATM accounting selection table entry with the **atm accounting selection** command.	`Switch(config-acct-sel)#`	To exit to global configuration mode, use the **exit** command. To exit directly to privileged EXEC mode, use the **end** command or press **Ctrl-Z**.
LANE configuration server database configuration	From global configuration mode, specify a LANE configuration server database name with the **lane database** command.	`Switch(lane-config-database)#`	To exit to global configuration mode, use the **exit** command. To exit directly to privileged EXEC mode, use the **end** command or press **Ctrl-Z**.
ATM E.164 translation table configuration	From global configuration mode, enter the **atm e164 translation-table** command	`Switch(config-atm-e164)#`	To exit to global configuration mode, use the **exit** command. To exit directly to privileged EXEC mode, use the **end** command or press **Ctrl-Z**.
ATM signalling diagnostics configuration	From global configuration mode, enter the **atm signalling diagnostics** command and an index to configure.	`Switch(cfg-atmsig-diag)#`	To exit to global configuration mode, use the **exit** command. To exit directly to privileged EXEC mode, use the **end** command or press **Ctrl-Z**.
Controller configuration	From global configuration mode, enter the **controller** command.	`Switch(config-controller)#`	To exit to global configuration mode, use the **exit** command. To exit directly to privileged EXEC mode, use the **end** command or press **Ctrl-Z**.

Table 2-2 Summary of Additional Command Modes (Catalyst 8540 MSR)

Command Mode	Access Method	Prompt	Exit Method
Redundancy configuration	From global configuration mode, enter the **redundancy** command.	`Switch(config-r)#`	To exit to global configuration mode, use the **exit** command. To exit directly to privileged EXEC mode, use the **end** command or press **Ctrl-Z**.
Main CPU configuration	From redundancy configuration mode, enter the **main-cpu** command.	`Switch(config-r-mc)#`	To exit to redundancy configuration mode, use the **exit** command. To exit directly to privileged EXEC mode, use the **end** command or press **Ctrl-Z**.

EXEC Mode

When you log in to the ATM switch router, you are in user EXEC, or simply EXEC, command mode. The EXEC commands available at the user level are a subset of those available at the privileged level. In general, the user-level EXEC commands allow you to connect to remote devices, change terminal settings on a temporary basis, perform basic tests, and list system information.

The user-level prompt consists of the ATM switch router's host name followed by the angle bracket (>):

```
Switch>
```

The default host name is *Switch*, unless it has been changed during using the **hostname** global configuration command.

Privileged EXEC Mode

The privileged EXEC command set includes all user-level EXEC mode commands and the **configure** command, through which you can access global configuration mode and the remaining configuration submodes. Privilege EXEC mode also includes high-level testing commands, such as **debug**, and commands that display potentially secure information.

To enter privileged EXEC mode from EXEC mode, use the **enable** command and enter your password; the prompt changes to the ATM switch router's host name followed by the pound sign (#):

```
Switch> enable
Password:
Switch#
```

To exit from privileged EXEC mode back to EXEC mode, use the **disable** command.

```
Switch# disable
Switch>
```

The system administrator uses the **enable password** global configuration command to set the password, which is case sensitive. If an enable password has not been set, privileged EXEC mode can only be accessed from the console.

ROM Monitor Mode

ROM monitor mode provides access to a basic system kernel, from which you can boot the ATM switch router or perform diagnostic tests. If a valid system image is not found, or if the configuration file is corrupted, the system might enter ROM monitor mode. The ROM monitor prompt is the angle bracket:

```
>
```

You can also enter ROM monitor mode by intentionally interrupting the boot sequence with the **Break** key during loading. For a description of this process, refer to the *Configuration Fundamentals Configuration Guide*.

To return to EXEC mode from ROM monitor mode, use the **continue** command:

```
> continue
Switch>
```

Global Configuration Mode

Global configuration mode provides access to commands that apply to the entire system. From global configuration mode you can also enter the other configuration modes described in the following subsections.

To enter global configuration mode from privileged EXEC mode, enter the **configure** command and specify the source of the configuration commands at the prompt; the prompt changes to the ATM switch router's hostname followed by (config)#:

```
Switch# configure
Configuring from terminal, memory, or network [terminal]? <CR>
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)#
```

You can specify either the terminal, nonvolatile memory (NVRAM), or a file stored on a network server as the source of configuration commands. For more information, see Chapter 22, "Managing Configuration Files, System Images, and Functional Images." The default is to enter commands from the terminal console.

As a shortcut for accessing the terminal method of configuration, enter the following:

```
Switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)#
```

To exit global configuration command mode and return to privileged EXEC mode, use the **exit** or **end** command, or press **Ctrl-Z**:

```
Switch(config)# end
Switch#
```

Interface Configuration Mode

Interface configuration mode provides access to commands that apply on a per-interface basis. These commands modify the operation of an interface such as an ATM, Ethernet, or asynchronous port.

To enter interface configuration mode from global configuration mode, use the **interface** command with a keyword indicating the interface type, followed by an interface number; the prompt changes to the ATM switch router's hostname followed by (config-if)#:

```
Switch(config)# interface atm 3/0/0
Switch(config-if)#
```

To exit interface configuration mode and return to global configuration mode, use the **exit** command:

```
Switch(config-if)# exit
Switch(config)#
```

To exit interface configuration mode and return to privileged EXEC mode, use the **end** command or press **Ctrl-Z**:

```
Switch(config-if)# end
Switch#
```

Interface Addressing Formats (Catalyst 8540)

In the ATM switch router chassis, you specify interfaces in slots 0 through 3 and 9 through 12 using the *card/subcard/port* format. Slots 4 and 8 each contain a CPU (multiservice route processor). Because the configurations on the primary and secondary route processors are automatically synchronized, they are configured via a single network interface, specified as **atm0** or **ethernet0**. There is no need to configure the secondary separately from the primary, but some show commands allow you to display information about the secondary route processor; in these cases, you specify the interface as **atm-sec0** or **ethernet-sec0**. Slots 5 through 7 contain the switch processors, which have no interfaces. Table 2-3 summarizes this addressing scheme, assuming that slot 4 is the primary route processor and slot 8 is the secondary route processor.

Table 2-3 Interface Addressing Formats (Catalyst 8540)

Slot	Addressing Format
0	*card/subcard/port*
1	*card/subcard/port*
2	*card/subcard/port*
3	*card/subcard/port*
4	**atm0** or **ethernet0**
5	-
6	-
7	-
8	**atm-sec0** or **ethernet-sec0**
9	*card/subcard/port*
10	*card/subcard/port*
11	*card/subcard/port*
12	*card/subcard/port*

The following example shows how to enter interface configuration mode to configure the Ethernet interface on the CPU:

```
Switch(config)# interface ethernet0
Switch(config-if)#
```

CPU Interface Address Format (Catalyst 8510 MSR and LightStream 1010)

With this release of the ATM switch router software, addressing the interface on the processor (CPU) has changed. The ATM interface is now called atm0, and the Ethernet interface is now called ethernet0.

The following example shows how to enter interface configuration mode to configure the Ethernet interface on the processor:

```
Switch(config)# interface ethernet0
Switch(config-if)#
```

Note The old formats (atm 2/0/0 and ethernet 2/0/0) are still supported in this release.

Subinterface Configuration Mode

Subinterface configuration mode allows access to commands that affect logical interfaces, also called subinterfaces. Subinterfaces are used, for example, to configure multiple VP tunnels on a single interface.

To enter subinterface configuration command mode from global configuration or interface configuration mode, use the **interface** command with a keyword indicating the interface type, followed by an interface and subinterface number; the prompt changes to the ATM switch router's hostname followed by (config-subif)#:

```
Switch(config)# interface atm 0/0/0.99
Switch(config-subif)#
```

To exit subinterface configuration mode and return to global configuration mode, use the **exit** command:

```
Switch(config-subif)# exit
Switch(config)#
```

To exit interface configuration mode and return to privileged EXEC mode, use the **end** command or press **Ctrl-Z**:

```
Switch(config-subif)# end
Switch#
```

Line Configuration Mode (Catalyst 8540 MSR)

Line configuration mode on the Catalyst 8540 MSR provides access to commands that modify the operation of individual terminal lines. These commands are used to configure the console, and vty connections, set up modem connections, and so on.

To enter line configuration mode from global configuration mode, use the **line** command followed by a line type (**console** or **vty**) and a line number or range; the prompt changes to the ATM switch router's hostname followed by (config-line)#:

```
Switch(config)# line vty 0
Switch(config-line)#
```

For detailed line configuration instructions, refer to the *Configuration Fundamentals Configuration Guide*.

To exit line configuration mode and return to global configuration mode, use the **exit** command:

```
Switch(config-line)# exit
Switch(config)#
```

To exit line configuration mode and return to privileged EXEC mode, use the **end** command or press **Ctrl-Z**:

```
Switch(config-line)# end
Switch#
```

Line Configuration Mode (Catalyst 8510 MSR and LightStream 1010)

Line configuration mode on the Catalyst 8510 MSR and LightStream 1010 ATM switch router provides access to commands that modify the operation of individual terminal lines. These commands are used to configure the console, auxiliary, and vty connections, set up modem connections, and so on.

To enter line configuration mode from global configuration mode, use the **line** command followed by a line type (**aux**, **console**, or **vty**) and a line number or range; the prompt changes to the ATM switch router's hostname followed by (config-line)#:

```
Switch(config)# line vty 0
Switch(config-line)#
```

For detailed line configuration instructions, refer to the *Configuration Fundamentals Configuration Guide*.

To exit line configuration mode and return to global configuration mode, use the **exit** command:

```
Switch(config-line)# exit
Switch(config)#
```

To exit line configuration mode and return to privileged EXEC mode, use the **end** command or press **Ctrl-Z**:

```
Switch(config-line)# end
Switch#
```

Map-List Configuration Mode

Map-list configuration mode provides access to commands used to statically map protocol addresses of remote hosts or switches to permanent virtual connections (PVCs) or switched virtual connections (SVCs).

To enter map-list configuration mode from global configuration mode, use the **map-list** command followed by a map-list name to configure; the prompt changes to the ATM switch router's hostname followed by (config-map-list)#:

```
Switch(config)# map-list newlist
Switch(config-map-list)#
```

You can also use the **map-list** command to enter map-list configuration mode directly from map-class configuration mode, without first returning to global configuration mode:

```
Switch(config-map-class)# map-list newlist
Switch(config-map-list)#
```

To exit map-list configuration mode and return to global configuration mode, use the **exit** command:

```
Switch(config-map-list)# exit
Switch(config)#
```

To exit map-list configuration mode and return to privileged EXEC mode, use the **end** command or press **Ctrl-Z**:

```
Switch(config-map-list)# end
Switch#
```

Map-Class Configuration Mode

Map-class configuration mode provides access to command used to define the traffic parameters when specifying a request for a switched virtual channel (SVC).

To enter map-class configuration mode from global configuration mode, enter the **map-class** command followed by a class name to configure; the prompt changes to the ATM switch router's hostname followed by (config-map-class)#:

```
Switch(config)# map-class atm newclass
Switch(config-map-class)#
```

You can also use the **map-class** command to enter map-class configuration mode directly from map-list configuration mode, without first returning to global configuration mode:

```
Switch(config-map-list)# map-class atm newclass
Switch(config-map-class)#
```

To exit map-class configuration mode and return to global configuration mode, use the **exit** command:

```
Switch(config-map-class)# exit
Switch(config)#
```

To exit map-class configuration mode and return to privileged EXEC mode, use the **end** command or press **Ctrl-Z**:

```
Switch(config-map-class)# end
Switch#
```

ATM Router Configuration Mode

ATM router configuration mode provides access to commands used to configure Private Network-Network Interface (PNNI) routing.

To enter ATM router configuration mode from global configuration mode, use the **atm router pnni** command; the prompt changes to the ATM switch router's hostname followed by (config-atm-router)#:

```
Switch(config)# atm router pnni
Switch(config-atm-router)#
```

To exit ATM router configuration mode and return to global configuration mode, use the **exit** command:

```
Switch(config-atm-router)# exit
Switch(config)#
```

To exit ATM router configuration mode and return to privileged EXEC mode, use the **end** command or press **Ctrl-Z**:

```
Switch(config-atm-router)# end
Switch#
```

For detailed information on configuring PNNI routing, see Chapter 10, "Configuring ATM Routing and PNNI."

PNNI Node Configuration Mode

The PNNI node configuration mode is a submode of ATM router configuration mode and provides access to commands you use to configure PNNI nodes on the ATM switch router.

To enter PNNI node configuration mode from ATM router configuration mode, use the **node** command followed by a node index; the prompt changes to the ATM switch router's hostname followed by (config-pnni-node)#:

```
Switch(config-atm-router)# node 1
Switch(config-pnni-node)#
```

To exit PNNI node configuration mode and return to ATM router configuration mode, use the **exit** command:

```
Switch(config-pnni-node)# exit
Switch(config-atm-router)#
```

To exit PNNI node configuration mode and return to privileged EXEC mode, use the **end** command or press **Ctrl-Z**:

```
Switch(config-pnni-node)# end
Switch#
```

For detailed information on configuring PNNI nodes, see Chapter 10, "Configuring ATM Routing and PNNI."

PNNI Explicit Path Configuration Mode

The PNNI explicit path configuration mode provides access to commands used to manually configure fully specified or partially specified paths for routing soft permanent virtual channel (soft PVC) and soft permanent virtual path (soft PVP) connections.

To enter the PNNI explicit path configuration mode from global configuration mode, use the **atm pnni explicit-path** command followed by an explicit path name or path-id number; the prompt changes to the ATM switch router's hostname followed by (cfg-pnni-expl-path)#:

```
Switch(config)# atm pnni explicit-path name newexplicit-path
Switch(cfg-pnni-expl-path)#
```

To exit PNNI explicit path configuration mode and return to global configuration mode, use the **exit** command:

```
Switch(cfg-pnni-expl-path)# exit
Switch(config)#
```

To exit PNNI explicit path configuration mode and return to privileged EXEC mode, use the **end** command or press **Ctrl-Z**:

```
Switch(cfg-pnni-expl-path)# end
Switch#
```

For detailed information on configuring PNNI explicit paths, see Chapter 10, "Configuring ATM Routing and PNNI."

ATM Accounting File Configuration Mode

ATM accounting file configuration mode provides access to commands used to configure a file for accounting and billing of virtual circuits (VCs).

To enter ATM accounting file configuration mode from global configuration mode, use the **atm accounting file** command followed by an accounting filename; the prompt changes to the ATM switch router hostname followed by (config-acct-file)#:

```
Switch(config)# atm accounting file acctng_file1
Switch(config-acct-file)#
```

To exit ATM accounting file configuration mode and return to global configuration mode, use the **exit** command:

```
Switch(config-acct-file)# exit
Switch(config)#
```

To exit ATM accounting file configuration mode and return to privileged EXEC mode, use the **end** command or press **Ctrl-Z**:

```
Switch(config-acct-file)# end
Switch#
```

For detailed information on configuring ATM accounting, see Chapter 14, "Configuring ATM Accounting and ATM RMON."

ATM Accounting Selection Configuration Mode

ATM accounting selection configuration mode provides access to commands used to specify the connection data to be gathered from the ATM switch router.

To enter ATM accounting selection configuration mode, use the **atm accounting selection** command and specify an accounting selection index; the prompt changes to the ATM switch router's hostname followed by (config-acct-sel)#:

```
Switch(config)# atm accounting selection 1
Switch(config-acct-sel)#
```

To exit ATM accounting selection configuration mode and return to global configuration mode, use the **exit** command:

```
Switch(config-acct-sel)# exit
Switch(config)#
```

To exit ATM accounting selection configuration mode and return to privileged EXEC mode, use the **end** command or press **Ctrl-Z**:

```
Switch(config-acct-sel)# end
Switch#
```

For detailed information on configuring ATM accounting selections, see Chapter 14, "Configuring ATM Accounting and ATM RMON."

LANE Configuration Server Database Configuration Mode

LAN emulation (LANE) configuration server database configuration mode provides access to commands used to define the LANE configuration server database.

To enter LANE configuration server database configuration mode from global configuration mode, use the **lane database** command and specify a database name; the prompt changes to the ATM switch router's hostname followed by (lane-config-database)#:

```
Switch(config)# lane database lecsdb
Switch(lane-config-database)#
```

To exit LANE configuration server database configuration mode and return to global configuration mode, use the **exit** command:

```
Switch(lane-config-database)# exit
Switch(config)#
```

To exit LANE configuration server database configuration mode and return to privileged EXEC mode, use the **end** command or press **Ctrl-Z**:

```
Switch(lane-config-database)# end
Switch#
```

For detailed information on configuring the LAN emulation configuration server database, see Chapter 13, "Configuring LAN Emulation."

ATM E.164 Translation Table Configuration Mode

ATM E.164 translation table configuration mode provides access to commands used to configure the translation table that maps native E.164 format addresses to ATM end system (AESA) format addresses.

To enter ATM E.164 translation table configuration mode from global configuration mode, use the **atm e164 translation-table** command; the prompt changes to the ATM switch router's hostname followed by (config-atm-e164)#:

```
Switch(config)# atm e164 translation-table
Switch(config-atm-e164)
```

To exit ATM E.164 translation table configuration mode and return to global configuration mode, use the **exit** command:

```
Switch(config-atm-e164)# exit
Switch(config)#
```

To exit ATM E.164 translation table configuration mode and return to privileged EXEC mode, use the **end** command or press **Ctrl-Z**:

```
Switch(config-atm-e164)# end
Switch#
```

For detailed information on configuring E.164 addresses, see the "Configuring E.164 Addresses" section on page 16-4.

ATM Signalling Diagnostics Configuration Mode

ATM signalling diagnostics configuration mode provides access to commands used to configure the signalling diagnostics table.

To enter ATM signalling diagnostics configuration mode from global configuration mode, use the **atm signalling diagnostics** command and specify an index for the filter table; the prompt changes to the ATM switch router's hostname followed by (cfg-atmsig-diag):

```
Switch(config)# atm signalling diagnostics 1
Switch(cfg-atmsig-diag)
```

To exit ATM signalling diagnostics configuration mode and return to global configuration mode, use the **exit** command:

```
Switch(cfg-atmsig-diag)# exit
Switch(config)#
```

To exit ATM signalling diagnostics configuration mode and return to privileged EXEC mode, use the **end** command or press **Ctrl-Z**:

```
Switch(cfg-atmsig-diag)# end
Switch#
```

For detailed information on configuring signalling diagnostics, see the "Configuring Signalling Diagnostics Tables" section on page 16-12.

Controller Configuration Mode

Controller configuration mode provides access to commands used to configure physical and logical parameters of a channelized interface.

To enter ATM controller configuration mode from global configuration mode, use the **controller** command with a channel type and interface:

```
Switch(config)# controller e1 1/0/0
Switch(config-controller)#
```

To exit ATM controller configuration mode and return to global configuration mode, use the **exit** command:

```
Switch(config-controller)# exit
Switch(config)#
```

To exit ATM controller configuration mode and return to privileged EXEC mode, use the **end** command or press **Ctrl-Z**:

```
Switch(config-controller)# end
Switch#
```

For detailed information on configuring channel groups on a Frame Relay/FUNI interface, see Chapter 19, "Configuring Frame Relay to ATM Interworking Port Adapter Interfaces."

Redundancy Configuration Mode (Catalyst 8540 MSR)

Redundancy configuration mode provides access to commands used to configure system redundancy and EHSA operation.

To enter redundancy configuration mode from global configuration mode, use the **redundancy** command; the prompt changes to the ATM switch router's hostname followed by (config-r):

```
Switch(config)# redundancy
Switch(config-r)#
```

To exit ATM redundancy configuration mode and return to global configuration mode, use the **exit** command:

```
Switch(config-r)# exit
Switch(config)#
```

To exit ATM redundancy configuration mode and return to privileged EXEC mode, use the **end** command or press **Ctrl-Z**:

```
Switch(config-r)# end
Switch#
```

For detailed information on configuring system redundancy, see the "Configuring Redundancy and Enhanced High System Availability (Catalyst 8540 MSR)" section on page 3-23.

Main CPU Configuration Mode (Catalyst 8540 MSR)

Main CPU configuration mode provides access to commands used to synchronize the configuration of the primary and secondary route processors.

To enter main CPU configuration mode from redundancy configuration mode, use the **main-cpu** command; the prompt changes to the ATM switch router's hostname followed by (config-r-mc):

```
Switch(config-r)# main-cpu
Switch(config-r-mc)#
```

To exit ATM main CPU configuration mode and return to redundancy configuration mode, use the **exit** command:

```
Switch(config-r-mc)# exit
Switch(config-r)#
```

To exit ATM main cpu configuration mode and return to privileged EXEC mode, use the **end** command or press **Ctrl-Z**:

```
Switch(config-r-mc)# end
Switch#
```

For detailed information on synchronizing configurations, see the "Configuring Redundancy and Enhanced High System Availability (Catalyst 8540 MSR)" section on page 3-23.

Additional Cisco IOS CLI Features

Because the ATM switch router's operating system is based on Cisco IOS software, its interface provides a number of features that help you use the CLI with greater flexibility, ease, and power. These features includes the following:

- Context-sensitive help—allows you to obtain a list of commands available for each command mode or a list of available options for a specific command by entering a question mark (**?**).
- Command history—records a history of commands, allowing you to recall previously entered long or complex commands.
- Editing—provides the ability to move around the command line, cut and paste entries, control scrolling, create keyboard macros, and so on.

For information on using these and other features of Cisco IOS software, refer to the *Configuration Fundamentals Configuration Guide*.

CHAPTER 3

Initially Configuring the ATM Switch Router

This chapter discusses specific steps used to initially configure the ATM switch router.

Note This chapter provides advanced configuration instructions for the Catalyst 8540 MSR, Catalyst 8510 MSR, and LightStream 1010 ATM switch routers. For conceptual and background information, refer to the *Guide to ATM Technology*. For complete descriptions of the commands mentioned in this chapter, refer to the *ATM Switch Router Command Reference* publication.

This chapter includes the following sections:

Methods for Configuring the ATM Switch Router

The ATM switch router defaults to a working configuration suitable for most networks. However, you might need to customize the configuration for your network.

Note If your Telnet station or SNMP network management workstation is on a different network from the switch, you must add a static routing table entry to the routing table. See the "Configuring Static Routes" section on page 10-6.

Terminal Line Configuration (Catalyst 8540 MSR)

The Catalyst 8540 MSR has a console terminal line that might require configuration. For line configuration, you must first set up the line for the terminal or the asynchronous device attached to it. For a complete description of configuration tasks and commands used to set up your terminal line and settings, refer to the *Configuration Fundamentals Configuration Guide* and *Dial Solutions Configuration Guide*.

You can connect a modem to the console port. The following settings on the modem are required:

- Enable auto answer mode
- Suppress result codes

You can configure your modem by setting the DIP switches on the modem or by connecting the modem to terminal equipment. Refer to the user manual provided with your modem for the correct configuration information.

Note Because there are no hardware flow control signals available on the console port, the console port terminal characteristics should match the modem settings.

Terminal Line Configuration (Catalyst 8510 MSR and LightStream 1010)

The ATM switch has two types of terminal lines: a console line and an auxiliary line. For line configuration, you must first set up the lines for the terminals or other asynchronous devices attached to them. For a complete description of configuration tasks and commands used to set up your lines, modems, and terminal settings, refer to the *Configuration Fundamentals Configuration Guide* and *Dial Solutions Configuration Guide*.

Configuration Prerequisites

Consider the following information you might need before you configure your ATM switch router:

- If you want to configure a BOOTP server to inform the switch of its Ethernet IP address and mask, you need the Media Access Control (MAC) address of the Ethernet port.
- If you want to configure a new ATM address for the switch (an autoconfigured ATM address is assigned by Cisco), you need an ATM address assigned by your system administrator.
- If you are not using BOOTP, you need an IP address and a netmask address.

Verifying Software and Hardware Installed on the ATM Switch Router

When you first power up your console and ATM switch router, a screen similar to the following from a Catalyst 8540 MSR appears:

```
Restricted Rights Legend

Use, duplication, or disclosure by the Government is
subject to restrictions as set forth in subparagraph
(c) of the Commercial Computer Software - Restricted
Rights clause at FAR sec. 52.227-19 and subparagraph
(c) (1) (ii) of the Rights in Technical Data and Computer
Software clause at DFARS sec. 252.227-7013.

           cisco Systems, Inc.
           170 West Tasman Drive
           San Jose, California 95134-1706



Cisco Internetwork Operating System Software
IOS (tm) PNNI Software (cat8540m-WP-M), Version 12.0(4a)W5(10.44),  INTERIM TEST
 SOFTWARE
Copyright (c) 1986-1999 by cisco Systems, Inc.
Compiled Tue 17-Aug-99 03:18 by
Image text-base: 0x60010930, data-base: 0x60936000



CUBI Driver subsystem initializing ...

primary interrupt reg read FFC00
secondary interrupt reg read EA800
*** this cpu is the primary
Enabling the MS timer



Switch Fabric Driver subsystem initializing ...

found
 smid=0
 smid=2
 smid=4
 smid=6
 smid=1
 smid=3
 smid=5
 smid=7
in cfc_init

... DONE
```

```
IDPROM in slot 0 not properly programmed
cisco C8540MSR (R5000) processor with 262144K bytes of memory.
R5000 processor, Implementation 35, Revision 2.1 (512KB Level 2 Cache)
Last reset from power-on
3 Ethernet/IEEE 802.3 interface(s)
11 ATM network interface(s)
507K bytes of non-volatile configuration memory.

20480K bytes of Flash PCMCIA card at slot 0 (Sector size 128K).
8192K bytes of Flash PCMCIA card at slot 1 (Sector size 128K).
8192K bytes of Flash internal SIMM (Sector size 256K).
%ENABLING INTERFACES.PLEASE WAIT...
%Secondary CPU has not booted IOS


Press RETURN to get started!
```

Note If an rommon> prompt appears, your switch requires a manual boot to recover. Refer to the *Configuration Fundamentals Configuration Guide* for instructions on manually booting from Flash memory.

Configuring the BOOTP Server

The BOOTP protocol automatically assigns an Ethernet IP address by adding the MAC and IP addresses of the Ethernet port to the BOOTP server configuration file. When the switch boots, it automatically retrieves the IP address from the BOOTP server.

The switch performs a BOOTP request *only* if the current IP address is set to 0.0.0.0. (This is the default for a new switch or a switch that has had its startup-config file cleared using the **erase** command.)

To allow your ATM switch router to retrieve its IP address from a BOOTP server, you must first determine the MAC address of the switch and add that MAC address to the BOOTP configuration file on the BOOTP server. The following steps provide an example of creating a BOOTP server configuration file:

	Command	Purpose
Step 1	—	Installs the BOOTP server code on the workstation, if it is not already installed.
Step 2	—	Determines the MAC address from the label on the chassis.
Step 3	—	Adds an entry in the BOOTP configuration file (usually */usr/etc/bootptab*) for each switch. Press **Return** after each entry to create a blank line between each entry. See the example BOOTP configuration file that follows.
Step 4	Switch# **reload**	Restarts the ATM switch router to automatically request the IP address from the BOOTP server.

Example

The following example BOOTP configuration file shows the added entry:

```
# /etc/bootptab: database for bootp server (/etc/bootpd)
#
# Blank lines and lines beginning with '#' are ignored.
#
# Legend:
#
#       first field -- hostname
#                       (may be full domain name and probably should be)
#
#       hd -- home directory
#       bf -- bootfile
#       cs -- cookie servers
#       ds -- domain name servers
#       gw -- gateways
#       ha -- hardware address
#       ht -- hardware type
#       im -- impress servers
#       ip -- host IP address
#       lg -- log servers
#       lp -- LPR servers
#       ns -- IEN-116 name servers
#       rl -- resource location protocol servers
#       sm -- subnet mask
#       tc -- template host (points to similar host entry)
#       to -- time offset (seconds)
#       ts -- time servers
#
<information deleted>
#
#########################################################################
# Start of individual host entries
#########################################################################
➔ Switch:         tc=netcisco0:   ha=0000.0ca7.ce00:      ip=172.31.7.97:
dross:          tc=netcisco0:   ha=00000c000139:        ip=172.31.7.26:

<information deleted>
```

Configuring the ATM Address

The ATM switch router ships with a preconfigured ATM address. The Integrated Local Management Interface (ILMI) protocol uses the first 13 bytes of this address as the switch prefix that it registers with end systems. Autoconfiguration also allows the ATM switch router to establish itself as a node in a single-level Private Network-Network Interface (PNNI) routing domain.

Note If you chose to manually change any ATM address, it is important to maintain the uniqueness of the address across large networks. Refer to the *Guide to ATM Technology* for PNNI address considerations and for information on obtaining registered ATM addresses.

For a description of the autoconfigured ATM address and considerations when assigning a new address, refer to the *Guide to ATM Technology*.

Manually Setting the ATM Address

To configure a new ATM address that replaces the previous ATM address when running IISP software only, see the "Configuring the ATM Address" section on page 10-4.

To configure a new ATM address that replaces the previous ATM address and generates a new PNNI node ID and peer group ID, see the "Configuring an ATM Address and PNNI Node Level" section on page 10-9.

Modifying the Physical Layer Configuration of an ATM Interface

Each of the ATM switch router's physical interfaces has a default configuration, listed in Chapter 17, "Configuring Interfaces." You can accept the defaults, or you can override them by reconfiguring the physical interface.

The following example describes modifying an OC-3c interface from the default settings to the following:

- Disable scrambling cell-payload.
- Disable scrambling STS-streaming.
- Change Synchronous Optical Network (SONET) mode of operation from Synchronous Time Stamp level 3c (STS-3c) mode to Synchronous Transfer Module level 1 (STM-1).

To change the configuration of the example interface, perform the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **interface atm** *card*/*subcard*/*port* Switch(config-if)#	Selects the physical interface to be configured.
Step 2	Switch(config-if)# **no scrambling cell-payload**	Disables cell-payload scrambling.
Step 3	Switch(config-if)# **no scrambling sts-stream**	Disables STS-stream scrambling.
Step 4	Switch(config-if)# **sonet stm-1**	Configures SONET mode as SDH/STM-1.

Example

The following example shows how to disable cell-payload scrambling and STS-stream scrambling and changes the SONET mode of operation to Synchronous Digital Hierarchy/Synchronous Transfer Module 1 (SDH/STM-1) of OC-3c physical interface ATM 0/0/0:

```
Switch(config)# interface atm 0/0/0
Switch(config-if)# no scrambling cell-payload
Switch(config-if)# no scrambling sts-stream
Switch(config-if)# sonet stm-1
```

To change any of the other physical interface default configurations, refer to the commands in the *ATM Switch Router Command Reference* publication.

To display the physical interface configuration, use the following privileged EXEC commands:

Command	Purpose
show controllers atm *card/subcard/port*	Shows the physical layer configuration.
more system:running-config	Shows the physical layer scrambling configuration.

Examples

The following example demonstrates using the **show controllers** command to display the OC-3c physical interface configuration after modification of the defaults:

```
Switch# show controllers atm 0/0/0
IF Name: ATM0/0/0    Chip Base Address: A8808000
Port type: 155UTP    Port rate: 155 Mbps    Port medium: UTP
Port status:SECTION LOS    Loopback:None    Flags:8300
TX Led: Traffic Pattern    RX Led: Traffic Pattern  TX clock source:  network-derived
Framing mode:  stm-1
Cell payload scrambling off
Sts-stream scrambling off

<information deleted>
```

The following example displays the OC-3c physical layer scrambling configuration after modification of the defaults using the **more system:running-config** command:

```
Switch# more system:running-config
!
version XX.X
<information deleted>
!
interface ATM0/0/0
 no keepalive
 atm manual-well-known-vc
 atm access-group tod1 in
 atm pvc 0 35 rx-cttr 3 tx-cttr 3  interface  ATM0 0 any-vci  encap qsaal
 sonet stm-1
 no scrambling sts-stream
 no scrambling cell-payload
!
<information deleted>
```

Configuring the IP Interface

IP addresses can be configured on the multiservice route processor interfaces. Each IP address is configured for one of the following types of connections:

- Ethernet port—Can be configured either from the BOOTP server or by using the **ip address** command in interface configuration mode.
- Classical IP over ATM—See Chapter 12, “Configuring IP over ATM.”
- LANE client—See Chapter 13, “Configuring LAN Emulation.”
- Serial Line Internet Protocol/Point-to-Point Protocol (SLIP/PPP)—Refer to the *Dial Solutions Configuration Guide*.

Note These IP connections are used only for network management.

To configure the switch to communicate via the Ethernet interface, provide the IP address and subnet mask bits for the interface.

This section includes the following:

- Configuring IP Address and Subnet Mask Bits, page 3-8
- Testing the Ethernet Connection, page 3-9

Configuring IP Address and Subnet Mask Bits

Define subnet mask bits as a decimal number between 0 and 22 for Class A addresses, between 0 and 14 for Class B addresses, or between 0 and 6 for Class C addresses. Do not specify 1 as the number of bits for the subnet field. That specification is reserved by Internet conventions.

To configure the IP address, perform the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **interface ethernet 0** Switch(config-if)#	Selects the interface to be configured.
Step 2	Switch(config-if)# **ip address** *ip-address mask*	Configures the IP and subnetwork address.

Note Since release 12.0(1a)W5(5b) of the ATM switch software, addressing the interface on the processor (CPU) has changed. The ATM interface is now called atm 0, and the Ethernet interface is now called ethernet 0. The old formats (atm 2/0/0 and ethernet 2/0/0) are still supported.

Example

The following example shows how to configure interface ethernet 0 with IP address 172.20.40.93 and subnetwork mask 255.255.255.0:

```
Switch(config)# interface ethernet 0
Switch(config-if)# ip address 172.20.40.93 255.255.255.0
```

Displaying the IP Address

To display the IP address configuration, use the following privileged EXEC commands:

Command	Purpose
show interfaces ethernet 0	Displays the Ethernet interface IP address.
more system:running-config	Shows the physical layer scrambling configuration.

Examples

The following example shows how to use the **show interfaces** command to display the IP address of interface ethernet 0:

```
Switch# show interfaces ethernet 0
Ethernet0 is up, line protocol is up
  Hardware is SonicT, address is 0040.0b0a.1080 (bia 0040.0b0a.1080)
  Internet address is 172.20.40.93/24
  <information deleted>
```

The following example uses the **more system:running-config** command to display the IP address of interface ethernet 0:

```
Switch# more system:running-config
!
version XX.X
<information deleted>
!
interface Ethernet0
 ip address 172.20.40.93 255.255.255.0
!
<information deleted>
```

Testing the Ethernet Connection

After you have configured the IP address(es) for the Ethernet interface, test for connectivity between the switch and a host. The host can reside anywhere in your network. To test for Ethernet connectivity, use the following EXEC command:

Command	Purpose
ping ip *ip-address*	Tests the configuration using the ping command. The ping command sends an echo request to the host specified in the command line.

The following example show how to test the Ethernet connectivity from the switch to a workstation with an IP address of 172.20.40.201:

```
Switch# ping ip 172.20.40.201

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.20.40.201, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/202/1000 ms
```

Configuring Network Clocking

This section describes network clocking configuration of the ATM switch router. Properly synchronized network clocking is important in the transmission of constant bit rate (CBR) and variable bit rate real time (VBR-RT) data. For an overview of network clocking and network clock configuration issues, refer to the chapter “Network Clock Synchronization” in the *Guide to ATM Technology*.

This section includes the following:

Network Clocking Features

Different types of network clock sources are available on the ATM switch router, both internal and external. Table 3-1 provides a summary of network clocking features.

Table 3-1 Network Clocking Feature Summary

Platform	Up/Down Detection	Loss of Synchronization Detection	Phase Adjustment Cutover	Stratum 3 Clock	BITS[1] Port	Clock Source Preference
Catalyst 8540 MSR with network clock module	Yes	Yes	Yes	Yes	Yes	Best
Catalyst 8510 MSR	Yes	Yes	Yes	No	No	Medium
LightStream 1010 with FC-PFQ	Yes	Yes	Yes	No	No	Medium
Catalyst 8540 MSR without network clock module	Yes	No	No	No	No	Poor
LightStream 1010 without FC-PFQ	Yes	No	No	No	No	Poor

1. BITS = Building Integrated Timing Supply

Configuring Network Clock Sources and Priorities (Catalyst 8540 MSR)

To configure the network clocking priorities and sources, use the following command in global configuration mode:

Command	Purpose
network-clock-select {*priority* {{**atm** | **cbr**} *card*/*subcard*/*port*} | **bits** {**0** | **1**} | **system**} | **bits** {**e1** | **t1**} | **revertive**	Configures the network clock priority.

Note Specifying the keyword **system** with the **network-clock-select command** selects the route processor reference clock (a stratum 4 clock source) or the network clock module (a stratum 3 clock source), if present.

Systems equipped with the network clock module can derive clocking from a Building Integrated Timing Supply (BITS) source. To specify the line type attached to the BITS ports on the network clock module and to assign a priority to a port, use the following commands in global configuration mode:

Command	Purpose
network-clock-select bits {**t1** | **e1**}	Selects the line type. This command applies to both BITS ports.
network-clock-select *priority* **bits** {**0** | **1**}	Selects the priority for a BITS port.

Examples

The following example shows how to configure the network clock priorities:

```
Switch(config)# network-clock-select 1 atm 0/0/0
Switch(config)# network-clock-select 2 atm 0/0/3
```

Note This configuration assumes that a full-width module, such as the 4-port OC-12c module, is being used to derive clocking. If port adapters inserted into carrier modules are used, the priority 1 and 2 source ports must be on different port adapters.

The following example shows how to configure the network clock to revert to the highest priority clock source after a failure and takeover by the source with the next lowest priority.

```
Switch(config)# network-clock-select revertive
```

Configuring Network Clock Sources and Priorities (Catalyst 8510 MSR and LightStream 1010)

To configure the network clocking priorities and sources, use the following command in global configuration mode:

Command	Purpose
network-clock-select {*priority* {{**atm** | **cbr**} *card*/*subcard*/*port*} | **system**} | **revertive**	Configures the network clock priority.

Note Specifying the keyword **system** with the **network-clock-select command** selects the processor card reference clock (a stratum 4 clock source).

Examples

The following example shows how to configure the network clock priorities:

```
Switch(config)# network-clock-select 1 atm 0/0/0
Switch(config)# network-clock-select 2 atm 0/0/3
```

The following example shows how to configure the network clock to revert to the highest priority clock source after a failure and takeover by the source with the next lowest priority.

```
Switch(config)# network-clock-select revertive
```

Configuring the Transmit Clocking Source

To configure where each interface receives its transmit clocking, perform the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **interface atm** *card*/*subcard*/*port* Switch(config-if)#	Selects the interface to be configured.
Step 2	Switch(config-if)# **clock source** {**free-running** | **loop-timed** | **network-derived**}	Configures the interface clock source.

Caution If the Network Clock Distribution Protocol (NCDP) is running on an interface, you should not override that port's clock source by configuring it to free-running or loop-timed. Doing so could cause synchronization problems, particularly in the case of loop-timed, which could cause a clocking loop to be formed on a link. See the "Configuring Network Clocking with NCDP" section on page 3-14.

Example

The following example configures ATM interface 3/0/0 to receive its transmit clocking from a network-derived source:

```
Switch(config)# interface atm 3/0/0
Switch(config-if)#clock source network-derived
```

Displaying the Network Clocking Configuration

To show the switch's network clocking configuration, use the following privileged EXEC commands:

Command	Purpose
show network-clocks	Shows the network clocking configuration.
more system:running-config	Shows the interface clock source configuration.
show controllers [**atm** *card*/*subcard*/*port*]	Shows the interface controller status.

Examples

The following example shows the configured network clock sources on a Catalyst 8510 MSR or LightStream 1010:

```
Switch# show network-clocks
clock configuration is NON-Revertive
Priority 1 clock source: ATM1/0/0
Priority 2 clock source: ATM1/1/0
Priority 3 clock source: No clock
Priority 4 clock source: No clock
Priority 5 clock source: System clock
Current clock source:System clock, priority:5
```

Note A source listed as "No clock" indicates that no clock source configured at that priority.

The following example shows the switch clock source configuration with the network clock module installed:

```
Switch# show network-clocks
Network clocking information:
---------------------------------------
Source switchover mode:    revertive
Netclkd state:             Active
Source selection method:   provisioned
NCLKM hardware status:     installed & usable
NCLKM status:              software enabled
Primary   clock source:    ATM0/0/0
Secondary clock source:    not configured
Present   clock source:    NCLKM Stratum 3 osc (0)
```

The following example shows the clock source configuration stored in the running configuration:

```
Switch# more system:running-config
!
<information deleted>
!
network-clock-select revertive
network-clock-select 1 ATM0/0/0
<information deleted>
```

Configuring Network Clocking with NCDP

The Network Clock Distribution Protocol (NCDP) provides a means by which a network can synchronize automatically to a primary reference source (PRS). To do so, NCDP constructs and maintains a spanning network clock distribution tree. This tree structure is superimposed on the network nodes by the software, resulting in an efficient, synchronized network suitable for transport of traffic with inherent synchronization requirements, such as voice and video.

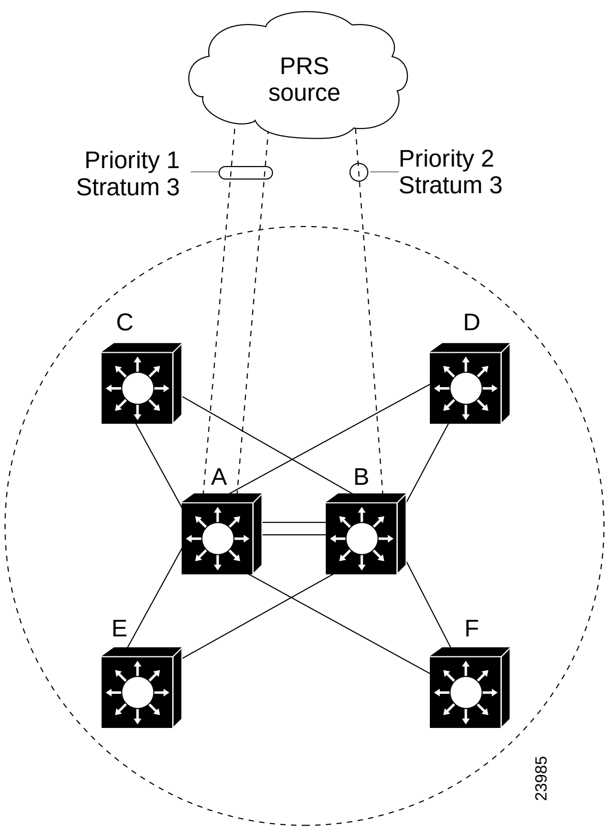
The following sections provide instructions for configuring NCDP. For a description of how NCDP works, refer to the *Guide to ATM Technology*.

Note The NCDP is intended for use on ATM switch routers equipped with FC-PFQ or with the network clock module.

NCDP Network Example

Figure 3-1 shows a network of six ATM switch routers with clocking derived from a stratum 3 PRS. Node A is configured to receive priority 1 clocking on two of its ports, while node B is configured to receive priority 2 clocking on one of its ports.

Figure 3-1 Network Configuration for NCDP



Enabling NCDP

To enable NCDP, use the following global configuration command for each node that you want to configure for NCDP:

Command	Purpose
ncdp	Enables NCDP.

Configuring Network Clock Sources and Priorities

You must specify the clocking sources, their priorities, and associated stratums used by NCDP in constructing the clock distribution tree. To do so, use the following command in global configuration mode:

Command	Purpose
ncdp source *priority* {{**atm** | **cbr**} *card/subcard/port stratum* | **bits**[1] {**0** | **1**} *stratum* | **system**}	Specifies a priority and source (stratum level or system) for this interface.

1. Allows you to specify a Building Integrated Timing Supply (BITS) source. This option is available only on the Catalyst 8540 MSR equipped with the network clock module.

If you do not configure a clock source, NCDP advertises its default source of network clock, which is its local oscillator; if no nodes in the network have a clock source configured, the tree is built so that it is rooted at the switch having the highest stratum oscillator (lowest numerical value) and lowest ATM address.

Example

The following example demonstrates configuring the network clock source, priority, and stratum on node A in Figure 3-1.

```
Switch(config)# ncdp source 1 atm 1/0/0 3
Switch(config)# ncdp source 1 atm 3/0/0 3
```

Configuring Optional NCDP Global Parameters

Optional NCDP parameters you can configure at the global level include the maximum number of hops between any two nodes, revertive behavior, and the values of the NCDP timers. To change any of these parameters from their defaults, use the following commands in global configuration mode:

Command	Purpose
ncdp max-diameter *hops*	Specifies the maximum network diameter for the protocol. The default maximum network diameter is 20.
ncdp revertive	Specifies the NCDP as revertive.
ncdp timers {**hello** | **hold**} *time-in-msec jitter-percent*	Specifies the values to be used by the NCDP timers.

When you specify a maximum diameter, you constrain the diameter of the spanning tree by specifying the maximum number of hops between any two nodes that participate in the protocol. Each node must be configured with the same maximum network diameter value for NCDP to operate correctly.

When you configure the NCDP as revertive, a clock source that is selected and then fails is selected again once it has become operational for a period of time. On the Catalyst 8510 MSR and LightStream 1010 platforms, if NCDP is configured to be revertive, a failed clocking source node after a switchover is restored to use after it has been functioning correctly for at least 1 minute. On the Catalyst 8540 MSR the failed source is restored after about 25 seconds. The network clock is, by default, configured as nonrevertive. Nonrevertive prevents a failed source from being selected again.

Example

The following example shows setting the maximum number of hops to 11 and enabling revertive behavior:

```
Switch(config)# ncdp max-diameter 11
Switch(config)# ncdp revertive
```

Configuring Optional NCDP Per-Interface Parameters

On a per-interface basis, you can enable or disable NCDP, specify the cost metric associated with the port, and change the control virtual circuit used to transport protocol messages between adjacent protocol entities. To change any of these parameters from their defaults, use the following commands in interface configuration mode:

Command	Purpose
ncdp admin-weight *weight*	Specifies the cost metric associated with the given port.
ncdp control-vc *vpi vci*	Specifies the VPI/VCI values to use for control VCs on the physical interface. The default is 0, 34. **Note** To change the control VC to a VPI other than 0, the VPI must exist on the physical interface.
no ncdp	Disables NCDP on the interface.

Example

The following example demonstrates setting the administrative weight on an interface:

```
Switch(config)# interface atm 0/0/0
Switch(config-if)# ncdp admin-weight 75
```

Displaying the NCDP Configuration

To display the NCDP configuration, use the following EXEC commands:

Command	Purpose
show ncdp path root	Displays the NCDP clock path from the switch to the root source.
show ncdp ports	Displays NCDP port information.
show ncdp sources	Displays NCDP clock sources configured on the switch.
show ncdp status	Displays NCDP status.
show ncdp timers	Displays NCDP timer information.

Example

The following example shows the NCDP status:

```
Switch# show ncdp status
 = ncdp switch information ==== enabled ==============
 non-revertive
 root clock source priority:      1
 root clock source stratum level: 4
 root clock source prs id:        255
 stratum level of root switch:    4
 clocking root address:           4700918100000000E0F75D040100E0F75D040100
 hop count:                       0
 root path cost:                  0
 root port:                       0
 max age:                         5
 hello time:                      500
 priority      of best source:    1
 stratum level of best source:    4
 prs id        of best source:    255
 switch stratum level:            4
 address:                         4700918100000000E0F75D040100E0F75D040100
 switch max age:                  5
 switch hello time:               500
 switch hold time:                500
 max diameter:                    5
 converged root count:            359375
 converged:                       1
 total timer events:              687271
 total queue events:              0
 rx config messages:              0
 tx config messages:              363716
 rx tcn messages:                 0
 tx tcn messages:                 0
 rx non-participant messages:     0
 rx unknown messages:             0

Switch#
```

Network Clock Services for CES Operations and CBR Traffic

Circuit emulation services-interworking functions (CES-IWF) and constant bit rate (CBR) traffic relate to a quality of service (QoS) classification defined by the ATM Forum for Class A (ATM adaptation layer 1 [AAL1]) traffic in ATM networks. In general, Class A traffic pertains to voice and video transmissions, which have particular clocking requirements. For details, refer to Chapter 18, "Configuring Circuit Emulation Services."

Configuring Network Routing

The default software image for the ATM switch router contains the Private Network-Network Interface (PNNI) routing protocol. The PNNI protocol provides the route dissemination mechanism for complete plug-and-play capability. The following section, "Configuring ATM Static Routes for IISP or PNNI," describes modifications that can be made to the default PNNI or Interim-Interswitch Signalling Protocol (IISP) routing configurations.

For routing protocol configuration information, refer to Chapter 9, "Configuring ILMI,"and Chapter 10, "Configuring ATM Routing and PNNI."

Configuring ATM Static Routes for IISP or PNNI

Static route configuration allows ATM call setup requests to be forwarded on a specific interface if the addresses match a configured address prefix. To configure a static route, use the following command in global configuration mode:

Command	Purpose
atm route *addr-prefx* **atm** *card/subcard/port*	Specifies a static route to a reachable address prefix.

Note An interface must be User-Network Interface (UNI) or Interim Interswitch Signalling Protocol (IISP) to be configured with static route. Static routes configured as PNNI interfaces default as down.

The following example shows how to use the **atm route** command to configure the 13-byte peer group prefix = 47.0091.8100.567.0000.0ca7.ce01 at interface ATM 3/0/0:

```
Switch(config)# atm route 47.0091.8100.567.0000.0ca7.ce01 atm 3/0/0
Switch(config)#
```

Configuring System Information

Although not required, the system clock and hostname should be set as part of the initial system configuration. To set these system parameters, perform the following steps, beginning in privileged EXEC mode:

	Command	Purpose
Step 1	Switch# **clock set** *hh:mm:ss day month year*	Sets the system clock.
Step 2	Switch# **configure terminal** Switch(config)#	Enters global configuration mode from the terminal.
Step 3	Switch(config)# **hostname** *name*	Sets the system name.

Examples

The following example shows how to configure the time, date, and month using the **clock set** command, enter global configuration mode, and assign a hostname.

```
Switch# clock set 15:01:00 17 October 1999
Switch# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)# hostname Publications
Publications#
```

The following example shows how to confirm the clock setting using the **show clock** command:

```
Publications# show clock
*15:03:12.015 UTC Fri Oct 17 1999
```

Configuring Online Diagnostics (Catalyst 8540 MSR)

Online and insertion diagnostics detect and report hardware failures in the Catalyst 8540 MSR during system bootup and operation.

The online diagnostics on the Catalyst 8540 MSR provide the following types of tests:

- Access tests between the route processor and the switch processors, feature cards, port adapters, and interface modules
- Online insertion and removal (OIR) diagnostic tests
- Snake tests through the switch router to ensure connectivity between the ports

Note Online diagnostics tests only run on the primary route processor.

Access Test (Catalyst 8540 MSR)

The access tests ensure connectivity at a configurable interval between the primary route processor and the following:

- Active switch processors
- Standby switch processor, if it is present

- Feature cards
- Carrier modules
- ATM port adapters
- ATM and Layer 3 interface modules
- ATM router modules

When the access test detects a hardware failure, the system issues an error message to the console.

If the access test detects a hardware problem with an active switch processor, the standby switch processor, if it is present, automatically takes over and becomes an active switch processor. The system generates an SNMP trap when the switchover occurs.

Note The access test does not support the network clock module.

OIR Test (Catalyst 8540 MSR)

Online insertion and removal (OIR) tests check the functioning of the switch fabric and interfaces on a per-port basis. The switch router performs these tests when the system boots up and when you insert a port adapter or interface module into a slot. The OIR test sends a packet to the interface loopback and expects to receive it back within a certain time period. If the packet does not reach the port within the expected time period, or the route processor receives a corrupted packet, the system issues an error message to the console, generates an SNMP trap, and brings the port to an administrative down state.

Note The size of the packet used in the test is configurable.

The OIR tests support all ATM port adapters, all ATM interface modules, all ATM router modules, and all Layer 3 interface modules except the 8-port Gigabit Ethernet.

Snake Test (Catalyst 8540 MSR)

The snake test detects and reports port-to-port connectivity failures. The snake test establishes a connection across all the active ports in the switch router, originating and terminating at the primary route processor. The route processor establishes a connection by sending a packet to each port in turn, which then terminates at the route processor. If the packet does not reach the route processor within the expected time period, or the received packet is corrupted, further testing is performed to isolate and disable the port causing the problem.The size of the packet and frequency of the test are configurable to minimize the impact on system performance.

The snake test supports all ATM interface modules and enhanced Gigabit Ethernet interface modules. It does not support ATM port adapters, ATM router modules, 16-port 10/100 Fast Ethernet interface modules, 2-port Gigabit Ethernet interface modules, or 8-port Gigabit Ethernet interface modules.

Note The snake test does not support ATM port adapters because of a hardware limitation in the carrier module.

Configuring Online Diagnostics (Catalyst 8540 MSR)

To configure online diagnostics, use the following global configuration commands:

Command	Purpose
diag online	Enables all of the online diagnostic tests.
diag online access	Enables only the access diagnostic test.
diag online access freq [*seconds*]	Configures the frequency of the access diagnostic tests. The default frequency is every 10 seconds.
diag online oir	Enables only the OIR test.
diag online oir pktsize [*bytes*]	Specifies the packet size for the OIR test. The default size is 1000 bytes.
diag online snake	Enables only the snake test.
diag online snake timer [*seconds*]	Specifies the time interval for the snake test. The default interval is 60 seconds.
no diag online [**access** | **oir** | **snake**]	Disables the online diagnostic tests.
debug diag online [**access** | **oir** | **snake**]	Enables debugging of online diagnostic tests.
no debug diag online [**access** | **oir** | **snake**]	Disables debugging of online diagnostic tests.

Examples

The following example shows how to enable all online diagnostic tests:

```
Switch(config)# diag online
 ONLINE-DIAG: Enabling all Online Diagnostics tests
```

The following example shows how to change the frequency of the access test to 20 seconds:

```
Switch(config)# diag online access freq 20
ONLINE-DIAG: Online Access Test Frequency set to 20 sec
```

Displaying the Online Diagnostics Configuration and Results (Catalyst 8540 MSR)

To display the online diagnostics configuration and results, use the following EXEC command:

Command	Purpose
show diag online [**details** | **status**] [**access** | **oir** | **snake**]	Displays information about the online diagnostics test configuration and the test results.

Examples

The following example shows how to display detailed access test configuration and results:

```
Switch# show diag online details access
======== Online Access Test Details ========
Current Test Status : Test is Enabled
Current Frequency of Access Test : 20 seconds

Slot Card-Type       Iteration    Success    Failure    Last Failure
---- ----------      ----------   -------    -------    ------------
 0/* Super Cam       42998        42998      0          ----
 0/0 8T1 IMA PAM     42998        42998      0          ----
 0/1 8E1 IMA PAM     42998        42998      0          ----
 2/* ARM PAM         42998        42998      0          ----
 3/* ETHERNET PAM    42998        42998      0          ----
 5/* Switch Card     42998        42998      0          ----
 5/0 Feature Card    42998        42998      0          ----
 7/* Switch Card     42998        42998      0          ----
 7/0 Feature Card    42998        42998      0          ----
 9/* OC48c PAM       42998        42998      0          ----
10/* OCM Board       42998        42998      0          ----
10/0 QUAD 622 Generi 42998        42998      0          ----
======== Online Access Test Details End ========
```

The following example shows how to display the status of the OIR test:

```
Switch# show diag online status oir
======== Online OIR Test Status ========
Current Test Status : Test is Enabled
-------- Bootup OIR status --------
Port    Card Type   Pkt Size  Result                    Test Time LOOP
_______ ___________ _________ ___________________  ______________ ____
00/0/00 8T1 IMA PAM       300 OIR_SUCCESS                00:00:41  PIF
00/0/01 8T1 IMA PAM       300 OIR_SUCCESS                00:00:41  PIF
00/0/02 8T1 IMA PAM       300 OIR_SUCCESS                00:00:41  PIF
00/0/03 8T1 IMA PAM       300 OIR_SUCCESS                00:00:41  PIF
00/1/00 8E1 IMA PAM       300 OIR_SUCCESS                00:00:41  PIF
00/1/01 8E1 IMA PAM       300 OIR_SUCCESS                00:00:46  PIF
00/1/02 8E1 IMA PAM       300 OIR_SUCCESS                00:00:41  PIF
00/1/03 8E1 IMA PAM       300 OIR_SUCCESS                00:00:46  PIF

03/0/00 ETHERNET PA      1000 OIR_SUCCESS                00:01:54  PIF
03/0/01 ETHERNET PA      1000 OIR_SUCCESS                00:01:52  PIF
03/0/02 ETHERNET PA      1000 OIR_SUCCESS                00:01:50  PIF
03/0/03 ETHERNET PA      1000 OIR_SUCCESS                00:01:48  PIF
03/0/04 ETHERNET PA      1000 OIR_SUCCESS                00:01:55  PIF
03/0/05 ETHERNET PA      1000 OIR_SUCCESS                00:01:53  PIF
03/0/06 ETHERNET PA      1000 OIR_SUCCESS                00:01:51  PIF
03/0/07 ETHERNET PA      1000 OIR_SUCCESS                00:01:49  PIF
03/0/08 ETHERNET PA      1000 OIR_SUCCESS                00:02:02  PIF
03/0/09 ETHERNET PA      1000 OIR_SUCCESS                00:02:00  PIF
03/0/10 ETHERNET PA      1000 OIR_SUCCESS                00:01:58  PIF
03/0/11 ETHERNET PA      1000 OIR_SUCCESS                00:01:56  PIF
03/0/12 ETHERNET PA      1000 OIR_SUCCESS                00:02:03  PIF
03/0/13 ETHERNET PA      1000 OIR_SUCCESS                00:02:01  PIF
03/0/14 ETHERNET PA      1000 OIR_SUCCESS                00:01:59  PIF
03/0/15 ETHERNET PA      1000 OIR_SUCCESS                00:01:57  PIF

09/0/00 OC48c PAM         300 OIR_SUCCESS                00:00:46 Both

10/0/00 QUAD 622 Ge       300 OIR_SUCCESS                00:00:46 Both
10/0/01 QUAD 622 Ge       300 OIR_SUCCESS                00:00:46 Both
10/0/02 QUAD 622 Ge       300 OIR_SUCCESS                00:00:46 Both
10/0/03 QUAD 622 Ge       300 OIR_SUCCESS                00:00:46 Both
```

The following example shows how to display the details and status of the snake test:

```
8540MSR#show diag online snake
========  Online Snake Test Status and Details ========
--------  Test Status --------
Current Test Status    : Test is Enabled
Current Test Type      : Normal Snake
Last Test Status       : Pass
Last Test Run Time     : 1w1d
Last Test Success Time : 1w1d

--------  Test Details --------
Snake Test Pkt Size    : 30 bytes
Default Test Period    : 60 seconds
Current Test Period    : 60 seconds

----------------------------------
      Statistics from Bootup
----------------------------------
Total Test Runs                       : 17311
Number Normal Snake Test Runs         : 17311
Number of Successive Normal Snake Test : 14083
Number of Incrimental Snake Test Runs  : 0

------------------------------------------
      Ports Test Stat in Last Iteration
------------------------------------------

Port      Card Type          Result         Test Time
_______   ________________   __________     _________
09/0/00   OC48c PAM          PORT_OK        1w1d
10/0/00   QUAD 622 Generic   PORT_OK        1w1d
11/0/00   OC48c PAM          PORT_OK        1w1d
12/0/00   QUAD 622 Generic   PORT_OK        1w1d

-----------------------------------------
      Ports Failed Stat from Bootup
-----------------------------------------
     No Port failed from Bootup
```

Configuring Redundancy and Enhanced High System Availability (Catalyst 8540 MSR)

The Catalyst 8540 MSR supports redundant CPU operation with dual route processors. In addition, Enhanced High System Availability (EHSA) is provided in the switching fabric when three switch processors are installed in the chassis. These features and their configuration are described in the following sections:

Route Processor Redundant Operation (Catalyst 8540 MSR)

The Catalyst 8540 MSR supports fault tolerance by allowing a secondary route processor to take over if the primary fails. This secondary, or redundant, route processor runs in standby mode. In standby mode, the secondary route processor is partially booted with the Cisco IOS software; however, no configuration is loaded.

At the time of a switchover, the secondary route processor takes over as primary and loads the configuration as follows:

- If the running configuration between the primary and secondary route processors match, the new primary uses the running configuration file
- If the running configuration between the primary and secondary route processors do not match, the new primary uses the last saved configuration file in its nonvolatile random-access memory (NVRAM) (not the NVRAM of the former primary)

The former primary then becomes the secondary route processor.

Note If the secondary route processor is unavailable, a major alarm is reported. Use the **show facility-alarm status** command to display the redundancy alarm status.

When the Catalyst 8540 MSR is powered on, the two route processors go through an arbitration to determine which is the primary route processor and which is the secondary. The following rules apply during arbitration:

- A newly inserted route processor card always comes up as the secondary, except in cases where the newly inserted card is the only one present.
- If the configuration is corrupted, one of the route processors comes up as primary, allowing you to correct the situation manually.
- The primary route processor at the time the Catalyst 8540 MSR is powered off continues as the primary when the Catalyst 8540 MSR is powered on.
- If none of the above conditions is true, the route processor in slot 4 becomes the primary.

During normal operation, the primary route processor is booted completely. The secondary CPU is partially up, meaning it stops short of parsing the configuration. From this point, the primary and secondary processors communicate periodically to synchronize any system configuration changes.

The following situations can cause a switchover of the primary route processor:

- The primary route processor is removed or swapped. When a route processor functioning as primary is removed, the secondary takes over. The Catalyst 8540 MSR is now nonredundant until a second route processor is inserted.
- The primary route processor is rebooted. When a route processor functioning as primary is rebooted, the secondary takes over.
- The primary route processor fails. The secondary route processor takes over as primary, using the last saved configuration (or the current running configuration if they have been synchronized with the **sync config** command).
- A switchover is manually forced with the **redundancy force-failover main-cpu** command.

When a switchover occurs, permanent virtual connections (PVCs) are preserved.

Configuring Route Processor Redundancy (Catalyst 8540 MSR)

For redundant operation, the following requirements must be met:

- Two route processors and three switch cards are required.
- The route processors must have identical hardware configurations. This includes variables such as DRAM size, presence or absence of network clock modules, and so on.
- Both route processors must have the same functional image. For more information, see the "Maintaining Functional Images (Catalyst 8540 MSR)" section on page 22-5.
- Both route processors must be running the same system image.
- Both route processors must be set to autoboot (a default setting).

If these requirements are met, the Catalyst 8540 MSR runs in redundant mode by default. The tasks described in the following sections are optional and used only to change nondefault values.

Forcing a Route Processor Switchover (Catalyst 8540 MSR)

You can manually force the secondary route processor to take over as primary. To do so, use the following privileged EXEC command:

Command	Purpose
redundancy force-failover main-cpu	Forces a route processor switchover.

As long as you have not changed the default configuration register setting, which is set to autoboot by default, the secondary route processor (formerly the primary) completes the boot process from standby mode.

If you have changed the default configuration register value, you can change it back to autoboot, and ensure that the correct system image is used at startup, by performing the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **config-register 0x2102**	Sets the config register for autoboot.
Step 2	Switch(config)# **boot system** {[*device*:]*filename* [*hostname* | *ip-address*] | **flash** [*device*:][*filename*] | **mop** *filename* [*type*] [*card*/*subcard*/*port*] | **rcp** *filename* [*ip-address*] | **rom** | **tftp** *filename* [*hostname* | *ip-address*]}	Specifies the system image file to load at startup.
Step 3	Switch(config)# **end** Switch#	Returns to privileged EXEC mode.
Step 4	Switch# **copy system:running-config nvram:startup-config**	Saves the configuration to NVRAM.

Note If the secondary route processor remains in ROM monitor mode, you can manually boot the processor from either the bootflash or Flash PC card.

Caution If no system image is specified in the startup configuration, the ROM monitor automatically boots the first system image on the Flash PC card in slot0. If there is no system image on the Flash PC card, or the Flash PC card is not available, the ROM monitor boots the first system image in bootflash. If there is no system image in bootflash, the switch remains in ROM monitor mode.

Displaying the Configuration Register Value

To display the configuration register value, use the following privileged EXEC command:

Command	Purpose
show version	Displays the configuration register value.

The following example shows the configuration register value:

```
Switch# show version
Cisco Internetwork Operating System Software
IOS (tm) PNNI Software (cat8540m-WP-M), Version XX.X(X)WX(X),  RELEASE SOFTWARE
Copyright (c) 1986-19XX by cisco Systems, Inc.
Compiled Mon XX-XXX-XX 10:15 by integ
Image text-base: 0x60010930, data-base: 0x606CE000

ROM: System Bootstrap, Version XX.XXX.X(X)WX(X) [BLD-JAGUAR120-4.0.9 ], E

Switch uptime is 3 weeks, 5 days, 23 hours, 30 minutes
System restarted by bus error at PC 0x6007EF24, address 0xFC
System image file is "bootflash:cat8540m-wp-mz.XXX-X.X.WX.X.XX"

cisco C8540MSR (R5000) processor with 65536K/256K bytes of memory.
R5000 processor, Implementation 35, Revision X.X (512KB Level 2 Cache)
Last reset from power-on
1 Ethernet/IEEE 802.3 interface(s)
9 ATM network interface(s)
507K bytes of non-volatile configuration memory.

8192K bytes of Flash PCMCIA card at slot 0 (Sector size 128K).
8192K bytes of Flash internal SIMM (Sector size 256K).
Secondary is up
Secondary has 0K bytes of memory.

Configuration register is 0x100 (will be 0x2102 at next reload)
```

Synchronizing the Configurations (Catalyst 8540 MSR)

During normal operation, the startup and running configurations are synchronized by default between the two route processors. In the event of a switchover, the new primary route processor uses the current configuration. Configurations synchronize either immediately from the command line or during route processor switchover.

Immediately Synchronizing Route Processor Configurations (Catalyst 8540 MSR)

To immediately synchronize the configurations used by the two route processors, use the following privileged EXEC command on the primary route processor:

Command	Purpose
redundancy manual-sync {**startup-config** | **running-config** | **both**}	Immediately synchronizes the configuration.

Example

In the following example, both the startup and running configurations are synchronized immediately:

```
Switch# redundancy manual-sync both
```

Synchronizing the Configurations During Switchover (Catalyst 8540 MSR)

To synchronize the configurations used by the two route processors during a switchover, perform the following steps on the primary route processor, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **redundancy** Switch(config-r)#	Enters redundancy configuration mode.
Step 2	Switch(config-r)# **main-cpu** Switch(config-r-mc)#	Enters main CPU configuration submode.
Step 3	Switch(config-r-mc)# **sync config** {**startup** | **running** | **both**}[1]	Synchronizes either or both configurations during switchover or writing the files to NVRAM.
Step 4	Switch(config-r-mc)# **end** Switch#	Returns to privileged EXEC mode.
Step 5	Switch# **copy system:running-config nvram:startup-config**	Forces a manual synchronization of the configuration files in NVRAM. **Note** This step is unnecessary to synchronize the running configuration file in DRAM.

1. Alternatively, you can force an immediate synchronization by entering the **redundancy manual-sync** command in privileged EXEC mode.

Example

In the following example, both the startup and running configurations are synchronized:

```
Switch(config)# redundancy
Switch(config-r)# main-cpu
Switch(config-r-mc)# sync config both
Switch(config-r-mc)# end
Switch# copy system:running-config nvram:startup-config
```

Displaying the Route Processor Redundancy Configuration (Catalyst 8540 MSR)

To display the route processor redundancy configuration, use the following privileged EXEC command:

Command	Purpose
show redundancy	Displays the redundancy configuration.

In the following example shows the route processor redundancy configuration:

```
Switch# show redundancy

This CPU is the PRIMARY
Primary
-------
Slot:                          4
Uptime:                        1 day, 18 hours, 40 minutes
Image:                         PNNI Software (cat8540m-WP-M), Version 12.0(4a)W5(10.44)

Time Since :
  Last Running Config. Sync:   3 hours, 13 minutes
  Last Startup Config. Sync:   Never
Last Restart Reason:           Normal Boot

Secondary
---------
State:                         UP
Slot:                          8
Uptime:                        3 hours, 16 minutes
Image:                         PNNI Software (cat8540m-WP-M), Version 12.0(4a)W5(10.46)
```

Preparing a Route Processor for Removal (Catalyst 8540 MSR)

Before removing a route processor that is running the IOS in secondary mode, it is necessary to change it to ROM monitor mode. You could use the reload command to force the route processor to ROM monitor mode but the automatic reboot would occur and you would interrupt switch traffic.

Caution If you fail to prepare the secondary route processor for removal, the traffic through the switch could be interrupted.

To change the secondary route processor to ROM monitor mode and eliminate the automatic reboot prior to removal, perform the following steps, beginning in privileged EXEC mode:

	Command	Purpose
Step 1	Switch# **copy system:running-config nvram:startup-config**	Forces a manual synchronization of the configuration files in NVRAM.
Step 2	Switch)# **redundancy prepare-for-cpu-removal**	Changes the current route processor to ROM monitor mode prior to removal.

Example

The following example shows how to change the current route processor to ROM monitor mode prior to removal:

```
Switch# copy system:running-config nvram:startup-config
Destination filename [startup-config]?
Building configuration...

EHSA:Syncing monvars to secondary, : BOOT=
EHSA:Syncing monvars to secondary, : CONFIG_FILE=
EHSA:Syncing monvars to secondary, : BOOTLDR=[OK]
Switch#
Switch# redundancy prepare-for-cpu-removal
This command will cause this CPU to go to the
rom monitor through a forced crash.
After this cpu goes to the rom monitor prompt, it is
safe to remove it from the chassis
Please DO NOT REBOOT this cpu before removing it
Do you want to remove it?[confirm]y

Queued messages:
1d22h: %SYS-3-LOGGER_FLUSHING: System pausing to ensure console debugging outpu.

*** System received a reserved exception ***
signal= 0x9, code= 0x0, context= 0x61818df8
PC = 0x600b62e0, Cause = 0x20, Status Reg = 0x34008702
AT: be840000, V0: 9, V1: 0
A0: 2b, A1: 9, A2: 0
A3: 61818df8, T0: 30, T1: 34008701
T2: 34008700, T3: ffff00ff, T4: 61059f88
T5: 7f, T6: 0, T7: 0
S0: 34008701, S1: 1, S2: 9
S3: 0, S4: 61818df8, S5: 611f8540
S6: 611e3740, S7: 61363710, T8: 47d1
T9: 618189d8, K0: 61612634, K1: 600b7e30
GP: 61177fa0, SP: 61818da8, S8: 611e3740
RA: 600a81b8
STATUS: 34008702
mdlo_hi: 0, mdlo: 0
mdhi_hi: 0, mdhi: 0
bvaddr_hi: ffffffff, bvaddr_lo: ffffffff
cause: 20, epc_hi: 0, epc:600b62e0
err_epc_hi: 0, err_epc: 200004
TIGER Masked Interrupt Register = 0x0000007f
TIGER Interrupt Value Register = 0x00000020

monitor: command "boot" @Ø--<ÒagZç
rommon 3 >
```

Configuring Switch Fabric Enhanced High System Availability Operation (Catalyst 8540 MSR)

Slots 5, 6, and 7 in the Catalyst 8540 MSR chassis can accommodate either two or three switch processor cards, with a switching capacity of 10 Gbps each. The possible configurations are as follows:

- Two switch processors—20 Gbps non-EHSA switching fabric (no spare)
- Three switch processors—20 Gbps EHSA switching fabric (one spare)

When three switch processors are installed, two are active at any time, while the third runs in standby mode. By default, switch processors 5 and 7 are active and switch processor 6 is the standby. To force the standby switch processor to become active, use the **redundancy preferred-switch-card-slots** command.

Caution Do not hot swap an active switch processor module before putting it in standby mode. Removing an active switch processor breaks active connections and stops the flow of traffic through the switch. Put an active switch in standby mode using the **redundancy preferred-switch-card-slots** command before removing it from the chassis.

When a switchover to the standby switch processor occurs, the system resets and all connections are lost. When the system comes up again, all PVCs and SVCs are reestablished automatically.

Configuring Preferred Switching Processors (Catalyst 8540 MSR)

To configure which two of the three switch processors are active and which runs in standby mode, use the following privileged EXEC command on the primary route processor:

Command	Purpose
r**edundancy preferred-switch-card-slots** {**5** | **6** | **7**} {**5** | **6** | **7**}	Configures the active and standby switch processors.

Example

In the following example, the preferred switch processors are configured to be in slots 5 and 7 with the slot 6 switch processor running in standby mode:

```
Switch# redundancy preferred-switch-card-slots 5 7
The preferred switch cards selected are already active
```

Note The preferred switch card slot configuration reverts to the default configuration when the switch is power cycled.

Displaying the Preferred Switch Processor Redundancy Configuration (Catalyst 8540 MSR)

To display the preferred switch processor redundancy configuration, use the following privileged EXEC commands:

Command	Purpose
show preferred-switch-card-slots	Displays the preferred switch processor configuration.
show switch fabric	Displays the switch processor status.

The following example shows the preferred switch processor configuration and status:

```
  Switch# show preferred-switch-card-slots
→ The currently preferred switch card slots are slot: 5 and slot: 7
→ The currently active switch card slots are slot: 5 and slot: 7
  Switch# show switch fabric
  swc_presence_mask: 0x5
  Switch mode: NR_20G
  Number of Switch Cards present in the Chassis: 2

  SWC SLOT             SWC_TYPE         SWC_STATUS
  =================================================

→       5               EVEN              ACTIVE
→       6               NOT-PRESENT       NOT-PRESENT
→       7               ODD               ACTIVE

  <information deleted>
```

Displaying the Switch Processor EHSA Configuration (Catalyst 8540 MSR)

To display the switch processor EHSA configuration, use the following privileged EXEC command:

Command	Purpose
show capability {**primary** | **secondary**}	Displays the switch redundancy configuration.

The following shows the primary switch processor EHSA configuration:

```
Switch# show capability primary
 Dram Size is :64 MB
 Pmem Size is :4 MB
 Nvram Size is :512 KB
 BootFlash Size is :8 MB
 ACPM hw version 5.2
 ACPM functional version 4.0
 Netclk Module present flag :16
 NCLK hw version 3.1
 NCLK func version 8.0

 Printing the parameters for Switch card: 0
 SWC0 HW version 7.2
 SWC0 Functional version 1.2
 SWC0 Table memory size: 0 MB
```

```
SWC0 Feat Card Present Flag: 0
SWC0 Feat Card HW version 0.0
SWC0 Feat Card Functional version 0.0

Printing the parameters for Switch card: 1
SWC1 HW version 0.0
SWC1 Functional version 0.0
SWC1 Table memory size: 0 MB
SWC1 Feat Card Present Flag: 0
SWC1 Feat Card HW version 0.0
SWC1 Feat Card Functional version 0.0

Printing the parameters for Switch card: 2
SWC2 HW version 7.2
SWC2 Functional version 1.2
SWC2 Table memory size: 0 MB
SWC2 Feat Card Present Flag: 0
SWC2 Feat Card HW version 0.0
SWC2 Feat Card Functional version 0.0

Number of Controller supported in IOS: 7

Driver 0 type: 2560 super cam Functional Version 1.3

Driver 1 type: 2562 OC12 SPAM Functional Version 5.1

Driver 2 type: 2564 OC mother board Functional Version 5.1

Driver 3 type: 258 Switch Card Functional Version 1.0

Driver 4 type: 259 Switch Feature Card Functional Version 4.0
```

Configuring SNMP and RMON

SNMP is an application-layer protocol that allows an SNMP manager, such a network management system (NMS), and an SNMP agent on the managed device to communicate. You can configure SNMPv1, SNMPv2, or both, on the ATM switch router. Remote Monitoring (RMON) allows you to see the activity on network nodes. By using RMON in conjunction with the SNMP agent on the ATM switch router, you can monitor traffic through network devices, segment traffic that is not destined for the ATM switch router, and create alarms and events for proactive traffic management.

For detailed instructions on SNMP and general RMON configuration, refer to the *Configuration Fundamentals Configuration Guide*. For instructions on configuring ATM RMON, refer to Chapter 14, “Configuring ATM Accounting and ATM RMON.”

Storing the Configuration

When autoconfiguration and any manual configurations are complete, you should copy the configuration into nonvolatile random-access memory (NVRAM). If you should power off your ATM switch router prior to saving the configuration in NVRAM, all manual configuration changes are lost.

To save the running configuration to NVRAM, use the following command in privileged EXEC mode:

Command	Purpose
copy system:running-config nvram:startup-config	Copies the running configuration in system memory to the startup configuration stored in NVRAM.

Testing the Configuration

The following sections describe tasks you can perform to confirm the hardware, software, and interface configuration:

- Confirming the Hardware Configuration (Catalyst 8540 MSR), page 3-34
- Confirming the Hardware Configuration (Catalyst 8510 MSR and LightStream 1010), page 3-34
- Confirming the Software Version, page 3-35
- Confirming Power-on Diagnostics, page 3-35
- Confirming the Ethernet Configuration, page 3-37
- Confirming the ATM Address, page 3-37
- Testing the Ethernet Connection, page 3-38
- Confirming the ATM Connections, page 3-38
- Confirming the ATM Interface Configuration, page 3-39
- Confirming the Interface Status, page 3-39
- Confirming Virtual Channel Connections, page 3-40
- Confirming the Running Configuration, page 3-41
- Confirming the Saved Configuration, page 3-42

Note The following examples differ depending on whether the switch processor feature card is present. (Catalyst 8540 MSR)

Note The following examples differ depending on the feature card installed on the processor. (Catalyst 8510 MSR and LightStream 1010)

Confirming the Hardware Configuration (Catalyst 8540 MSR)

Use the **show hardware** and **show capability** commands to confirm the correct hardware installation:

```
Switch# show hardware

C8540 named Switch, Date: 08:36:44 UTC Fri May 21 1999

Slot Ctrlr-Type    Part No.   Rev  Ser No   Mfg Date   RMA No. Hw Vrs  Tst EEP
---- ------------  ---------- -- -------- --------- -------- ------- --- ---
 0/* Super Cam     73-2739-02 02 07287xxx Mar 31 98            3.0
 0/0 155MM PAM     73-1496-03 06 02180424 Jan 16 96 00-00-00   3.0     0   2
 0/1 155MM PAM     73-1496-03 00 02180455 Jan 17 96 00-00-00   3.0     0   2
 4/* Route Proc    73-2644-05 A0 03140NXK Apr 04 99 0          5.7
 4/0 Netclk Modul  73-2868-03 A0 03140NSU Apr 04 99 0          3.1
 5/* Switch Card   73-3315-08 B0 03170SMB May 03 99 0          8.3
 5/0 Feature Card  73-3408-04 B0 03160S4H May 03 99 0          4.1
 7/* Switch Card   73-3315-08 B0 03160SDT May 03 99 0          8.3
 7/0 Feature Card  73-3408-04 B0 03160RQV May 03 99 0          4.1
 8/* Route Proc    73-2644-05 A0 03140NXH Apr 04 99 0          5.7
 8/0 Netclk Modul  73-2868-03 A0 03140NVT Apr 04 99 0          3.1

DS1201 Backplane EEPROM:
Model  Ver.  Serial  MAC-Address  MAC-Size  RMA  RMA-Number   MFG-Date
------ ---- -------- ------------ --------  ---  ----------  -----------
C8540  2   6315484 00902156D800   1024      0        0      Mar 23 1999
cubi version : F

Power Supply:
Slot Part No.         Rev  Serial No.  RMA No.     Hw Vrs  Power Consumption
---- ---------------- ---- ----------- ----------- ------- -----------------
0          34-0829-02 A000 APQ0225000R 00-00-00-00   1.0            2746 cA
```

See the “Displaying the Switch Processor EHSA Configuration (Catalyst 8540 MSR)” section on page 3-31 for an example of the **show capability** command.

Confirming the Hardware Configuration (Catalyst 8510 MSR and LightStream 1010)

Use the **show hardware** command to confirm the correct hardware installation:

```
Switch# show hardware

LS1010 named ls1010_c5500, Date: XX:XX:XX UTC Thu Jan 8 1998
Feature Card’s FPGA Download Version: 10

Slot Ctrlr-Type    Part No.   Rev  Ser No   Mfg Date   RMA No. Hw Vrs  Tst EEP
---- ------------  ---------- -- -------- --------- -------- ------- --- ---
0/0  T1 PAM        12-3456-78 00 00000022 Aug 01 95 00-00-00   0.4     0   2
0/1  T1 PAM        12-3456-78 00 00000025 Aug 01 95 00-00-00   0.4     0   2
1/0  155MM PAM     73-1496-03 06 02180446 Jan 17 96 00-00-00   3.0     0   2
1/1  QUAD DS3 PAM  73-2197-02 00 03656116 Dec 18 96 00-00-00   1.0     0   2
3/0  155MM PAM     73-1496-03 00 02180455 Jan 17 96 00-00-00   3.0     0   2
2/0  ATM Swi/Proc  73-1402-06 D0 07202996 Dec 20 97 00-00-00   4.1     0   2
2/1  FeatureCard1  73-1405-05 B0 07202788 Dec 20 97 00-00-00   3.2     0   2

DS1201 Backplane EEPROM:
Model  Ver.  Serial  MAC-Address  MAC-Size  RMA  RMA-Number   MFG-Date
------ ---- -------- ------------ --------  ---  ----------  -----------
LS1010  2   69000050 00400B0A2E80   256      0        0      Aug 01 1995
```

Confirming the Software Version

Use the **show version** command to confirm the correct version and type of software and the configuration register are installed:

```
Switch# show version
Cisco Internetwork Operating System Software
IOS (tm) PNNI Software (cat8540m-WP-M), Version XX.X(X), RELEASE SOFTWARE
Copyright (c) 1986-1998 by cisco Systems, Inc.
Compiled XXX XX-XXX-XX XX:XX by
Image text-base: 0x600108B4, data-base: 0x6057A000

ROM: System Bootstrap, Version XX.X(X) RELEASE SOFTWARE

Switch uptime is 1 hour, 1 minute
System restarted by reload
System image file is "tftp://cat8540m-wp-mz_nimmu"

cisco C8540MSR (R5000) processor with 65536K/256K bytes of memory.
R5000 processor, Implementation 35, Revision 2.1 (512KB Level 2 Cache)
Last reset from power-on
1 Ethernet/IEEE 802.3 interface(s)
8 ATM network interface(s)
507K bytes of non-volatile configuration memory.

16384K bytes of Flash PCMCIA card at slot 0 (Sector size 128K).
8192K bytes of Flash internal SIMM (Sector size 256K).
Configuration register is 0x0
```

Confirming Power-on Diagnostics

Power-on diagnostics test the basic hardware functionality of the system when it is power cycled, when it is reloaded with a new version of power-on diagnostics software, or when you online insert and remove (OIR) a module. The power-on diagnostics test the route processors, switch processors, port adapters, interface modules.

Example (Catalyst 8540 MSR)

The following example displays the power-on diagnostic tests results for the Catalyst 8540 MSR:

```
Switch# show diag power-on
Cat8540 Power-on Diagnostics Status (.=Pass,F=Fail,U=Unknown,N=Not Applicable)
-----------------------------------------------------------------------------
   Last Power-on Date: 1999/07/28   Time: 11:06:12

   BOOTFLASH:  .   PCMCIA-Slot0: .   PCMCIA-Slot1: .
   CPU-IDPROM: .   NVRAM-Config: .
   ETHSRAM:    .   DRAM:         .   SARSRAM:       .

   PS0:        .   PS2:          N   PS (12V):     .
   FAN:        .   Temperature:  .   Bkp-IDPROM:   .


   Ethernet-port Access:   .         Ethernet-port CAM-Access: .
   Ethernet-port Loopback: .         Ethernet-port Loadgen:    .

Power-on Diagnostics Passed.
```

Example (Catalyst 8510 MSR and LightStream 1010)

The following example displays the power-on diagnostic tests results for the Catalyst 8510 MSR and LightStream 1010:

```
NewLs1010# show diag power-on
LS1010 Power-on Diagnostics Status (.=Pass,F=Fail,U=Unknown,N=Not Applicable)
-----------------------------------------------------------------------------
   Last Power-on Diags  Date: 99/07/09   Time: 07:52:17   By: V 4.51

   BOOTFLASH:  .   PCMCIA-Slot0: .   PCMCIA-Slot1: N
   CPU-IDPROM: .   FCard-IDPROM: .   NVRAM-Config: .
   SRAM:       .   DRAM:         .

   PS1:        .   PS2:          N   PS (12V):     .
   FAN:        .   Temperature:  .   Bkp-IDPROM:   .

   MMC-Switch Access: .              Accordian Access: .
   LUT: .   ITT: .   OPT: .   OTT: .   STK: .   LNK: .   ATTR: .   Queue: .
   Cell-Memory:  .

   FC-PFQ
    Access: .
     RST: .    REG: .    IVC: .    IFILL: .    OVC: .    OFILL: .

    TEST:
     CELL: .   SNAKE: .   RATE: .   MCAST: .   SCHED: .
     TGRP: .   UPC  : .   ABR : .   RSTQ : .

Access/Interrupt/Loopback/CPU-MCast/Port-MCast/FC-MCast/FC-TMCC Test Status:
Ports                      0         1         2         3
----------------------------------------------------------------------------
PAM 0/0 (IMA8T1)        .....NN   .....NN   .....NN   .....NN
    Port 4  to 7 :      .....NN   .....NN   .....NN   .....NN
PAM 0/1 (IMA8E1)        .....NN   .....NN   .....NN   .....NN
    Port 4  to 7 :      .....NN   .....NN   .....NN   .....NN
PAM 1/0 (FR4CE1)       .....NN   .....NN   .....NN   .....NN
PAM 1/1 (155UTP)        .....NN   .....NN   .....NN   .....NN
PAM 3/0 (T1)            .....NN   .....NN   .....NN   .....NN
PAM 3/1 (E1CEUTP)       .....NN   .....NN   .....NN   .....NN
PAM 4/0 (DS3)           .....NN   .....NN      N         N
PAM 4/1 (25M)           .....NN   .....NN   .....NN   .....NN
    Port 4  to 7 :      .....NN   .....NN   .....NN   .....NN
    Port 8  to 11:      .....NN   .....NN   .....NN   .....NN


FRPAM#          ING-SSRAM ING-SDRAM EGR-SSRAM EGR-SDRAM LOOPBACK
------------------------------------------------------------------
PAM 1/0 (FR4CE1)  .        .         .         .         .
   Ethernet-port Access:    .        Ethernet-port CAM-Access: .
   Ethernet-port Loopback:  .        Ethernet-port Loadgen:    .
   GEPAM Microcode:         .        GEPAM Access:            .
   GEPAM CAM Access:        .

Power-on Diagnostics Passed.
```

Confirming the Ethernet Configuration

Use the **show interfaces** command to confirm that the Ethernet interface on the route processor is configured correctly:

```
Switch# show interfaces ethernet 0
Ethernet0 is up, line protocol is up
  Hardware is SonicT, address is 0000.0000.0000 (bia 0000.0000.0000)
  Internet address is 172.20.52.20/26
  MTU 1500 bytes, BW 10000 Kbit, DLY 1000 usec, rely 255/255, load 1/255
  Encapsulation ARPA, loopback not set, keepalive set (10 sec)
  ARP type: ARPA, ARP Timeout 04:00:00
  Last input 00:00:00, output 00:00:00, output hang never
  Last clearing of "show interface" counters never
  Queueing strategy: fifo
  Output queue 0/40, 0 drops; input queue 0/75, 0 drops
  5 minute input rate 1000 bits/sec, 2 packets/sec
  5 minute output rate 0 bits/sec, 1 packets/sec
     69435 packets input, 4256035 bytes, 0 no buffer
     Received 43798 broadcasts, 0 runts, 0 giants, 0 throttles
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
     0 input packets with dribble condition detected
     203273 packets output, 24079764 bytes, 0 underruns
     0 output errors, 0 collisions, 2 interface resets
     0 babbles, 0 late collision, 0 deferred
     0 lost carrier, 0 no carrier
     0 output buffer failures, 0 output buffers swapped out
```

Confirming the ATM Address

Use the **show atm addresses** command to confirm correct configuration of the ATM address for the ATM switch router:

```
Switch# show atm addresses

Switch Address(es):
  47.00918100000000100000001.000100000001.00 active

Soft VC Address(es):
  47.0091.8100.0000.0001.0000.0001.4000.0c80.9000.00 ATM1/1/0
  47.0091.8100.0000.0001.0000.0001.4000.0c80.9010.00 ATM1/1/1
  47.0091.8100.0000.0001.0000.0001.4000.0c80.9020.00 ATM1/1/2
  47.0091.8100.0000.0001.0000.0001.4000.0c80.9030.00 ATM1/1/3
  47.0091.8100.0000.0001.0000.0001.4000.0c81.8000.00 ATM3/0/0
  47.0091.8100.0000.0001.0000.0001.4000.0c81.8000.63 ATM3/0/0.99
  47.0091.8100.0000.0001.0000.0001.4000.0c81.8010.00 ATM3/0/1
  47.0091.8100.0000.0001.0000.0001.4000.0c81.8020.00 ATM3/0/2
  47.0091.8100.0000.0001.0000.0001.4000.0c81.8030.00 ATM3/0/3
  47.0091.8100.0000.0001.0000.0001.4000.0c81.9000.00 ATM3/1/0
  47.0091.8100.0000.0001.0000.0001.4000.0c81.9010.00 ATM3/1/1
  47.0091.8100.0000.0001.0000.0001.4000.0c81.9020.00 ATM3/1/2
  47.0091.8100.0000.0001.0000.0001.4000.0c81.9030.00 ATM3/1/3

 <information deleted>

ILMI Switch Prefix(es):
  47.0091.8100.0000.0001.0000.0001

ILMI Configured Interface Prefix(es):

LECS Address(es):
```

Testing the Ethernet Connection

After you have configured the IP address(es) for the Ethernet interface, test for connectivity between the switch and a host. The host can reside anywhere in your network. To test for Ethernet connectivity, use the following user EXEC command:

Command	Purpose
ping ip *ip-address*	Tests the configuration using the **ping** command. The **ping** command sends an echo request to the host specified in the command.

For example, to test Ethernet connectivity from the switch to a workstation with an IP address of 172.20.40.201, enter the command **ping ip 172.20.40.201**. If the switch receives a response, the following message displays:

```
Switch# ping ip 172.20.40.201

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.20.40.201, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/202/1000 ms
```

Confirming the ATM Connections

Use the **ping atm interface** command to confirm that the ATM connections are configured correctly:

```
Switch# ping atm interface atm 3/0/0 0 5 seg-loopback

Type escape sequence to abort.
Sending Seg-Loopback 5, 53-byte OAM Echoes to a neighbour,timeout is 5 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/4 ms
Switch#
```

Confirming the ATM Interface Configuration

Use the **show atm interface** command to confirm the ATM interfaces are configured correctly:

```
Switch# show atm interface atm 1/0/0

Interface:      ATM1/0/0        Port-type:      oc3suni
IF Status:      UP              Admin Status:   up
Auto-config:    disabled        AutoCfgState:   not applicable
IF-Side:        Network         IF-type:        NNI
Uni-type:       not applicable  Uni-version:    not applicable
Max-VPI-bits:   8               Max-VCI-bits:   14
Max-VP:         255             Max-VC:         16383
ConfMaxSvpcVpi: 255             CurrMaxSvpcVpi: 255
ConfMaxSvccVpi: 255             CurrMaxSvccVpi: 255
ConfMinSvccVci: 35              CurrMinSvccVci: 35
Svc Upc Intent: pass            Signalling:     Enabled
ATM Address for Soft VC: 47.0091.8100.0000.00e0.4fac.b401.4000.0c80.8000.00
Configured virtual links:
  PVCLs SoftVCLs   SVCLs   TVCLs   PVPLs SoftVPLs   SVPLs Total-Cfgd Inst-Conns
      4        0       0       0       1        0       0          5          3
Logical ports(VP-tunnels):     1
Input cells:    263109          Output cells:   268993
5 minute input rate:             0 bits/sec,       0 cells/sec
5 minute output rate:         1000 bits/sec,       2 cells/sec
Input AAL5 pkts: 171788, Output AAL5 pkts: 174718, AAL5 crc errors: 0
```

Confirming the Interface Status

Use the **show atm status** command to confirm the status of ATM interfaces:

```
Switch# show atm status
NUMBER OF INSTALLED CONNECTIONS: (P2P=Point to Point, P2MP=Point to MultiPoint)

Type       PVCs  SoftPVCs      SVCs      PVPs  SoftPVPs      SVPs      Total
P2P          30         0         0         1         1         0         32
P2MP          0         0         0         1         0         0          1
                                    TOTAL INSTALLED CONNECTIONS =         33

PER-INTERFACE STATUS SUMMARY AT 16:07:59 UTC Wed Nov 5 1997:
   Interface      IF        Admin  Auto-Cfg    ILMI Addr     SSCOP    Hello
     Name       Status      Status   Status    Reg State     State    State
------------- -------- ------------ -------- ------------ --------- --------
ATM1/1/0          DOWN         down  waiting          n/a      Idle      n/a
ATM1/1/1          DOWN         down  waiting          n/a      Idle      n/a
ATM1/1/2          DOWN         down  waiting          n/a      Idle      n/a
ATM1/1/3          DOWN         down  waiting          n/a      Idle      n/a
ATM0                UP           up      n/a  UpAndNormal      Idle      n/a
ATM3/0/0            UP           up      n/a  UpAndNormal    Active  LoopErr
ATM3/0/0.99         UP           up  waiting  WaitDevType      Idle      n/a
ATM3/0/1            UP           up     done  UpAndNormal    Active  LoopErr
ATM3/0/2            UP           up      n/a  UpAndNormal    Active  LoopErr
ATM3/0/3            UP           up     done  UpAndNormal    Active  LoopErr
ATM3/1/0            UP           up     done  UpAndNormal    Active  LoopErr
ATM3/1/1            UP           up     done  UpAndNormal    Active  LoopErr
ATM3/1/2            UP           up     done  UpAndNormal    Active  LoopErr
ATM3/1/3            UP           up     done  UpAndNormal    Active  LoopErr
<information deleted>
```

Confirming Virtual Channel Connections

Use the **show atm vc** command to confirm the status of ATM virtual channel connections:

```
Switch# show atm vc
Interface    VPI   VCI   Type    X-Interface  X-VPI X-VCI  Encap Status
ATM1/1/0     0     5      PVC     ATM0         0     52    QSAAL  DOWN
ATM1/1/0     0     16     PVC     ATM0         0     32    ILMI   DOWN
ATM1/1/1     0     5      PVC     ATM0         0     53    QSAAL  DOWN
ATM1/1/1     0     16     PVC     ATM0         0     33    ILMI   DOWN
ATM1/1/2     0     5      PVC     ATM0         0     54    QSAAL  DOWN
ATM1/1/2     0     16     PVC     ATM0         0     34    ILMI   DOWN
ATM1/1/3     0     5      PVC     ATM0         0     55    QSAAL  DOWN
ATM1/1/3     0     16     PVC     ATM0         0     35    ILMI   DOWN
ATM0         0     32     PVC     ATM1/1/0     0     16    ILMI   DOWN
ATM0         0     33     PVC     ATM1/1/1     0     16    ILMI   DOWN
ATM0         0     34     PVC     ATM1/1/2     0     16    ILMI   DOWN
ATM0         0     35     PVC     ATM1/1/3     0     16    ILMI   DOWN
ATM0         0     36     PVC     ATM3/0/0     0     16    ILMI   UP
ATM0         0     37     PVC     ATM3/0/1     0     16    ILMI   UP
ATM0         0     38     PVC     ATM3/0/2     0     16    ILMI   UP
ATM0         0     39     PVC     ATM3/0/3     0     16    ILMI   UP
ATM0         0     40     PVC     ATM3/1/0     0     16    ILMI   UP
ATM0         0     41     PVC     ATM3/1/1     0     16    ILMI   UP
ATM0         0     42     PVC     ATM3/1/2     0     16    ILMI   UP
ATM0         0     43     PVC     ATM3/1/3     0     16    ILMI   UP
<information deleted>
```

Use the **show atm vc interface** *card*/*subcard*/*port* command to confirm the status of ATM virtual channels on a specific interface:

```
Switch# show atm vc interface atm 3/0/0
Interface    VPI   VCI   Type    X-Interface  X-VPI X-VCI  Encap Status
ATM3/0/0     0     5      PVC     ATM0         0     56    QSAAL  UP
ATM3/0/0     0     16     PVC     ATM0         0     36    ILMI   UP
ATM3/0/0     0     18     PVC     ATM0         0     85    PNNI   UP
ATM3/0/0     50    100    PVC     ATM3/0/1     60    200          DOWN
                                  ATM3/0/2     70    210          UP
                                  ATM3/0/3     80    220          UP
ATM3/0/0     100   200    SoftVC  NOT CONNECTED
```

Use the **show atm vc interface atm** *card*/*subcard*/*port vpi vci* command to confirm the status of a specific ATM interface and virtual channel connection.

```
Switch# show atm vc interface atm 0/0/0 0 16

Interface: ATM0/0/0, Type: oc3suni
VPI = 0  VCI = 16
Status: DOWN
Time-since-last-status-change: 1w5d
Connection-type: PVC
Cast-type: point-to-point
Packet-discard-option: enabled
Usage-Parameter-Control (UPC): pass
Wrr weight: 15
Number of OAM-configured connections: 0
OAM-configuration: disabled
OAM-states:  Not-applicable
Cross-connect-interface: ATM0, Type: Unknown
Cross-connect-VPI = 0
Cross-connect-VCI = 35
Cross-connect-UPC: pass
Cross-connect OAM-configuration: disabled
```

```
Cross-connect OAM-state:  Not-applicable
Encapsulation: AAL5ILMI
Threshold Group: 6, Cells queued: 0
Rx cells: 0, Tx cells: 0
Tx Clp0:0,  Tx Clp1: 0
Rx Clp0:0,  Rx Clp1: 0
Rx Upc Violations:0, Rx cell drops:0
Rx pkts:0, Rx pkt drops:0
Rx connection-traffic-table-index: 3
Rx service-category: VBR-RT (Realtime Variable Bit Rate)
Rx pcr-clp01: 424
Rx scr-clp01: 424
Rx mcr-clp01: none
Rx      cdvt: 1024 (from default for interface)
Rx       mbs: 50
Tx connection-traffic-table-index: 3
Tx service-category: VBR-RT (Realtime Variable Bit Rate)
Tx pcr-clp01: 424
Tx scr-clp01: 424
Tx mcr-clp01: none
Tx      cdvt: none
Tx       mbs: 50
```

Confirming the Running Configuration

Use the **more system:running-config** command to confirm that the current configuration is correct:

```
Switch# more system:running-config
version XX.X
no service pad
no service password-encryption
!
hostname Switch
!
<information deleted>
!
interface Ethernet0
 ip address 172.20.52.11 255.255.255.224
 no ip directed-broadcast
!
interface ATM-E0
 no ip address
 no ip directed-broadcast
 atm pvc 0 29 pd on wrr-weight 15 rx-cttr 3 tx-cttr 3  interface  ATM0 0 any-vci
 wrr-weight 15  encap
!
interface Async1
 no ip address
 no ip directed-broadcast
 hold-queue 10 in
!
logging buffered 4096 debugging
!
line con 0
 exec-timeout 0 0
 transport input none
line vty 0 4
 exec-timeout 0 0
 no login
!
end
```

Confirming the Saved Configuration

Use the **more nvram:startup-config** command to confirm that the configuration saved in NVRAM is correct:

```
Switch# more nvram:startup-config
version XX.X
no service pad
no service password-encryption
!
hostname Switch
!
<information deleted>
!
interface Ethernet0
 ip address 172.20.52.11 255.255.255.224
 no ip directed-broadcast
!
interface ATM-E0
 no ip address
 no ip directed-broadcast
!
interface Async1
 no ip address
 no ip directed-broadcast
 hold-queue 10 in
!
logging buffered 4096 debugging
!
line con 0
 exec-timeout 0 0
 transport input none
line vty 0 4
 exec-timeout 0 0
 no login
!
end
```

CHAPTER 4

Configuring System Management Functions

This chapter describes the basic tasks for configuring general system features, such as access control and basic switch management.

Note This chapter provides advanced configuration instructions for the Catalyst 8540 MSR, Catalyst 8510 MSR, and LightStream 1010 ATM switch routers. For complete descriptions of the commands mentioned in this chapter, refer to the *ATM Switch Router Command Reference* publication.

The following sections describe basic tasks for configuring general system features, such as access control and basic switch management tasks:

System Management Tasks

The role of the administration interface is to provide a simple command-line interface to all internal management and debugging facilities of the ATM switch router.

Configuring Terminal Lines and Modem Support (Catalyst 8540 MSR)

The Catalyst 8540 MSR has a console terminal line that might require configuration. For line configuration, you must first set up the line for the terminal or the asynchronous device attached to it. For a complete description of configuration tasks and commands used to set up your terminal line and settings, refer to the *Dial Solutions Configuration Guide* and *Dial Solutions Command Reference* publications.

You can connect a modem to the console port. The following settings on the modem are required:

- Enable auto answer mode
- Suppress result codes

You can configure your modem by setting the dual in-line package (DIP) switches on the modem or by connecting the modem to terminal equipment. Refer to the user manual provided with your modem for the correct configuration information.

Note Because there are no hardware flow control signals available on the console port, the console port terminal characteristics should match the modem settings.

Configuring Terminal Lines and Modem Support (Catalyst 8510 MSR and LightStream 1010)

The Catalyst 8510 MSR and LightStream 1010 ATM switch routers have two types of terminal lines: a console line and an auxiliary line. For line configuration, you must first set up the lines for the terminals or other asynchronous devices attached to them. For a complete description of configuration tasks and commands used to set up your lines, modems, and terminal settings, refer to the *Dial Solutions Configuration Guide* and *Dial Solutions Command Reference* publications.

Configuring Alias

You can create aliases for commonly used or complex commands. Use word substitutions or abbreviations to tailor command syntax. For detailed instructions on performing these tasks, refer to the *Configuration Fundamentals Configuration Guide* publication.

Configuring Buffers

To make adjustments to initial buffer pool settings and to the limits at which temporary buffers are created and destroyed, use the following global configuration command:

Command	Purpose
buffers {**small** | **middle** | **big** | **verybig** | **large** | **huge** | *type number*}	Configures buffers; the default huge buffer size is 18,024 bytes.
show buffers [**all** | **assigned** [**dump**]]	Displays statistics for the buffer pools on the network server.

To display the buffer pool statistics, use the following privileged EXEC command:

Command	Purpose
show buffers [**address** *hex-addr* | **all** | **assigned** | **free** | **input-interface** *type card*/*subcard*/*port* | **old** | **pool** *name* [**dump** | **header** | **packet**]] [**failures**]	Displays statistics for the buffer pools on the network server.

Configuring Cisco Discovery Protocol

To specify how often your ATM switch router sends Cisco Discovery Protocol (CDP) updates, perform the following tasks in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **cdp holdtime** *seconds*	Specifies the hold time in seconds, to be sent in packets.
Step 2	Switch(config)# **cdp timer** *seconds*	Specifies how often your ATM switch router will send CDP updates.
Step 3	Switch(config)# **cdp run**	Enables CDP.

To reset CDP traffic counters to zero (0) on your ATM switch router, perform the following tasks in privileged EXEC mode:

	Command	Purpose
Step 1	Switch# **clear cdp counters**	Clears CDP counters.
Step 2	Switch# **clear cdp table**	Clears CDP tables.

To show the CDP configuration, use the following privileged EXEC commands:

Command	Purpose
show cdp	Displays global CDP information.
show cdp *entry-name* [**protocol** | **version**]	Displays information about a neighbor device listed in the CDP table.
show cdp interface [*interface-type interface-number*]	Displays interfaces on with CDP enabled.
show cdp neighbors [*interface-type interface-number*] [**detail**]	Displays CDP neighbor information.
show cdp traffic	Displays CDP traffic information.

Configuring Enable Passwords

To log on to the ATM switch router at a specified level, use the following EXEC command:

Command	Purpose
enable *level*	Enables login.

To configure the enable password for a given level, use the following global configuration command:

Command	Purpose
enable password [**level** *number*] [*encryption-type*] *password*	Configures the enable password.

Configuring Load Statistics Interval

To change the length of time for which data is used to compute load statistics, perform the following tasks, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **interface** {**atm** | **ethernet**} **0** Switch(config-if)#	Selects the route processor interface to be configured.
Step 2	Switch(config-if)# **load-interval** *seconds*	Configures the load interval.

Configuring Logging

To log messages to a syslog server host, use the following global configuration commands:

Command	Purpose
logging *host*	Configures the logging name or IP address of the host to be used as a syslog server.
logging buffered [*level* | *size*]	Logs messages to an internal buffer, use the **logging buffered** global configuration command. The **no logging buffered** command cancels the use of the buffer and writes messages to the console terminal, which is the default.
logging console *level*	Limits messages logged to the console based on severity, use the **logging console** global configuration command.
logging facility *type*	Configures the syslog facility in which error messages are sent, use the **logging facility** global configuration command. To revert to the default of local, use the **no logging facility** global configuration command.

Command	Purpose
logging monitor *level*	Limits messages logged to the terminal lines (monitors) based on severity, use the **logging monitor** global configuration command. This command limits the logging messages displayed on terminal lines other than the console line to messages with a level at or above *level*. The **no logging monitor** command disables logging to terminal lines other than the console line.
logging on	Controls logging of error messages, use the **logging on** global configuration command. This command enables or disables message logging to all destinations except the console terminal. The **no logging on** command enables logging to the console terminal only.
logging trap *level*	Limits messages logged to the syslog servers based on severity, use the **logging trap** global configuration command. The command limits the logging of error messages sent to syslog servers to only those messages at the specified level. The **no logging trap** command disables logging to syslog servers.
logging source-interface *type identifier*	Specifies the interface for source address in logging transactions.

Configuring Login Authentication

To enable TACACS+ authentication for logins, perform the following steps, beginning in global configuration mode:

Command	Purpose
line [**aux** | **console** | **vty**] *line-number* [*ending-line-number*]	Selects the line to configure.
login [**local** | **tacacs**]	Configures login authentication.

Configuring Scheduler Attributes

To control the maximum amount of time that can elapse without running the lowest-priority system processes, use the following global configuration commands:

Command	Purpose
scheduler allocate *msecs*	Configures the guaranteed CPU time for processes, in milliseconds. The minimum interval is 500 ms; the maximum value is 6000 ms.
scheduler process-watchdog {**hang** | **normal** | **reload** | **terminate**}	Configures scheduler process-watchdog action for looping processes.
scheduler interval *msecs*	Specifies maximum time in milliseconds that can elapse without running system processes.

Configuring Services

To configure miscellaneous system services, use the following global configuration commands:

Command	Purpose
service alignment	Configures alignment correction and logging.
service compress-config	Compresses the configuration file.
service config	Loads config TFTP files.
service disable-ip-fast-frag	Disables IP particle-based fast fragmentation.
service exec-callback	Enables EXEC callback.
service exec-wait	Configures a delay of the start-up of the EXEC on noisy lines.
service finger	Allows Finger protocol requests (defined in RFC 742) from the network server.
service hide-telnet-addresses	Hides destination addresses in Telnet command.
service linenumber	Enables a line number banner for each EXEC.
service nagle	Enables the Nagle congestion control algorithm.
service old-slip-prompts	Allows old scripts to operate with SLIP/PPP.
service pad	Enables Packet Assembler Dissembler commands.
service password-encryption	Enables encrypt passwords.
service prompt	Enables a mode-specific prompt.
service slave-log	Enables log capability on slave IPs.
service tcp-keepalives {**in** | **out**}	Configures keepalive packets on idle network connections.
service tcp-small-servers	Enables small TCP servers (for example, ECHO).

Command	Purpose
service telnet-zero-idle	Sets the TCP window to zero (0) when the Telnet connection is idle.
service timestamps	Displays timestamp debug/log messages.
service udp-small-servers	Enables small UDP servers (for example, ECHO).

Configuring SNMP

To create or update an access policy, use the following global configuration commands:

Command	Purpose
snmp-server access-policy *destination-party source-party context privileges*	Configures global access policy.
snmp-server chassis-id *text*	Provides a message line identifying the SNMP server serial number.
snmp-server community *string* [**RO** | **RW**] [*number*]	Configures the SNMP community access string.
snmp-server contact *text*	Configures the system contact (syscontact) string.
snmp-server context *context-name context-oid view-name*	Configures a context record.
snmp-server enable	Enables SNMP traps or informs.
snmp-server host *name community-string* [**envmon**] [**frame-relay**] [**sdlc**] [**snmp**] [**tty**] [**x25**]	Configures the recipient of an SNMP trap operation.
snmp-server location *text*	Configures a system location string.
snmp-server packetsize *byte-count*	Configures the largest SNMP packet size permitted when the SNMP server is receiving a request or generating a reply.
snmp-server party *party-name party-oid* [*protocol-address*] [**packetsize** *size*] [**local** | **remote**] [**authentication** {**md5** *key* [**clock** *clock*] [**lifetime** *lifetime*] | **snmpv1** *string*}]	Configures a party record.
snmp-server queue-length *length*	Configures the message queue length for each trap host.
snmp-server system-shutdown	Enables use of the SNMP reload command.
snmp-server trap-authentication [**snmpv1** | **snmpv2**]	Configures trap message authentication.
snmp-server trap-timeout *seconds*	Configures how often to resend trap messages on the retransmission queue.
snmp-server view *view-name mib-tree* {**included** | **excluded**}	Configures view entry.

To display the SNMP status, use the following EXEC command:

Command	Purpose
show snmp	Checks the status of communications between the SNMP agent and SNMP manager.

Username Commands

To establish a username-based authentication system at login, use the following global configuration commands:

Command	Purpose
username *name* [**dnis**] [**nopassword** | **password** [*encryption-type*] *password*]	Configures username-based authentication system at login.
username *name* **password** *secret*	Configures username-based CHAP authentication system at login.
username *name* **autocommand** *command*	Configures username-based authentication system at login with an additional command to be added.
username *name* **nohangup**	Configures username-based authentication system at login and prevents Cisco IOS from disconnecting after the automatic command is completed.
username *name* **noescape**	Configures username-based authentication system at login but prevents the user from issuing an escape character on the switch.
username *name* **privilege** *level*	Sets user privilege level.

Configuring the Privilege Level

This section describes configuring and displaying the privilege level access to the ATM switch router. The access privileges can be configured at the global level or at the line level for a specific line.

Configuring Privilege Level (Global)

To set the privilege level for a command, use the following global configuration command:

Command	Purpose
privilege *mode* **level** *number command* [*type*]	Sets the privilege level.

To display your current level of privilege, use the following privileged EXEC command:

Command	Purpose
show privilege	Displays the privilege level.

Configuring Privilege Level (Line)

To set the default privilege level for a line, perform the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **line** [**aux** | **console** | **vty**] *line-number* [*ending-line-number*]	Selects the line to configure.
Step 2	Switch(config-line)# **privilege level** *number*	Configures the default privilege level.

To display your current level of privilege, use the following privileged EXEC command:

Command	Purpose
show privilege	Displays the privilege level.

Configuring the Network Time Protocol

This section describes configuring the Network Time Protocol (NTP) on the ATM switch router.

To control access to the system NTP services, use the following **ntp** global configuration commands. To remove access control to the system's NTP services, use the **no ntp** command. See the example configuration at the end of this section and the "Displaying the NTP Configuration" section on page 4-12 to confirm the NTP configuration.

To see a list of the NTP commands enter a **?** in EXEC configuration mode. The following example shows the list of commands available for NTP configuration:

```
Switch(config)# ntp ?
  access-group        Control NTP access
  authenticate        Authenticate time sources
  authentication-key  Authentication key for trusted time sources
  broadcastdelay      Estimated round-trip delay
  clock-period        Length of hardware clock tick
  master              Act as NTP master clock
  max-associations    Set maximum number of associations
  peer                Configure NTP peer
  server              Configure NTP server
  source              Configure interface for source address
  trusted-key         Key numbers for trusted time sources
  update-calendar     Periodically update calendar with NTP time
```

To control access to the system NTP services, use the following global configuration command:

Command	Purpose
ntp access-group {**query-only** | **serve-only** | **serve** | **peer**} *access-list-number*	Configures an NTP access group.

To enable NTP authentication, perform the following steps in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **ntp authenticate**	Enables NTP authentication.
Step 2	Switch(config)# **ntp authentication-key** *number* **md5** *value*	Defines an authentication key.

To specify that a specific interface should send NTP broadcast packets, perform the following steps, beginning to global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **interface** *type card*/*subcard*/*port* Switch(config-if)#	Selects the physical interface to be configured.
Step 2	Switch(config-if)# **ntp broadcast** [**client** | **destination** | **key** | **version**]	Configures the system to receive NTP broadcast packets.

As NTP compensates for the error in the system clock, it keeps track of the correction factor for this error. The system automatically saves this value into the system configuration using the **ntp clock-period** global configuration command.

Caution Do not enter the **ntp clock-period** command; it is documented for informational purposes only. The system automatically generates this command as NTP determines the clock error and compensates.

To prevent an interface from receiving NTP packets, perform the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **interface** *type card/subcard/port* Switch(config-if)#	Selects the physical interface to be configured.
Step 2	Switch(config-if)# **ntp disable**	Disables the NTP receive interface.

To configure the ATM switch router as a NTP master clock to which peers synchronize themselves when an external NTP source is not available, use the following global configuration command:

Command	Purpose
ntp master [*stratum*]	Configures NTP master clock.

To configure the ATM switch router as a NTP peer that receives its clock synchronization from an external NTP source, use the following global configuration command:

Command	Purpose
ntp peer *ip-address* [**version** *number*] [**key** *keyid*] [**source** *interface*] [**prefer**]	Configures the system clock to synchronize a peer or to be synchronized by a peer.

To allow the ATM switch router system clock to be synchronized by a time server, use the following global configuration command:

Command	Purpose
ntp server *ip-address* [**version** *number*] [**key** *keyid*] [**source** *interface*] [**prefer**]	Configures the system clock to allow it to be synchronized by a time server.

To use a particular source address in NTP packets, use the following global configuration command:

Command	Purpose
ntp source *interface type card/subcard/port*	Configures a particular source address in NTP packets.

To authenticate the identity of a system to which NTP will synchronize, use the following global configuration command:

Command	Purpose
ntp trusted-key *key-number*	Configures an NTP synchronize number.

To periodically update the ATM switch router calendar from NTP, use the following global configuration command:

Command	Purpose
ntp update-calendar	Updates an NTP calendar.

Example

The following example configures the ATM switch router to synchronize its clock and calendar to an NTP server, using ethernet0, and other features:

```
Switch# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)# ntp server 198.92.30.32
Switch(config)# ntp source ethernet0
Switch(config)# ntp authenticate
Switch(config)# ntp max-associations 2000
Switch(config)# ntp trusted-key 22507
Switch(config)# ntp update-calendar
```

Displaying the NTP Configuration

To show the status of NTP associations, use the following privileged EXEC commands:

Command	Purpose
show ntp associations [**detail**]	Displays NTP associations.
show ntp status	Displays the NTP status.

Examples

The following example displays detail NTP configuration:

```
Switch# show ntp associations detail
198.92.30.32 configured, our_master, sane, valid, stratum 3
ref ID 171.69.2.81, time B6C04E67.6E779000 (18:18:15.431 UTC Thu Feb 27 1997)
our mode client, peer mode server, our poll intvl 128, peer poll intvl 128
root delay 109.51 msec, root disp 377.38, reach 377, sync dist 435.638
delay -3.88 msec, offset 7.7674 msec, dispersion 1.57
precision 2**17, version 3
org time B6C04F19.437D8000 (18:21:13.263 UTC Thu Feb 27 1997)
rcv time B6C04F19.41018C62 (18:21:13.253 UTC Thu Feb 27 1997)
xmt time B6C04F19.41E3EB4B (18:21:13.257 UTC Thu Feb 27 1997)
filtdelay =    -3.88   -3.39   -3.49   -3.39   -3.36   -3.46   -3.37   -3.16
filtoffset =    7.77    6.62    6.60    5.38    4.13    4.43    6.28   12.37
filterror =     0.02    0.99    1.48    2.46    3.43    4.41    5.39    6.36
```

The following example displays the NTP status:

```
Switch# show ntp status
Clock is synchronized, stratum 4, reference is 198.92.30.32
nominal freq is 250.0000 Hz, actual freq is 249.9999 Hz, precision is 2**24
reference time is B6C04F19.41018C62 (18:21:13.253 UTC Thu Feb 27 1997)
clock offset is 7.7674 msec, root delay is 113.39 msec
root dispersion is 386.72 msec, peer dispersion is 1.57 msec
```

Configuring the Clock and Calendar

If no other source of time is available, you can manually configure the current time and date after the system is restarted. The time will remain accurate until the next system restart. Cisco recommends that you use manual configuration only as a last resort.

Note If you have an outside source to which the ATM switch router can synchronize, you do not need to manually set the system clock.

Configuring the Clock

To configure, read, and set the ATM switch router as a time source for a network based on its calendar, perform the following steps in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **clock calendar-valid**	Sets the ATM switch router as the default clock.
Step 2	Switch(config)# **clock summer-time** *zone* **recurring** [*week day month hh:mm week day month hh:mm* [*offset*]]	Configures the system to automatically switch to summer time (daylight savings time), use one of the formats of the **clock summer-time** configuration command.
Step 3	Switch(config)# **clock timezone** *zone hours* [*minutes*]	Configures the system time zone.

To manually read and set the calendar into the ATM switch router system clock, perform the following steps in privileged EXEC mode:

	Command	Purpose
Step 1	Switch# **clock read-calendar**	Reads the calendar.
Step 2	Switch# **clock set** *hh:mm:ss day month year*	Manually sets the system clock.
Step 3	Switch# **clock update-calendar**	Sets the calendar.

To display the system clock information, use the following EXEC command:

Command	Purpose
show clock [*detail*]	Displays the system clock.

Configuring the Calendar

To set the system calendar, use the following privileged EXEC command:

Command	Purpose
calendar set *hh:mm:ss day month year*	Configures the calendar.

To display the system calendar information, use the following EXEC command:

Command	Purpose
show calendar	Displays the calendar setting.

Configuring TACACS

You can configure the ATM switch router to use one of three special TCP/IP protocols related to TACACS: regular TACACS, extended TACACS, or AAA/TACACS+. TACACS services are provided by and maintained in a database on a TACACS server running on a workstation. You must have access to and configure a TACACS server before configuring the TACACS features described in this publication on your Cisco device. Cisco's basic TACACS support is modeled after the original Defense Data Network (DDN) application.

A comparative description of the supported versions follows. Table 4-1 compares the versions by commands.

- TACACS—Provides password checking, authentication, and notification of user actions for security and accounting purposes.
- Extended TACACS—Provides information about protocol translator and ATM switch router use. This information is used in UNIX auditing trails and accounting files.

> **Note** The extended TACACS software is available using FTP (refer to the README file in the ftp.cisco.com directory).

- AAA/TACACS+—Provides more detailed accounting information as well as more administrative control of authentication and authorization processes.

You can establish TACACS-style password protection on both user and privileged levels of the system EXEC.

Table 4-1 TACACS Command Comparison

Command	TACACS	Extended TACACS	TACACS+
aaa accounting			X
aaa authentication arap			X
aaa authentication enable default			X
aaa authentication login			X

Table 4-1 TACACS Command Comparison (continued)

Command	TACACS	Extended TACACS	TACACS+
aaa authentication local override			X
aaa authentication ppp			X
aaa authorization			X
aaa new-model			X
arap authentication			X
arap use-tacacs	X	X	
enable last-resort	X	X	
enable use-tacacs	X	X	
login authentication			X
login tacacs	X	X	
ppp authentication	X	X	X
ppp use-tacacs	X	X	X
tacacs-server attempts	X	X	X
tacacs-server authenticate	X	X	
tacacs-server extended		X	
tacacs-server host	X	X	X
tacacs-server key			X
tacacs-server last-resort	X	X	
tacacs-server notify	X	X	
tacacs-server optional-passwords	X	X	
tacacs-server retransmit	X	X	X
tacacs-server timeout	X	X	X

Note Many original TACACS and extended TACACS commands cannot be used after you have initialized AAA/TACACS+. To identify which commands can be used with the three versions, refer to Table 4-1.

Configuring AAA Access Control with TACACS+

To enable the AAA access control model that includes TACACS+, use the following global configuration command:

Command	Purpose
aaa new-model	Enables the AAA access control model.

Configuring AAA Accounting

To enable the AAA accounting of requested services for billing or security purposes when using TACACS+, perform the following steps in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **aaa accounting system**	Performs accounting for all system-level events not associated with users, such as reloads.
Step 2	Switch(config)# **aaa accounting network**	Runs accounting for all network-related service requests, including SLIP, PPP, PPP NCPs, and ARAP.
Step 3	Switch(config)# **aaa accounting connection**	Runs accounting for outbound Telnet and rlogin.
Step 4	Switch(config)# **aaa accounting exec**	Runs accounting for Execs (user shells). This keyword might return user profile information such as **autocommand** information.
Step 5	Switch(config)# **aaa accounting commands** *level*	Runs accounting for all commands at the specified privilege level.

Configuring TACACS Server

Refer to the *Security Configuration Guide* for details about the TACACS configuration tasks that include:

- Setting the number of login attempts allowed to the TACACS server
- Enabling extended TACACS mode
- Configuring a TACACS host

Configuring PPP Authentication

Refer to the *Dial Solutions Configuration Guide* for details about the PPP Authentication configuration tasks that include:

- Enabling Challenge Handshake Authentication Protocol (CHAP) or Password Authentication Protocol (PAP)
- Enabling an AAA authentication method on an interface

Testing the System Management Functions

This section describes the commands used to monitor and display the system management functions.

Displaying Active Processes

To display information about the active processes, use the following privileged EXEC commands:

Command	Purpose
show processes	Displays active process statistics.
show processes cpu	Displays active process CPU utilization.
show processes memory	Displays active process memory utilization.

Displaying Protocols

To display the configured protocols, use the following privileged EXEC command:

Command	Purpose
show protocols *type card*/*subcard*/*port*	Displays the global and interface-specific status of any configured Level 3 protocol; for example, IP, DECnet, Internet Packet Exchange (IPX), and AppleTalk.

Displaying Stacks

To monitor the stack utilization of processes and interrupt routines, use the following privileged EXEC command:

Command	Purpose
show stacks *number*	Displays system stack trace information.

The **show stacks** display includes the reason for the last system reboot. If the system was reloaded because of a system failure, a saved system stack trace is displayed. This information is of use only to Cisco engineers analyzing crashes in the field. It is included here in case you need to read the displayed statistics to an engineer over the phone.

Displaying Routes

To discover the IP routes that the ATM switch router packets will actually take when traveling to their destination, use the following EXEC command:

Command	Purpose
traceroute [*protocol*] [*destination*]	Displays packets through the network.

Displaying Environment

To display temperature and voltage information on the ATM switch router console, use the following EXEC command:

Command	Purpose
show environment	Displays temperature and voltage information.

Checking Basic Connectivity (Catalyst 8540 MSR)

To diagnose basic ATM network connectivity on the Catalyst 8540 MSR, use the following privileged EXEC command:

Command	Purpose
ping atm interface atm *card*/*subcard*/*port* *vpi* [*vci*] {**end-loopback** [*destination*] | **ip-address** *ip-address* | **seg-loopback** [*destination*]}	Uses **ping** to check the ATM network connection.

Checking Basic Connectivity (Catalyst 8510 MSR and LightStream 1010)

To diagnose basic ATM network connectivity on the Catalyst 8510 MSR and LightStream 1010 ATM switch routers, use the following privileged EXEC command:

Command	Purpose
ping atm interface atm *card*/*subcard*/*port* *vpi* [*vci*] {**atm-prefix** *prefix* | **end-loopback** [*destination*] | **ip-address** *ip-address* | **seg-loopback** [*destination*]}	Uses **ping** to check the ATM network connection.

CHAPTER 5

Configuring ATM Network Interfaces

This chapter describes how to explicitly configure ATM network interface types. Explicitly configuring interfaces is the alternative to Integrated Local Management Interface (ILMI) autoconfiguration, which senses the peer interface type and appropriately configures the interface on the ATM switch router.

Note This chapter provides advanced configuration instructions for the Catalyst 8540 MSR, Catalyst 8510 MSR, and LightStream 1010 ATM switch routers. For a discussion and examples of ATM network interface types, refer to the *Guide to ATM Technology*. For complete descriptions of the commands mentioned in this chapter, refer to the *ATM Switch Router Command Reference* publication.

The network configuration tasks described in this chapter are used to explicitly change your ATM switch router operation from the defaults, which are suitable for most networks. The following sections are included:

Disabling Autoconfiguration

Autoconfiguration determines an interface type when the interface initially comes up. To change the configuration of the interface type (such as UNI, NNI, or IISP), side, or version, you must first disable autoconfiguration.

Note When you change the interface type, side, or version, ATM signalling and ILMI are restarted on the interface. When ATM signalling is restarted, all switched virtual connections (SVCs) across the interface are cleared; permanent virtual connections are not affected.

To disable autoconfiguration on an interface, perform the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **interface atm** *card*/*subcard*/*port*[**.***vpt#*] Switch(config-if)#	Selects the interface to be configured.
Step 2	Switch(config-if)# **no atm auto-configuration**	Disables autoconfiguration on the interface.

Example

The following example shows how to disable autoconfiguration on interface ATM 1/0/0:

```
Switch(config)# interface atm 1/0/0
Switch(config-if)# no atm auto-configuration
Switch(config-if)#
%ATM-6-ILMINOAUTOCFG: ILMI(ATM1/0/0): Auto-configuration is disabled, current interface
parameters will be used at next interface restart.
```

Displaying the Autoconfiguration

To confirm that autoconfiguration is disabled for the interface, use the following EXEC command:

Command	Purpose
show atm interface atm *card*/*subcard*/*port*	Shows the ATM interface configuration.

Example

The following example shows the autoconfiguration status of ATM interface 1/0/0 as disabled:

```
Switch# show atm interface atm 1/0/0

Interface:      ATM1/0/0        Port-type:      oc3suni
IF Status:      UP              Admin Status:   up
Auto-config:    disabled        AutoCfgState:   not applicable
IF-Side:        Network         IF-type:        NNI
Uni-type:       not applicable  Uni-version:    not applicable
Max-VPI-bits:   8               Max-VCI-bits:   14
Max-VP:         255             Max-VC:         16383
ConfMaxSvpcVpi: 255             CurrMaxSvpcVpi: 255
ConfMaxSvccVpi: 255             CurrMaxSvccVpi: 255
ConfMinSvccVci: 35              CurrMinSvccVci: 35
Svc Upc Intent: pass            Signalling:     Enabled
ATM Address for Soft VC: 47.0091.8100.0000.00e0.4fac.b401.4000.0c80.8000.00
Configured virtual links:
  PVCLs SoftVCLs   SVCLs   TVCLs   PVPLs SoftVPLs   SVPLs Total-Cfgd Inst-Conns
      4        0       0       0       1        0       0          5          3
Logical ports(VP-tunnels): 0
Input cells:    263250          Output cells:   269783
5 minute input rate:             0 bits/sec,       0 cells/sec
5 minute output rate:            0 bits/sec,       0 cells/sec
Input AAL5 pkts: 171880, Output AAL5 pkts: 175134, AAL5 crc errors: 0
```

Configuring UNI Interfaces

The User-Network Interface (UNI) specification defines communications between ATM end stations (such as workstations and routers) and ATM switches in private ATM networks.

To configure a UNI interface, perform the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **interface atm** *card*/*subcard*/*port*[**.***vpt#*] Switch(config-if)#	Selects the interface to be configured.
Step 2	Switch(config-if)# **no atm auto-configuration**	Disables autoconfiguration on the interface.
Step 3	Switch(config-if)# **atm uni** [**side** {**network** | **user**}] [**type** {**private** | **public**}] [**version** {**3.0** | **3.1** | **4.0**}]	Configures the ATM UNI interface.

Example

The following example shows how to disable autoconfiguration on ATM interface 0/1/0 and configure the interface as the user side of a private UNI running version 4.0:

```
Switch(HB-1)(config)# interface atm 0/1/0
Switch(HB-1)(config-if)# no atm auto-configuration
Switch(HB-1)(config-if)#
%ATM-6-ILMINOAUTOCFG: ILMI(ATM0/1/0): Auto-configuration is disabled, current interface
parameters will be used at next interface restart.
Switch(HB-1)(config-if)# atm uni side user type private version 4.0
Switch(HB-1)(config-if)#
%ATM-5-ATMSOFTSTART: Restarting ATM signalling and ILMI on ATM0/1/0.
```

Displaying the UNI Interface Configuration

To show the UNI configuration for an ATM interface, use the following EXEC command:

Command	Purpose
show atm interface atm *card*/*subcard*/*port*[**.***vpt#*]	Shows the ATM interface configuration.

Example

The following example shows the ATM interface 0/1/0 UNI configuration:

```
Switch(HB-1)# show atm interface atm 0/1/0

Interface:      ATM0/1/0       Port-type:      oc3suni
IF Status:      UP             Admin Status:   up
Auto-config:    disabled       AutoCfgState:   not applicable
IF-Side:        Network        IF-type:        UNI
Uni-type:       private        Uni-version:    V4.0
<information deleted)
```

Configuring NNI Interfaces

The Network-Network Interface (NNI) specification defines communications between two ATM switches in a private ATM network.

You must configure NNI connections to allow for route discovery and topology analysis between the ATM switch routers. To configure the NNI interface, perform the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **interface atm** *card*/*subcard*/*port*[**.***vpt#*] Switch(config-if)#	Selects the interface to be configured.
Step 2	Switch(config-if)# **no atm auto-configuration**	Disables autoconfiguration on the interface.
Step 3	Switch(config-if)# **atm nni**	Configures the ATM NNI interface.

Example

The following example shows how to configure ATM interface 3/0/0 as an NNI interface:

```
Switch(HB-1)(config)# interface atm 3/0/0
Switch(HB-1)(config-if)# no atm auto-configuration
Switch(HB-1)(config-if)#
%ATM-6-ILMINOAUTOCFG: ILMI(ATM3/0/0): Auto-configuration is disabled, current interface
parameters will be used at next interface restart.
Switch(HB-1)(config-if)# atm nni
Switch(HB-1)(config-if)#
%ATM-5-ATMSOFTSTART: Restarting ATM signalling and ILMI on ATM3/0/0.
```

Displaying the NNI Interface Configuration

To show the NNI configuration for an ATM interface, use the following EXEC command:

Command	Purpose
show atm interface atm *card*/*subcard*/*port*[**.***vpt#*]	Shows the ATM interface configuration.

Example

The following example shows the configuration of the NNI interface ATM 3/0/0 on the ATM switch router-1 (HB-1) located in the headquarters building:

```
Switch(HB-1)# show atm interface atm 3/0/0

Interface:      ATM3/0/0        Port-type:    oc3suni
IF Status:      UP              Admin Status: up
Auto-config:    disabled        AutoCfgState: not applicable
→ IF-Side:      Network         IF-type:      NNI
Uni-type:       not applicable  Uni-version:  not applicable

<information deleted>
```

Configuring a 12-Bit VPI NNI Interface (Catalyst 8540 MSR)

The Catalyst 8540 MSR ATM switch router can accommodate up to six interfaces per module for maxvpi-bits greater than the standard 8-bit configuration. If you try to configure more than the maximum number of allowed interfaces with 12-bit virtual path identifiers (VPIs), follow these precautions:

- When you must remove an interface (for example, hot-swapping a port adapter) that is configured for a maxvpi-bit, the number of interfaces (with maxvpi-bit value greater than 8) on the module is decremented. This allows you to then configure other interfaces on the same module for maxvpi-bits greater than eight bits.
- If a port adapter with interfaces configured with a maxvpi-bits value of eight is reinserted into a module location that previously held a port adapter with maxvpi-bits greater than eight bits, the VCs with VPIs greater than 255 remain in "No HW RESOURCES" state. An interface can be reconfigured to maxvpi-bits greater than eight, by changing the value to less than or equal to eight bits on a different interface. The VCs can be restored from "No HW RESOURCES" state by toggling the interface state using the **shutdown** and **no shutdown** commands.

When you need a 12-bit VPI range greater than 255, change the maximum VPI bits configuration. Perform the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **interface atm** *card*/*subcard*/*port* Switch(config-if)#	Selects the interface to be configured.
Step 2	Switch(config-if)# **no atm auto-configuration**	Disables autoconfiguration on the interface.
Step 3	Switch(config-if)# **atm nni**	Configures the ATM NNI interface.
Step 4	Switch(config-if)# **atm maxvpi-bits** *max-vpi-bits*	Modifies the maximum VPI bits configuration.

Note 12-bit VPI support is only available on ATM NNI interfaces.

Example

The following example shows that if you are unable to configure a port with a maximum 12-bit VPI value greater than 8, you receive a message prompting you to reconfigure the port:

```
Switch(config)# interface atm 0/0/0
Switch(config-if)# no atm auto-configuration
Switch(config-if)# atm nni
Switch(config-if)# atm maxvpi-bits 12

This port can not be configured for vpi bits greater than 8, unless one
of the following ports is reconfigured for 8 bits vpi

interface a11/0/0
interface a11/0/1
interface a11/0/2
interface a11/0/3
interface a12/0/0
interface a12/0/1
```

Displaying the 12-Bit VPI NNI Interface Configuration (Catalyst 8540 MSR)

To display the 12-bit VPI NNI interface configuration, use the following EXEC commands:

Command	Purpose
show switch module interface atm *card/subcard/port*	Displays the maxvpi-bits for the specified ATM interface.
show atm interface atm *card/subcard/port*	Shows the ATM interface configuration.

Examples

The following example shows the maxvpi-bits for interface ATM 0/0/0:

```
Switch# show switch module interface atm 0/0/0
Module ID  Interface  Maxvpi-bits  State
----------------------------------------
0            ATM0/0/0   8            UP
             ATM0/0/4   8          DOWN
             ATM0/0/1   8          DOWN
             ATM0/0/5   8          DOWN
             ATM0/0/2   8            UP
             ATM0/0/6   8          DOWN
             ATM0/0/3   8            UP
             ATM0/0/7   8          DOWN
========================================
```

The following example shows how to display the configuration information for interface ATM 0/0/0:

```
Switch# show atm interface atm 0/0/0

Interface:      ATM0/0/0        Port-type:      oc3suni
IF Status:      DOWN            Admin Status:   down
Auto-config:    enabled         AutoCfgState:   waiting for response from peer
IF-Side:        Network         IF-type:        UNI
Uni-type:       Private         Uni-version:    V3.0
Max-VPI-bits:   8               Max-VCI-bits:   14
Max-VP:         255             Max-VC:         16383
ConfMaxSvpcVpi: 100             CurrMaxSvpcVpi: 100
ConfMaxSvccVpi: 100             CurrMaxSvccVpi: 100
ConfMinSvccVci: 60              CurrMinSvccVci: 60
Svc Upc Intent: pass            Signalling:     Enabled
ATM Address for Soft VC: 47.0091.8100.0000.0040.0b0a.2a81.4000.0c80.0000.00
Configured virtual links:
  PVCLs SoftVCLs   SVCLs   TVCLs   PVPLs SoftVPLs   SVPLs Total-Cfgd Inst-Conns
      3        0       0       0       0        0       0          3          0
Logical ports(VP-tunnels):     0
Input cells:    0               Output cells:   0
5 minute input rate:             0 bits/sec,       0 cells/sec
5 minute output rate:            0 bits/sec,       0 cells/sec
Input AAL5 pkts: 0, Output AAL5 pkts: 0, AAL5 crc errors: 0
```

Configuring IISP Interfaces

The Interim Interswitch Signalling Protocol (IISP) defines a static routing protocol for use between ATM switches. IISP provides support for switched virtual connections (SVCs) on switches that do not support the Private Network-Network Interface (PNNI) protocol. For further information, see Chapter 10, "Configuring ATM Routing and PNNI."

To configure an IISP interface, perform the following tasks, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **interface atm** *card*/*subcard*/*port*[**.***vpt#*] Switch(config-if)#	Selects the interface to be configured.
Step 2	Switch(config-if)# **no atm auto-configuration**	Disables autoconfiguration on the interface.
Step 3	Switch(config-if)# **atm iisp** [**side** {**network** | **user**}] [**version** {**3.0** | **3.1** | **4.0**}]	Configures the ATM IISP interface.
Step 4	Switch(config-if)# **exit** Switch(config)#	Exits interface configuration mode.
Step 5	Switch(config)# **atm route** *addr-prefix* **atm** *card*/*subcard*/*port*[**.***subinterface#*]	Configures the ATM route address prefix.

Example

The following example shows how to configure ATM interface 3/0/0 on the ATM switch router (SB-1) as user side IISP and specifies an ATM route address prefix:

```
Switch(SB-1)(config)# interface atm 3/0/0
Switch(SB-1)(config-if)# no atm auto-configuration
Switch(SB-1)(config-if)#
%ATM-6-ILMINOAUTOCFG: ILMI(ATM3/0/0): Auto-configuration is disabled, current interface
parameters will be used at next interface restart.
Switch(SB-1)(config-if)# atm iisp side user
Switch(SB-1)(config-if)#
%ATM-5-ATMSOFTSTART: Restarting ATM signalling and ILMI on ATM3/0/0.
Switch(SB-1)(config-if)# exit
Switch(SB-1)(config)# atm route 47.0091.8100.0000.0000.0ca7.ce01 atm 3/0/0
```

Displaying the IISP Configuration

To show the interface IISP configuration, use the following EXEC command:

Command	Purpose
show atm interface atm *card*/*subcard*/*port*[**.***vpt#*]	Shows the interface configuration.

Example

The following example shows the configuration of ATM interface 3/0/0 on the ATM switch router (SB-1):

```
Switch(SB-1)# show atm interface atm 3/0/0

Interface:      ATM3/0/0        Port-type:      oc3suni
IF Status:      UP              Admin Status:   up
Auto-config:    disabled        AutoCfgState:   not applicable
IF-Side:        User            IF-type:        IISP
Uni-type:       not applicable  Uni-version:    V3.0
Max-VPI-bits:   8               Max-VCI-bits:   14
Max-VP:         255             Max-VC:         16383
ConfMaxSvpcVpi: 255             CurrMaxSvpcVpi: 255
ConfMaxSvccVpi: 255             CurrMaxSvccVpi: 255
ConfMinSvccVci: 35              CurrMinSvccVci: 35
Svc Upc Intent: pass            Signalling:     Enabled
ATM Address for Soft VC: 47.0091.8100.0000.00e0.4fac.b401.4000.0c80.8000.00
Configured virtual links:
  PVCLs SoftVCLs   SVCLs   TVCLs   PVPLs SoftVPLs   SVPLs Total-Cfgd Inst-Conns
      3        0       0       0       0        0       0          3          2
Logical ports(VP-tunnels):     0
Input cells:    264089          Output cells:   273253
5 minute input rate:             0 bits/sec,       0 cells/sec
5 minute output rate:            0 bits/sec,       0 cells/sec
Input AAL5 pkts: 172421, Output AAL5 pkts: 176993, AAL5 crc errors: 0
```

CHAPTER 6

Configuring Virtual Connections

This chapter describes how to configure virtual connections (VCs) in a typical ATM network after autoconfiguration has established the default network connections. The network configuration modifications described in this chapter are used to optimize your ATM network operation.

Note This chapter provides advanced configuration instructions for the Catalyst 8540 MSR, Catalyst 8510 MSR, and LightStream 1010 ATM switch routers. For an overview of virtual connection types and applications, refer to the *Guide to ATM Technology*. For complete descriptions of the commands mentioned in this chapter, refer to the *ATM Switch Router Command Reference* publication.

The tasks to configure virtual connections are described in the following sections:

Characteristics and Types of Virtual Connections

This section lists the various virtual connections (VC) types in Table 6-1.

Table 6-1 Supported VC Types

Connection	Point-to-Point	Point-to-Multipoint	Transit	Terminate
Permanent virtual channel link (PVCL)	x	x	—	—
Permanent virtual path link (PVPL)	x	x	—	—
Permanent virtual channel (PVC)	x	x	x	x
Permanent virtual path (PVP)	x	x	x	—
Soft permanent virtual channel (Soft PVC)	x	—	x	—
Soft permanent virtual path (Soft PVP)	x	—	x	—
Switched virtual channel (SVC)	x	x	x	x
Switched virtual path (SVP)	x	x	x	—

Configuring Virtual Channel Connections

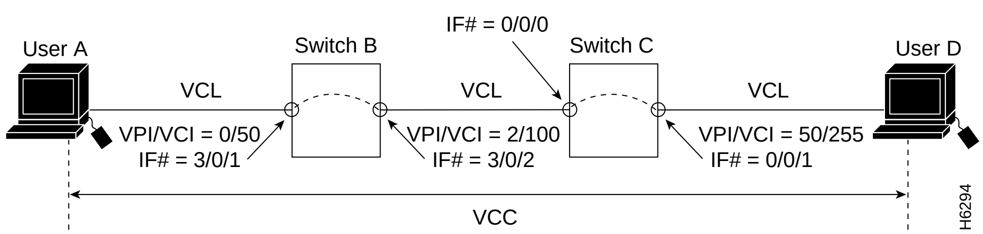
This section describes configuring virtual channel connections (VCCs) on the ATM switch router. A VCC is established as a bidirectional facility to transfer ATM traffic between two ATM layer users. Figure 6-1 shows an example VCC between ATM user A and user D.

An end-to-end VCC, as shown in Figure 6-1 between user A and user D, has two parts:

- Virtual channel links, labelled VCL. These are the interconnections between switches, either directly or through VP tunnels.
- Internal connections, shown by the dotted line in the switch. These connections are also sometimes called cross-connections or cross-connects.

The common endpoint between an internal connection and a link occurs at the switch interface. The endpoint of the internal connection is also referred to as a *connection leg* or *half-leg*. A cross-connect connects two legs together.

Figure 6-1 VCC Example



Note The value of the VPIs and VCIs can change as the traffic is relayed through the ATM network.

To configure a point-to-point VCC, perform the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **interface atm** *card*/*subcard*/*port* Switch(config-if)#	Selects the interface to be configured.
Step 2	Switch(config-if)# **atm pvc** *vpi-A* [*vci-A* | **any-vci**[1]] [**rx-cttr** *index*] [**tx-cttr** *index*] **interface atm** *card*/*subcard*/*port*[**.***vpt#*] *vpi-B* [*vci*-B | **any-vci**[1]]	Configures the PVC.

1. The **any-vci** parameter is only available for interface atm0.

Note The row index for **rx-cttr** and **tx-cttr** must be configured before using this optional parameter. See the "Configuring the Connection Traffic Table" section on page 8-10.

Note When configuring PVC connections, begin with lower VCI numbers. Using low VCI numbers allows more efficient use of the switch fabric resources.

Examples

The following example shows how to configure the internal cross-connect PVC on Switch B between interface ATM 3/0/1 (VPI = 0, VCI = 50) and interface ATM 3/0/2 (VPI = 2, VCI = 100) (see Figure 6-1):

```
Switch-B(config)# interface atm 3/0/1
Switch-B(config-if)# atm pvc 0 50 interface atm 3/0/2 2 100
```

The following example shows how to configure the internal cross-connect PVC on Switch C between interface ATM 0/0/0, VPI = 2, VCI = 100, and interface ATM 0/0/1, VPI 50, VCI = 255:

```
Switch-C(config)# interface atm 0/0/0
Switch-C(config-if)# atm pvc 2 100 interface atm 0/0/1 50 255
```

Each subsequent VC cross-connection and link must be configured until the VC is terminated to create the entire VCC.

Note The above examples show how to configure cross-connections using one command. This is the preferred method, but it is also possible to configure each leg separately, then connect them with the **atm pvc** *vpi vci* **interface atm** *card*/*subcard*/*port vpi vci* command. This alternative method requires more steps, but might be convenient if each leg has many additional configuration parameters or if you have configured individual legs with SNMP commands and you want to connect them with one CLI command.

Displaying VCCs

To show the VCC configuration, use the following EXEC commands:

Command	Purpose
show atm interface [**atm** *card*/*subcard*/*port*]	Shows the ATM interface configuration.
show atm vc [**interface atm** *card*/*subcard*/*port* *vpi vci*]	Shows the PVC interface configuration.

Note The following examples differ depending on the feature card installed on the processor.

Examples

The following example shows the Switch B PVC configuration on ATM interface 3/0/1:

```
Switch-B# show atm interface

Interface:      ATM3/0/1        Port-type:    oc3suni
IF Status:      UP              Admin Status:   up
Auto-config:    enabled         AutoCfgState:   completed
IF-Side:        Network         IF-type:        NNI
Uni-type:       not applicable  Uni-version:    not applicable
Max-VPI-bits:   8               Max-VCI-bits:   14
Max-VP:         255             Max-VC:         16383
ConfMaxSvpcVpi: 255             CurrMaxSvpcVpi: 255
ConfMaxSvccVpi: 255             CurrMaxSvccVpi: 255
ConfMinSvccVci: 35              CurrMinSvccVci: 35
Svc Upc Intent: pass            Signalling:     Enabled
ATM Address for Soft VC: 47.0091.8100.0000.00e0.4fac.b401.4000.0c80.8000.00
Configured virtual links:
  PVCLs SoftVCLs   SVCLs   TVCLs   PVPLs SoftVPLs   SVPLs Total-Cfgd Inst-Conns
      4        0       0       0       0        0       0          4          2
Logical ports(VP-tunnels):     0
Input cells:    264330          Output cells:   273471
5 minute input rate:             0 bits/sec,       0 cells/sec
5 minute output rate:            0 bits/sec,       0 cells/sec
Input AAL5 pkts: 172613, Output AAL5 pkts: 177185, AAL5 crc errors: 0
```

The following example shows the Switch B PVC configuration on ATM interface 3/0/1:

```
Switch-B# show atm vc interface atm 3/0/1
Interface    VPI   VCI   Type    X-Interface  X-VPI X-VCI  Encap Status
ATM3/0/1     0     5      PVC     ATM0          0     57    QSAAL  UP
ATM3/0/1     0     16     PVC     ATM0          0     37    ILMI   UP
ATM3/0/1     0     18     PVC     ATM0          0     73    PNNI   UP
ATM3/0/1     0     50     PVC     ATM3/0/2      2     100          UP
ATM3/0/1     1     50     PVC     ATM0          0     80    SNAP   UP
```

The following example shows the Switch B PVC configuration on ATM interface 3/0/1, VPI = 0, VCI = 50, with the switch processor feature card installed:

```
Switch-B# show atm vc interface atm 3/0/1 0 50

Interface: ATM3/0/1, Type: oc3suni
VPI = 0  VCI = 50
Status: UP
Time-since-last-status-change: 4d02h
Connection-type: PVC
Cast-type: point-to-point
Packet-discard-option: disabled
Usage-Parameter-Control (UPC): pass
Wrr weight: 32
Number of OAM-configured connections: 0
OAM-configuration: disabled
OAM-states:  Not-applicable
Cross-connect-interface: ATM3/0/2, Type: oc3suni
Cross-connect-VPI = 2
Cross-connect-VCI = 100
Cross-connect-UPC: pass
Cross-connect OAM-configuration: disabled
Cross-connect OAM-state:  Not-applicable
Threshold Group: 5, Cells queued: 0
Rx cells: 0, Tx cells: 0
Tx Clp0:0,  Tx Clp1: 0
Rx Clp0:0,  Rx Clp1: 0
Rx Upc Violations:0, Rx cell drops:0
Rx Clp0 q full drops:0, Rx Clp1 qthresh drops:0
Rx connection-traffic-table-index: 1
Rx service-category: UBR (Unspecified Bit Rate)
Rx pcr-clp01: 7113539
Rx scr-clp01: none
Rx mcr-clp01: none
Rx      cdvt: 1024 (from default for interface)
Rx       mbs: none
Tx connection-traffic-table-index: 1
Tx service-category: UBR (Unspecified Bit Rate)
Tx pcr-clp01: 7113539
Tx scr-clp01: none
Tx mcr-clp01: none
Tx      cdvt: none
Tx       mbs: none
```

Deleting VCCs from an Interface

This section describes how to delete a VCC configured on an interface. To delete a VCC, perform the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **interface atm** *card*/*subcard*/*port* Switch(config-if)#	Selects the interface to be configured.
Step 2	Switch(config-if)# **no atm pvc** *vpi vci*	Deletes the PVC.

Example

The following example shows how to delete the VCC on ATM interface 3/0/0, VPI = 20, VCI = 200:

```
Switch(config-if)# interface atm 3/0/0
Switch(config-if)# no atm pvc 20 200
```

Confirming VCC Deletion

To confirm the deletion of a VCC from an interface, use the following EXEC command before and after deleting the VCC:

Command	Purpose
show atm vc interface atm *card*/*subcard*/*port* [*vpi vci*]	Shows the PVCs configured on the interface.

Example

The following example shows how to confirm that the VCC is deleted from the interface:

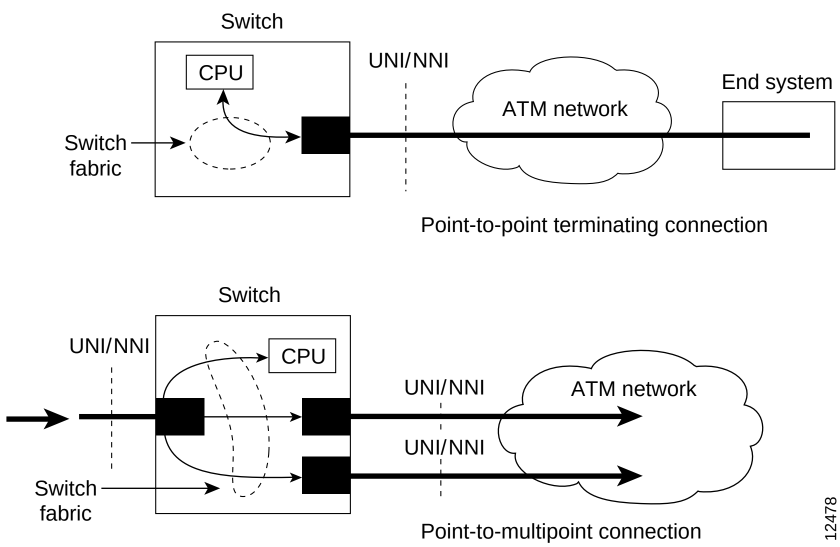
```
Switch# show atm vc interface atm 3/0/0
Interface         VPI  VCI   Type   X-Interface     X-VPI X-VCI Encap  Status
ATM3/0/0          0    5     PVC    ATM2/0/0         0    77    QSAAL  UP
ATM3/0/0          0    16    PVC    ATM2/0/0         0    55    ILMI   UP
ATM3/0/0          0    18    PVC    ATM2/0/0         0    152   PNNI   UP
➔ ATM3/0/0        0    34    PVC    ATM2/0/0         0    151   NCDP   UP
➔ ATM3/0/0        20   200   PVC    ATM1/1/1         10   100          DOWN
Switch# configure terminal
Switch(config)# interface atm 3/0/0
Switch(config-if)# no atm pvc 20 200
Switch(config-if)# end
Switch# show atm vc interface atm 3/0/0
Interface         VPI  VCI   Type   X-Interface     X-VPI X-VCI Encap  Status
ATM3/0/0          0    5     PVC    ATM2/0/0         0    77    QSAAL  UP
ATM3/0/0          0    16    PVC    ATM2/0/0         0    55    ILMI   UP
ATM3/0/0          0    18    PVC    ATM2/0/0         0    152   PNNI   UP
➔ ATM3/0/0        0    34    PVC    ATM2/0/0         0    151   NCDP   UP
```

Configuring Terminating PVC Connections

This section describes configuring point-to-point and point-to-multipoint terminating permanent virtual channel (PVC) connections. Terminating connections provide the connection to the ATM switch router's route processor for LAN emulation (LANE), IP over ATM, and control channels for Integrated Local Management Interface (ILMI), signalling, and Private Network-Network Interface (PNNI) plus network management.

Figure 6-2 shows an example of transit and terminating connections.

Figure 6-2 Terminating PVC Types



Point-to-point and point-to-multipoint are two types of terminating connections. Both terminating connections are configured using the same commands as transit connections (discussed in the previous sections). However, all switch terminating connections use interface atm0 to connect to the route processor.

Note Since release 12.0(1a)W5(5b) of the system software, addressing the interface on the processor (CPU) has changed. The ATM interface is now called atm0, and the Ethernet interface is now called ethernet0. The old formats (atm 2/0/0 and ethernet 2/0/0) are still supported.

To configure both point-to-point and point-to-multipoint terminating PVC connections, perform the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **interface atm** *card-A/subcard-A/port-A*[**.***vpt#*] Switch(config-if)#	Selects the interface to be configured.
Step 2	Switch(config-if)# **atm pvc** *vpi-A* [*vci-A* | **any-vci**[1]] [**cast-type** *type*] [**rx-cttr** *index*] [**tx-cttr** *index*] **interface atm** *card-B/subcard-B/port-B*[**.***vpt#*] *vpi-B* [*vci-B* | **any-vci**[1]] [**encap** *type*] [**cast-type** *type*]	Configures the PVC between ATM switch router connections.

1. The any-vci feature is only available for interface atm 0.

When configuring point-to-multipoint PVC connections using the **atm pvc** command, the root point is port A and the leaf points are port B.

Note The row index for **rx-cttr** and **tx-cttr** must be configured before using this optional parameter. See the "Configuring the Connection Traffic Table" section on page 8-10.

Examples

The following example shows how to configure the internal cross-connect PVC between interface ATM 3/0/1, VPI = 1, VCI = 50, and the terminating connection at the route processor interface ATM 0, VPI = 0, and VCI unspecified:

```
Switch-B(config)# interface atm 3/0/1
Switch-B(config-if)# atm pvc 1 50 interface atm0 0 any-vci encap aal5snap
```

The following example shows how to configure the route processor leg of any terminating PVC:

```
Switch(config)# interface atm0
Switch(config-if)# atm pvc 0 any-vci
```

When configuring the route processor leg of a PVC that is not a tunnel, the VPI should be configured as 0. The preferred method of VCI configuration is to select the **any-vci** parameter, unless a specific VCI is needed as a parameter in another command, such as **map-list**.

Note If configuring a specific VCI value for the route processor leg, select a VCI value higher than 300 to prevent a conflict with an automatically assigned VCI for well-known channels if the ATM switch router reboots.

Displaying the Terminating PVC Connections

To display the terminating PVC configuration VCs on the interface, use the following EXEC command:

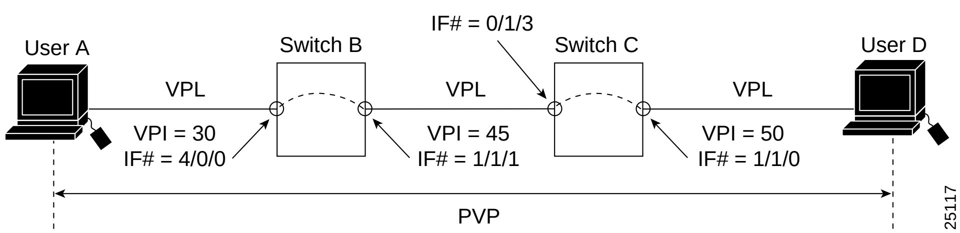
Command	Purpose
show atm vc interface atm *card/subcard/port vpi vci*	Shows the PVC configured on the interface.

See the "Displaying VCCs" section on page 6-4 for examples of the **show atm vc** commands.

Configuring PVP Connections

This section describes configuring a permanent virtual path (PVP) connection. Figure 6-3 shows an example of PVPs configured through the ATM switch routers.

Figure 6-3 Virtual Path Connection Example



To configure a PVP connection, perform the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **interface atm** *card/subcard/port* Switch(config-if)#	Selects the physical interface to be configured.
Step 2	Switch(config-if)# **atm pvp** *vpi-A* [**rx-cttr** *index*] [**tx-cttr** *index*] **interface atm** *card/subcard/port vpi-B*	Configures the interface PVP.

Note The row index for **rx-cttr** and **tx-cttr** must be configured before using this optional parameter. See the "Configuring the Connection Traffic Table" section on page 8-10.

Note When configuring PVP connections, begin with lower virtual path identifier (VPI) numbers. Using low VPI numbers allows more efficient use of the switch fabric resources.

Examples

The following example shows how to configure the internal cross-connect PVP within Switch B between interfaces 4/0/0, VPI = 30, and interface ATM 1/1/1, VPI = 45:

```
Switch-B(config)# interface atm 4/0/0
Switch-B(config-if)# atm pvp 30 interface atm 1/1/1 45
```

The following example shows how to configure the internal cross-connect PVP within Switch C between interfaces 0/1/3, VPI = 45, and interface ATM 1/1/0, VPI = 50:

```
Switch-C(config)# interface atm 0/1/3
LS1010(config-if)# atm pvp 45 interface atm 1/1/0 50
```

Each subsequent PVP cross connection and link must be configured until the VP is terminated to create the entire PVP.

Displaying PVP Configuration

To show the ATM interface configuration, use the following EXEC command:

Command	Purpose
show atm vp [**interface atm** *card*/*subcard*/*port vpi*]	Shows the ATM VP configuration.

Example

The following example shows the PVP configuration of Switch B:

```
Switch-B# show atm vp
Interface    VPI    Type  X-Interface     X-VPI     Status
ATM1/1/1     45      PVP     ATM4/0/0     30        UP
ATM4/0/0     30      PVP     ATM1/1/1     45        UP
```

The following example shows the PVP configuration of Switch B with the switch processor feature card installed:

```
Switch-B# show atm vp interface atm 4/0/0 30

Interface: ATM4/0/0, Type: ds3suni
VPI = 30
Status: UP
Time-since-last-status-change: 00:09:02
Connection-type: PVP
Cast-type: point-to-point
Usage-Parameter-Control (UPC): pass
Wrr weight: 2
Number of OAM-configured connections: 0
OAM-configuration: disabled
OAM-states:  Not-applicable
Cross-connect-interface: ATM1/1/1, Type: oc3suni
Cross-connect-VPI = 45
Cross-connect-UPC: pass
Cross-connect OAM-configuration: disabled
Cross-connect OAM-state:  Not-applicable
Threshold Group: 5, Cells queued: 0
Rx cells: 0, Tx cells: 0
Tx Clp0:0,  Tx Clp1: 0
```

```
Rx Clp0:0,  Rx Clp1: 0
Rx Upc Violations:0, Rx cell drops:0
Rx Clp0 q full drops:0, Rx Clp1 qthresh drops:0
Rx connection-traffic-table-index: 1
Rx service-category: UBR (Unspecified Bit Rate)
Rx pcr-clp01: 7113539
Rx scr-clp01: none
Rx mcr-clp01: none
Rx      cdvt: 1024 (from default for interface)
Rx       mbs: none
Tx connection-traffic-table-index: 1
Tx service-category: UBR (Unspecified Bit Rate)
Tx pcr-clp01: 7113539
Tx scr-clp01: none
Tx mcr-clp01: none
Tx      cdvt: none
Tx       mbs: none
```

Deleting PVPs from an Interface

This section describes how to delete a PVP configured on an interface. To delete a PVP, perform the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **interface atm** *card*/*subcard*/*port* Switch(config-if)#	Selects the interface to be configured.
Step 2	Switch(config-if)# **no atm pvp** *vpi*	Deletes the PVP.

Example

The following example shows how to delete the PVP on ATM interface 1/1/0, VPI = 200:

```
Switch(config-if)# interface atm 1/1/0
Switch(config-if)# no atm pvp 200
```

Confirming PVP Deletion

To confirm the deletion of a PVP from an interface, use the following EXEC command before and after deleting the PVP:

Command	Purpose
show atm vp interface atm [*card*/*subcard*/*port* *vpi*]	Shows the PVCs configured on the interface.

Example

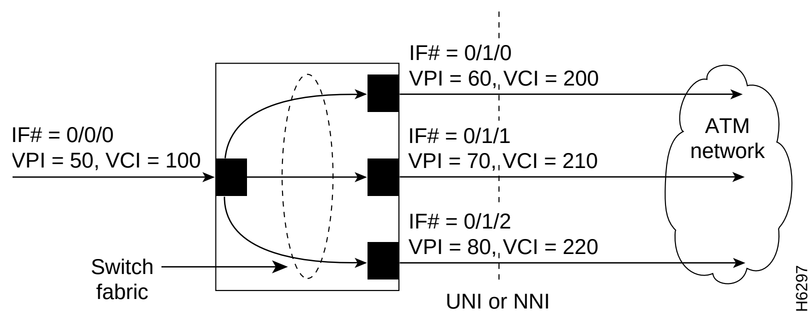
The following example shows how to confirm that the PVP is deleted from the interface:

```
Switch# show atm vp
Interface         VPI  Type  X-InterfaceX-VPI     Status
ATM1/1/0          113  PVP    TUNNEL
➔ ATM1/1/0        200  PVP    ATM1/1/1100 DOWN
ATM1/1/1          1    PVP    SHAPED TUNNEL
➔ ATM1/1/1        100  PVP    ATM1/1/0200 DOWN
Switch# configure terminal
Switch(config)# interface atm 1/1/0
Switch(config-if)# no atm pvp 200
Switch(config-if)# end
Switch# show atm vp
Interface         VPI  Type  X-InterfaceX-VPI     Status
ATM1/1/0          113  PVP    TUNNEL
ATM1/1/1          1    PVP    SHAPED TUNNEL
Switch#
```

Configuring Point-to-Multipoint PVC Connections

This section describes configuring point-to-multipoint PVC connections. In Figure 6-4, cells entering the ATM switch router at the root point (on the left side at interface ATM 0/0/0, VPI = 50, VCI = 100) are duplicated and switched to the leaf points (output interfaces) on the right side of the figure.

Figure 6-4 Point-to-Multipoint PVC Example



Note If desired, one of the leaf points can terminate in the ATM switch router at the route processor interface ATM 0.

To configure the point-to-multipoint PVC connections shown in Figure 6-4, perform the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **interface atm** *card*/*subcard*/*port*[**.***vpt#*] Switch(config-if)#	Selects the interface to be configured.
Step 2	Switch(config-if)# **atm pvc** *vpi-A vci-A* [**cast-type** *type-A*] [**rx-cttr** *index*] [**tx-cttr** *index*] **interface atm** *card*/*subcard*/*port*[**.***vpt#*] *vpi-B vci-B* [**cast-type** *type-B*]	Configures the PVC between ATM switch router connections.

To configure the point-to-multipoint PVC connections using the **atm pvc** command, the root point is port A and the leaf points are port B.

Note The row index for **rx-cttr** and **tx-cttr** must be configured before using this optional parameter. See the "Configuring the Connection Traffic Table" section on page 8-10.

Examples

The following example shows how to configure the root-point PVC on ATM switch router interface ATM 0/0/0, VPI = 50, VCI = 100, to the leaf-point interfaces (see Figure 6-4):

```
Switch(config)# interface atm 0/0/0
Switch(config-if)# atm pvc 50 100 cast-type p2mp-root interface atm 0/1/0 60 200 cast-type p2mp-leaf
Switch(config-if)# atm pvc 50 100 cast-type p2mp-root interface atm 0/1/1 70 210 cast-type p2mp-leaf
Switch(config-if)# atm pvc 50 100 cast-type p2mp-root interface atm 0/1/2 80 220 cast-type p2mp-leaf
```

Displaying Point-to-Multipoint PVC Configuration

To display the point-to-multipoint PVC configuration, use the following EXEC mode command:

Command	Purpose
show atm vc interface atm *card*/*subcard*/*port*	Shows the PVCs configured on the interface.
show atm vc interface atm *card*/*subcard*/*port vpi vci*	Shows the PVCs configured on the interface.

Examples

The following example shows the PVC configuration of the point-to-multipoint connections on ATM interface 0/0/0:

```
Switch# show atm vc interface atm 0/0/0
Interface         VPI  VCI   Type   X-Interface      X-VPI X-VCI Encap  Status
ATM0/0/0          0    5     PVC    ATM2/0/0          0    70    QSAAL  UP
ATM0/0/0          0    16    PVC    ATM2/0/0          0    46    ILMI   UP
ATM0/0/0          0    18    PVC    ATM2/0/0          0    120   PNNI   UP
ATM0/0/0          0    34    PVC    ATM2/0/0          0    192   NCDP   UP
ATM0/0/0          50   100   PVC    ATM0/1/0          60   200          UP
                                    ATM0/1/1          70   210          UP
                                    ATM0/1/2          80   220          UP
```

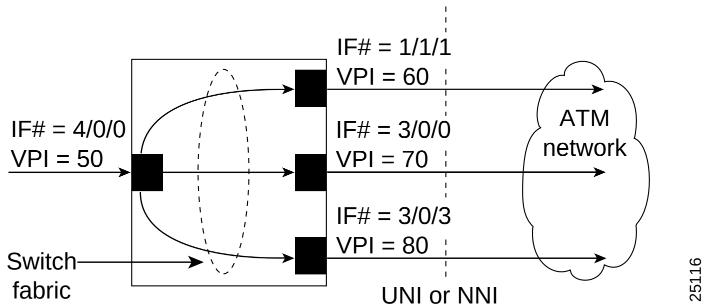
The following example shows the VC configuration on interface ATM 0/0/0, VPI = 50, VCI = 100, with the switch processor feature card installed:

```
Switch# show atm vc interface atm 0/0/0 50 100

Interface: ATM0/0/0, Type: oc3suni
VPI = 50  VCI = 100
Status: UP
Time-since-last-status-change: 00:07:06
Connection-type: PVC
Cast-type: point-to-multipoint-root
Packet-discard-option: disabled
Usage-Parameter-Control (UPC): pass
Wrr weight: 32
Number of OAM-configured connections: 0
OAM-configuration: disabled
OAM-states:  Not-applicable
Cross-connect-interface: ATM0/1/0, Type: oc3suni
Cross-connect-VPI = 60
Cross-connect-VCI = 200
Cross-connect-UPC: pass
Cross-connect OAM-configuration: disabled
Cross-connect OAM-state:  Not-applicable
Cross-connect-interface: ATM0/1/1
Cross-connect-VPI = 70
Cross-connect-VCI = 210
Cross-connect-interface: ATM0/1/2
Cross-connect-VPI = 80
Cross-connect-VCI = 220
Threshold Group: 5, Cells queued: 0
Rx cells: 0, Tx cells: 0
Tx Clp0:0,  Tx Clp1: 0
Rx Clp0:0,  Rx Clp1: 0
Rx Upc Violations:0, Rx cell drops:0
Rx Clp0 q full drops:0, Rx Clp1 qthresh drops:0
Rx connection-traffic-table-index: 1
Rx service-category: UBR (Unspecified Bit Rate)
Rx pcr-clp01: 7113539
Rx scr-clp01: none
Rx mcr-clp01: none
Rx      cdvt: 1024 (from default for interface)
Rx       mbs: none
Tx connection-traffic-table-index: 1
Tx service-category: UBR (Unspecified Bit Rate)
Tx pcr-clp01: 7113539
Tx scr-clp01: none
Tx mcr-clp01: none
Tx      cdvt: none
Tx       mbs: none
```

Configuring Point-to-Multipoint PVP Connections

This section describes configuring point-to-multipoint PVP connections. Figure 6-5 provides an example of point-to-multipoint PVP connections.

Figure 6-5 Point-to-Multipoint PVP Example



In Figure 6-5, cells entering the ATM switch router at the root point (the left side at interface ATM 4/0/0), VPI = 50, are duplicated and switched to the leaf points (output interfaces), on the right side of the figure.

To configure point-to-multipoint PVP connections, perform the following steps, beginning in global configuration mode:

Command	Purpose
interface atm *card-A*/*subcard-A*/*port-A*	Selects the interface to be configured.

To configure the point-to-multipoint PVP connections using the **atm pvp** command, the root point is port A and the leaf points are port B.

Note The row index for **rx-cttr** and **tx-cttr** must be configured before using this optional parameter. See the "Configuring the Connection Traffic Table" section on page 8-10.

Examples

The following example shows how to configure the root-point PVP on ATM switch router interface ATM 4/0/0 (VPI = 50), to the leaf point interfaces ATM 1/1/1 (VPI = 60), ATM 3/0/0 (VPI = 70), and ATM 3/0/3 (VPI = 80) (see Figure 6-5):

```
Switch(config)# interface atm 4/0/0
Switch(config-if)# atm pvp 50 cast-type p2mp-root interface atm 1/1/1 60 cast-type p2mp-leaf
Switch(config-if)# atm pvp 50 cast-type p2mp-root interface atm 3/0/0 70 cast-type p2mp-leaf
Switch(config-if)# atm pvp 50 cast-type p2mp-root interface atm 3/0/3 80 cast-type p2mp-leaf
```

Displaying Point-to-Multipoint PVP Configuration

To display the ATM interface configuration, use the following EXEC command:

Command	Purpose
show atm vp [**interface atm** *card*/*subcard*/*port* *vpi*]	Shows the ATM VP configuration.

Examples

The following example shows the PVP configuration of the point-to-multipoint PVP connections on ATM interface 4/0/0:

```
Switch# show atm vp interface atm 4/0/0
Interface    VPI    Type  X-Interface     X-VPI     Status
ATM4/0/0     50      PVP     ATM1/1/1     60        UP
                             ATM3/0/0     70        UP
                             ATM3/0/3     80        UP
```

The following example shows the PVP configuration of the point-to-multipoint PVP connections on ATM interface 4/0/0, VPI = 50, with the switch processor feature card installed:

```
Switch# show atm vp interface atm 4/0/0 50

Interface: ATM4/0/0, Type: ds3suni
VPI = 50
Status: UP
Time-since-last-status-change: 00:01:51
Connection-type: PVP
Cast-type: point-to-multipoint-root
Usage-Parameter-Control (UPC): pass
Wrr weight: 2
Number of OAM-configured connections: 0
OAM-configuration: disabled
OAM-states:  Not-applicable
Cross-connect-interface: ATM1/1/1, Type: oc3suni
Cross-connect-VPI = 60
Cross-connect-UPC: pass
Cross-connect OAM-configuration: disabled
Cross-connect OAM-state:  Not-applicable
Cross-connect-interface: ATM3/0/0
Cross-connect-VPI = 70
Cross-connect-interface: ATM3/0/3
Cross-connect-VPI = 80
Threshold Group: 5, Cells queued: 0
Rx cells: 0, Tx cells: 0
Tx Clp0:0,  Tx Clp1: 0
Rx Clp0:0,  Rx Clp1: 0
Rx Upc Violations:0, Rx cell drops:0
Rx Clp0 q full drops:0, Rx Clp1 qthresh drops:0
Rx connection-traffic-table-index: 1
Rx service-category: UBR (Unspecified Bit Rate)
Rx pcr-clp01: 7113539
Rx scr-clp01: none
Rx mcr-clp01: none
Rx      cdvt: 1024 (from default for interface)
Rx       mbs: none
```

```
Tx connection-traffic-table-index: 1
Tx service-category: UBR (Unspecified Bit Rate)
Tx pcr-clp01: 7113539
Tx scr-clp01: none
Tx mcr-clp01: none
Tx       cdvt: none
Tx        mbs: none
```

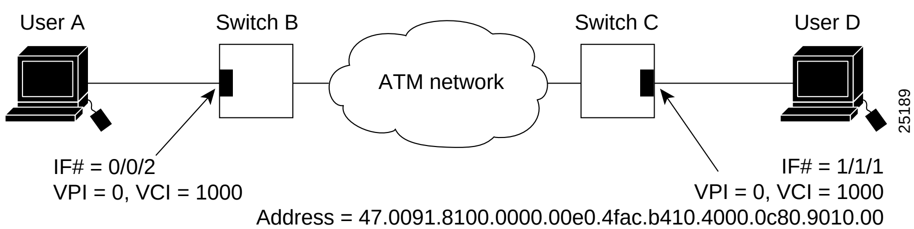
Configuring Soft PVC Connections

This section describes configuring soft permanent virtual channel (PVC) connections, which provide the following features:

- Connection to another host or ATM switch router that supports signalling
- Configuration of PVCs without the manual configuration steps described in the "Configuring Virtual Channel Connections" section on page 6-2
- Configuration of PVCs with the reroute or retry capabilities when a failure occurs in the network

Figure 6-6 illustrates the soft PVC connections used in the following examples.

Figure 6-6 Soft PCV Connection Example



Guidelines for Creating Soft PVCs

Perform the following steps when you configure soft PVCs:

Step 1 Determine which two ports you want to define as participants in the soft PVC.

Step 2 Decide which of these two ports you want to designate as the destination (or passive) side of the soft PVC.

This decision is arbitrary—it makes no difference which port you define as the destination end of the circuit.

Step 3 Retrieve the ATM address of the destination end of the soft PVC using the **show atm address** command.

Step 4 Retrieve the VPI/VCI values for the circuit using the **show atm vc** command.

Step 5 Configure the source (active) end of the soft PVC. At the same time, complete the soft PVC setup using the information derived from Step 3 and Step 4. Be sure to select an unused VPI/VCI value (one that does not appear in the **show atm vc** display).

Configuring Soft PVCs

To configure a soft PVC connection, perform the following steps, beginning in privileged EXEC mode:

	Command	Purpose
Step 1	Switch# **show atm addresses**	Determines the destination ATM address.
Step 2	Switch# **configure terminal** Switch(config)#	At the privileged EXEC prompt, enters configuration mode from the terminal.
Step 3	Switch(config)# **interface atm** *card*/*subcard*/*port*[**.***vpt#*] Switch(config-if)#	Selects the interface to be configured.
Step 4	Switch(config-if)# **atm soft-vc** *source-vpi source-vci* **dest-address** *atm-address dest-vpi dest-vci* [**enable** | **disable**] [**upc** *upc*] [**pd** *pd*] [**rx-cttr** *index*] [**tx-cttr** *index*] [**retry-interval** [**first** *interval*] [**maximum** *interval*]] [**redo-explicit** [**explicit-path** *precedence* {**name** *path-name* | **identifier** *path-id*} [**upto** *partial-entry-index*]] [**only-explicit**]]	Configures the soft PVC connection.

Note The row index for **rx-cttr** and **tx-cttr** must be configured before using this optional parameter. See the "Configuring the Connection Traffic Table" section on page 8-10.

Examples

The following example shows the destination ATM address of the interface connected to User D:

```
Switch-C# show atm addresses

Switch Address(es):
  47.00918100000000400B0A2A81.00400B0A2A81.00 active
  47.00918100000000E04FACB401.00E04FACB401.00

Soft VC Address(es):

<Information deleted>

  47.0091.8100.0000.00e0.4fac.b401.4000.0c80.9000.00 ATM1/1/0
  47.0091.8100.0000.00e0.4fac.b401.4000.0c80.9010.00 ATM1/1/1
  47.0091.8100.0000.00e0.4fac.b401.4000.0c80.9020.00 ATM1/1/2

<Information deleted>
```

The following example shows how to configure a soft PVC on Switch B between interface ATM 0/0/2, source VPI = 0, VCI = 1000; and Switch C, destination VPI = 0, VCI = 1000 with a specified ATM address (see Figure 6-6):

```
Switch-B(config)# interface atm 0/0/2
Switch-B(config-if)# atm soft-vc 0 1000 dest-address 47.0091.8100.0000.00e0.4fac.b401.4000.0c80.9010.00 0
1000
```

Displaying Soft PVC Configuration

To display the soft PVC configuration at either end of a ATM switch router, use the following EXEC commands:

Command	Purpose
show atm vc interface atm *card*/*subcard*/*port*	Shows the VCs configured on the ATM interface.
show atm vc interface atm *card*/*subcard*/*port* *vpi vci*	Shows the soft PVC interface configuration.

Examples

The following example shows the soft PVC configuration of Switch B, on interface ATM 0/0/2 out to the ATM network:

```
Switch-B# show atm vc interface atm 0/0/2
Interface         VPI  VCI   Type   X-Interface      X-VPI X-VCI Encap  Status
ATM0/0/2          0    5     PVC    ATM0              0    45    QSAAL  UP
ATM0/0/2          0    16    PVC    ATM0              0    37    ILMI   UP
ATM0/0/2          0    18    PVC    ATM0              0    52    PNNI   UP
ATM0/0/2          0    34    PVC    ATM0              0    51    NCDP   UP
ATM0/0/2          0    35    SVC    ATM0/0/2          0    1000         UP
➔ ATM0/0/2        0    1000  SoftVC ATM0/0/2          0    35           UP
```

The following example shows the soft PVC configuration of Switch C, on interface ATM 1/1/1 out to the ATM network:

```
Switch-C# show atm vc interface atm 1/1/1
Interface         VPI  VCI   Type   X-Interface      X-VPI X-VCI Encap  Status
ATM1/1/1          0    5     PVC    ATM2/0/0          0    74    QSAAL  UP
ATM1/1/1          0    16    PVC    ATM2/0/0          0    44    ILMI   UP
ATM1/1/1          0    18    PVC    ATM2/0/0          0    109   PNNI   UP
ATM1/1/1          0    34    PVC    ATM2/0/0          0    120   NCDP   UP
ATM1/1/1          0    123   SVC    ATM1/1/1          0    1000         UP
➔ ATM1/1/1        0    1000  SoftVC ATM1/1/1          0    123          UP
ATM1/1/1          2    100   PVC    ATM2/0/0          0    103   SNAP   UP
```

The following example shows the soft PVC configuration of Switch B, on interface ATM 0/0/2 (VPI = 0, VCI = 1000) out to the ATM network with the switch processor feature card installed:

```
Switch-B# show atm vc interface atm 0/0/2 0 1000

Interface: ATM0/0/2, Type: oc3suni
➔ VPI = 0  VCI = 1000
Status: UP
Time-since-last-status-change: 21:56:48
Connection-type: SoftVC
Cast-type: point-to-point
 Soft vc location: Source
➔ Remote ATM address: 47.0091.8100.0000.0040.0b0a.2a81.4000.0c80.9010.00
 Remote VPI: 0
 Remote VCI: 1000
 Soft vc call state: Active
 Number of soft vc re-try attempts: 0
 First-retry-interval: 5000 milliseconds
 Maximum-retry-interval: 60000 milliseconds
 Aggregate admin weight: 10080
 TIME STAMPS:
 Current Slot:2
```

```
  Outgoing Setup     May 25 10:38:50.718
  Incoming Connect   May 25 10:38:50.762

Packet-discard-option: disabled
Usage-Parameter-Control (UPC): pass
Wrr weight: 2
Number of OAM-configured connections: 0
OAM-configuration: disabled
OAM-states:  Not-applicable
Cross-connect-interface: ATM0/0/2, Type: oc3suni
Cross-connect-VPI = 0
Cross-connect-VCI = 35
Cross-connect-UPC: pass
Cross-connect OAM-configuration: disabled
Cross-connect OAM-state:  Not-applicable
Threshold Group: 5, Cells queued: 0
Rx cells: 0, Tx cells: 0
Tx Clp0:0,  Tx Clp1: 0
Rx Clp0:0,  Rx Clp1: 0
Rx Upc Violations:0, Rx cell drops:0
Rx Clp0 q full drops:0, Rx Clp1 qthresh drops:0
Rx connection-traffic-table-index: 1
Rx service-category: UBR (Unspecified Bit Rate)
Rx pcr-clp01: 7113539
Rx scr-clp01: none
Rx mcr-clp01: none
Rx      cdvt: 1024 (from default for interface)
Rx       mbs: none
Tx connection-traffic-table-index: 1
Tx service-category: UBR (Unspecified Bit Rate)
Tx pcr-clp01: 7113539
Tx scr-clp01: none
Tx mcr-clp01: none
Tx      cdvt: none
Tx       mbs: none
```

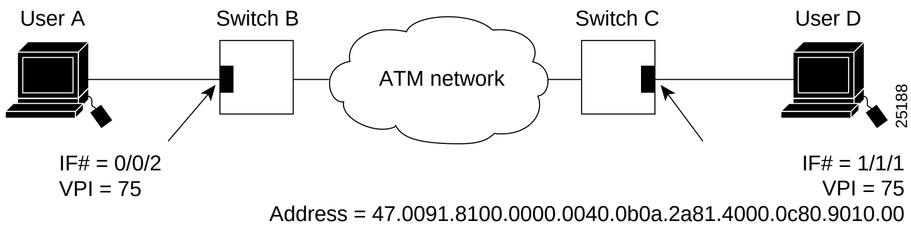
Configuring Soft PVP Connections

This section describes configuring soft permanent virtual path (PVP) connections, which provide the following features:

- Connection to another host or ATM switch router that does supports signalling
- Configuration of PVPs without the manual configuration steps described in the "Configuring Virtual Channel Connections" section on page 6-2.
- Configuration of PVPs with the reroute or retry capabilities when a failure occurs within the network

Figure 6-7 is an illustration of the soft PVP connections used in the examples in this section.

Figure 6-7 Soft PVP Connection Example



To configure a soft PVP connection, perform the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **interface atm** *card*/*subcard*/*port* Switch(config-if)#	Selects the interface to be configured.
Step 2	Switch(config-if)# **atm soft-vp** *source-vpi* **dest-address** *atm-address dest-vpi* [**enable** | **disable**] [**upc** *upc*] [**rx-cttr** *index*] [**tx-cttr** *index*] [**retry-interval** [**first** *interval*] [**maximum** *interval*]] [**redo-explicit** [**explicit-path** *precedence* {**name** *path-name* | **identifier** *path-id*} [**upto** *partial-entry-index*]] [**only-explicit**]]	Configures the soft PVP connection.

The row index for **rx-cttr** and **tx-cttr** must be configured before using this optional parameter. See the "Configuring the Connection Traffic Table" section on page 8-10.

Example

The following example shows how to configure a soft PVP on Switch B between interface ATM 0/0/2, source VPI = 75; and Switch C, destination VPI = 75, with a specified ATM address (see Figure 6-7):

```
Switch-B(config)# interface atm 0/0/2
Switch-B(config-if)# atm soft-vp 75 dest-address 47.0091.8100.0000.0040.0b0a.2a81.4000.0c80.9010.00 75
```

Displaying Soft PVP Connections

To display the ATM soft PVP configuration, use the following EXEC command:

Command	Purpose
show atm vp [**interface atm** *card*/*subcard*/*port vpi*]	Shows the soft PVP configuration.

Examples

The following example shows the soft PVP configuration at Switch B, on interface ATM 0/0/2 out to the ATM network:

```
Switch-B# show atm vp
Interface         VPI  Type  X-Interface         X-VPI     Status
ATM0/0/2          1    SVP    ATM0/0/2          75  UP
ATM0/0/2          75   SoftVP ATM0/0/2          1   UP
```

The following example shows the soft PVP configuration on interface ATM 1/1/1 at Switch C out to the ATM network:

```
Switch-C# show atm vp
Interface         VPI  Type  X-Interface         X-VPI     Status
ATM1/1/1          1    SVP    ATM1/1/1          75  UP
ATM1/1/1          75   SoftVP ATM1/1/1          1   UP
```

The following example shows the soft PVP configuration at Switch B on interface ATM 0/0/2 (VPI = 75) out to the ATM network with the switch processor feature card installed:

```
Switch-B# show atm vp interface atm 0/0/2 75

Interface: ATM0/0/2, Type: oc3suni
VPI = 75
Status: UP
Time-since-last-status-change: 00:09:46
Connection-type: SoftVP
Cast-type: point-to-point
 Soft vp location: Source
 Remote ATM address: 47.0091.8100.0000.0040.0b0a.2a81.4000.0c80.9010.00
 Remote VPI: 75
 Soft vp call state: Active
 Number of soft vp re-try attempts: 0
 First-retry-interval: 5000 milliseconds
 Maximum-retry-interval: 60000 milliseconds
 Aggregate admin weight: 10080
 TIME STAMPS:
 Current Slot:2
  Outgoing Setup     May 26 09:45:30.292
  Incoming Connect   May 26 09:45:30.320
<information deleted>
```

Configuring the Soft PVP or Soft PVC Route Optimization Feature

This section describes the soft PVP or soft PVC route optimization feature. Most soft PVPs or soft PVCs have a much longer lifetime than SVCs. The route chosen during the soft connection setup remains the same even though the network topology might change.

Soft connections, with the route optimization percentage threshold set, provide the following features:

- When a better route is available, soft PVPs or PVCs are dynamically rerouted
- Route optimization can be triggered manually

Note Soft PVC route optimization should not be configured with constant bit rate (CBR) connections.

Route optimization is directly related to administrative weight, which is similar to hop count. For a description of administrative weight, see the "Configuring the Global Administrative Weight Mode" section on page 10-39.

Configuring soft PVP or soft PVC route optimization is described in the following sections:

- Enabling Soft PVP or Soft PVC Route Optimization, page 6-23
- Configuring a Soft PVP/PVC Interface with Route Optimization, page 6-23

For overview information about the route optimization feature refer to the *Guide to ATM Technology*.

Enabling Soft PVP or Soft PVC Route Optimization

Soft PVP or soft PVC route optimization must be enabled and a threshold level configured to determine the point when a better route is identified and the old route is reconfigured.

To enable and configure route optimization, use the following global configuration command:

Command	Purpose
atm route-optimization percentage-threshold *percent*	Configures route optimization.

Example

The following example enables route optimization and sets the threshold percentage to 85 percent:

```
Switch(config)# atm route-optimization percentage-threshold 85
```

Configuring a Soft PVP/PVC Interface with Route Optimization

Soft PVP or soft PVC route optimization must be enabled and configured to determine the point at which a better route is found and the old route is reconfigured.

To enable and configure a soft PVC/PVP interface with route optimization, perform the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **interface** [**atm** *card*/*subcard*/*port* | **serial** *card*/*subcard*/*port*:*cgn*] Switch(config-if)#	Selects the interface to configure. Enter the interface number of the source end of the soft PVC/PVP. Route optimization works for the source end of a soft PVC/PVP only and is ignored if configured on the destination interface.
Step 2	Switch(config-if)# **atm route-optimization soft-connection** [**interval** *minutes*] [**time-of-day** {**anytime** | *start-time end-time*}]	Configures the interface for route optimization.

Example

The following example shows how to configure an interface with a route optimization interval configured as every 30 minutes between the hours of 6:00 P.M. and 5:00 A.M.:

```
Switch(config)# interface atm 0/0/0
Switch(config-if)# atm route-optimization soft-connection interval 30 time-of-day 18:00 5:00
```

Displaying an Interface Route Optimization Configuration

To display the interface route optimization configuration, use the following EXEC command:

Command	Purpose
show atm interface [**atm** *card*/*subcard*/*port* | **serial** *card*/*subcard*/*port***:***cgn*]	Shows the interface configuration route optimization configuration.

Example

The following example shows the route optimization configuration of ATM interface 0/0/0:

```
Switch# show atm interface atm 0/0/0
IF Status:      UP              Admin Status:   up
Auto-config:    enabled         AutoCfgState:   completed
IF-Side:        Network         IF-type:        NNI
Uni-type:       not applicable  Uni-version:    not applicable
Max-VPI-bits:   8               Max-VCI-bits:   14
Max-VP:         255             Max-VC:         16383
ConfMaxSvpcVpi: 255             CurrMaxSvpcVpi: 255
ConfMaxSvccVpi: 255             CurrMaxSvccVpi: 255
ConfMinSvccVci: 35              CurrMinSvccVci: 35
Svc Upc Intent: pass            Signalling:     Enabled
➔ Soft vc route optimization is enabled
➔ Soft vc route optimization interval = 30 minutes
➔ Soft vc route optimization time-of-day range = (18:0 - 5:0)
ATM Address for Soft VC: 47.0091.8100.0000.00e0.4fac.b401.4000.0c80.8000.00
<information deleted>
```

Configuring Soft PVCs with Explicit Paths

Normally, soft PVCs and soft PVPs are automatically routed by PNNI over paths that meet the traffic parameter objectives. However, for cases where manually configured paths are needed, PNNI explicit paths can optionally be specified for routing the soft PVC or soft PVP. For detailed information on configuring PNNI explicit paths, see the “Configuring Explicit Paths” section on page 10-36.

The explicit paths are assigned using precedence numbers 1 through 3. The precedence 1 path is tried first and if it fails the soft connection is routed using the precedence 2 path and so forth. If all of the explicit paths fail, standard on-demand PNNI routing is tried unless the **only-explicit** keyword is specified.

If the soft connection destination address is reachable at one of the included entries in an explicit path, any following entries in that path are automatically disregarded. This allows longer paths to be reused for closer destinations. Alternatively, the **upto** keyword can be specified for an explicit path in order to disregard later path entries.

Example

The following example shows how to configure a soft PVC between ATM switch router dallas_1 and an address on ATM switch router new_york_3 using either of the two explicit paths new_york.path1 and new_york.path2. If both explicit paths fail, the ATM switch router uses PNNI on-demand routing to calculate the route.

```
dallas_1(config)# interface atm 0/0/0
dallas_1(config)# atm soft-vc 0 201 dest-address 47.0091.8100.0000.1061.3e7b.2f99.4000.0c80.0030.00 0 101
explicit-path 1 name new_york.path1 explicit-path 2 name new_york.path2
```

Changing Explicit Paths for an Existing Soft PVC

Explicit paths can be added, modified or removed without tearing down existing soft PVCs by using the **redo-explicit** keyword. Only the source VPI and VCI options need to be specified. All applicable explicit path options are replaced by the respecified explicit path options.

The soft PVC is not immediately rerouted using the new explicit path. However, reroutes using the new explicit path can happen for the following four reasons:

1. A failure occurs along the current path.
2. The EXEC command **atm route-optimization soft-connection** is entered for the soft PVC.
3. **route-optimization** is enabled and the retry time interval has expired.
4. The soft PVC is disabled and then reenabled using the **disable** and **enable** keywords.

Example

The following example shows how to change the explicit path configuration for an existing soft PVC on the ATM switch router dallas_1 without tearing down the connection. The new configuration specifies the two explicit paths, new_york.path3 and new_york.path4, and uses the only-explicit option.

```
dallas_1(config)# interface atm 0/0/0
dallas_1(config)# atm soft-vc 0 201 redo-explicit explicit-path 1 name new_york.path3
explicit-path 2 name new_york.path4 only-explicit
```

Note The configuration displayed for soft connections with explicit paths is always shown as two separate lines using the **redo-explicit** keyword on the second line, even if it is originally configured using a single command line.

Displaying Explicit Path for Soft PVC Connections

To display a soft PVC connection successfully routed over an explicit path, use the following EXEC command:

Command	Purpose
show atm vc interface atm *card/subcard/port vpi vci*	Displays the soft PVC connection status including the PNNI explicit path routing status for the last setup attempt.

Example

The following example shows the last explicit path status for a soft PVC using the **show atm vc interface** EXEC command. Note that the first listed explicit path new_york.path2 shows an unreachable result, but the second explicit path new_york.path1 succeeded.

```
Switch# show atm vc interface atm 0/1/3 0 40
VPI = 0  VCI = 40
Status:UP
Time-since-last-status-change:00:00:03
Connection-type:SoftVC
Cast-type:point-to-point
 Soft vc location:Source
 Remote ATM address:47.0091.8100.0000.0060.705b.d900.4000.0c81.9000.00
 Remote VPI:0
 Remote VCI:40
 Soft vc call state:Active
 Number of soft vc re-try attempts:0
 First-retry-interval:5000 milliseconds
 Maximum-retry-interval:60000 milliseconds
 Aggregate admin weight:15120
 TIME STAMPS:
 Current Slot:4
  Outgoing Release   February 26 17:02:45.940
  Incoming Rel comp  February 26 17:02:45.944
  Outgoing Setup     February 26 17:02:45.948
  Incoming Connect   February 26 17:02:46.000
  Outgoing Setup     February 23 11:54:17.587
  Incoming Release   February 23 11:54:17.591
  Outgoing Setup     February 23 11:54:37.591
  Incoming Release   February 23 11:54:37.611
  Outgoing Setup     February 23 11:55:17.611
  Incoming Connect   February 23 11:55:17.655

→  Explicit-path 1:result=6  PNNI_DEST_UNREACHABLE  (new_york.path2)
→  Explicit-path 2:result=1  PNNI_SUCCESS  (new_york.path1)
 Only-explicit
Packet-discard-option:disabled
Usage-Parameter-Control (UPC):pass
Number of OAM-configured connections:0
OAM-configuration:disabled
OAM-states: Not-applicable
Cross-connect-interface:ATM0/0/3.4, Type:oc3suni
Cross-connect-VPI = 4
Cross-connect-VCI = 35
Cross-connect-UPC:pass
Cross-connect OAM-configuration:disabled
Cross-connect OAM-state: Not-applicable
Rx cells:0, Tx cells:0
Rx connection-traffic-table-index:1
Rx service-category:UBR (Unspecified Bit Rate)
Rx pcr-clp01:7113539
Rx scr-clp01:none
Rx mcr-clp01:none
Rx      cdvt:1024 (from default for interface)
Rx       mbs:none
Tx connection-traffic-table-index:1
Tx service-category:UBR (Unspecified Bit Rate)
Tx pcr-clp01:7113539
Tx scr-clp01:none
Tx mcr-clp01:none
Tx      cdvt:none
Tx       mbs:none
```

Configuring Nondefault Well-Known PVCs

Normally the default well-known VCs are automatically created with default virtual channel identifiers (VCIs). However, for the unusual instances where the ATM switch router interfaces with nonstandard equipment, you can configure nondefault well-known VCI values on a per-interface basis.

For overview information about the well-known PVCs, refer to the *Guide to ATM Technology*.

Table 6-2 lists the default well-known VCs and their default configuration.

Table 6-2 Well-Known Virtual Channels

Channel Type	Virtual Path Identifier	Virtual Channel Identifier
Signalling	0	5
ILMI	0	16
PNNI	0	18
Tag switching	0	32

Caution Do not change the well-known channels to use a VC where the remote end is sending AAL5 messages not intended for the well-known VC. For example, do not swap VC values between two types of well-known VCs.

Overview of Nondefault PVC Configuration

Following is an overview of the steps needed to configure nondefault well-known VCs:

Step 1 Enable manual well-known VC configuration.

Step 2 Delete any existing automatically created well-known VCs.

Step 3 Configure the individual encapsulation type as follows:

- Signalling (QSAAL)
- ILMI
- PNNI
- Tag switching

Step 4 Copy the running-configuration file to the startup-configuration file.

Configuring Nondefault PVCs

To configure the nondefault PVCs for signalling, ILMI, and PNNI, perform the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **interface atm** *card*/*subcard*/*port* Switch(config-if)#	Selects the interface to be configured.
Step 2	Switch(config-if)# **atm manual-well-known-vc** {**keep** | **delete**}	Enters manual-well-known-vc mode.
Step 3	Switch(config-if)# **atm pvc** *vpi vci* [**rx-cttr** *index*] [**tx-cttr** *index*] **interface atm** *card*/*subcard*/*port* **any-vci** [**encap** {**ilmi | pnni | qsaal**}] or Switch(config-if)# **tag-switching atm control-vc** *vpi vci*	Configures the nondefault PVC for encapsulation type.
Step 4	Switch(config-if)# **end** Switch#	Returns to privileged EXEC mode.
Step 5	Switch# **copy system:running-config nvram:startup-config**	Copies the running configuration file to the startup configuration file.

Note An error condition occurs if either the signalling or ILMI well-known VCs remain unconfigured when an interface is enabled.

Example

The following example shows the nondefault VC configuration steps:

Step 1 Use the **show atm vc interface atm** command to display the configuration of the existing default well-known VCs for ATM interface 0/0/0.

Step 2 Change to interface configuration mode for ATM interface 0/0/0.

Step 3 Enter manual well-known-vc mode and delete the existing default well-known VCs using the **atm manual-well-known-vc delete** command.

Step 4 Confirm deletion by entering **y.**

Step 5 Configure the nondefault VC for signalling from 5 (the default) to 35 using the **atm pvc** command.

Step 6 Configure the ILMI VC, then configure the PNNI VC if needed using the same procedure.

Step 7 Save the new running configuration to the startup configuration.

An example of this procedure follows:

```
Switch# show atm vc interface atm 0/0/0
Interface    VPI   VCI   Type    X-Interface  X-VPI X-VCI  Encap Status
ATM0/0/0     0     5      PVC     ATM0          0     49    QSAAL  UP
ATM0/0/0     0     16     PVC     ATM0          0     33    ILMI   UP
ATM0/0/0     0     18     PVC     ATM0          0     65    PNNI   UP
Switch#
Switch# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)# interface atm 0/0/0
Switch(config-if)# atm manual-well-known-vc delete

Okay to delete well-known VCs for this interface? [no]: y
Switch(config-if)# atm pvc 1 35 interface atm0 any-vci encap qsaal
Switch(config-if)# end
Switch#
%SYS-5-CONFIG_I: Configured from console by console
Switch# show atm vc interface atm 0/0/0
Interface    VPI   VCI   Type    X-Interface  X-VPI X-VCI  Encap Status
ATM0/0/0     1    35     PVC     ATM0         0     150   QSAAL  UP
Switch# copy system:running-config nvram:startup-config
Building configuration...
[OK]
```

Configuring a VPI/VCI Range for SVPs and SVCs

You can configure a virtual path identifier/virtual channel identifier (VPI/VCI) range for switched virtual channels and switched virtual paths (SVCs and SVPs). ILMI uses the specified range to negotiate the VPI/VCI range parameters with peers. This feature allows you to:

- Specify ranges for SVPs/SVCs.
- Avoid VPI/VCI conflicts when attempting to set up soft PVPs or soft PVCs.

You can still configure PVPs and PVCs in any supported range, including any VPI/VCI range you configured for SVPs/SVCs.

Note This feature is supported in ILMI 4.0.

The default maximum switched virtual path connection (SVPC) VPI is equal to 255. You can change the maximum SVPC VPI by entering the **atm svpc vpi max** *value* command. See Table 6-3 for the allowable ranges.

Table 6-3 Maximum SVPC VPI Range

VPI Bit Type	Maximum Value Range
8-bit VPI	0 to 255
12-bit VPI[1]	0 to 4095

1. Only available on ATM NNI interfaces.

Note The maximum value specified applies to all interfaces except logical interfaces, which have a fixed value of 0.

For further information and examples of using VPI/VCI ranges for SVPs/SVCs, refer to the *Guide to ATM Technology*.

Every interface negotiates the local values for the maximum SVPC VPI, maximum SVCC VPI, and minimum SVCC VCI with the peer's local value during ILMI initialization. The negotiated values determine the ranges for SVPs and SVCs. If the peer interface does not support these objects or autoconfiguration is turned off on the local interface, the local values determine the range.

To configure a VPI/VCI range for SVCs/SVPs, perform the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **interface atm** *card*/*subcard*/*port* Switch(config-if)#	Selects the physical interface to be configured.
Step 2	Switch(config-if)# **atm svpc vpi max** *value*	Configures the maximum VPI value for a SVPC.
Step 3	Switch(config-if)# **atm svcc vpi max** *value*	Configures the maximum VPI value for a SVCC.
Step 4	Switch(config-if)# **atm svcc vci min** *value*	Configures the minimum VCI value for a SVCC.

The following example shows configuring ATM interface 0/0/0 with the SVPC and SVCC VPI maximum set to 100, and SVCC VCI minimum set to 60.

```
Switch(config)# interface atm 0/0/0
Switch(config-if)# atm svpc vpi max 100
Switch(config-if)# atm svcc vpi max 100
Switch(config-if)# atm svcc vci min 60
```

Displaying the VPI/VCI Range Configuration

To confirm the VPI or VCI range configuration, use one of the following commands:

Command	Purpose
show atm interface atm *card*/*subcard*/*port*	Shows the ATM interface configuration.
show atm ilmi-status atm *card*/*subcard*/*port*	Shows the ILMI status on the ATM interface.

Examples

The following example shows how to confirm the VPI and VCI range configuration on an ATM interface. The values displayed for ConfMaxSvpcVpi, ConfMaxSvccVpi, and ConfMinSvccVci are local values. The values displayed for CurrMaxSvpcVpi, CurrMaxSvccVpi, and CurrMinSvccVci are negotiated values.

```
Switch# show atm interface atm 0/0/0
Interface:      ATM0/0/0        Port-type:      oc3suni
IF Status:      DOWN            Admin Status:   down
Auto-config:    enabled         AutoCfgState:   waiting for response from peer
IF-Side:        Network         IF-type:        UNI
Uni-type:       Private         Uni-version:    V3.0
Max-VPI-bits:   8               Max-VCI-bits:   14
Max-VP:         255             Max-VC:         16383
ConfMaxSvpcVpi: 100             CurrMaxSvpcVpi: 100
ConfMaxSvccVpi: 100             CurrMaxSvccVpi: 100
ConfMinSvccVci: 60              CurrMinSvccVci: 60
Svc Upc Intent: pass            Signalling:     Enabled
ATM Address for Soft VC: 47.0091.8100.0000.0040.0b0a.2a81.4000.0c80.0000.00
Configured virtual links:
  PVCLs SoftVCLs   SVCLs   TVCLs   PVPLs SoftVPLs   SVPLs Total-Cfgd Inst-Conns
      3        0       0       0       0        0       0          3          0
Logical ports(VP-tunnels):     0
Input cells:    0               Output cells:   0
5 minute input rate:             0 bits/sec,       0 cells/sec
5 minute output rate:            0 bits/sec,       0 cells/sec
Input AAL5 pkts: 0, Output AAL5 pkts: 0, AAL5 crc errors: 0
```

The following example shows how to confirm the peer's local values for VPI and VCI range configuration by displaying the ILMI status on an ATM interface:

```
Switch# show atm ilmi-status atm 0/0/0

Interface : ATM0/0/0 Interface Type : Private NNI
ILMI VCC : (0, 16) ILMI Keepalive : Disabled
Addr Reg State:   UpAndNormal
Peer IP Addr:     172.20.40.232   Peer IF Name:     ATM0/0/0
Peer MaxVPIbits:  8               Peer MaxVCIbits:  14
Peer MaxVPCs:     255             Peer MaxVCCs:     16383
Peer MaxSvccVpi:  255             Peer MinSvccVci:  255
Peer MaxSvpcVpi:  48
Configured Prefix(s) :
47.0091.8100.0000.0010.11ba.9901
```

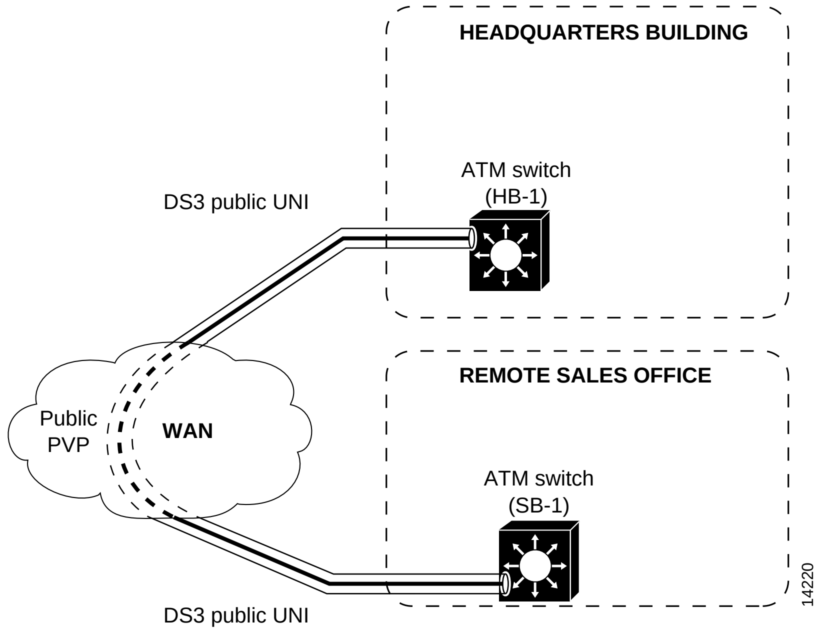
Note Note that the **show atm ilmi-status** command displays the information above only if the peer supports it.

Configuring VP Tunnels

This section describes configuring virtual path (VP) tunnels, which provide the ability to interconnect ATM switch routers across public networks using PVPs. You can configure a VP tunnel to carry a single service category, or you can configure a VP tunnel to carry multiple service categories, including merged VCs.

Figure 6-8 shows a public UNI interface over a DS3 connection between the ATM switch router (HB-1) in the Headquarters building and the ATM switch router (SB-1) in the Remote Sales building. To support signalling across this connection, a VP tunnel must be configured.

Figure 6-8 Public VP Tunnel Network Example



Configuring a VP Tunnel for a Single Service Category

The type of VP tunnel described in this section is configured as a VP of a single service category. Only virtual circuits (VCs) of that service category can transit the tunnel.

To configure a VP tunnel connection for a single service category, perform the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **atm connection-traffic-table-row** [**index** *row-index*] [{**vbr-rt** | **vbr-nrt**} **pcr** *pcr_value* {**scr0** | **scr10**} *scr_value* [**mbs** *mbs_value*] [**cdvt** *cdvt_value*] | [**cbr pcr** *pcr_value* [**cdvt** *cdvt_value*] | [**abr pcr** *pcr_value* [**mcr** *mcr_value*] [**cdvt** *cdvt_value*] | [**ubr pcr** *pcr_value* [**mcr** *mcr_value*] [**cdvt** *cdvt_value*]]	Configures the connection-traffic-table-row index for any nondefault traffic values (optional).
Step 2	Switch(config)# **interface atm** *card*/*subcard*/*port* Switch(config-if)#	Selects the physical interface to be configured.
Step 3	Switch(config-if)# **atm pvp** *vpi* [**rx-cttr** *index*] [**tx-cttr** *index*]	Configures an interface permanent virtual path (PVP) leg.

	Command	Purpose
Step 4	Switch(config-if)# **exit** Switch(config)#	Exits interface configuration mode.
Step 5	Switch(config)# **interface atm** *card/subcard/port.vpt#* Switch(config-subif)#	Creates a VP tunnel using a VP tunnel number that matches the PVP leg virtual path identifier (VPI).

Note The row index for nondefault **rx-cttr** and **tx-cttr** must be configured before these optional parameters are used.

Examples

The following example shows how to configure the ATM VP tunnel on the ATM switch router (HB-1) at interface ATM 1/0/0, VPI 99:

```
Switch(HB-1)(config)# interface atm 1/0/0
Switch(HB-1)(config-if)# atm pvp 99
Switch(HB-1)(config-if)# exit
Switch(HB-1)(config)# interface atm 1/0/0.99
Switch(HB-1)(config-subif)# end
Switch(HB-1)#
```

The following example shows how to configure the ATM VP tunnel on the ATM switch router (SB-1) interface ATM 0/0/0, VPI 99:

```
Switch(SB-1)(config)# interface atm 0/0/0
Switch(SB-1)(config-if)# atm pvp 99
Switch(SB-1)(config-if)# exit
Switch(SB-1)(config)# interface atm 0/0/0.99
Switch(SB-1)(config-subif)# end
Switch(SB-1)#
```

Displaying the VP Tunnel Configuration

To show the ATM virtual interface configuration, use the following EXEC command:

Command	Purpose
show atm interface atm *card/subcard/port.vpt#*	Shows the ATM interface configuration.

The following example shows the ATM virtual interface configuration for interface ATM 1/0/0.99:

```
Switch# show atm interface atm 1/0/0.99
Interface:      ATM1/0/0.99     Port-type:      vp tunnel
IF Status:      UP              Admin Status:   up
Auto-config:    enabled         AutoCfgState:   waiting for response from peer
IF-Side:        Network         IF-type:        UNI
Uni-type:       Private         Uni-version:    V3.0
<information deleted>
```

Configuring a Shaped VP Tunnel

This section describes configuring a shaped VP tunnel for a single service category with rate-limited tunnel output on a switch.

A shaped VP tunnel is configured as a VP of the CBR service category. By default, this tunnel can carry VCs only of the CBR service category. However, you can configure this VP tunnel to carry VCs of other service categories. The overall output of this VP tunnel is rate-limited by hardware to the peak cell rate (PCR) of the tunnel.

Note Shaped VP tunnels are supported only on systems with the FC-PFQ feature card. (Catalyst 8510 MSR and LightStream 1010)

A shaped VP tunnel is defined as a CBR VP with a PCR. The following limitations apply:

- A maximum of 64 shaped VP tunnels can be defined on each of the following interface groups: (0/0/*x*, 1/0/*x*), (0/1/*x*, 1/1/*x*), (2/0/*x*, 3/0/*x*), (2/1/*x*, 3/1/*x*), (9/0/*x*, 10/0/*x*), (9/1/*x*, 10/1/*x*), (11/0/*x*, 12/0/*x*), and (11/1/*x*, 12/1/*x*). (Catalyst 8540 MSR)
- A maximum of 64 shaped VP tunnels can be defined on interfaces *x*/0/*y*; similarly, a maximum of 64 shaped VP tunnels can be defined on interfaces *x*/1/*y*. (Catalyst 8510 MSR and LightStream 1010)
- The bandwidth of the shaped VP tunnel is shared by the active VCs inside the tunnel in strict round-robin (RR) fashion.
- Even though the shaped VP tunnel is defined as CBR, it can carry VCs of another service category by substituting the new service category after the tunnel interface has been initially configured. For configuration information, see the "Configuring Interface Service Category Support" section on page 8-35.
- Shaped VP tunnels do not support merged VCs for tag switching.
- UBR+ and ABR VCs with non-zero MCR are not allowed on a shaped VP tunnel interface.
- A maximum of 128 VCs can transit a shaped VP tunnel interface.
- Shaped VP tunnels support interface overbooking. For configuration information, see the "Configuring Interface Overbooking" section on page 8-37.
- Shaped VP tunnels cannot be configured with ATM router modules because CBR scheduling is not supported on those interfaces.

Configuring a Shaped VP Tunnel on an Interface

To configure a shaped VP tunnel, perform the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **atm connection-traffic-table-row** [**index** *row-index*] **cbr pcr** *rate*	Configures the connection-traffic-table row for the desired PVP CBR cell rate.
Step 2	Switch(config)# **interface atm** *card*/*subcard*/*port* Switch(config-if)#	Selects the physical interface to configure.

	Command	Purpose
Step 3	Switch(config-if)# **atm pvp** *vpi* **shaped rx-cttr** *index* **tx-cttr** *index*	Configures an interface PVP leg.
Step 4	Switch(config-if)# **exit** Switch(config)#	Exits interface configuration mode.
Step 5	Switch(config)# **interface atm** *card/subcard/port.vpt#* Switch(config-subif)#	Creates a shaped VP tunnel using a VP tunnel number that matches the PVP leg VPI.

Note The **rx-cttr** and **tx-cttr** row indexes must be configured before they are used.

Example

The following example shows how to configure a shaped VP tunnel with a VPI of 99 as ATM interface 0/0/0.99

```
Switch(config)# interface atm 0/0/0
Switch(config-if)# atm pvp 99 shaped rx-cttr 100 tx-cttr 100
Switch(config-if)# exit
Switch(config-if)# interface atm 0/0/0.99
Switch(config-subif)#
```

Displaying the Shaped VP Tunnel Configuration

To display the shaped VP tunnel interface configuration, use the following EXEC command:

Command	Purpose
show atm interface atm *card/subcard/port.vpt#*	Shows the ATM VP interface configuration.

For an example display from the **show atm interface** command, see the "Displaying the Hierarchical VP Tunnel Configuration" section on page 6-38.

Configuring a Hierarchical VP Tunnel for Multiple Service Categories

This section describes configuring a hierarchical VP tunnel for multiple service categories with rate-limited tunnel output.

A hierarchical VP tunnel allows VCs of multiple service categories to pass through the tunnel. In addition, the overall output of the VP tunnel is rate-limited to the PCR of the tunnel. There is no general limit on the number of connections allowed on a such a tunnel. Hierarchical VP tunnels can also support merged VCs for tag switching. See the "Configuring VC Merge" section on page 15-12.

Service categories supported include the following:

- Constant bit rate (CBR)
- Variable bit rate (VBR)

- Available bit rate (ABR) with a nonzero minimum cell rate (MCR)
- Unspecified bit rate (UBR+) with a nonzero MCR

Note Hierarchical VP tunnels are supported only on systems with the FC-PFQ feature card. (Catalyst 8510 MSR and LightStream 1010)

While capable of carrying any traffic category, a hierarchical VP tunnel is itself defined as CBR with a PCR. The following limitations apply on the Catalyst 8540 MSR:

- Hierarchical VP tunnels can be defined only on interfaces in slots 0, 2, 9, and 11.
- For carrier module port adapters, interfaces 0/*x*/*y*, 2/*x*/*y*, 9/*x*/*y*, and 11/*x*/*y* can each support 30 hierarchical VP tunnels, for a combined total of 120. For OC-12 full-width modules, ports 0/0/[0-1], 0/0/[2-3], 2/0/[0-1], 2/0/[2-3], 9/0/[0-1], 9/0/[2-3], 11/0/[0-1], and 11/0/[2-3] can each support 30 hierarchical VP tunnels, for a combined total of 240.

The following limitations apply on the Catalyst 8510 MSR and LightStream 1010:

- A maximum of 30 hierarchical VP tunnels can be defined on interfaces 0/0/x and 3/0/x. A maximum of 30 hierarchical VP tunnels can be defined on interfaces 0/1/x and 3/1/x.
- Hierarchical VP tunnels can be defined only on interfaces in slots 0 and 3.

The following limitations apply on the Catalyst 8540 MSR, Catalyst 8510 MSR and LightStream 1010:

- Only hierarchical VPs are allowed on the interface (not other VCs or VPs).
- Bandwidth allocated on output to a hierarchical VP cannot be used by another hierarchical VP.
- At system boot, when global hierarchical scheduling is enabled, the switch router initializes the slot pairs according to the following restrictions:
  - Hierarchical scheduling is disabled for any slot pair that contains an ATM router module or Ethernet interface module. On the Catalyst 8540 MSR, the slot pairs are slots 0 and 1, slots 2 and 3, slots 9 and 10, and slots 11 and 12. On the Catalyst 8510 MSR and LightStream 1010, the slot pairs are slots 0 and 1 and slots 3 and 4.
  - Hierarchical scheduling is enabled for any slot pair that has an ATM port adapter or interface module in one slot and the other slot empty, or ATM port adapters or interface modules in both slots.
  - If a slot pair is empty, the hierarchical scheduling mode is determined by the first port adapter or interface module that is installed in the slot pair. If you insert an ATM port adapter or interface module first, hierarchical scheduling is enabled; if you insert an ATM router module or Ethernet interface module first, hierarchical scheduling is disabled.
- If hierarchical scheduling is enabled for a slot pair, ATM router modules or Ethernet interface modules inserted into the slot pair do not function.
- If hierarchical scheduling is disabled for a slot pair, ATM port adapters or interface modules inserted into the slot pair do not support hierarchical VP tunnels, and any hierarchical VP tunnels configured for the slot pair do not function.
- Hierarchical VP tunnels support interface overbooking. For configuration information, see the "Configuring Interface Overbooking" section on page 8-37.

Enabling Hierarchical Mode

Before configuring a hierarchical VP tunnel, you must first enable hierarchical mode, then reload the ATM switch router. Perform the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **atm hierarchical-tunnel**	Enables hierarchical mode.
Step 2	Switch(config)# **exit** Switch#	Exits global configuration mode.
Step 3	Switch# **copy system:running-config nvram:startup-config**	Saves the running configuration to the startup configuration.
Step 4	Switch# **reload**	Reloads the operating system.

Note Enabling hierarchical mode causes the minimum rate allocated for guaranteed bandwidth to a connection to be increased.

Example

The following example shows how to enable hierarchical mode, then save and reload the configuration.

```
Switch(config)# atm hierarchical-tunnel
Switch(config)# exit
Switch# copy system:running-config nvram:startup-config
Switch# reload
```

Configuring a Hierarchical VP Tunnel on an Interface

To configure a hierarchical VP tunnel, perform the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **atm connection-traffic-table-row** [**index** *row-index*] **cbr pcr** *rate*	Configures the connection-traffic-table row for the desired PVP CBR cell rate.
Step 2	Switch(config)# **interface atm** *card*/*subcard*/*port* Switch(config-if)#	Selects the physical interface to be configured.
Step 3	Switch(config-if)# **atm pvp** *vpi* **hierarchical rx-cttr** *index* **tx-cttr** *index*	Configures an interface PVP leg.
Step 4	Switch(config-if)# **exit** Switch(config)#	Exits interface configuration mode.
Step 5	Switch(config)# **interface atm** *card*/*subcard*/*port***.***vpt#* Switch(config-subif)#	Creates a hierarchical VP tunnel using a VP tunnel number that matches the PVP leg VPI.

Note The **rx-cttr** and **tx-cttr** row indexes must be configured before they are used.

Example

The following example shows how to configure a hierarchical VP tunnel with a PVP of 99 as ATM interface 0/0/0.99

```
Switch(config)# interface atm 0/0/0
Switch(config-if)# atm pvp 99 hierarchical rx-cttr 100 tx-cttr 100
Switch(config-if)# exit
Switch(config-if)# interface atm 0/0/0.99
Switch(config-subif)#
```

Displaying the Hierarchical VP Tunnel Configuration

To display the hierarchical VP tunnel interface configuration, use the following EXEC command:

Command	Purpose
show atm interface atm *card/subcard/port.vpt#*	Shows the ATM VP interface configuration.

Example

The following example shows the VP tunnel configuration on interface ATM 1/0/0 with PVP 99:

```
Switch# show atm interface atm 1/0/0.99
Interface:      ATM1/0/0.99     Port-type:      vp tunnel
IF Status:      UP              Admin Status:   up
Auto-config:    enabled         AutoCfgState:   waiting for response from peer
IF-Side:        Network         IF-type:        UNI
Uni-type:       Private         Uni-version:    V3.0
Max-VPI-bits:   0               Max-VCI-bits:   14
Max-VP:         0               Max-VC:         16383
ConfMaxSvpcVpi: 0               CurrMaxSvpcVpi: 0
ConfMaxSvccVpi: 0               CurrMaxSvccVpi: 0
ConfMinSvccVci: 35              CurrMinSvccVci: 35
Signalling:     Enabled
ATM Address for Soft VC: 47.0091.8100.0000.0060.3e64.fe01.4000.0c81.9000.63
Configured virtual links:
  PVCLs SoftVCLs   SVCLs   TVCLs Total-Cfgd Inst-Conns
      4        0       0       0          4          4
```

Configuring an End-Point PVC to a PVP Tunnel

To configure an end point of a permanent virtual channel (PVC) to a previously created PVP tunnel, perform the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **interface atm** *card*/*subcard*/*port* Switch(config-if)#	Selects the physical interface to be configured.
Step 2	Switch(config-if)# **atm pvc** *vpi-a vci-a* [**upc** *upc*] [**pd** *pd*] [**rx-cttr** *index*] [**tx-cttr** *index*] **interface atm** *card*/*subcard*/*port***.***vpt# vpi-b vci-b* [**upc** *upc*]	Configures the PVC with the VPI of the tunnel leg matching the tunnel VP tunnel number.

The following restrictions apply to an end point of a PVC-to-PVP tunnel subinterface:

- The VPI number of the tunnel leg of any PVC connection must match the VP tunnel number of the tunnel.
- For single service-category VP tunnels, the service class specified by the connection-traffic-table row (CTTR) of any PVC connections must match the service category for the row(s) selected for the tunnel PVP (for simple VP tunnels), or the configured service category (for shaped VP tunnels). This restriction does not apply to VP tunnels configured for multiple service categories (hierarchical VP tunnels).
- For service classes other than UBR, the PCRs of all PVCs must be within the peak cell rate of the tunnel PVP. This setup requires new CTTR rows to be defined for CBR or VBR PVCs, with peak cell rates that are less than the intended tunnel PVP.

Example

The following example shows how to configure the example tunnel ATM 1/0/0.99 with a PVC from ATM interface 0/0/1 to the tunnel at ATM interface 1/0/0.99:

```
Switch(HB-1)(config)# interface atm 0/0/1
Switch(HB-1)(config-if)# atm pvc 0 50 interface atm 1/0/0.99 99 40
```

Displaying PVCs

To confirm PVC interface configuration, use the following EXEC command:

Command	Purpose
show atm vc interface atm *card*/*subcard*/*port*	Shows the ATM VC interface configuration.

Example

The following example shows the configuration of ATM subinterface 1/0/0.99 on the ATM switch router Switch(HB-1):

```
Switch(HB-1)# show atm vc interface atm 0/0/1
Interface    VPI   VCI   Type    X-Interface  X-VPI X-VCI  Encap Status
ATM0/0/1     0     5      PVC     ATM2/0/0     0     41    QSAAL  UP
ATM0/0/1     0     16     PVC     ATM2/0/0     0     33    ILMI   UP
ATM0/0/1     0     50     PVC     ATM1/0/0.99  99    40           UP
```

Configuring Signalling VPCI for VP Tunnels

You can specify the value of the virtual path connection identifier (VPCI) that is to be carried in the signalling messages within a VP tunnel. The connection identifier information element (IE) is used in signalling messages to identify the corresponding user information flow. The connection identifier IE contains the VPCI and VCI.

Note By default, the VPCI is the same as the VPI on the ATM switch router.

This feature can also be used to support connections over a virtual UNI.

To configure a VP tunnel connection signalling VPCI, perform the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **interface atm** *card/subcard/port***.***vpt#* Switch(config-if)#	Selects the subinterface.
Step 2	Switch(config-if)# **atm signalling vpci** *vpci-number*	Configures the ATM signalling VPCI number 0 to 255.

Example

The following example configures a VP tunnel on ATM interface 0/0/0, PVP 99, and then configures the connection ID VCPI as 0.

```
Switch(config)# interface atm 1/0/0
Switch(config-if)# atm pvp 99
Switch(config-if)# exit
Switch(config)# interface atm 1/0/0.99
Switch(config-subif)# atm signalling vpci 0
Switch(config-subif)# end
```

Displaying the VP Tunnel VPCI Configuration

To confirm the VP tunnel VPCI configuration, use the following privileged EXEC command:

Command	Purpose
more system:running-config	Shows the VP tunnel subinterface configuration.

Deleting VP Tunnels

To delete a VP tunnel connection, perform the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **no interface atm** *card*/*subcard*/*port***.***vpt#*	Deletes the subinterface.
Step 2	Switch(config)# **interface atm** *card*/*subcard*/*port* Switch(config-if)#	Selects the physical interface to be modified.
Step 3	Switch(config-if)# **no atm pvp** *vpi*	Deletes the interface PVP half-leg.

Example

The following example shows deleting subinterface 99 at ATM interface 1/0/0 and then PVP half-leg 99:

```
Switch(HB-1)(config)# no interface atm 1/0/0.99
Switch(HB-1)(config)# interface atm 1/0/0
Switch(HB-1)(config-if)# no atm pvp 99
```

Confirming VP Tunnel Deletion

To confirm the ATM virtual interface deletion, use the following EXEC command:

Command	Purpose
show atm interface [**atm** *card*/*subcard*/*port*[**.***vpt#*]]	Shows the ATM interface configuration.

Example

The following example shows that ATM subinterface 1/0/0.99 on the ATM switch router (HB-1) has been deleted:

```
Switch(HB-1)# show interfaces atm 1/0/0
IF Status:      UP              Admin Status:   up
Auto-config:    disabled        AutoCfgState:   not applicable
IF-Side:        Network         IF-type:        NNI
Uni-type:       not applicable  Uni-version:    not applicable
Max-VPI-bits:   8               Max-VCI-bits:   14
Max-VP:         255             Max-VC:         16383
ConfMaxSvpcVpi: 255             CurrMaxSvpcVpi: 255
ConfMaxSvccVpi: 255             CurrMaxSvccVpi: 255
ConfMinSvccVci: 35              CurrMinSvccVci: 35
Svc Upc Intent: pass            Signalling:     Enabled
ATM Address for Soft VC: 47.0091.8100.0000.00e0.4fac.b401.4000.0c80.8000.00
Configured virtual links:
  PVCLs SoftVCLs   SVCLs   TVCLs   PVPLs SoftVPLs   SVPLs Total-Cfgd Inst-Conns
      4        0       0       0       0        0       0          4          3
Logical ports(VP-tunnels):     0
Input cells:    263843          Output cells:   273010
5 minute input rate:             0 bits/sec,       0 cells/sec
5 minute output rate:            0 bits/sec,       0 cells/sec
Input AAL5 pkts: 172265, Output AAL5 pkts: 176838, AAL5 crc errors: 0
```

Configuring Interface and Connection Snooping

Snooping allows the cells from all connections, in either receive or transmit direction, on a selected physical port to be transparently mirrored to a snoop test port where an external ATM analyzer can be attached. Unlike shared medium LANs, an ATM system requires a separate port to allow nonintrusive traffic monitoring on a line.

Note Only cells that belong to existing connections are sent to the snoop test port. Any received cells that do not belong to existing connections are not copied. In addition, the STS-3c (or other) overhead bytes transmitted at the test port are not copies of the overhead bytes at the monitored port.

Snooping Test Ports (Catalyst 8510 MSR and LightStream 1010)

With the FC-PCQ installed, only the highest port on the last module in the ATM switch router can be configured as a snoop test port. Table 6-4 lists the interface number of the allowed snoop test port for the various port adapter types. If you specify an incorrect snoop test port for the currently installed port adapter type, an error appears on the console. The feature card per-class queuing (FC-PCQ) also does not support per-connection snooping.

The port number of the test port depends on the card type. Table 6-4 lists the allowed snoop test port number for the supported interfaces.

Table 6-4 Allowed ATM Snoop Ports with FC-PCQ

Interface	Port Number
25-Mbps	4/1/11[1]
OC-3	4/1/3
OC-12	4/1/0
DS3/E3	Not supported
CES	Not supported

1. Both transmit and receive interfaces must be on 25-Mbps port adapters.

Effect of Snooping on Monitored Port

There is no effect on cell transmission, interface or VC status and statistics, front panel indicators, or any other parameters associated with a port being monitored during snooping. Any port, other than the highest port, that contains a port adapter type with a bandwidth less than or equal to the port adapter bandwidth for the test port can be monitored by snooping.

Shutting Down Test Port for Snoop Mode Configuration

The port being configured as a test port must be shut down before configuration. While the test port is shut down and after snoop mode has been configured, no cells are transmitted from the test port until it is reenabled using the **no shutdown** command. A test port can be put into snoop mode even if there are existing connections to it; however, those connections remain “Down” even after the test port is reenabled using the **no shutdown** command. This includes any terminating connections for ILMI, PNNI, or signalling channels on the test port.

If you use a **show atm interface** command while the test port is enabled in snoop mode, the screen shows the following:

- Interface state appears as "Snooping" instead of "up" or "down."
- Other ATM layer information for the test port is still displayed.
- Any previously configured connections on the test port remain installed, but are listed as Connection Status = down.
- Data for transmitted cells and output rates indicates the snooping cells are being transmitted.
- Counts for receive cells should remain unchanged and the input rate should be 0.

Other Configuration Options for Snoop Test Port

Most inapplicable configurations on the test port interface are disregarded while in snoop mode. However, the following configuration options are not valid when specified for the snoop test port and may affect the proper operation of the snoop mode on the test port:

- Diagnostic and PIF loopbacks of the snoop test port. These types of loopbacks do not function in snooping mode since the PIF receive side signals are disabled.
- Other physical layer loopbacks (line, cell, or payload) function normally when in snooping mode since they loop toward the line and are unaffected by the lack of PIF receive input.
- Interface pacing (with the rate for the snoop test port lower than the rate for the monitored port).
- Network-derived clock source using the snoop test port.
- Clock-source = loop-timed for the snoop test port.

Caution You should ensure that all options are valid and configured correctly while in the snoop mode.

Configuring Interface Snooping

The **atm snoop interface atm** command enables a snoop test port. Cells transmitted from the snoop test port are copies of cells from a single direction of a monitored port.

When in snoop mode, any prior permanent virtual connections to the snoop test port remain in the down state.

To configure interface port snooping, perform the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **interface atm** *card*/*subcard*/*port* Switch(config-if)#	Selects the interface to be configured.
Step 2	Switch(config-if)# **atm snoop interface atm** *card*/*subcard*/*port* **direction** [**receive** | **transmit**]	Specifies the interface and direction to be snooped.

Example

The following example shows how to configure ATM interface 12/1/3 as the port in snoop mode to monitor ATM interface 3/0/0, tested in the receive direction:

```
Switch(config)# interface atm 12/1/3
Switch(config-if)# atm snoop interface atm 3/0/0 direction receive
```

Displaying Interface Snooping

To display the test port information, use the following EXEC command:

Command	Purpose
show atm snoop	Displays the snoop configuration.

Example

The following example shows the snoop configuration on the OC-3c port and the actual register values for the highest interface:

```
Switch# show atm snoop
Snoop Test Port Name:  ATM12/1/3 (interface status=SNOOPING)
Snoop option:          (configured=enabled) (actual=enabled)
Monitored Port Name:   (configured=ATM3/0/0) (actual=ATM3/0/0)
Snoop direction:       (configured=receive) (actual=receive)
```

Configuring Per-Connection Snooping

With per-connection snooping you must specify both the snooped connection endpoint and the snooping connection endpoint. The Cisco IOS software adds the snooping connection endpoint as a leaf to the snooped connection. The root of the temporary multicast connection depends on the direction being snooped. Snooping in the direction of leaf to root is not allowed for multicast connections. Per-connection snooping features are as follows:

- Per-VC snooping
- Per-VP snooping

The snooping connection can be configured on any port when there is no VPI/VCI collision for the snoop connection with the existing connections on the port. Also the port should have enough resources to satisfy the snoop connection resource requirements. In case of failure, due to VPI/VCI collision or resource exhaustion, a warning message is displayed, and you can reconfigure the connection on a different port.

To snoop both transmit and receive directions of a connection, you need to configure two different snoop connections.

Note Per-connection snooping is available only with the switch processor feature card.

Nondisruptive per-connection snooping is achieved by dynamically adding a leaf to an existing connection (either unicast or multicast). This can lead to cell discard if the added leaf cannot process the snooped cells fast enough. For a multicast connection, the queue buildup is dictated by the slowest leaf in the connection. The leaf added for snooping inherits the same traffic characteristics as the other connection leg. This ensures that the added leaf does not become the bottleneck and affect the existing connection.

To configure connection snooping, perform the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **interface atm** *card*/*subcard*/*port* Switch(config-if)#	Selects the interface to be configured.
Step 2	Switch(config-if)# **atm snoop-vc** [*a-vpi a-vci*] **interface atm** *card*/*subcard*/*port x-vpi x-vci* [**direction** {**receive** | **transmit**}]	Configures the virtual channel to be snooped. *a* denotes the snooping connection. *x* denotes the snooped connection.
Step 3	Switch(config-if)# **atm snoop-vp** [*a-vpi*] **interface atm** *card*/*subcard*/*port x-vpi* [**direction** {**receive** | **transmit**}]	Configures the virtual path to be snooped.

Examples

The following example shows how to configure VC 100 200 on ATM interface 3/1/0 to snoop VC 200 150 on ATM interface 1/0/0:

```
Switch(config)# interface atm 3/1/0
Switch(config-if)# atm snoop-vc 100 200 interface atm 1/0/0 200 150 direction receive
```

The following example shows how to configure VP 100 on ATM interface 3/1/0 to snoop VP 200 on ATM interface 1/0/0:

```
Switch(config)# interface atm 3/1/0
Switch(config-if)# atm snoop-vp 100 interface atm 1/0/0 200 direction receive
```

Displaying Per-Connection Snooping

To display the test per-connection information, use the following EXEC commands:

Command	Purpose
show atm snoop-vc [**interface atm** *card*/*subcard*/*port* [*vpi vci*]]	Displays the snoop VC information.
show atm snoop-vp [**interface atm** *card*/*subcard*/*port* [*vpi*]]	Displays the snoop VP information.

Examples

The following example shows all VC snoop connections on the ATM switch router:

```
Switch> show atm snoop-vc
      Snooping                               Snooped
Interface    VPI   VCI   Type    X-Interface  X-VPI X-VCI Dir    Status
ATM0/0/2     0     5     PVC     ATM0/1/1     0     5     Rx     DOWN
ATM0/0/2     0     16    PVC     ATM0/1/1     0     16    Rx     DOWN
ATM0/1/2     0     5     PVC     ATM0/0/1     0     5     Tx     DOWN
ATM0/1/2     0     16    PVC     ATM0/0/1     0     16    Tx     DOWN
ATM0/1/2     0     18    PVC     ATM0/0/1     0     18    Tx     UP
ATM0/1/2     0     100   PVC     ATM0/0/1     0     100   Tx     DOWN
ATM0/1/2     0     201   PVC     ATM0/0/1     0     201   Tx     DOWN
ATM0/1/2     0     202   PVC     ATM0/0/1     0     202   Tx     DOWN
ATM0/1/2     0     300   PVC     ATM0/0/1     0     300   Tx     DOWN
ATM0/1/2     0     301   PVC     ATM0/0/1     0     301   Tx     DOWN
```

The following example shows the VC snoop connections on ATM interface 0/1/2:

```
Switch> show atm snoop-vc interface atm 0/1/2
      Snooping                               Snooped
Interface    VPI   VCI   Type    X-Interface  X-VPI X-VCI Dir    Status
ATM0/1/2     0     5     PVC     ATM0/0/1     0     5     Tx     DOWN
ATM0/1/2     0     16    PVC     ATM0/0/1     0     16    Tx     DOWN
ATM0/1/2     0     18    PVC     ATM0/0/1     0     18    Tx     UP
ATM0/1/2     0     100   PVC     ATM0/0/1     0     100   Tx     DOWN
ATM0/1/2     0     201   PVC     ATM0/0/1     0     201   Tx     DOWN
ATM0/1/2     0     202   PVC     ATM0/0/1     0     202   Tx     DOWN
ATM0/1/2     0     300   PVC     ATM0/0/1     0     300   Tx     DOWN
ATM0/1/2     0     301   PVC     ATM0/0/1     0     301   Tx     DOWN
```

The following example shows the VC snoop connection 0, 55 on ATM interface 0/0/2 in extended mode with the switch processor feature card installed:

```
Switch> show atm snoop-vc interface atm 0/0/2 0 55
Interface: ATM0/0/2, Type: oc3suni
VPI = 0  VCI = 55
Status: DOWN
Time-since-last-status-change: 00:01:59
Connection-type: PVC
Cast-type: snooping-leaf
Packet-discard-option: disabled
Usage-Parameter-Control (UPC): pass
Wrr weight: 32
Number of OAM-configured connections: 0
OAM-configuration: disabled
OAM-states:  Not-applicable
Cross-connect-interface: ATM0/1/1, Type: oc3suni
Cross-connect-VPI = 0
Cross-connect-VCI = 5
Cross-connect-UPC: pass
Cross-connect OAM-configuration: disabled
Cross-connect OAM-state:  Not-applicable
Threshold Group: 6, Cells queued: 0
Rx cells: 0, Tx cells: 0
Tx Clp0:0,  Tx Clp1: 0
Rx Clp0:0,  Rx Clp1: 0
Rx Upc Violations:0, Rx cell drops:0
Rx Clp0 q full drops:0, Rx Clp1 qthresh drops:0
Rx connection-traffic-table-index: 3
Rx service-category: VBR-RT (Realtime Variable Bit Rate)
Rx pcr-clp01: 424
Rx scr-clp01: 424
```

```
Rx mcr-clp01: none
Rx      cdvt: 1024 (from default for interface)
Rx       mbs: none
Tx connection-traffic-table-index: 3
Tx service-category: VBR-RT (Realtime Variable Bit Rate)
Tx pcr-clp01: 424
Tx scr-clp01: 424
Tx mcr-clp01: none
Tx      cdvt: none
Tx       mbs: none
```

The following example shows all VP snoop connections on the ATM switch router:

```
Switch> show atm snoop-vp
      Snooping                    Snooped
Interface    VPI   Type    X-Interface  X-VPI Dir    Status
ATM0/1/2     57    PVP     ATM0/0/1     57    Tx     DOWN
```

The following example shows all VP snoop connections on ATM interface 0/1/2, VPI = 57, in extended mode with the switch processor feature card installed:

```
Switch> show atm snoop-vp interface atm 0/1/2 57
Interface: ATM0/1/2, Type: oc3suni
VPI = 57
Status: DOWN
Time-since-last-status-change: 00:14:46
Connection-type: PVP
Cast-type: snooping-leaf
Usage-Parameter-Control (UPC): pass
Wrr weight: 32
Number of OAM-configured connections: 0
OAM-configuration: disabled
OAM-states:  Not-applicable
Cross-connect-interface: ATM0/0/2, Type: oc3suni
Cross-connect-VPI = 57
Cross-connect-UPC: pass
Cross-connect OAM-configuration: disabled
Cross-connect OAM-state:  Not-applicable
Threshold Group: 5, Cells queued: 0
Rx cells: 0, Tx cells: 0
Tx Clp0:0,  Tx Clp1: 0
Rx Clp0:0,  Rx Clp1: 0
Rx Upc Violations:0, Rx cell drops:0
Rx Clp0 q full drops:0, Rx Clp1 qthresh drops:0
Rx connection-traffic-table-index: 1
Rx service-category: UBR (Unspecified Bit Rate)
Rx pcr-clp01: 7113539
Rx scr-clp01: none
Rx mcr-clp01: none
Rx      cdvt: 1024 (from default for interface)
Rx       mbs: none
Tx connection-traffic-table-index: 1
Tx service-category: UBR (Unspecified Bit Rate)
Tx pcr-clp01: 7113539
Tx scr-clp01: none
Tx mcr-clp01: none
Tx      cdvt: none
Tx       mbs: none
```

Configuring Operation, Administration, and Maintenance

This chapter describes the Operation, Administration, and Maintenance (OAM) implementation on the ATM switch router.

Note This chapter provides advanced configuration instructions for the Catalyst 8540 MSR, Catalyst 8510 MSR, and LightStream 1010 ATM switch routers. For complete descriptions of the commands mentioned in this chapter, refer to the *ATM Switch Router Command Reference* publication.

This chapter includes the following sections:

OAM Overview

OAM performs fault management and performance management functions at the ATM management (M)-plane layer.

Note Current OAM implementation supports only the fault management function, which includes connectivity verification and alarm surveillance.

The ATM switch router has full support for the following ATM OAM cell flows:

- F4 flows—OAM information flows between network elements (NEs) used within virtual paths to report an unavailable path or a virtual path (VP) that cannot be guaranteed.
- F5 flows—OAM information flows between network elements (NEs) used within virtual connections to report degraded virtual channel (VC) performance such as late arriving cells, lost cells, and cell insertion problems.

Both F4 and F5 flows can be configured as either end-to-end or segment-loopback and used with alarm indication signal (AIS) and remote defect indication (RDI) functions. An AIS is a signal transmitted downstream informing the destination that an upstream failure has been detected. An RDI signal indicates that a failure has occurred at the far end of an ATM network.

Note Cells can be sent either on demand or periodically to verify link and connection integrity.

In addition to the standard OAM functions, the ATM switch router can also send OAM pings. OAM cells containing the ATM node addresses or IP addresses of intermediate switches allow network administrators to determine the integrity of a chosen connection at any intermediate point along the connection, allowing for network connection debugging and troubleshooting.

OAM software implements ATM Layer F4 and F5 OAM fault management functions. OAM performs standard loopback (end-to-end or segment) and fault detection and notification (AIS and RDI) for each connection. It also maintains a group of timers for the OAM functions. When there is an OAM state change such as loopback failure, OAM software notifies the connection management software.
The network operator can enable or disable OAM operation for the following switch components:

- The entire switch
- A specific ATM interface
- A specific ATM connection

If OAM operation is disabled, outgoing OAM cells (AIS, RDI and loopbacks) are not generated and AIS and RDI cells that arrive at connection endpoints are discarded.

To support various OAM operations, the ATM switch router hardware provides OAM cell routing functions on a per-connection basis for each direction and for different OAM cell spans (segment and end-to-end). The hardware OAM cell routing determines the destination of an OAM cell received from the link or the network and then determines whether OAM cells are processed by the switch software.

The hardware can perform the following functions on OAM cells:

- Intercept—Intercepted to the CPU queue and processed by the ATM switch router software
- Relay—Relayed along with user cell by hardware without any software processing
- Discard—Discarded by hardware

An ATM connection consists of a group of network points that form the edges of each ATM switch or end system.

Each point can be one of the following:

- Connection end point—The end of a connection where the user ATM cells are terminated
- Segment end point—The end of a connection segment
- Connecting point—The middle point of a connection segment

The following sections describe the OAM tasks:

- Configuring OAM Functions, page 7-3
- Checking the ATM Connection (Catalyst 8510 MSR and LightStream 1010), page 7-6
- Displaying the OAM Configuration, page 7-7

Configuring OAM Functions

This section describes OAM commands in EXEC, global, and interface configuration mode.

Configuring OAM for the Entire Switch (Catalyst 8540 MSR)

To enable OAM operations for the Catalyst 8540 MSR, use the global configuration command, as shown in the following table:

Command	Purpose
atm oam [**ais**] [**end-loopback**] [**max-limit** *number*] [**rdi**] [**seg-loopback**]	Enables or disables OAM operations for the entire switch.

Note The number of maximum OAM configured connections allowed ranges from 1 to 3200; the default is 3200.

Examples

The following example shows how to enable AIS and segment loopback for the entire switch:

```
Switch(config)# atm oam ais seg-loopback
% OAM: Switch level seg loopback is enabled

% OAM: Switch level ais is enabled
```

The following example shows how to configure the ATM OAM connection maximum to 1600:

```
Switch(config)# atm oam max-limit 1600
```

Configuring OAM for the Entire Switch (Catalyst 8510 MSR and LightStream 1010)

To enable OAM operations for the entire Catalyst 8510 MSR and LightStream 1010 ATM switch router, use the global configuration command, as shown in the following table:

Command	Purpose
atm oam [**ais**] [**end-loopback**] [**intercept end-to-end**] [**max-limit** *number*] [**rdi**] [**seg-loopback**]	Enables or disables OAM operations for the entire switch.

Note The number of maximum OAM configured connections allowed ranges from 1 to 3200; the default is 3200.

Examples

The following example shows how to enable AIS and segment loopback for the entire switch:

```
Switch(config)# atm oam ais seg-loopback
% OAM: Switch level seg loopback is enabled

% OAM: Switch level ais is enabled
```

The following example shows how to configure the ATM OAM connection maximum to 1600:

```
Switch(config)# atm oam max-limit 1600
```

Configuring the Interface-Level OAM

To enable OAM operations on an interface, perform the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **interface atm** *card*/*subcard*/*port*[**.***vpt#*] Switch(config-if)#	Selects the interface to be configured.
Step 2	Switch(config-if)# **atm oam** [**interface atm** *card*/*subcard*/*port*[**.***vpt#*]] [*vpi* [*vci*]] [**ais**] [**end-loopback**] [**rdi**] [**seg-loopback**]	Configures interface OAM operations.
Step 3	Switch(config-if)# **atm oam** *vpi* [*vci*] **loopback-timer** *tx-timer-value*	Configures the OAM loopback transmit timer.

Examples

The following example shows how to enable OAM AIS and end-to-end loopback on interface 3/0/0:

```
Switch(config)# interface atm 3/0/0
Switch(config-if)# atm oam ais end-loopback
% OAM: Interface level end to end loopback is enabled

% OAM: Interface level ais is enabled
```

The following example shows how to enable OAM AIS and end-to-end loopback on interface 3/0/0, VPI = 50, VCI = 100:

```
Switch(config)# interface atm 3/0/0
Switch(config-if)# atm oam 50 100 ais end-loopback
% OAM: Connection level end to end loopback is enabled

% OAM: Connection level ais is enabled
```

Note You can use only VPI values to configure OAM operations on VP connections.

In interface configuration command mode, you can enable or disable OAM operations on existing connections on different interfaces by specifying **interface atm** *card*/*subcard*/*port*. The following example disables OAM AIS flows at interface 1/0/0 while in interface 3/0/0:

```
Switch(config)# interface atm 3/0/0
Switch(config-if)# no atm oam interface atm 1/0/0 ais
% OAM: Interface level ais is disabled
```

Checking the ATM Connection (Catalyst 8540 MSR)

To check ATM connection reachability and network connectivity on the Catalyst 8540 MSR, use the **ping** EXEC command, as shown in the following table:

Command	Purpose
ping atm interface atm *card*/*subcard*/*port vpi* [*vci*] {**end-loopback** [*destination*] | **seg-loopback** [*destination*]}	Checks the connection.

You can ping a neighbor switch by selecting the segment loopback option. In privileged EXEC mode, you can select various other parameters such as repeat count and timeout values.

Examples

The following example shows the **ping** command used in normal mode to check a virtual channel connection (VCC) with a segment loopback flow:

```
Switch# ping atm interface atm 3/0/0 50 100 seg-loopback

Type escape sequence to abort.
Sending Seg-Loopback 5, 53-byte OAM Echoes to a neighbor, timeout is 5 seconds:
.....
Success rate is 0 percent (0/5)
```

The following example shows the **ping** command used in extended mode to check a VCC with end-to-end loopback flow:

```
Switch# ping
Protocol [ip]: atm
Interface [card/sub-card/port]: 3/0/0
VPI [0]: 0
VCI [0]: 16
Send OAM-Segment-Loopback ? [no]:
Target IP address:
Target NSAP Prefix:
Repeat count [5]:
Timeout in seconds [5]:
Type escape sequence to abort.
Sending end-Loopback 5, 53-byte OAM Echoes to a connection end point, timeout is
 5 seconds:
.....
Success rate is 0 percent (0/5)
```

Checking the ATM Connection (Catalyst 8510 MSR and LightStream 1010)

To check ATM connection reachability and network connectivity on the Catalyst 8510 MSR and LightStream 1010 ATM switch router, use the **ping** EXEC command, as shown in the following table:

Command	Purpose
ping atm interface atm *card*/*subcard*/*port* *vpi* [*vci*] {[**atm-prefix** *prefix*] | **end-loopback** [*destination*] | **ip-address** *ip-address* | **seg-loopback** [*destination*]}	Checks the connection.

You can use either an ATM address prefix or an IP address as a ping destination. You can ping a neighbor switch by selecting the segment loopback option. In privileged EXEC mode, you can select various other parameters such as repeat count and timeout values.

Examples

The following example shows the **ping** command used in normal mode to check a VCC with a segment loopback flow:

```
Switch# ping atm interface atm 3/0/0 50 100 seg-loopback

Type escape sequence to abort.
Sending Seg-Loopback 5, 53-byte OAM Echoes to a neighbor, timeout is 5 seconds:
.....
Success rate is 0 percent (0/5)
```

The following example shows the **ping** command used in extended mode to check a VCC with end-to-end loopback flow:

```
Switch# ping
Protocol [ip]: atm
Interface [card/sub-card/port]: 3/0/0
VPI [0]: 0
VCI [0]: 16
Send OAM-Segment-Loopback ? [no]:
Target IP address:
Target NSAP Prefix:
Repeat count [5]:
Timeout in seconds [5]:
Type escape sequence to abort.
Sending end-Loopback 5, 53-byte OAM Echoes to a connection end point, timeout is
 5 seconds:
.....
Success rate is 0 percent (0/5)
```

Note If you do not enable the OAM segment loopback option, the **ping** command uses an OAM end-to-end loopback cell. If you do not provide a target address, the connection end point becomes the target.

Displaying the OAM Configuration

To display the OAM configuration, use the following EXEC command:

Command	Purpose
more system:running-config	Displays the OAM configuration.

Example

The OAM configuration is displayed in the following example:

```
Switch# more system:running-config
Building configuration...

Current configuration:
!
version XX.X
no service pad
service udp-small-servers
service tcp-small-servers
!
hostname Switch
!
boot system flash slot0:rhino/ls1010-wi-m_1.083.bin.Z
!
ip rcmd remote-username doug
atm oam max-limit 1600
atm over-subscription-factor 16
atm service-category-limit cbr 3000
atm qos uni3-default  cbr max-cell-loss-ratio 12
atm lecs-address 47.0091.0000.0000.0000.0000.0000.0000.0000.0000.00
atm address 47.0091.8100.0000.0060.3e5a.db01.0060.3e5a.db01.00
!
interface ATM0/0/0
 no keepalive
 map-group atm-1
 no atm auto-configuration
 no atm address-registration
 no atm ilmi-enable
 no atm ilmi-lecs-implied
 atm iisp side user
 atm pvp 99
 atm oam 0 5 seg-loopback  end-loopback  rdi
 atm oam 0 16 seg-loopback  end-loopback  rdi
 atm oam 0 18 seg-loopback  end-loopback  rdi
!
interface ATM0/0/0.99 point-to-point
 no atm auto-configuration
 no atm address-registration
 no atm ilmi-enable
 no atm ilmi-lecs-implied
 atm maxvp-number 0
 atm oam 99 5 end-loopback  rdi
 atm oam 99 16 end-loopback  rdi
 atm oam 99 18 end-loopback  rdi
!
--More--

<information deleted>
```

CHAPTER 8

Configuring Resource Management

This chapter describes resource management, which involves modeling and managing switch, interface, and connection resources. Such resources include equivalent bandwidth and buffering to support the provision of specified traffic classes.

Note This chapter provides advanced configuration instructions for the Catalyst 8540 MSR, Catalyst 8510 MSR, and LightStream 1010 ATM switch routers. For detailed descriptions of traffic management mechanisms and their operation, refer to the *Guide to ATM Technology*. For complete descriptions of the commands mentioned in this chapter, refer to the *ATM Switch Router Command Reference* publication.

This chapter includes the following sections:

Note The traffic and resource management features of the ATM switch router are presented in a different order in this guide and in the *Guide to ATM Technology*. In this guide the sequence of features follows configuration scope and proceeds from global to per-interface features. In the *Guide to ATM Technology* the sequence of features follows the phases of a connection and proceeds from traffic contract to management of hardware resources.

Resource Management Functions

The ATM switch router resource management software provides the following functions:

- Network management interface—Includes operational configuration changes (take place immediately), proposed configuration changes (take place on restart), user interface, and status.
- Default quality of service (QoS) objective table management—Since User-Network Interface 3 (UNI 3) signalling does not provide information elements to signal QoS values, resource management provides a table that contains default values for QoS.
- Connection Traffic Table (CTT) management—Rather than store traffic parameters for each connection in that connection's data structure, resource management manages a table of connection traffic parameters, used by network and connection management.
- Hardware resource management (Catalyst 8540 MSR)—The switch processor feature card provides functionality that include statistic collection, and traffic policing usage parameter control (UPC). See the "Configuring Global Resource Management" section on page 8-4 for detailed information.
- Hardware resource management (Catalyst 8510 MSR and LightStream 1010)—Different sets of functionality are available with feature card per-class queueing (FC-PCQ) and feature card per-flow queueing (FC-PFQ). FC-PCQ features include switch cell priority limits, interface queue sizes, and thresholds. FC-PFQ features include threshold group configuration. The interface pacing feature is available with both feature cards. See the "Processor Feature Card Functionality (Catalyst 8510 MSR and LightStream 1010)" section on page 8-3 for detailed information.
- Resource Call Admission Control (RCAC)—Determines whether a virtual channel connection/virtual path connection (VCC/VPC) can be admitted (allowed to be set up), based on the available connection resources and requested traffic characteristics.
- Logical interface creation and deletion.
- Private Network-Network Interface (PNNI) metrics—resource management supplies PNNI with link metrics for connection routing.

Switch Fabric Functionality (Catalyst 8540 MSR)

The switch fabric for the Catalyst 8540 MSR provides the required ATM Forum Traffic Management features as described in Table 8-1.

Table 8-1 Switch Processor Feature Card

Feature	Description
Traffic classes:	CBR[1], VBR-RT[2], VBR-NRT[3], UBR[4], ABR[5](EFCI)[6]
Output queuing	Per-VC or per-VP
Output scheduling	RS[7] and WRR[8]
Intelligent early packet discard	Multiple dynamic thresholds
Intelligent tail (partial) packet discard	Supported
Selective cell marking and discard	Multiple, weighted, dynamic thresholds

Table 8-1 Switch Processor Feature Card (continued)

Feature	Description
Shaping	Per-port pacing, per-CBR VC, per-CBR transit VP, per-shaped CBR VP tunnel (128 shaped VP tunnels total), and hierarchical VP tunnels
Policing (UPC[9])[10]	Dual leaky bucket
Frame mode VC-merge	Supported
Point-to-multipoint VC (multicast)	Multiple leafs per output port, per point-to-multipoint
Network clock switchover[10]	Programmable clock selection criteria
Nondisruptive snooping	Per-VC or per-VP
Hierarchical VP tunnel	Maximum of 240 VP tunnels.

1. CBR = constant bit rate
2. VBR-RT = variable bit rate real time
3. VBR-NRT = variable bit rate non-real time
4. UBR = unspecified bit rate
5. ABR = available bit rate
6. EFCI = explicit forward congestion indication
7. RS = rate scheduling
8. WRR = weighted round-robin
9. UPC = usage parameter control
10. Performed by feature card

Processor Feature Card Functionality (Catalyst 8510 MSR and LightStream 1010)

Two types of feature cards are available for the Catalyst 8510 MSR and LightStream 1010 ATM switch routers: FC-PCQ and FC-PFQ. Each card provides the required ATM Forum Traffic Management features. FC-PCQ contains a subset of the FC-PFQ features, as described in Table 8-2.

Note To determine which feature card you have installed, enter the **show hardware** EXEC command. Either FeatureCard1, for FC-PCQ, or FC-PFQ displays in the Ctrlr-Type column.

Table 8-2 FC-PCQ and FC-PFQ Feature Comparison

Feature	FC-PCQ	FC-PFQ
Traffic classes	CBR[1], VBR-RT[2], VBR-NRT[3], ABR[4] (EFCI[5] and RR[6]), UBR[7]	CBR, VBR-RT, VBR-NRT, ABR (EFCI and RR), UBR
Output queuing	Four classes per port	Per-VC or per-VP
Output scheduling	SP[8]	RS[9] and WRR[10]
Intelligent early packet discard	Multiple fixed thresholds	Multiple dynamic thresholds

Table 8-2 FC-PCQ and FC-PFQ Feature Comparison (continued)

Feature	FC-PCQ	FC-PFQ
Intelligent tail (partial) packet discard	Supported	Supported
Selective cell marking and discard	Multiple fixed thresholds	Multiple, weighted, dynamic thresholds
Shaping	Per-port (pacing)	Per-port pacing, per-CBR VC, per-CBR transit VP, per-shaped CBR VP tunnel (128 shaped VP tunnels total), and hierarchical VP tunnels
Policing (UPC[11])	Dual mode, single leaky bucket	Dual leaky bucket
Point-to-multipoint VC (multicast)	One leaf per output port, per point-to-multipoint	Multiple leafs per output port, per point-to-multipoint
Network clock switch over	Automatic upon failure	Programmable clock selection criteria
Nondisruptive snooping	Per-port transmit or receive	Per-VC or per-VP
Hierarchical VP tunnel[12]	–	Maximum of 62 VP tunnels

1. CBR = constant bit rate
2. VBR-NT = variable bit rate real time
3. VBR-NRT = variable bit rate non-real time
4. ABR = available bit rate
5. EFCI = explicit forward congestion indication
6. RR = relative rate
7. UBR = unspecified bit rate
8. SP = strict priority
9. RS = rate scheduling
10. WRR = weighted round-robin
11. UPC = usage parameter control
12. Available with FC-PFQ only

Configuring Global Resource Management

Global resource management configurations affect all interfaces on the switch. The following sections describe global resource management tasks:

- Configuring the Default QoS Objective Table, page 8-5
- Configuring the Switch Oversubscription Factor (Catalyst 8510 MSR and LightStream 1010), page 8-6
- Configuring the Service Category Limit (Catalyst 8510 MSR and LightStream 1010), page 8-7
- Configuring the ABR Congestion Notification Mode (Catalyst 8510 MSR and LightStream 1010), page 8-8
- Configuring the Connection Traffic Table, page 8-10

Configuring the Default QoS Objective Table

Resource management provides a table of default objective values for quality of service (QoS) for guaranteed service categories. These values—either metrics or attributes—are used as the criteria for connection setup requirements.

Note Default objective values for QoS for guaranteed service categories can be configured for UNI 4.0 signalling.

Table 8-3 lists the default values of the QoS objective table.

Table 8-3 Default QoS Objective Table Row Contents

Service Category	Max Cell Transfer Delay (clp01)	Peak-to-Peak Cell Delay Variation (clp01)	Cell Loss Ratio (clp0)	Cell Loss Ratio (clp0+1)
CBR	Undefined	Undefined	Undefined	Undefined
VBR-RT	Undefined	Undefined	Undefined	Undefined
VBR-NRT	—	—	Undefined	Undefined

Each objective can have a defined or undefined value. If undefined, the objective is not considered in connection setup. The table should be configured with the same values for an entire network.

To configure the default QoS objective table, perform the following tasks in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **atm qos default** {**cbr** | **vbr-rt**} **max-cell-transfer-delay** {*microseconds* | **any**}	Selects the ATM QoS default CBR or VBR-RT maximum cell transfer delay.
Step 2	Switch(config)# **atm qos default** {**cbr** | **vbr-rt**} **peak-to-peak- cell-delay variation** {*microseconds* | **any**}	Selects the ATM QoS default CBR or VBR-RT peak-to-peak cell delay variation.
Step 3	Switch(config)# **atm qos default** {**cbr** | **vbr-rt** | **vbr-nrt**} **max-cell-loss-ratio** [**clp0** | **clp1plus0**] {*loss-ratio-exponent* | **any**}	Selects the ATM QoS default CBR, VBR-RT, or VBR-NRT maximum cell loss ratio.

Example

The following example shows how to change the constant bit rate (CBR) maximum cell loss ratio objective for cell loss priority (CLP) = 0+1 to 10^{-12} cells per second:

```
Switch(config)# atm qos default cbr max-cell-loss-ratio clp1plus0 12
```

Displaying the ATM QoS Objective Table

To display the default QoS objective table, use the following EXEC command:

Command	Purpose
show atm qos-defaults	Displays the ATM QoS objective table configuration.

The per-service category, maximum cell transfer delay, peak-to-peak cell delay variation, and maximum cell loss ratio objectives are displayed.

Example

The ATM QoS objective table configuration is displayed in the following example:

```
Switch> show atm qos-defaults
Default QoS objective table:
    Max cell transfer delay (in microseconds): any cbr, any vbr-rt
    Peak-to-peak cell  delay variation (in microseconds): any cbr, any vbr-rt
    Max cell loss ratio for CLP0 cells: any cbr, any vbr-rt, any vbr-nrt
    Max cell loss ratio for CLP0+1 cells:  10**(-12) cbr, any vbr-rt, any vbr-nrt
```

Configuring the Switch Oversubscription Factor (Catalyst 8510 MSR and LightStream 1010)

The switch oversubscription factor (OSF) feature on the Catalyst 8510 MSR and LightStream 1010 ATM switch routers is used in determining initial port maximum queue sizing for variable bit rate non-real time (VBR-NRT) and available bit rate/unspecified bit rate (ABR/UBR) queues.

Note Over subscription factor configuration is only possible on switches with FC-PCQ installed.

The size of the VBR-NRT queue and ABR/UBR queues is determined by the following equations, where the default size of the CBR and VBR-RT queues vary by interface type, as listed in Table 8-4:

```
Default Size (VBR-NRT) = 0.25 * ((OSF * 2048) - DefaultSize(CBR) - DefaultSize (VBR-RT))
Default Size (ABR-UBR) = 0.75 * ((OSF * 2048) - DefaultSize(CBR) - DefaultSize (VBR-RT))
```

Table 8-4 Default CBR and VBR Determined by Interface Type

Interface Type	Default Max Size CBR Queue	Default Max Size Type VBR-RT Queue
SONET	256	256
DS3/E3	256	512

To configure the OSF, use the following global configuration command:

Command	Purpose
atm over-subscription-factor *o-value*	Configures the switch OSF from 1 to 32.

Note This value can be changed at any time, but it is only used at start-up and when a module is hot-swapped from the chassis.

Example

The following example shows how to set the switch oversubscription factor to 16:

```
Switch(config)# atm over-subscription-factor 16
```

Displaying the OSF Configuration (Catalyst 8510 MSR and LightStream 1010)

To display the OSF configuration, use the following EXEC command:

Command	Purpose
show atm resource	Displays the OSF configuration.

Note The following examples differ depending on the feature card installed in your switch.

Examples

The following example shows the switch OSF configuration with FC-PCQ installed:

```
Switch> show atm resource
Resource configuration:
    Over-subscription-factor 16  Sustained-cell-rate-margin-factor 1%
    Abr-mode:   relative-rate
    Atm service-category-limit (in cells):
         64544 cbr 64544 vbr-rt 64544 vbr-nrt 64544 abr-ubr
Resource state:
    Cells per service-category:
         0 cbr 0 vbr-rt 0 vbr-nrt 0 abr-ubr
```

Configuring the Service Category Limit (Catalyst 8510 MSR and LightStream 1010)

The service category limit configuration restricts the number of cells admitted into the switch, as determined by the type of output queues.

Note Service category limit configuration is only possible on switches with FC-PCQ installed.

Caution Setting a service category limit to 0 causes the connection requests for the associated service categories to be rejected.

To configure the service category limits, use the following global configuration command:

Command	Purpose
atm service-category-limit {**cbr** | **vbr-rt** | **vbr-nrt** | **abr-ubr**} *value*	Configures ATM service category limits for a specific output queue.

Note The **atm service-category-limit** command affects all connections, including those already established.

Example

The following example shows how to change the service category limit for the CBR cells within the switch fabric to 3000 cells:

```
Switch(config)# atm service-category-limit cbr 3000
```

Displaying the Service Category Limit Configuration (Catalyst 8510 MSR and LightStream 1010)

To display the service category limit configuration, use the following EXEC command:

Command	Purpose
show atm resource	Displays the service category limits configuration.

Example

The following example shows the service category limits configuration:

```
Switch> show atm resource
Resource configuration:
    Over-subscription-factor 16  Sustained-cell-rate-margin-factor 1%
    Abr-mode:   relative-rate
➔   Atm service-category-limit (in cells):
         3000 cbr 64544 vbr-rt 64544 vbr-nrt 64544 abr-ubr
Resource state:
    Cells per service-category:
         0 cbr 0 vbr-rt 0 vbr-nrt 0 abr-ubr
```

Configuring the ABR Congestion Notification Mode (Catalyst 8510 MSR and LightStream 1010)

The available bit rate (ABR) congestion notification mode changes the type of notification used on ABR connections to alert the end station of congestion. ABR mode configuration determines whether ABR uses explicit forward congestion indication (EFCI) marking, relative-rate marking, or both, for rate management on ABR connections.

The global configuration function is used to modify the ABR mode selection for all ABR connections.

To configure the ABR mode, use the following global configuration command:

Command	Purpose
atm abr-mode {**efci** | **relative-rate** | **all**}	Configures ABR congestion notification mode.

Note The **atm abr-mode** command affects all connections, including those already established.

Example

The following example shows how to configure the entire switch to set the EFCI bit whenever a cell arrives on a congested ABR connection:

```
Switch(config)# atm abr-mode efci
```

Displaying the ABR Congestion Notification Mode Configuration (Catalyst 8510 MSR and LightStream 1010)

To display the ABR congestion notification mode configuration, use the following EXEC command:

Command	Purpose
show atm resource	Displays the ABR congestion notification mode configuration.

Note The following examples differ depending on the feature card installed in your switch.

Examples

The following example shows the ABR mode configuration with FC-PCQ installed:

```
Switch> show atm resource
Resource configuration:
    Over-subscription-factor 16  Sustained-cell-rate-margin-factor 1%
➔   Abr-mode: efci
    Atm service-category-limit (in cells):
         3000 cbr 64544 vbr-rt 64544 vbr-nrt 64544 abr-ubr
Resource state:
    Cells per service-category:
         0 cbr 0 vbr-rt 0 vbr-nrt 0 abr-ubr
```

The following example shows the ABR mode configuration with FC-PFQ installed:

```
Switch> show atm resource
Resource configuration:
    Over-subscription-factor 8  Sustained-cell-rate-margin-factor 1%
➔   Abr-mode:   efci
    Service Category to Threshold Group mapping:
     cbr 1 vbr-rt 2 vbr-nrt 3 abr 4 ubr 5
    Threshold Groups:
    Group Max    Max Q  Min Q  Q thresholds  Cell  Name
          cells  limit  limit  Mark Discard  count
          instal instal instal
    ---------------------------------------------------
     1    65535  63     63     25 %  87 %      0     cbr-default-tg
     2    65535  127    127    25 %  87 %      0     vbrrt-default-tg
     3    65535  511    31     25 %  87 %      0     vbrnrt-default-tg
     4    65535  511    31     25 %  87 %      0     abr-default-tg
     5    65535  511    31     25 %  87 %      0     ubr-default-tg
     6    65535  1023   1023   25 %  87 %      0     well-known-vc-tg
```

Configuring the Connection Traffic Table

A row in the connection traffic table (CTT) must be created for each unique combination of traffic parameters. Virtual path links (VPLs) and virtual channel links (VCLs) then specify traffic by specifying a row in the table per flow (receive and transmit). Many VCL/VPLs can refer to the same row in the traffic table.

The following two subsections outline the differences in the CTT feature according to platform and feature card.

CTT Supported Features (Catalyst 8540 MSR)

The rows corresponding to various service categories support the following features on the Catalyst 8540 MSR.

- Non-zero minimum cell rate (MCR) for UBR+ service categories. UBR+ is a variant of UBR, in which peak cell rate (PCR), MCR, and cell delay variation tolerance (CDVT) are specified in the traffic contract, with a guarantee on MCR.
- Both CDVT and maximum burst size (MBS) for VBR rows. Dual-leaky-bucket UPC is allowed.
- Whether SCR applies to either the CLP0 or CLP0+1 flow of cells. Only one or the other of these flows can be policed.

CTT Supported Features (Catalyst 8510 MSR and LightStream 1010)

ATM switch routers with feature card per-flow queuing (FC-PFQ) and software version 11.2(8) or later have more rows of various service categories that allow you to specify the following features:

- Non-zero minimum cell rate (MCR) for ABR and UBR+ service categories. UBR+ is a variant of UBR, in which peak cell rate (PCR), MCR, and cell delay variation tolerance (CDVT) are specified in the traffic contract, with a guarantee on MCR.
- Both CDVT and maximum burst size (MBS) for VBR rows. FC-PFQ allows dual-leaky-bucket UPC.
- Whether SCR applies to either the CLP0 or CLP0+1 flow of cells. FC-PFQ can police one or the other of these flows.

If your switch has FC-PCQ installed on the route processor you cannot take advantage of these new capabilities. CTT rows specifying these new parameters can be configured with FC-PCQ installed, with the following effect:

- Non-zero MCR is not supported. Requests for connections specifying non-zero MCR are rejected.
- On VBR connections, only SCR and MBS are used for UPC, and policing is done only on the CLP0+1 flow of cells.

PVC Connection Traffic Rows

The CTT in a permanent virtual channel (PVC) setup requires storing PVC traffic values in a CTT data structure. Rows used for PVCs are called stable rows, and contain traffic parameters.

SVC Connection Traffic Rows

The CTT in a switched virtual channel (SVC) setup provides a row identifier that Simple Network Management Protocol (SNMP) or the user interface can use to read or display SVC traffic parameters. A CTT row index is stored in the connection-leg data structure for each flow of the connection.

Note Rows cannot be deleted while in use by a connection.

CTT Row Allocations and Defaults

To make CTT management software more efficient, the CTT row-index space is split into rows allocated as a result of signalling and rows allocated from the command-line interface (CLI) and SNMP. Table 8-5 describes the row-index range for both.

Table 8-5 CTT Row-Index Allocation

Allocated by	Row-index range
ATOMMIB Traffic Descriptor Table or CLI connection-traffic-table-row creation	1 through 1,073,741,823
Signalling VxL creation	1,073,741,824 through 2,147,483,647

Table 8-6 describes the well-known, predefined ATM CTT rows.

Table 8-6 Default ATM Connection Traffic Table Rows

CTT Row Index	Service Category	Peak-Cell-Rate (clp01)	Sustained-Cell-Rate (clp01)	Tolerance	Use
1	UBR	7,113,539	—	None	Default PVP/PVC row index
2	CBR	424 kbps	—	None	CBR tunnel well-known (WK) VCs

Table 8-6 Default ATM Connection Traffic Table Rows (continued)

CTT Row Index	Service Category	Peak-Cell-Rate (clp01)	Sustained-Cell-Rate (clp01)	Tolerance	Use
3	VBR-RT	424 kbps	424 kbps	50	Physical interface/VBR-RT WK VCs
4	VBR-NRT	424 kbps	424 kbps	50	VBR-NRT tunnel WK VCs
5	ABR	424 kbps	—	None	—
6	UBR	424 kbps	—	None	UBR tunnel WK VCs

The **atm connection-traffic-table-row** command supports these service categories: CBR, VBR-RT, VBR-NRT, ABR, and UBR. To create or delete an ATM CTT row, perform the following tasks in global configuration mode:

Note Your CTT feature set depends on the type of feature card that is installed on the Catalyst 8510 MSR and LightStream 1010 ATM switch routers route processor.

	Command	Purpose
Step 1	Switch(config)# **atm connection-traffic-table-row** [**index** *row-index*] {**vbr-rt** | **vbr-nrt**} **pcr** *pcr-value* {**scr0** | **scr10**} *scr-value* [**mbs** *mbs-value*] [**cdvt** *cdvt_value*]	Configures an ATM CTT VBR row.
Step 2	Switch(config)# **atm connection-traffic-table-row** [**index** *row-index*] **cbr pcr** *pcr-value* [**cdvt** *cdvt-value*]	Configures an ATM CTT CBR row.
Step 3	Switch(config)# **atm connection-traffic-table-row** [**index** *row-index*] **abr pcr** *pcr-value* [**mcr** *mcr-value*] [**cdvt** *cdvt-value*]	Configures an ATM CTT ABR row.
Step 4	Switch(config)# **atm connection-traffic-table-row** [**index** *row-index*] **ubr pcr** *pcr-value* [**mcr** *mcr-value*] [**cdvt** *cdvt-value*]	Configures an ATM CTT UBR row.

If you do not specify an index row number, the system software determines if one is free and displays it in the allocated index field if the command is successful.

Example

The following example shows how to configure an ATM CTT row with an ABR peak cell rate of 30,000 kbps:

```
Switch(config)# atm connection-traffic-table-row abr pcr 30000
 Allocated index = 63999
```

Displaying the ATM Connection Traffic Table

To display the CTT configuration, use the following EXEC command:

Command	Purpose
show atm connection-traffic-table [**row** *row-index* | **from-row** *row-index*]	Displays the CTT configuration.

Example

The following example shows how to display the CTT configuration table:

```
Switch> show atm connection-traffic-table
Row      Service-category    pcr        scr/mcr       mbs         cdvt
1            ubr          7113539        none                     none
2            cbr              424                                 none
3            vbr-rt           424         424          50         none
4            vbr-nrt          424         424          50         none
5            abr              424           0                     none
6            ubr              424        none                     none
64000        cbr             1741                                 none
2147483645*  ubr                0        none                     none
2147483646*  ubr                1        none                     none
2147483647*  ubr          7113539        none                     none
```

Configuring the Sustainable Cell Rate Margin Factor

The sustained cell rate margin factor determines the aggressiveness of weighting sustainable cell rate (SCR) compared to peak cell rate (PCR). It uses the connection admission control algorithm in admitting VBR connections.

To configure the SCR for your ATM switch router, use the following global configuration command:

Command	Purpose
atm sustained-cell-rate-margin-factor *s-value*	Configures the sustained cell rate margin factor.

Note The **atm sustained-cell-rate-margin-factor** command affects subsequent connections but not connections that are already established.

Example

The following example shows how to configure the SCR margin factor as 85 percent of maximum:

```
Switch(config)# atm sustained-cell-rate-margin-factor 85
```

Displaying the SCR Margin Configuration

To display the SCR margin factor configuration, use the following EXEC command:

Command	Purpose
show atm resource	Displays the SCR margin factor configuration.

Example

The following example shows the SCR margin factor configuration:

```
Switch> show atm resource
Resource configuration:
➔ Sustained-cell-rate-margin-factor 85%
    Abr-mode:   EFCI
    Service Category to Threshold Group mapping:
     cbr 1 vbr-rt 2 vbr-nrt 3 abr 4 ubr 5
    Threshold Groups:
 Module   Group Max     Max Q  Min Q  Q thresholds  Cell  Name
  ID            cells   limit  limit  Mark Discard  count
                instal instal instal
 ------------------------------------------------------------
  1        1    131071  63     63      25 %  87 %      0     cbr-default-tg
           2    131071  127    127     25 %  87 %      0     vbrrt-default-tg
           3    131071  511    31      25 %  87 %      0     vbrnrt-default-tg
           4    131071  511    31      25 %  87 %      0     abr-default-tg
           5    131071  511    31      25 %  87 %      0     ubr-default-tg
           6    131071  1023   1023    25 %  87 %      0     well-known-vc-tg
===========================================================
  2        1    131071  63     63      25 %  87 %      0     cbr-default-tg
           2    131071  127    127     25 %  87 %      0     vbrrt-default-tg
           3    131071  511    31      25 %  87 %      0     vbrnrt-default-tg
           4    131071  511    31      25 %  50 %      0     abr-default-tg
           5    131071  511    31      25 %  87 %      0     ubr-default-tg
           6    131071  1023   1023    25 %  87 %      0     well-known-vc-tg
===========================================================
  7        1    131071  63     63      25 %  87 %      0     cbr-default-tg
           2    131071  127    127     25 %  87 %      0     vbrrt-default-tg
           3    131071  511    31      25 %  87 %      0     vbrnrt-default-tg
           4    131071  511    31      25 %  87 %      0     abr-default-tg
           5    131071  511    31      25 %  87 %      0     ubr-default-tg
           6    131071  1023   1023    25 %  87 %      0     well-known-vc-tg
===========================================================
  8        1    131071  63     63      25 %  87 %      0     cbr-default-tg
           2    131071  127    127     25 %  87 %      0     vbrrt-default-tg
           3    131071  511    31      25 %  87 %      0     vbrnrt-default-tg
           4    131071  511    31      25 %  87 %      0     abr-default-tg
           5    131071  511    31      25 %  87 %      0     ubr-default-tg
           6    131071  1023   1023    25 %  87 %      0     well-known-vc-tg
===========================================================
```

Overview of Threshold Groups

Threshold groups combine VCs/VPs to determine per-connection thresholds, based on the use of memory by the group.

Note Threshold groups are supported on the Catalyst 8540 MSR, and on the Catalyst 8510 MSR and LightStream 1010 ATM switch routers equipped with the FC-PFQ feature card.

The initial default configuration of per-VC queueing on the switch has all connections of a service category assigned to one threshold group. However, the assignment of service categories to threshold groups is configurable. A service category cannot be mapped to more than one threshold group. If you configure a service category to a threshold group more than once, the last configuration stays in effect. The default assigns each service category to a different threshold group. However, you can assign more than one service category to a threshold group.

Note The configuration of threshold groups is static, not dynamic.

For a description of how the threshold group feature works, refer to the *Guide to ATM Technology*.

Table 8-7 lists the configuration parameter defaults.

Table 8-7 Threshold Group Configuration Parameter Defaults

Group	Maximum Cells[1]	Maximum Queue Limit[2]	Minimum Queue Limit[3]	Mark Threshold[4]	Discard Threshold[5]	Use
1	65,535	63	63	25%	87%	CBR
2	65,535	127	127	25%	87%	VBR-RT
3	65,535	511	31	25%	87%	VBR-NRT
4	65,535	511	31	25%	87%	ABR
5	65,535	511	31	25%	87%	UBR
6	65,535	1023	1023	25%	87%	well-known VCs

1. Maximum number of cells in threshold group
2. Maximum (uncongested) per-VC queue limit in cells
3. Minimum (congested) per-VC queue limit in cells
4. Marking threshold percent full of per-VC queue
5. Discard threshold percent full of per-VC queue

Configuring the Threshold Group

To configure the threshold groups on a ATM switch router, perform the following tasks in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **atm threshold-group service** {**cbr** | **vbr-rt** | **vbr-nrt** | **abr** | **ubr**} *group*	Assigns a service category to a threshold group.
Step 2	Switch(config)# **atm threshold-group** [**module-id** *module*][1] *group* **max-cells** *number*	Configures the maximum number of cells queued for all connections that are members of the threshold group.
Step 3	Switch(config)# **atm threshold-group** [**module-id** *module*][1] *group* **discard-threshold** *percent*	Configures the threshold of per-connection queue-full at which the queue is considered full for CLP[2] discard and EPD[3].
Step 4	Switch(config)# **atm threshold-group** [**module-id** *module*][1] *group* **max-queue-limit** *number*	Configures the largest per-VC queue limit that is applied to connections in the threshold group.
Step 5	Switch(config)# **atm threshold-group** [**module-id** *module*][1] *group* **min-queue-limit** *number*	Configures the smallest per-VC queue-limit that is applied to connections in the threshold group.
Step 6	Switch(config)# **atm threshold-group** [**module-id** *module*][1] *group* **name** *name*	Configures the name associated with a threshold group.
Step 7	Switch(config)# **atm threshold-group** [**module-id** *module*][1] *group* **max-cells** *number*	Configures the maximum number of cells queued for specified threshold group for all module-ids.[4] Optionally, configure for the specified threshold group for the specified module-id.
Step 8	Switch(config)# **atm threshold-group** [**module-id** *module*][1] *group* **marking-threshold** *percent*	Configures the threshold of per-connection queue-full at which the queue is considered full for EFCI marking and ABR relative-rate marking.

1. The **module-id** identifier is only supported on the Catalyst 8540 MSR.
2. CLP = cell loss priority.
3. EPD = early packet discard.
4. Each module on the Catalyst 8540 MSR has its own cell memory and threshold groups. There are eight of these modules in a 20-gigabyte configuration. Each module has a 64-kbps cell memory, and the threshold groups can be configured per module. By default, all the threshold groups of all the modules are configured identically.

Example

The following example shows how to configure ATM threshold group 5 with a maximum number of cells before the cells are discarded:

```
Switch(config)# atm threshold-group 5 max-cells 50000
```

Displaying the Threshold Group Configuration

To display the threshold group configuration, use the following user EXEC command:

Command	Purpose
show atm resource	Displays the threshold group configuration.

Example

The following example displays the threshold group configuration:

```
Switch> show atm resource
Resource configuration:
Sustained-cell-rate-margin-factor 1%
    Abr-mode:   EFCI
    Service Category to Threshold Group mapping:
     cbr 1 vbr-rt 2 vbr-nrt 3 abr 4 ubr 5
    Threshold Groups:
 Module   Group Max    Max Q  Min Q  Q thresholds  Cell  Name
  ID            cells  limit  limit  Mark Discard  count
                instal instal instal
 ------------------------------------------------------------
  1        1    131071  63     63     25 %  87 %      0     cbr-default-tg
           2    131071  127    127    25 %  87 %      0     vbrrt-default-tg
           3    131071  511    31     25 %  87 %      0     vbrnrt-default-tg
           4    131071  511    31     25 %  87 %      0     abr-default-tg
           5    131071  511    31     25 %  87 %      0     ubr-default-tg
           6    131071  1023   1023   25 %  87 %      0     well-known-vc-tg
===========================================================
  2        1    131071  63     63     25 %  87 %      0     cbr-default-tg
           2    131071  127    127    25 %  87 %      0     vbrrt-default-tg
           3    131071  511    31     25 %  87 %      0     vbrnrt-default-tg
           4    131071  511    31     25 %  50 %      0     abr-default-tg
           5    131071  511    31     25 %  87 %      0     ubr-default-tg
           6    131071  1023   1023   25 %  87 %      0     well-known-vc-tg
===========================================================
<information deleted>
```

Configuring Physical Interfaces

Physical interface resource management configurations affect only specific interfaces on the switch. The following sections describe physical interface configuration resource management tasks:

Configuring the Interface Maximum Queue Size (Catalyst 8510 MSR and LightStream 1010)

Maximum queue size feature on the Catalyst 8510 MSR and LightStream 1010 ATM switch routers is used to determine the following:

- Maximum number of cells in the switch fabric queue
- Maximum cell transfer delay (CTD)
- Peak-to-peak cell delay variation (CDV) provided on an output switch interface

Note Interface maximum queue size configuration is only possible on switches with FC-PCQ installed on your route processor.

Because not all queue size values are supported by the switch fabric, the value installed is displayed, as well as the configuration value requested. The value installed is always greater than or equal to that requested.

To configure the maximum queue size, perform the following tasks, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **interface atm** *card/subcard/port* Switch(config-if)#	Selects the interface to be configured.
Step 2	Switch(config-if)# **atm output-queue** [**force**] {**cbr** | **vbr-rt** | **vbr-nrt** | **abr-ubr**} **max-size** *number*	Configures the ATM output queue maximum size.

Note The **atm output-queue** command affects all connections, including those already established.

This command is not applicable for subinterface level configuration. For other restrictions, refer to the *ATM Switch Router Command Reference* publication.

If the interface status is up, the **force** parameter is required before the request is completed. If the request is forced, output on the interface is briefly disabled, cells on the output queue are discarded, and the queue size is changed to the new limit. Any impact on existing connections by the implicit change in guaranteed maximum CTD and peak-to-peak CDV is not considered before making the change. Subsequent setup of switched virtual channel (SVC) connections will be affected.

Note The queue must be momentarily disabled to change the threshold.

Example

The following example shows how to configure the CBR ATM output queue maximum size to 30,000 cells:

```
Switch(config)# interface atm 3/0/0
Switch(config-if)# atm output-queue force cbr max-size 30000
```

Displaying the Output Queue Maximum Configuration (Catalyst 8510 MSR and LightStream 1010)

To display the output queue maximum size configuration, use the following user EXEC command:

Command	Purpose
show atm interface resource atm *card/subcard/port*	Displays the output queue maximum size configuration.

Example

The following example displays the interface output queue maximum size configuration with FC-PCQ installed:

```
Switch> show atm interface resource atm 3/0/0
Resource Management configuration:
    Output queues:
➔        Max sizes(explicit cfg): 30000  cbr, none vbr-rt, none vbr-nrt, none abr-ubr
        Max sizes(installed): 30208 cbr, 256 vbr-rt, 4096 vbr-nrt, 12032 abr-ubr
        Efci threshold: 25% cbr, 25% vbr-rt, 25% vbr-nrt, 25% abr, 25% ubr
        Discard threshold: 87% cbr, 87% vbr-rt, 87% vbr-nrt, 87% abr, 87% ubr
        Abr-relative-rate threshold: 25% abr
    Pacing: disabled   0 Kbps rate configured, 0 Kbps rate installed
    Service Categories supported: cbr,vbr-rt,vbr-nrt,abr,ubr
    Link Distance: 0 kilometers
    Controlled Link sharing:
        Max aggregate guaranteed services: none RX,  none TX
        Max bandwidth: none cbr RX, none cbr TX, none vbr RX, none vbr TX,
                       none abr RX, none abr TX, none ubr RX, none ubr TX
        Min bandwidth: none cbr RX, none cbr TX, none vbr RX, none vbr TX,
                       none abr RX, none abr TX, none ubr RX, none ubr TX
    Best effort connection limit: disabled  0 max connections
    Max traffic parameters by service (rate in Kbps, tolerance in cell-times):
        Peak-cell-rate RX: none cbr, none vbr, none abr, none ubr
        Peak-cell-rate TX: none cbr, none vbr, none abr, none ubr
        Sustained-cell-rate: none vbr RX, none vbr TX
        Minimum-cell-rate RX: none abr, none ubr
        Minimum-cell-rate TX: none abr, none ubr
        CDVT RX: none cbr, none vbr, none abr, none ubr
        CDVT TX: none cbr, none vbr, none abr, none ubr
        MBS: none vbr RX, none vbr TX
Resource Management state:
    Cell-counts: 0 cbr, 0 vbr-rt, 0 vbr-nrt, 0 abr-ubr
    Available bit rates (in Kbps):
        147743 cbr RX, 147743 cbr TX, 147743 vbr RX, 147743 vbr TX,
        0 abr RX, 0 abr TX, 0 ubr RX, 0 ubr TX
    Allocated bit rates:
        0 cbr RX, 0 cbr TX, 0 vbr RX, 0 vbr TX,
        0 abr RX, 0 abr TX, 0 ubr RX, 0 ubr TX
    Best effort connections: 1 pvcs,  0 svcs
```

Configuring the Interface Queue Thresholds per Service Category (Catalyst 8510 MSR and LightStream 1010)

The queue thresholds can be specified for the different levels of service and configured on each interface queue. The following queue thresholds can be configured:

- Output queue EFCI threshold
- Output queue cell loss priority (CLP) and packet discard (PD) threshold
- ABR relative rate threshold

Note Interface queue threshold per-service category configuration is only possible on switches with FC-PCQ installed on your route processor.

These queue thresholds can be changed at any time. The result changes the threshold for all connections of that service category using the interface for output and for any subsequent connections.

Note The CLP and PD discard threshold and ABR relative rate threshold have finer granularity than the explicit forward congestion indication (EFCI) threshold.

To configure the output threshold, perform the following tasks, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **interface atm** *card/subcard/port* Switch(config-if)#	Selects the interface to be configured.
Step 2	Switch(config-if)# **atm output-threshold** {**cbr** | **vbr-rt** | **vbr-nrt** | **abr** | **ubr**} **discard-threshold** *disc-thresh-num*	Configures the ATM output discard threshold.
Step 3	Switch(config-if)# **atm output-threshold** {**cbr** | **vbr-rt** | **vbr-nrt** | **abr** | **ubr**} **efci-threshold** *efci-thresh-number*	Configures the ATM output threshold.
Step 4	Switch(config-if)# **atm output-threshold abr relative-rate** *abr-thresh-number*	Configures the ATM output threshold ABR.

Note These commands affect all connections, including those already established.

These commands are not applicable for subinterface level configurations. For other restrictions, refer to the *ATM Switch Router Command Reference* publication.

Examples

The following example shows how to configure the interface output threshold CBR discard threshold to 87 percent of maximum size:

```
Switch(config)# interface atm 3/0/0
Switch(config-if)# atm output-threshold cbr discard 87
```

The following example shows how to configure the interface output discard threshold for CBR EFCI threshold to 50 percent of maximum size:

```
Switch(config)# interface atm 3/0/0
Switch(config-if)# atm output-threshold cbr efci 50
```

Displaying the Output Threshold Maximum Configuration (Catalyst 8510 MSR and LightStream 1010)

To display the output threshold maximum size configuration, use the following user EXEC command:

Command	Purpose
show atm interface resource atm *card*/*subcard*/*port*	Displays the output threshold maximum size configuration.

Example

The following example shows the interface output threshold maximum size configuration with FC-PCQ installed:

```
Switch> show atm interface resource atm 3/0/0
Resource Management configuration:
    Output queues:
        Max sizes(explicit cfg): 30000  cbr, none vbr-rt, none vbr-nrt, none abr-ubr
        Max sizes(installed): 30208 cbr, 256 vbr-rt, 4096 vbr-nrt, 12032 abr-ubr
➔       Efci threshold: 50% cbr, 25% vbr-rt, 25% vbr-nrt, 25% abr, 25% ubr
        Discard threshold: 87% cbr, 87% vbr-rt, 87% vbr-nrt, 87% abr, 87% ubr
        Abr-relative-rate threshold: 25% abr
    Pacing: disabled   0 Kbps rate configured, 0 Kbps rate installed
    Service Categories supported: cbr,vbr-rt,vbr-nrt,abr,ubr
    Link Distance: 0 kilometers
    Controlled Link sharing:
        Max aggregate guaranteed services: none RX,  none TX
        Max bandwidth: none cbr RX, none cbr TX, none vbr RX, none vbr TX,
                       none abr RX, none abr TX, none ubr RX, none ubr TX
        Min bandwidth: none cbr RX, none cbr TX, none vbr RX, none vbr TX,
                       none abr RX, none abr TX, none ubr RX, none ubr TX
    Best effort connection limit: disabled  0 max connections
    Max traffic parameters by service (rate in Kbps, tolerance in cell-times):
        Peak-cell-rate RX: none cbr, none vbr, none abr, none ubr
        Peak-cell-rate TX: none cbr, none vbr, none abr, none ubr
        Sustained-cell-rate: none vbr RX, none vbr TX
        Minimum-cell-rate RX: none abr, none ubr
        Minimum-cell-rate TX: none abr, none ubr
        CDVT RX: none cbr, none vbr, none abr, none ubr
        CDVT TX: none cbr, none vbr, none abr, none ubr
        MBS: none vbr RX, none vbr TX
<information deleted>
```

Configuring Interface Output Pacing

Output pacing is used to artificially reduce the output speed of an interface in kbps. Output pacing can be changed at any time, enabled, or disabled. When an output pacing change request is made, resource management determines if the change will not provide the guaranteed bandwidth at the outbound port for the existing virtual channels or virtual paths (VCs or VPs). Guaranteed bandwidth is reserved for constant bit rate (CBR) and variable bit rate (VBR) connections.

Note Pacing is only allowed for carrier module ports on the Catalyst 8540 MSR.

To enable or change an interface output pacing rate, perform the following tasks, beginning in global configuration mode:

Command	Purpose
interface atm *card*/*subcard*/*port*	Selects the interface to be configured.
atm pacing *kbps* [**force**]	Configures the interface output pacing.

The **force** argument indicates that the change should be made even if it results in an output cell rate that does not provide sufficient bandwidth for guaranteed service on the interface transmit flow. The **force** argument has no effect on Catalyst 8510 MSR and LightStream 1010 ATM switch routers with FC-PFQ installed on the route processor.

Note The **atm pacing** command affects all connections, including those already established.

This command does not apply to the CPU interfaces (atm0 and ethernet0) or subinterfaces. For other restrictions, refer to the *ATM Switch Router Command Reference* publication.

Note Since the 12.0(1a)W5(5b) release of the system software, addressing the interface on the route processor (CPU) has changed. The ATM interface is now called atm0, and the Ethernet interface is now called ethernet0. Old formats (atm 2/0/0 and ethernet 2/0/0) are still supported.

Example

The following example shows how to configure the interface output pacing to 10,000 kbps:

```
Switch(config)# interface atm 3/0/0
Switch(config-if)# atm pacing 10000
```

Displaying the Output Pacing Configuration

To display the output pacing configuration, use the following EXEC command:

Command	Purpose
show atm interface resource atm *card*/*subcard*/*port*	Displays the output pacing configuration.

Example

The following example shows the interface output pacing configuration:

```
Switch> show atm interface resource atm 0/0/0
Resource Management configuration:
    Service Classes:
        Service Category map: c1 cbr, c2 vbr-rt, c3 vbr-nrt, c4 abr, c5 ubr
        Scheduling: RS c1 WRR c2, WRR c3, WRR c4, WRR c5
        WRR Weight: 8 c2, 1 c3, 1 c4, 1 c5
➔   Pacing: disabled   0 Kbps rate configured, 0 Kbps rate installed
    Link Distance: 0 kilometers
    Controlled Link sharing:
        Max aggregate guaranteed services: none RX,  none TX
        Max bandwidth: none cbr RX, none cbr TX, none vbr RX, none vbr TX
        Min bandwidth: none cbr RX, none cbr TX, none vbr RX, none vbr TX
    Best effort connection limit: disabled  0 max connections
    Max traffic parameters by service (rate in Kbps, tolerance in cell-times):
        Peak-cell-rate RX: none cbr, none vbr, none abr, none ubr
        Peak-cell-rate TX: none cbr, none vbr, none abr, none ubr
        Sustained-cell-rate: none vbr RX, none vbr TX,
        Tolerance RX: none cbr, none vbr, none abr, none ubr
        Tolerance TX: none cbr, none vbr, none abr, none ubr
<information deleted>
```

Configuring Controlled Link Sharing

Resource management allows fine-tuning of the connection admission control functions on a per-interface and direction (receive and transmit) basis. The reservations are specified with the following three parameters:

- Maximum aggregate guaranteed cell rate on an interface, which limits the guaranteed bandwidth that can be allocated on an interface
- Maximum guaranteed cell rates on an interface per-service category
- Minimum guaranteed cell rates on an interface per-service category

Table 8-8 shows the minimum and maximum parameter relationships.

Table 8-8 Connection Admission Control Parameter to Bandwidth Relationships

Service Category	Value	Service Category	Bandwidth
Minimum CBR	+	Minimum VBR	<= 95 percent
Minimum CBR	<=	Maximum CBR	<= 95 percent
Minimum VBR	<=	Maximum VBR	<= 95 percent
Minimum CBR	<=	Maximum Aggregate	<= 95 percent
Minimum VBR	<=	Maximum Aggregate	<= 95 percent
Maximum CBR	<=	Maximum Aggregate	<= 95 percent
Maximum VBR	<=	Maximum Aggregate	<= 95 percent

To configure controlled link sharing, perform the following tasks, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **interface atm** *card*/*subcard*/*port*[**.***vpt#*] Switch(config-if)#	Selects the interface to be configured.
Step 2	Switch(config-if)# **atm cac link-sharing max-guaranteed-service-bandwidth** {**receive** | **transmit**} *percent*	Configures controlled link sharing for the maximum guaranteed service bandwidth.
Step 3	Switch(config-if)# **atm cac link-sharing max-bandwidth** {**abr** | **cbr** | **ubr** | **vbr**} {**receive** | **transmit**} *percent*	Configures controlled link sharing for the maximum guaranteed service bandwidth by service category.
Step 4	Switch(config-if)# **atm cac link-sharing min-bandwidth** {**cbr** | **vbr** | **abr** | **ubr**} {**receive** | **transmit**} *percent*	Configures controlled link sharing for the minimum guaranteed service bandwidth by service category.

Note These commands affect subsequent connections but not connections that are already established.

For restrictions to these commands, refer to the *ATM Switch Router Command Reference* publication.

Example

The following example shows how to configure the controlled link sharing, maximum guaranteed service bandwidth, and receive configuration to 87 percent:

```
Switch(config)# interface atm 3/0/0
Switch(config-if)# atm cac link-sharing max-guaranteed-service-bandwidth receive 87
```

Displaying the Controlled Link Sharing Configuration

To display the controlled link sharing configuration, perform the following task in user EXEC mode:

Command	Purpose
show atm interface resource atm *card*/*subcard*/*port*	Displays the controlled link sharing configuration.

Example

The following example displays the controlled link sharing configuration:

```
Switch> show atm interface resource atm 0/0/0
Resource Management configuration:
    Service Classes:
        Service Category map: c1 cbr, c2 vbr-rt, c3 vbr-nrt, c4 abr, c5 ubr
        Scheduling: RS c1 WRR c2, WRR c3, WRR c4, WRR c5
        WRR Weight: 8 c2, 1 c3, 1 c4, 1 c5
    Pacing: disabled   0 Kbps rate configured, 0 Kbps rate installed
    Link Distance: 0 kilometers
→   Controlled Link sharing:
→       Max aggregate guaranteed services: none RX,  none TX
→       Max bandwidth: none cbr RX, none cbr TX, none vbr RX, none vbr TX
→       Min bandwidth: none cbr RX, none cbr TX, none vbr RX, none vbr TX
    Best effort connection limit: disabled  0 max connections
    Max traffic parameters by service (rate in Kbps, tolerance in cell-times):
        Peak-cell-rate RX: none cbr, none vbr, none abr, none ubr
        Peak-cell-rate TX: none cbr, none vbr, none abr, none ubr
        Sustained-cell-rate: none vbr RX, none vbr TX,
        Tolerance RX: none cbr, none vbr, none abr, none ubr
        Tolerance TX: none cbr, none vbr, none abr, none ubr
<information deleted>
```

Configuring the Scheduler and Service Class

A service class denotes one of the scheduling classes referred to as output virtual circuit (OVC) QoS classes. Up to eight service classes can be allocated to each physical interface (PIF) port. In scheduling the next cell to be transmitted from a port, the rate scheduler (RS) has first call on supplying an eligible cell. If RS does not have one, then weighted round-robin (WRR) scheduler chooses a service class with an OVC ready to transmit, and finally a VC within the service class is selected.

Note Scheduler and service class configuration is only possible on Catalyst 8510 MSR and LightStream 1010 ATM switch routers with FC-PFQ installed on your route processor.

ATM service categories are mapped statically to service classes, as shown in Table 8-9, where service class 2 has the highest scheduling priority.

Table 8-9 ATM Service Category to Service Class

Service Category	Service Class
VBR-RT	2
VBR-NRT	3
ABR	4
UBR	5

Each service class is assigned a weight. These weights are configurable, in the range of 1 to 15. The default weighting is {15,2,2,2} for classes {2,3,4,5}, respectively. The weighting is not modified dynamically.

Within service classes, individual PVCs are also weighted, again in the range of 1 to 15. A standard weight (2) is assigned to all PVCs in a service class. Optionally, PVCs can be configured with a specific weight per half-leg (applying to the transmit OVC weight). SVCs take the value 2.

Note For a detailed description of rate and WRR scheduling, refer to the *Guide to ATM Technology*.

To configure the interface service class and WRR value, perform the following tasks, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **interface atm** *card*/*subcard*/*port*[**.***vpt#*] Switch(config-if)#	Selects the interface to be configured.
Step 2	Switch(config-if)# **atm service-class** {**2** | **3** | **4** | **5**} **wrr-weight** *weight*	Configures the weight given to each service class.

Example

The following example shows how to configure service class 3 on interface ATM 0/1/0 with a WRR weight of 5:

```
Switch(config)# interface atm 0/1/0
Switch(config-if)# atm service-class 3 wrr-weight 5
```

Displaying the Interface Service Class Information

To display the configuration of an interface in a service class, use the following user EXEC command:

Command	Purpose
show atm interface resource {**atm** | **atm-p**} *card*/*subcard*/*port*	Displays the configured membership of the interface in a service class.

Example

The following example shows the configuration of the interface in a service class:

```
Switch> show atm interface resource atm 0/0/0
Resource Management configuration:
➔   Service Classes:
        Service Category map: c1 cbr, c2 vbr-rt, c3 vbr-nrt, c4 abr, c5 ubr
        Scheduling: RS c1 WRR c2, WRR c3, WRR c4, WRR c5
        WRR Weight: 8 c2, 1 c3, 1 c4, 1 c5
    Pacing: disabled   0 Kbps rate configured, 0 Kbps rate installed
    Link Distance: 0 kilometers
    Controlled Link sharing:
        Max aggregate guaranteed services: none RX,  none TX
        Max bandwidth: none cbr RX, none cbr TX, none vbr RX, none vbr TX
        Min bandwidth: none cbr RX, none cbr TX, none vbr RX, none vbr TX
    Best effort connection limit: disabled  0 max connections
    Max traffic parameters by service (rate in Kbps, tolerance in cell-times):
        Peak-cell-rate RX: none cbr, none vbr, none abr, none ubr
        Peak-cell-rate TX: none cbr, none vbr, none abr, none ubr
        Sustained-cell-rate: none vbr RX, none vbr TX,
        Tolerance RX: none cbr, none vbr, none abr, none ubr
        Tolerance TX: none cbr, none vbr, none abr, none ubr
<information deleted>
```

Configuring Physical and Logical Interface Parameters

The following sections describe interface configuration resource management tasks for both physical and logical interface types:

- Configuring the Interface Link Distance, page 8-27
- Configuring the Limits of Best-Effort Connections, page 8-29
- Configuring the Interface Maximum of Individual Traffic Parameters, page 8-30
- Configuring the ATM Default CDVT and MBS, page 8-33
- Configuring Interface Service Category Support, page 8-35

Configuring the Interface Link Distance

Specifying the physical link distance for the next ATM hop in the outbound direction allows you to increase the propagation delay. Propagation delay is used in determining the connection admission control (CAC) maximum cell transfer delay (CTD) provided on the output by a switch interface, which can affect the switched virtual channel (SVC) connection requests accepted.

Note For a detailed description of the CAC algorithm pseudo-code on the ATM switch router, refer to the *Guide to ATM Technology*.

To configure the interface link distance, perform the following tasks, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **interface atm** *card*/*subcard*/*port*[**.***vpt#*] Switch(config-if)#	Selects the interface to be configured.
Step 2	Switch(config-if)# **atm link-distance** *kilometers*	Configures the interface link distance for the interface.

Note The **atm link-distance** command affects subsequent connections but not connections that are already established.

Example

The following example shows how to configure the outbound link distance to 150 kilometers:

```
Switch(config-if)# atm link-distance 150
```

Displaying the Interface Link Distance Configuration

To display the interface link distance configuration, use the following EXEC command:

Command	Purpose
show atm interface resource atm *card*/*subcard*/*port*[**.***vpt#*]	Displays the interface link distance configuration.

Example

The following example shows the configuration of the interface link distance with switch processor feature card installed:

```
Switch> show atm interface resource atm 0/0/0
Resource Management configuration:
    Service Classes:
        Service Category map: c1 cbr, c2 vbr-rt, c3 vbr-nrt, c4 abr, c5 ubr
        Scheduling: RS c1 WRR c2, WRR c3, WRR c4, WRR c5
        WRR Weight: 8 c2, 1 c3, 1 c4, 1 c5
    Pacing: disabled   0 Kbps rate configured, 0 Kbps rate installed
    Service Categories supported: cbr,vbr-rt,vbr-nrt,abr,ubr
→   Link Distance: 150 kilometers
    Controlled Link sharing:
        Max aggregate guaranteed services: none RX,  none TX
        Max bandwidth: none cbr RX, none cbr TX, none vbr RX, none vbr TX,
                       none abr RX, none abr TX, none ubr RX, none ubr TX
        Min bandwidth: none cbr RX, none cbr TX, none vbr RX, none vbr TX,
                       none abr RX, none abr TX, none ubr RX, none ubr TX
    Best effort connection limit: disabled  0 max connections
    Max traffic parameters by service (rate in Kbps, tolerance in cell-times):
        Peak-cell-rate RX: none cbr, none vbr, none abr, none ubr
        Peak-cell-rate TX: none cbr, none vbr, none abr, none ubr
```

```
        Sustained-cell-rate: none vbr RX, none vbr TX
        Minimum-cell-rate RX: none abr, none ubr
        Minimum-cell-rate TX: none abr, none ubr
        CDVT RX: none cbr, none vbr, none abr, none ubr
        CDVT TX: none cbr, none vbr, none abr, none ubr
        MBS: none vbr RX, none vbr TX
<information deleted>
```

Configuring the Limits of Best-Effort Connections

Each interface can be configured to allow a specific number of best-effort available bit rate (ABR) and unspecified bit rate (UBR) connections.

To configure the number of best-effort connections, perform the following tasks, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **interface atm** *card*/*subcard*/*port*[**.***vpt#*] Switch(config-if)#	Selects the interface to be configured.
Step 2	Switch(config-if)# **atm cac best-effort-limit** *conn-value*	Configures the connection best-effort limit.

Note These commands affect subsequent connections but not connections that are already established.

Example

The following example shows how to configure the connection best-effort limit configuration to 2000:

```
Switch(config)# interface atm 3/0/0
Switch(config-if)# atm cac best-effort-limit 2000
```

Displaying the Interface Best-Effort Limit Configuration

To display the interface best-effort configuration, use the following EXEC command:

Command	Purpose
show atm interface resource atm *card*/*subcard*/*port*[**.***vpt#*]	Displays the subinterface best-effort configuration.

Example

The following example shows the interface best-effort configuration with the switch processor feature card installed:

```
Switch> show atm interface resource atm 3/0/0
Resource Management configuration:
    Service Classes:
        Service Category map: c1 cbr, c2 vbr-rt, c3 vbr-nrt, c4 abr, c5 ubr
        Scheduling: RS c1 WRR c2, WRR c3, WRR c4, WRR c5
        WRR Weight: 8 c2, 1 c3, 1 c4, 1 c5
    Pacing: disabled   0 Kbps rate configured, 0 Kbps rate installed
    Service Categories supported: cbr,vbr-rt,vbr-nrt,abr,ubr
    Link Distance: 0 kilometers
    Controlled Link sharing:
        Max aggregate guaranteed services: none RX,  none TX
        Max bandwidth: none cbr RX, none cbr TX, none vbr RX, none vbr TX,
                       none abr RX, none abr TX, none ubr RX, none ubr TX
        Min bandwidth: none cbr RX, none cbr TX, none vbr RX, none vbr TX,
                       none abr RX, none abr TX, none ubr RX, none ubr TX
➔   Best effort connection limit: enabled  2000 max connections
    Max traffic parameters by service (rate in Kbps, tolerance in cell-times):
        Peak-cell-rate RX: none cbr, none vbr, none abr, none ubr
        Peak-cell-rate TX: none cbr, none vbr, none abr, none ubr
        Sustained-cell-rate: none vbr RX, none vbr TX
        Minimum-cell-rate RX: none abr, none ubr
        Minimum-cell-rate TX: none abr, none ubr
        CDVT RX: none cbr, none vbr, none abr, none ubr
        CDVT TX: none cbr, none vbr, none abr, none ubr
        MBS: none vbr RX, none vbr TX
<information deleted>
```

Configuring the Interface Maximum of Individual Traffic Parameters

When a VCC is set up, you can specify per-flow (receive and transmit traffic) parameters. Traffic parameter limits may be configured independently by service category and traffic direction for the following:

- Maximum peak cell rate (PCR)
- Maximum sustained cell rate (SCR)
- Maximum cell delay variation tolerance (CDVT)
- Maximum burst size (MBS)
- Maximum minimum cell rate (MCR)

To configure the traffic parameters, perform the following tasks, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **interface atm** *card*/*subcard*/*port*[**.***vpt#*] Switch(config-if)#	Selects the interface to be configured.
Step 2	Switch(config-if)# **atm cac max-peak-cell-rate** {**cbr** | **vbr** | **abr** | **ubr**} {**receive** | **transmit**} *rate*	Configures the connection maximum PCR.

	Command	Purpose
Step 3	Switch(config-if)# **atm cac max-sustained-cell-rate** {**receive** | **transmit**} *rate*	Configures the connection SCR.
Step 4	Switch(config-if)# **atm cac max-cdvt** {**abr** | **cbr** | **ubr** | **vbr**} {**receive** | **transmit**} *cell-count*	Configures the connection maximum CDVT.
Step 5	Switch(config-if)# **atm cac max-mbs** {**receive** | **transmit**} *cell-count*	Configures the connection maximum MBS.
Step 6	Switch(config-if)# **atm cac max-min-cell-rate** {**abr** | **ubr**} {**receive** | **transmit**} *rate*	Configures the connection maximum MCR per service category flow.

Note These commands affect subsequent connections but not connections that are already established.

Examples

The following example shows how to configure the maximum PCR for constant bit rate (CBR) connections on interface 3/0/0, specified in receive mode, to 100,000 kbps:

```
Switch(config)# interface atm 3/0/0
Switch(config-if)# atm cac max-peak-cell-rate cbr receive 100000
```

The following example shows how to configure the maximum SCR for connections on interface 3/0/0, specified in receive mode, to 60,000 kbps:

```
Switch(config)# interface atm 3/0/0
Switch(config-if)# atm cac max-sustained-cell-rate receive 60000
```

The following example shows how to configure the maximum tolerance for CBR connections on interface 3/0/0, specified in receive mode, 75,000 kbps:

```
Switch(config)# interface atm 3/0/0
Switch(config-if)# atm cac max-cdvt cbr receive 75000
```

Displaying the Interface Maximum Individual Traffic Parameter Configuration

To display the interface maximum individual traffic parameter configuration, use the following EXEC command:

Command	Purpose
show atm interface resource atm [*card*/*subcard*/*port*[**.***vpt#*]]	Displays the controlled link sharing configuration.

Example

The following example shows the interface maximum individual traffic configuration with switch processor feature card installed:

```
Switch> show atm interface resource atm 3/0/0
Resource Management configuration:
    Service Classes:
        Service Category map: c1 cbr, c2 vbr-rt, c3 vbr-nrt, c4 abr, c5 ubr
        Scheduling: RS c1 WRR c2, WRR c3, WRR c4, WRR c5
        WRR Weight: 8 c2, 1 c3, 1 c4, 1 c5
    Pacing: disabled   0 Kbps rate configured, 0 Kbps rate installed
    Service Categories supported: cbr,vbr-rt,vbr-nrt,abr,ubr
    Link Distance: 0 kilometers
    Controlled Link sharing:
        Max aggregate guaranteed services: none RX,  none TX
        Max bandwidth: none cbr RX, none cbr TX, none vbr RX, none vbr TX,
                       none abr RX, none abr TX, none ubr RX, none ubr TX
        Min bandwidth: none cbr RX, none cbr TX, none vbr RX, none vbr TX,
                       none abr RX, none abr TX, none ubr RX, none ubr TX
    Best effort connection limit: enabled  2000 max connections
➔   Max traffic parameters by service (rate in Kbps, tolerance in cell-times):
        Peak-cell-rate RX: none cbr, none vbr, none abr, none ubr
        Peak-cell-rate TX: none cbr, none vbr, none abr, none ubr
        Sustained-cell-rate: none vbr RX, none vbr TX
        Minimum-cell-rate RX: none abr, none ubr
        Minimum-cell-rate TX: none abr, none ubr
        CDVT RX: none cbr, none vbr, none abr, none ubr
        CDVT TX: none cbr, none vbr, none abr, none ubr
        MBS: none vbr RX, none vbr TX
<information deleted>
```

Configuring the ATM Default CDVT and MBS

You can change the default cell delay variation tolerance (CDVT) and maximum burst size (MBS) to request for UPC of cells received on the interface for connections that do not individually request a CDVT or MBS value.

You can specify CDVT or MBS for PVCs through a connection traffic table row. If no CDVT or MBS is specified in the row, then a per-interface, per-service category default is applied for purposes of usage parameter control (UPC) on the connection.

Note For signalled connections, CDVT or MBS cannot be signalled and the defaults specified on the interface apply.

To configure the default CDVT and MBS parameters, perform the following task, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **interface atm** *card*/*subcard*/*port* Switch(config-if)#	Specifies an ATM interface and enter interface configuration mode.
Step 2	Switch(config-if)# **atm cdvt-default** {**cbr** | **vbr-rt** | **vbr-nrt** | **abr** | **ubr**} *number*	Configures the ATM CDVT default.
Step 3	Switch(config-if)# **atm mbs-default** {**vbr-rt** | **vbr-nrt**} *number*	Configures the ATM MBS default.

Example

The following example shows how to change the default tolerance for received cells on VBR-RT connections:

```
Switch(config)# interface atm 3/0/0
Switch(config-if)# atm cdvt-default vbr-rt 4000
```

Displaying the ATM CDVT and MBS Configuration

To display the ATM CDVT and MBS configuration, use the following EXEC commands:

Command	Purpose
show atm vc	Displays the ATM VC CDVT configuration.
show atm vp	Displays the ATM VP CDVT configuration.

Examples

The following example shows the ATM CDVT and MBS configuration of an ATM VC:

```
Switch> show atm vc interface atm 0/0/3 0 100

Interface: ATM0/0/3, Type: oc3suni
VPI = 0  VCI = 100
Status: UP
Time-since-last-status-change: 00:00:08
Connection-type: PVC
Cast-type: point-to-point
Packet-discard-option: disabled
Usage-Parameter-Control (UPC): pass
Wrr weight: 32
Number of OAM-configured connections: 0
OAM-configuration: disabled
OAM-states:  Not-applicable
Cross-connect-interface: ATM0/0/2, Type: oc3suni
Cross-connect-VPI = 0
Cross-connect-VCI = 100
Cross-connect-UPC: pass
Cross-connect OAM-configuration: disabled
Cross-connect OAM-state:  Not-applicable
Threshold Group: 2, Cells queued: 0
Rx cells: 0, Tx cells: 0
Tx Clp0:0,  Tx Clp1: 0
Rx Clp0:0,  Rx Clp1: 0
Rx Upc Violations:0, Rx cell drops:0
Rx Clp0 q full drops:0, Rx Clp1 qthresh drops:0
Rx connection-traffic-table-index: 9999
Rx service-category: VBR-RT (Realtime Variable Bit Rate)
Rx pcr-clp01: 40000
Rx scr-clp0 : 30000
Rx mcr-clp01: none
➔ Rx      cdvt: 1024 (from default for interface)
➔ Rx       mbs: 1024 (from default for interface)
Tx connection-traffic-table-index: 9999
Tx service-category: VBR-RT (Realtime Variable Bit Rate)
Tx pcr-clp01: 40000
Tx scr-clp0 : 30000
Tx mcr-clp01: none
➔ Tx      cdvt: none
➔ Tx       mbs: none
```

The following example shows the ATM CDVT and MBS configuration of an ATM VP:

```
Switch> show atm vp interface atm0/0/3 4

Interface: ATM0/0/3, Type: oc3suni
VPI = 4
Status: UP
Time-since-last-status-change: 00:00:10
Connection-type: PVP
Cast-type: point-to-point
Usage-Parameter-Control (UPC): pass
Wrr weight: 32
Number of OAM-configured connections: 0
OAM-configuration: disabled
OAM-states:  Not-applicable
Cross-connect-interface: ATM0/0/2, Type: oc3suni
Cross-connect-VPI = 4
Cross-connect-UPC: pass
Cross-connect OAM-configuration: disabled
Cross-connect OAM-state:  Not-applicable
Threshold Group: 5, Cells queued: 0
Rx cells: 0, Tx cells: 0
Tx Clp0:0,  Tx Clp1: 0
```

```
   Rx Clp0:0,  Rx Clp1: 0
   Rx Upc Violations:0, Rx cell drops:0
   Rx Clp0 q full drops:0, Rx Clp1 qthresh drops:0
   Rx connection-traffic-table-index: 1
   Rx service-category: UBR (Unspecified Bit Rate)
   Rx pcr-clp01: 7113539
   Rx scr-clp01: none
   Rx mcr-clp01: none
→  Rx      cdvt: 1024 (from default for interface)
→  Rx       mbs: none
   Tx connection-traffic-table-index: 1
   Tx service-category: UBR (Unspecified Bit Rate)
   Tx pcr-clp01: 7113539
   Tx scr-clp01: none
   Tx mcr-clp01: none
→  Tx      cdvt: none
→  Tx       mbs: none
```

Configuring Interface Service Category Support

You can configure which service categories connection admission control (CAC) allows on an interface. You can configure interface service category support only on physical interfaces and shaped and hierarchical logical virtual path (VP) tunnel interfaces.

Note For information on how to configure your physical and logical VP tunnel interfaces, see the "Configuring VP Tunnels" section on page 6-31.

The underlying service category for shaped and hierarchical VP tunnels is CBR. For VP shaped tunnels, interface service category support can be used to configure a service category other than CBR for VCs within the tunnel. For physical interfaces and hierarchical VP tunnels, all service category VCs (by default) can migrate across the interface. However, you can use the interface service category support feature to explicitly allow or prevent VCs of specified service categories to migrate across the interface.

Table 8-10 shows the service category of the shaped VP (always CBR), the service categories you can configure for transported VCs, and a suggested transit VP service category for the tunnel.

Table 8-10 Service Category Support for Physical and Logical Interfaces

Shaped VP Tunnel Service Category	VC Service Category	Suggested Transit VP Service Category
CBR	CBR	CBR
CBR	VBR	CBR or VBR
CBR	ABR[1]	CBR or VBR
CBR	UBR	Any service category

1. We recommend ABR only if the transit VP is set up so that congestion occurs at the shaped tunnel, not in the transit VP.

The following restrictions apply to interface service category support:

- This configuration is allowed on physical interfaces and shaped and hierarchical VP tunnel logical interfaces.
- On shaped VP tunnel logical interfaces, only one service category is permitted at a time. To replace CBR with another service category on these interfaces, you must first deny the CBR service category, then permit the chosen service category. To deny a service category, you must delete all user VCs of that service category on the interface.
- For ABR and UBR, only zero MCR is supported on VCs on a shaped VP tunnel.

To configure a service category on an interface, perform the following tasks, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **interface atm** *card*/*subcard*/*port*[**.***vpt#*] Switch(config-if)#	Selects the interface to be configured.
Step 2	**atm cac service-category** {**cbr** | **vbr-rt** | **vbr-nrt** | **abr** | **ubr**} {**permit** | **deny**}	Configures the service category on the interface.

Example

The following example shows how to configure the ABR service category on ATM interface 3/0/0:

```
Switch(config)# interface atm 3/0/0
Switch(config-if)# atm cac service-category cbr deny
Switch(config-if)# atm cac service-category abr permit
```

Displaying the Service Category on an Interface

To display the service category configured on an interface, use the following user EXEC command:

Command	Purpose
show atm interface resource atm *card/subcard/port*[**.***vpt#*]	Displays the controlled link sharing configuration.

Example

The following example shows the service category configuration:

```
Switch> show atm interface resource atm 3/0/0
Resource Management configuration:
    Service Classes:
        Service Category map: c1 cbr, c2 vbr-rt, c3 vbr-nrt, c4 abr, c5 ubr
        Scheduling: RS c1 WRR c2, WRR c3, WRR c4, WRR c5
        WRR Weight: 8 c2, 1 c3, 1 c4, 1 c5
    Pacing: disabled   0 Kbps rate configured, 0 Kbps rate installed
➔     Service Categories supported: cbr,vbr-rt,vbr-nrt,ubr
    Link Distance: 0 kilometers
    Controlled Link sharing:
        Max aggregate guaranteed services: none RX,  none TX
        Max bandwidth: none cbr RX, none cbr TX, none vbr RX, none vbr TX,
                       none abr RX, none abr TX, none ubr RX, none ubr TX
        Min bandwidth: none cbr RX, none cbr TX, none vbr RX, none vbr TX,
                       none abr RX, none abr TX, none ubr RX, none ubr TX
    Best effort connection limit: disabled  0 max connections
    Max traffic parameters by service (rate in Kbps, tolerance in cell-times):
        Peak-cell-rate RX: none cbr, none vbr, none abr, none ubr
        Peak-cell-rate TX: none cbr, none vbr, none abr, none ubr
        Sustained-cell-rate: none vbr RX, none vbr TX
        Minimum-cell-rate RX: none abr, none ubr
        Minimum-cell-rate TX: none abr, none ubr
        CDVT RX: none cbr, none vbr, none abr, none ubr
        CDVT TX: none cbr, none vbr, none abr, none ubr
<information deleted>
```

Configuring Interface Overbooking

The interface overbooking feature allows the available equivalent bandwidth of an interface to exceed the maximum cell rate (MaxCR) or physical line rate on ATM and inverse multiplexing over ATM (IMA) interfaces. The available equivalent bandwidth is by default limited by the MaxCR. Increasing the available equivalent bandwidth beyond the MaxCR allows the configuration of more connections on an interface than its physical bandwidth would allow. Overbooking allows more flexibility when configuring an interface when the traffic over the interface will be less than the MaxCR.

The following restrictions apply to interface overbooking:

- Regular VP tunnels do not support interface overbooking.
- You cannot add new hierarchical VP tunnels on a physical interface if the interface's bandwidth guarantees exceed the MaxCR regardless of any overbooking configured on that interface.
- On IMA interfaces, the available equivalent bandwidth for PVCs differs from the available equivalent bandwidth for SVCs. The available equivalent bandwidth for PVCs is based on the number of interfaces configured as part of the IMA group. The available equivalent bandwidth for

SVCs on an IMA interface is based on the number of interfaces that are active in the IMA group. Overbooking increases both the available equivalent bandwidth values by the same configured percentage.

- The MaxCR for transmit and receive flows might differ on output-paced physical interfaces. Configuring overbooking on such interfaces results in different maximum guaranteed services bandwidth values and available cell rates for service categories for transmit and receive flows. Maximum guaranteed services bandwidth is the maximum equivalent bandwidth allocated for guaranteed services on the interface.
- When an interface is overbooked with traffic, cell flow through the well-known VCs might be reduced.
- Although overbooking increases the available cell rates for various service categories on an interface, various traffic parameters of a connection are still limited by the MaxCR.
- If the overbooking configuration results in a maximum guaranteed services bandwidth that is below the currently allocated bandwidth guarantees on an interface, the configuration is rejected.

Caution Overbooking can cause interface traffic to exceed the guaranteed bandwidth that the switch can provide.

Note Interface overbooking configuration is not supported on switches with feature card per-flow queuing (FC-PCQ) installed.

To configure overbooking, perform the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	**interface atm** *card*/*subcard*/*slot* Switch(config-if)# or **interface atm** *card*/*subcard*/**ima***group* Switch(config-if)#	Specifies the physical interface to configure. Specifies the IMA group interface to configure.
Step 2	Switch(config-if)# **shutdown**	Shuts down the interface prior to configuring overbooking.
Step 3	Switch(config-if)# **atm cac overbooking** *percent*	Configures overbooking on an interface as a percentage of the maximum equivalent bandwidth available on the interface from 100 to 1000. A value of 100 disables overbooking on the interface.
Step 4	Switch(config-if)# **no shutdown**	Reenables the interface

Example

The following example shows how to set the interface overbooking percentage to 300:

```
Switch(config)# interface atm 4/1/0
Switch(config-if)# shutdown
Switch(config-if)# atm cac overbooking 300
Switch(config-if)# no shutdown
```

Displaying the Interface Overbooking Configuration

To display the interface overbooking configuration, use the following user EXEC command:

Command	Purpose
show atm interface resource atm *card*/*subcard*/*port*[**.***vpt#*]	Displays the interface overbooking configuration.

Example

The following example shows the interface overbooking configuration with FC-PFQ installed:

```
Switch> show atm interface resource atm 4/1/0
Resource Management configuration:
    Service Classes:
        Service Category map: c2 cbr, c2 vbr-rt, c3 vbr-nrt, c4 abr, c5 ubr
        Scheduling: RS c1 WRR c2, WRR c3, WRR c4, WRR c5
        WRR Weight: 15 c2, 2 c3, 2 c4, 2 c5
    CAC Configuration to account for Framing Overhead : Disabled
    Pacing: disabled   0 Kbps rate configured, 0 Kbps rate installed
➔   overbooking :  300
    Service Categories supported: cbr,vbr-rt,vbr-nrt,abr,ubr
    Link Distance: 0 kilometers
    Controlled Link sharing:
        Max aggregate guaranteed services: none RX,  none TX
        Max bandwidth: none cbr RX, none cbr TX, none vbr RX, none vbr TX,
                       none abr RX, none abr TX, none ubr RX, none ubr TX
        Min bandwidth: none cbr RX, none cbr TX, none vbr RX, none vbr TX,
                       none abr RX, none abr TX, none ubr RX, none ubr TX
    Best effort connection limit: disabled  0 max connections
    Max traffic parameters by service (rate in Kbps, tolerance in cell-times):
        Peak-cell-rate RX: none cbr, none vbr, none abr, none ubr
        Peak-cell-rate TX: none cbr, none vbr, none abr, none ubr
        Sustained-cell-rate: none vbr RX, none vbr TX
        Minimum-cell-rate RX: none abr, none ubr
        Minimum-cell-rate TX: none abr, none ubr
        CDVT RX: none cbr, none vbr, none abr, none ubr
        CDVT TX: none cbr, none vbr, none abr, none ubr
        MBS: none vbr RX, none vbr TX
Resource Management state:
    Available bit rates (in Kbps):
        72959 cbr RX, 72959 cbr TX, 72959 vbr RX, 72959 vbr TX,
        72959 abr RX, 72959 abr TX, 72959 ubr RX, 72959 ubr TX
    Allocated bit rates:
        0 cbr RX, 0 cbr TX, 0 vbr RX, 0 vbr TX,
        0 abr RX, 0 abr TX, 0 ubr RX, 0 ubr TX
    Best effort connections: 0 pvcs,  0 svcs
```

Configuring Framing Overhead

The interface framing overhead feature determines whether the MaxCR of a physical interface conforms to the actual physical line rate, including framing overhead. By default, the unframed rate is used for determining the MaxCR.

When framing overhead is considered, the MaxCR is less than the unframed rate and some previously configured connections might not be established. Table 8-11 provides the MaxCR values for the different framing modes, with and without framing overhead configured.

Table 8-11 MaxCR For Different Framing Overhead Configurations

Interface Type	Framing Mode	With Framing Overhead Configured	Without Framing Overhead Configured
OC-3	–	149,759 kbps	155,519 kbps
OC-12	–	599,032 kbps	622,079 kbps
OC-48c[1]	–	2,396,156 kbps	2,488,319 kbps
DS3	M23 ADM	44,209 kbps	44,735 kbps
	M23 PLCP	40,704 kbps	44,735 kbps
	CBIT ADM	44,209 kbps	44,735 kbps
	CBIT PLCP	40,704 kbps	44,735 kbps
E3	G 832 ADM	33,920 kbps	34,367 kbps
	G 751 ADM	34,009 kbps	34,367 kbps
	G 751 PLCP	30,528 kbps	34,367 kbps
E1	CRC4 ADM	1919 kbps	2047 kbps
	CRC4 PLCP	1785 kbps	2047 kbps
	PCM30 ADM	1919 kbps	2047 kbps
	PCM30 PLCP	1785 kbps	2047 kbps
T1	SF ADM	1535 kbps	1543 kbps
	SF PLCP	1413 kbps	1543 kbps
	ESF ADM	1535 kbps	1543 kbps
	ESF PLCP	1413 kbps	1543 kbps

1. OC-48c is only available on the Catalyst 8540 MSR.

The framing mode changes when you issue the **framing** command on an interface and the MaxCR is adjusted accordingly. If enabling framing overhead reduces the maximum guaranteed service bandwidth supported on a direction of an interface below the current allocation, use the **force** option to ensure that the configuration takes effect.

To configure framing overhead, use the following interface configuration commands:

	Command	Purpose
Step 1	Switch(config)# **interface atm** *card*/*subcard*/*slot* Switch(config-if)#	Specifies the physical interface to configure.
Step 2	Switch(config-if)# **atm cac framing overhead** [**force**]	Configures framing overhead on an interface

Example

The following example shows how to enable framing overhead on an interface:

```
Switch(config)# interface atm 4/1/0
Switch(config-if)# atm cac framing overhead
```

Displaying the Framing Overhead Configuration

To display the framing overhead configuration, use the following user EXEC command:

Command	Purpose
show atm interface resource atm *card*/*subcard*/*port*[**.***vpt#*]	Displays the interface framing overhead configuration.

Example

The following example shows the framing overhead configuration:

```
Switch> show atm interface resource atm 4/1/0
Resource Management configuration:
    Service Classes:
        Service Category map: c2 cbr, c2 vbr-rt, c3 vbr-nrt, c4 abr, c5 ubr
        Scheduling: RS c1 WRR c2, WRR c3, WRR c4, WRR c5
        WRR Weight: 15 c2, 2 c3, 2 c4, 2 c5
➔   CAC Configuration to account for Framing Overhead : Enabled
    Pacing: disabled   0 Kbps rate configured, 0 Kbps rate installed
    overbooking :  disabled
    Service Categories supported: cbr,vbr-rt,vbr-nrt,abr,ubr
    Link Distance: 0 kilometers
    Controlled Link sharing:
        Max aggregate guaranteed services: none RX,  none TX
        Max bandwidth: none cbr RX, none cbr TX, none vbr RX, none vbr TX,
                       none abr RX, none abr TX, none ubr RX, none ubr TX
        Min bandwidth: none cbr RX, none cbr TX, none vbr RX, none vbr TX,
                       none abr RX, none abr TX, none ubr RX, none ubr TX
    Best effort connection limit: disabled  0 max connections
<information deleted>
```

Configuring ILMI

This chapter describes the Integrated Local Management Interface (ILMI) protocol implementation within the ATM switch router.

Note This chapter provides advanced configuration instructions for the Catalyst 8540 MSR, Catalyst 8510 MSR, and LightStream 1010 ATM switch routers. For a description of the role of ILMI, refer to the *Guide to ATM Technology*. For complete descriptions of the commands mentioned in this chapter, refer to the *ATM Switch Router Command Reference* publication.

This chapter includes the following sections:

Configuring the Global ILMI System

This section describes configuring the ATM address and the LAN emulation configuration server (LECS) address, and displaying the ILMI configuration for the entire switch.

Configuring the ATM Address

The ATM switch router ships with an autoconfigured ATM address. Private Network-Network Interface (PNNI) uses the autoconfigured address to construct a flat PNNI topology. ILMI uses the first 13 bytes of this address as the switch prefix that it registers with end systems. For a description of the autoconfigured ATM address and considerations when assigning a new address, refer to the *Guide to ATM Technology*.

Note The most important rule in the addressing scheme is to maintain the uniqueness of the address across very large networks.

Multiple addresses can be configured for a single switch, and this configuration can be used during ATM address migration. ILMI registers end systems with multiple prefixes during this period until an old address is removed. PNNI automatically summarizes all of the switch's prefixes in its reachable address advertisement.

To configure a new ATM address that replaces the previous ATM address, see the "Configuring the ATM Address" section on page 10-4.

Configuring Global ILMI Access Filters

The ILMI access filter feature allows you to permit or deny certain ILMI registered addresses.

Note If you want to allow certain addresses to be registered via ILMI, but restrict those addressees from being advertised through PNNI, use the PNNI suppressed summary address feature instead. For additional information, see the "Configuring Redistribution" section on page 10-43, or the **summary-address** command in the *ATM Switch Router Command Reference* publication.

If end systems are allowed to register arbitrary addresses via ILMI, including addresses that do not match the ILMI prefixes used on the interface, a security hole may be opened. The ILMI access filter feature closes the security hole by permitting or denying ILMI registration of different classes of addresses.

The ILMI access filter allows you to configure two levels of access filters:

- Globally, to configure the switch default access filter
- At the interface level, to set the per-interface specific override

In either level, you can choose among the following options:

- Permit all—Any ATM end system address (AESA) registered by an attached end system is permitted.
- Permit prefix match—Only AESAs that match an ILMI prefix used on the interface are permitted.
- Permit prefix match and well-known group addresses—AESAs that match an ILMI prefix used on the interface as well as the well-known group addresses, including the old LECS address (47.0079.0000.0000.0000.0000.0000.00A0.3E00.0001.00) and any address matching the ATM Forum address prefix for well-known address (C5.0079.0000.0000.0000.0000.0000.00A0.3E) are permitted.
- Permit prefix match and all group addresses—All group addresses, including the well-known group addresses, as well as AESAs that match the ILMI prefix(es) used on the interface are permitted.

To configure global ILMI access filters, use the following global configuration command:

Command	Purpose
atm ilmi default-access permit {**all** | **matching-prefix** [**all-groups** | **wellknown-groups**]}	Configures an ILMI default access filter.

Note If you use Cisco's Simple Server Redundancy Protocol (SSRP) for LAN emulation in this network, ILMI registration of well-known group addresses should be permitted. This allows the active LECS to register the well-known LECS address with the switch. Either the **permit all**, **permit matching-prefix wellknown-groups**, or **permit matching-prefix all-groups** option should be configured.

Example

The following example configures the global default access filter for ILMI address registration to allow well-known group addresses and addresses with matching prefixes:

```
Switch(config)# atm ilmi default-access permit matching-prefix wellknown-groups
```

See the command **atm address-registration** in the *ATM Switch Router Command Reference* publication for information on configuration of the individual interface access filter override.

Display the ILMI Access Filter Configuration

To display the global ILMI default access configuration, use the following privileged EXEC command:

Command	Purpose
more system:running-config	Displays the global ILMI default access configuration.

Example

The following example displays the ILMI filter configuration for all ATM interfaces:

```
Switch# more system:running-config
Building configuration...
Current configuration:

<information deleted>

!
atm abr-mode efci
atm lecs-address-default 47.0091.8100.0000.0040.0b0a.1281.0040.0b4e.d023.00 1
atm lecs-address-default 47.0091.8100.0000.0040.0b0a.1281.0040.0b07.4023.00 2
➔ atm ilmi default-access permit matching-prefix
atm address 47.0091.8100.0000.0040.0b0a.2b81.0040.0b0a.2b81.00
atm address 47.0091.8100.0000.0060.3e5a.7901.0060.3e5a.7901.00
atm router pnni
 statistics call
 node 1 level 56 lowest
```

Configuring the LANE Configuration Server Address

To configure the LECS address advertised to the directly connected end nodes, use the following global configuration command:

Command	Purpose
atm lecs-address *lecs-address* [*sequence-number*]	Configures the switch LECS address.

The *sequence-number* provides the position of this address in the ordered LECS address table.

Example

The following example shows how to configure the LECS ATM address:

```
Switch(config)# atm lecs-address 47.0091.8100.0000.0000.0ca7.9e01.4000.0c81.9030.01
```

Displaying the ILMI Global Configuration

To display the switch ILMI configuration, use the following EXEC commands:

Command	Purpose
show atm addresses	Displays the ATM addresses.
show atm ilmi-configuration	Displays the ILMI configuration.
show atm ilmi-status	Displays the ILMI status.

Examples

The following example shows the ATM address and the LECS address:

```
Switch# show atm addresses

Switch Address(es):
  47.00918100000000000CA79E01.00000CA79E01.00 active
  88.88888888000000000000000.000000005151.00

Soft VC Address(es):
  47.0091.8100.0000.0000.0ca7.9e01.4000.0c81.0000.00 ATM0
  47.0091.8100.0000.0000.0ca7.9e01.4000.0c81.8000.00 ATM3/0/0
  47.0091.8100.0000.0000.0ca7.9e01.4000.0c81.8010.00 ATM3/0/1
  47.0091.8100.0000.0000.0ca7.9e01.4000.0c81.8020.00 ATM3/0/2
  47.0091.8100.0000.0000.0ca7.9e01.4000.0c81.8030.00 ATM3/0/3
  47.0091.8100.0000.0000.0ca7.9e01.4000.0c81.9000.00 ATM3/1/0
  47.0091.8100.0000.0000.0ca7.9e01.4000.0c81.9010.00 ATM3/1/1
  47.0091.8100.0000.0000.0ca7.9e01.4000.0c81.9020.00 ATM3/1/2
  47.0091.8100.0000.0000.0ca7.9e01.4000.0c81.9030.00 ATM3/1/3

ILMI Switch Prefix(es):
  47.0091.8100.0000.0000.0ca7.9e01
  88.8888.8888.0000.0000.0000.0000

ILMI Configured Interface Prefix(es):

LECS Address(es):
  47.0091.8100.0000.0000.0ca7.9e01.4000.0c81.9030.01
  47.0091.8100.0000.0000.0ca7.9e01.4000.0c81.9030.02
```

Note Since the 12.0(1a)W5(5b) release of the system software, addressing the interface on the route processor (CPU) has changed. The ATM interface is now called atm0, and the Ethernet interface is now called ethernet0. Old formats (atm 2/0/0 and ethernet 2/0/0) are still supported.

The following example shows the ILMI configuration:

```
Switch# show atm ilmi-configuration

Switch ATM Address (s) :
11223344556677889901122334455667788990000
LECS Address (s):
1122334455667788990011223344556677889900
ARP Server Address (s):
11223344556677889900112233445566778899000
```

The following example shows the ILMI status:

```
Switch# show atm ilmi-status

Interface : ATM0 Interface Type : Local
Configured Prefix(s) :
47.0091.8100.0000.0003.c386.b301

Interface : ATM3/0/0 Interface Type : Private NNI
ILMI VCC : (0, 16) ILMI Keepalive : Disabled
Configured Prefix(s) :
47.0091.8100.0000.0003.c386.b301

Interface : ATM3/0/3 Interface Type : Private NNI
ILMI VCC : (0, 16) ILMI Keepalive : Disabled
Configured Prefix(s) :
47.0091.8100.0000.0003.c386.b301
```

Configuring an ILMI Interface

To configure an ILMI interface, perform the following tasks, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **interface atm** *card*/*subcard*/*port*[**.***vpt#*] Switch(config-if)#	Specifies an ATM interface and enters interface configuration mode.
Step 2	Switch(config-if)# **atm auto-configuration**	Enables ILMI autoconfiguration, including determination of interface protocol, version, and side.
Step 3	Switch(config-if)# **atm address-registration**	Configures ILMI address registration for a specified interface.
Step 4	Switch(config-if)# **atm ilmi-keepalive** [*seconds* [**retry** *number*]]	Configures ILMI keepalive.

Note If the ILMI VC (by default VCI = 16) is disabled, then the ILMI is disabled.

Examples

The following example shows how to enable ILMI autoconfiguration on ATM interface 3/0/3:

```
Switch(config)# interface atm 3/0/3
Switch(config-if)# atm auto-configuration
```

The following example shows how to enable ATM address registration on ATM interface 3/0/3:

```
Switch(config)# interface atm 3/0/3
Switch(config-if)# atm address-registration
```

Note If you use the **no atm address-registration** command to disable ILMI on this interface, the keepalives and responses to incoming ILMI queries continue to function. If you want ILMI to be completely disabled at this interface, use the **no atm ilmi-enable** command.

The following example shows how to configure the ILMI ATM interface 3/0/3 with a keepalive time of 20 seconds and retry count of 3:

```
Switch(config)# interface atm 3/0/3
Switch(config-if)# atm ilmi-keepalive 20 retry 3
```

In this example, the peer network element is polled every 20 seconds.

Proceed to the following section to confirm the ILMI interface configuration.

Configuring Per-Interface ILMI Address Prefixes

The ATM switch router allows configuration of per-interface ILMI address prefixes, so different address prefixes can be registered with end systems attached to different interfaces. When any per-interface ILMI address prefixes are configured, they override the prefix(es) derived from the first 13 bytes of the switch ATM address(es) for that specific interface.

Multiple ILMI address prefixes can be configured on each interface; for example, during ATM address migration.

To configure a per-interface ILMI address prefix, perform the following tasks, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **interface atm** *card*/*subcard*/*port*[**.***vpt#*] Switch(config-if)#	Specifies an ATM interface and enters interface configuration mode.
Step 2	Switch(config-if)# **atm prefix** *13-byte-prefix*	Configures the ILMI address prefix.

Examples

The following example shows how to change the ATM address of the switch from the autoconfigured address 47.0091.8100.0000.0041.0b0a.1081.0041.0b0a.1081.00 to the new address 47.0091.8100.5670.0000.0000.1122.0041.0b0a.1081.00:

```
Switch(config)# atm address 47.0091.8100.5670.0000.0000.1122...
Switch(config)# no atm address 47.0091.8100.0000.0041.0b0a.1081...
```

The following example shows how to configure an additional ATM address manually, or address prefix 47.0091.8100.0000.0003.c386.b301 on ATM interface 0/0/1:

```
Switch(config)# interface atm 0/0/1
Switch(config-if)# atm prefix 47.0091.8100.0000.0003.c386.b301
```

Displaying ILMI Address Prefix

Use the **show atm addresses** command to display the ILMI address prefix configuration for all interfaces or a specific interface.

To display the ILMI address prefix configuration for all interfaces, use the following EXEC command:

Command	Purpose
show atm addresses	Displays the interface ILMI address prefix configuration.

Example

The following example shows the ILMI address prefix configuration for all ATM interfaces:

```
Switch# show atm addresses

Switch Address(es):
  47.00918100000000410B0A1081.00410B0A1081.00 active
  47.00918100000000603E5ADB01.00603E5ADB01.00
  47.0091810056700000000001122.00400B0A1081.00

Soft VC Address(es):
  47.0091.8100.0000.0041.0b0a.1081.4000.0c80.0000.00 ATM0/0/0
  47.0091.8100.0000.0041.0b0a.1081.4000.0c80.0000.63 ATM0/0/0.99
  47.0091.8100.0000.0041.0b0a.1081.4000.0c80.0010.00 ATM0/0/1
  47.0091.8100.0000.0041.0b0a.1081.4000.0c80.0020.00 ATM0/0/2
  47.0091.8100.0000.0041.0b0a.1081.4000.0c80.0030.00 ATM0/0/3
  47.0091.8100.0000.0041.0b0a.1081.4000.0c80.1000.00 ATM0/1/0
  47.0091.8100.0000.0041.0b0a.1081.4000.0c80.1010.00 ATM0/1/1
  47.0091.8100.0000.0041.0b0a.1081.4000.0c80.1020.00 ATM0/1/2
  47.0091.8100.0000.0041.0b0a.1081.4000.0c80.1030.00 ATM0/1/3
  47.0091.8100.0000.0041.0b0a.1081.4000.0c80.8000.00 ATM1/0/0
  47.0091.8100.0000.0041.0b0a.1081.4000.0c80.8010.00 ATM1/0/1
  47.0091.8100.0000.0041.0b0a.1081.4000.0c80.8020.00 ATM1/0/2
  47.0091.8100.0000.0041.0b0a.1081.4000.0c80.8030.00 ATM1/0/3
  47.0091.8100.0000.0041.0b0a.1081.4000.0c80.9000.00 ATM1/1/0
  47.0091.8100.0000.0041.0b0a.1081.4000.0c80.9010.00 ATM1/1/1
  47.0091.8100.0000.0041.0b0a.1081.4000.0c80.9020.00 ATM1/1/2
  47.0091.8100.0000.0041.0b0a.1081.4000.0c80.9030.00 ATM1/1/3

ILMI Switch Prefix(es):
  47.0091.8100.0000.0041.0b0a.1081
  47.0091.8100.0000.0060.3e5a.db01
  47.0091.8100.5670.0000.0000.1122

ILMI Configured Interface Prefix(es):

LECS Address(es):
```

Displaying the ILMI Interface Configuration

To show the ILMI interface configuration, use the following EXEC command:

Command	Purpose
show atm ilmi-status atm *card*/*subcard*/*port*	Shows the ILMI configuration on a per-port basis.

Example

The following example displays the ILMI status for ATM interface 3/0/0:

```
Switch# show atm ilmi-status atm 3/0/0

Interface : ATM3/0/0 Interface Type : Private NNI
ILMI VCC : (0, 16) ILMI Keepalive : Disabled
Configured Prefix(s) :
47.0091.8100.0000.0003.c386.b301
```

Configuring ATM Address Groups

ATM address groups allow more than one interface to have the same ATM address. These multiple connections provide load balancing for traffic from an end station.

Configure the interfaces in a group by performing the following tasks, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **interface atm** *card*/*subcard*/*port*[**.***vpt#*] Switch(config-if)#	Specifies an ATM interface and enters interface configuration mode.
Step 2	Switch(config-if)# **atm interface-group** *number*	Configures the ATM address group.

Example

The following example shows how to configure ATM interface 1/1/0 and ATM interface 3/0/1 in ATM address group 5:

```
Switch(config)# interface atm 1/1/0
Switch(config-if)# atm interface-group 5
Switch(config-if)# exit
Switch(config)# interface atm 3/0/1
Switch(config-if)# atm interface-group 5
```

Displaying ATM Address Group Configuration

To determine if an interface is a member of an ATM address group, use the following privileged EXEC command:

Command	Purpose
show running-config interface atm *card*/*subcard*/*port*	Shows the ILMI configuration on a per-port basis.

Example

The following example shows the ATM address group configuration for ATM interface 1/1/0 and ATM interface 3/0/1:

```
Switch# show running-config interface atm 1/1/0
Building configuration...

Current configuration:
!
➔ interface ATM1/1/0
 no ip address
 no ip directed-broadcast
 no atm ilmi-keepalive
 atm prefix 47.0091.8100.5670.0000.0000.1122...
➔  atm interface-group 5
 clock source free-running
end
Switch# show running-config interface atm 3/0/1
Building configuration...

Current configuration:
!
➔ interface ATM3/0/1
 no ip address
 no ip directed-broadcast
 no atm ilmi-keepalive
 atm prefix 47.0091.8100.5670.0000.0000.1122...
➔  atm interface-group 5
 clock source free-running
end
```

CHAPTER 10

Configuring ATM Routing and PNNI

This chapter describes the Interim Interswitch Signaling Protocol (IISP) and Private Network-Network Interface (PNNI) ATM routing protocol implementations on the ATM switch router.

Note This chapter provides advanced configuration instructions for the Catalyst 8540 MSR, Catalyst 8510 MSR, and LightStream 1010 ATM switch routers. For conceptual and background information, refer to the *Guide to ATM Technology*. For complete descriptions of the commands mentioned in this chapter, refer to the *ATM Switch Router Command Reference* publication.

This chapter includes the following sections:

Overview

To place calls between ATM end systems, signalling consults either IISP, a static routing protocol, or PNNI, a dynamic routing protocol. PNNI provides quality of service (QoS) routes to signalling based on the QoS requirements specified in the call setup request.

For detailed discussions of the following topics, refer to the *Guide to ATM Technology*:

- IISP routing
- PNNI signalling and routing
- Mechanisms and components of single-level and hierarchical PNNI

ATM Addresses

The autoconfigured ATM address of the ATM switch router suffices when implementing single-level PNNI. Hierarchical PNNI requires an addressing scheme to ensure global uniqueness of the ATM address and to plan for future network expansion.

For detailed discussions of the following related topics, refer to the *Guide to ATM Technology*:

- The autoconfigured ATM address for single-level PNNI
- E.164 AESA prefixes
- Designing an ATM address plan for hierarchical PNNI
- Obtaining registered ATM addresses

IISP Configuration

This section describes the procedures necessary for Interim Interswitch Signaling Protocol (IISP) configuration, and includes the following subsections:

- Configuring the Routing Mode, page 10-2
- Configuring the ATM Address, page 10-4
- Configuring Static Routes, page 10-6

Configuring the Routing Mode

The ATM routing software can be restricted to operate in static mode. In this mode, the call routing is restricted to only the static configuration of ATM routes, disabling operation of any dynamic ATM routing protocols, such as PNNI.

The **atm routing-mode** command is different from deleting all PNNI nodes using the **node** command and affects Integrated Local Management Interface (ILMI) autoconfiguration. If the switch is configured using static routing mode on each interface, the switch ILMI variable atmfAtmLayerNniSigVersion is set to IISP. This causes either of the following to happen:

- ILMI autoconfiguration on the interfaces between two switches determines the interface type as IISP.
- The switch on the other side indicates that the Network-Network Interface (NNI) signalling protocol is not supported.

Note The **atm routing-mode** command is activated only after the next software reload. The switch continues to operate in the current mode until the software is reloaded.

To configure the routing mode, perform these steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **atm routing-mode static**	Configures the ATM routing mode to static.
Step 2	Switch(config)# **end** Switch#	Exits configuration mode.
Step 3	Switch# **copy system:running-config nvram:startup-config**	Writes the running configuration to the startup configuration.
Step 4	Switch# **reload**	Reloads the switch software.

Example

The following example shows how to use the **atm routing-mode static** command to restrict the switch operation to static routing mode:

```
Switch(config)# atm routing-mode static
This Configuration Will Not Take Effect Until Next Reload.
Switch(config)# end
Switch# copy system:running-config nvram:startup-config
Building configuration...
[OK]
Switch# reload
```

The following example shows how to reset the switch operation back to PNNI if the switch is operating in static mode:

```
Switch(config)# no atm routing-mode static
This Configuration Will Not Take Effect Until Next Reload.
Switch(config)# end
Switch# copy system:running-config nvram:startup-config
Building configuration...
[OK]
Switch# reload
```

Displaying the ATM Routing Mode Configuration

To display the ATM routing mode configuration, use the following privileged EXEC command:

Command	Purpose
more system:running-config	Displays the ATM routing mode configuration.

Example

The following example shows the ATM routing mode configuration using the **more system:running-config** privileged EXEC command:

```
Switch# more system:running-config
Building configuration...

Current configuration:
!
version 11.2
<information deleted>
!
hostname Switch
!
username dtate
ip rcmd remote-username dplatz
!
atm e164 translation-table
 e164 address 1111111 nsap-address 11.111111111111111111111111.112233445566.11
 e164 address 2222222 nsap-address 22.222222222222222222222222.112233445566.22
 e164 address 3333333 nsap-address 33.333333333333333333333333.112233445566.33
!
➔ atm routing-mode static
atm address 47.0091.8100.0000.0040.0b0a.2b81.0040.0b0a.2b81.00
!
<information deleted>
```

Configuring the ATM Address

If you are planning to implement only a flat topology network (and have no future plans to migrate to PNNI hierarchy), you can skip this section and use the preconfigured ATM address assigned by Cisco Systems.

Note For information about ATM address considerations, see the “ATM Addresses” section on page 10-2.

To change the active ATM address, create a new address, verify that it exists, and then delete the current active address. Follow these steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **atm address** *new-address-template*	Configures the ATM address for the switch.
Step 2	Switch(config)# **end** Switch#	Returns to privileged EXEC mode.
Step 3	Switch# **show atm addresses**	Verifies the new address.
Step 4	Switch# **configure terminal** Switch(config)#	Enters configuration mode from the terminal.
Step 5	Switch(config)# **no atm address** *old-address-template*	Removes the old ATM address from the switch.

Example

The following example shows how to add the ATM address prefix 47.0091.8100.5670.000.0ca7.ce01. Using the ellipses (...) adds the default Media Access Control (MAC) address as the last six bytes.

```
Switch(config)# atm address 47.0091.8100.5670.0000.0ca7.ce01...
Switch(config)# no atm address 47.0091.8100.0000.0041.0b0a.1081...
```

Displaying the ATM Address Configuration

To display the ATM address configuration, use the following EXEC command:

Command	Purpose
show atm addresses	Displays the ATM address configuration.

Example

The following example shows the ATM address configuration using the **show atm addresses** EXEC command:

```
Switch# show atm addresses

Switch Address(es):
➔ 47.00918100000000410B0A1081.00410B0A1081.00 active
  47.009181005670000000CA7CE01.00410B0A1081.00

Soft VC Address(es):
  47.0091.8100.0000.0041.0b0a.1081.4000.0c80.0000.00 ATM0/0/0
  47.0091.8100.0000.0041.0b0a.1081.4000.0c80.0000.63 ATM0/0/0.99
  47.0091.8100.0000.0041.0b0a.1081.4000.0c80.0010.00 ATM0/0/1
  47.0091.8100.0000.0041.0b0a.1081.4000.0c80.0020.00 ATM0/0/2
  47.0091.8100.0000.0041.0b0a.1081.4000.0c80.0030.00 ATM0/0/3
  47.0091.8100.0000.0041.0b0a.1081.4000.0c80.1000.00 ATM0/1/0
  47.0091.8100.0000.0041.0b0a.1081.4000.0c80.1010.00 ATM0/1/1
  47.0091.8100.0000.0041.0b0a.1081.4000.0c80.1020.00 ATM0/1/2
  47.0091.8100.0000.0041.0b0a.1081.4000.0c80.1030.00 ATM0/1/3
  47.0091.8100.0000.0041.0b0a.1081.4000.0c80.8000.00 ATM1/0/0
  47.0091.8100.0000.0041.0b0a.1081.4000.0c80.8010.00 ATM1/0/1
  47.0091.8100.0000.0041.0b0a.1081.4000.0c80.8020.00 ATM1/0/2
  47.0091.8100.0000.0041.0b0a.1081.4000.0c80.8030.00 ATM1/0/3
  47.0091.8100.0000.0041.0b0a.1081.4000.0c80.9000.00 ATM1/1/0
  47.0091.8100.0000.0041.0b0a.1081.4000.0c80.9010.00 ATM1/1/1
  47.0091.8100.0000.0041.0b0a.1081.4000.0c80.9020.00 ATM1/1/2
  47.0091.8100.0000.0041.0b0a.1081.4000.0c80.9030.00 ATM1/1/3

ILMI Switch Prefix(es):
  47.0091.8100.0000.0041.0b0a.1081
  47.0091.8100.0000.0060.3e5a.db01

ILMI Configured Interface Prefix(es):

LECS Address(es):
```

Configuring Static Routes

Use the **atm route** command to configure a static route. A static route attached to an interface allows all ATM addresses matching the configured address prefix to be reached through that interface.

Note For private User-Network Interface (UNI) interfaces where ILMI address registration is not used, internal-type static routes should be configured to a 19-byte address prefix representing the attached end system.

To configure a static route, use the following global configuration command:

Command	Purpose
atm route *addr-prefix* **atm** *card*/*subcard*/*port* [**e164-address** *address-string* [**number-type** *numtype*]] [**internal**] [**scope** *org-scope*] [**aesa-gateway** *aesa-address*]	Specifies a static route to a reachable address prefix.

Examples

The following example uses the **atm route** command to configure a static route to the 13-byte switch prefix 47.00918100000000410B0A1081 to ATM interface 0/0/0:

```
Switch(config)# atm route 47.0091.8100.0000.0041.0B0A.1081 atm 0/0/0
```

The following example uses the **atm route** command to configure a static route to the 13-byte switch prefix 47.00918100000000410B0A1081 to ATM interface 0/0/0 configured with a scope 1 associated:

```
Switch(config)# atm route 47.0091.8100.0000.0041.0B0A.1081 atm 0/0/0 scope 1
```

Displaying the Static Route Configuration

To display the ATM static route configuration, use the following EXEC command:

Command	Purpose
show atm route	Displays the static route configuration.

Examples

The following example shows the ATM static route configuration using the **show atm route** privileged EXEC command:

```
Switch# show atm route

Codes: P - installing Protocol (S - Static, P - PNNI, R - Routing control),
       T - Type (I - Internal prefix, E - Exterior prefix, SE -
                 Summary Exterior prefix, SI - Summary Internal prefix,
                 ZE - Suppress Summary Exterior, ZI - Suppress Summary Internal)

P  T Node/Port        St Lev Prefix
~ ~~ ~~~~~~~~~~~~~~~~ ~~ ~~~ ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
S  E 1   ATM0/0/0     DN 56  47.0091.8100.0000/56
S  E 1   ATM0/0/0     DN 0   47.0091.8100.0000.00/64
                             (E164 Address 1234567)
R SI 1   0            UP 0   47.0091.8100.0000.0041.0b0a.1081/104
R  I 1   ATM0         UP 0   47.0091.8100.0000.0041.0b0a.1081.0041.0b0a.1081/152
R  I 1   ATM0         UP 0   47.0091.8100.0000.0041.0b0a.1081.4000.0c/128
R SI 1   0            UP 0   47.0091.8100.5670.0000.0000.0000/104
R  I 1   ATM0         UP 0   47.0091.8100.5670.0000.0000.0000.0040.0b0a.1081/152
R  I 1   ATM0         UP 0   47.0091.8100.5670.0000.0000.0000.4000.0c/128
```

Configuring ATM Address Groups

ATM address groups allow more than one interface to have the same internal address prefix for the same static route. These multiple static routes provide load balancing for traffic from an end station.

Configure the interfaces in a group by performing the following tasks, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **interface atm** *card*/*subcard*/*port*[**.***vpt#*] Switch(config-if)#	Specifies an ATM interface and enters interface configuration mode.
Step 2	Switch(config-if)# **atm interface-group** *number*	Configures the ATM address group.

Example

The following example shows how to configure ATM interface 1/1/0 and ATM interface 3/0/1 in ATM address group 5:

```
Switch(config)# interface atm 1/1/0
Switch(config-if)# atm interface-group 5
Switch(config-if)# exit
Switch(config)# interface atm 3/0/1
Switch(config-if)# atm interface-group 5
```

Displaying ATM Address Group Configuration

To determine if an interface is a member of an ATM address group, use the following privileged EXEC command:

Command	Purpose
show running-config interface atm *card*/*subcard*/*port*	Shows the ILMI configuration on a per-port basis.

Example

The following example shows the ATM address group configuration for ATM interface 1/1/0 and ATM interface 3/0/1:

```
Switch# show running-config interface atm 1/1/0
Building configuration...

Current configuration:
!
➔ interface ATM1/1/0
 no ip address
 no ip directed-broadcast
 no atm ilmi-keepalive
 atm prefix 47.0091.8100.5670.0000.0000.1122...
➔ atm interface-group 5
 clock source free-running
end
Switch# show running-config interface atm 3/0/1
Building configuration...

Current configuration:
!
➔ interface ATM3/0/1
 no ip address
 no ip directed-broadcast
 no atm ilmi-keepalive
 atm prefix 47.0091.8100.5670.0000.0000.1122...
➔ atm interface-group 5
 clock source free-running
end
```

Basic PNNI Configuration

This section describes all the procedures necessary for a basic PNNI configuration and includes the following subsections:

- Configuring PNNI without Hierarchy, page 10-9
- Configuring the Lowest Level of the PNNI Hierarchy, page 10-9
- Configuring Higher Levels of the PNNI Hierarchy, page 10-16

Configuring PNNI without Hierarchy

The ATM switch router defaults to a working PNNI configuration suitable for operation in isolated flat topology ATM networks. The switch comes with a globally unique preconfigured ATM address. Manual configuration is not required if you:

- Have a flat network topology
- Do not plan to connect the switch to a service provider network
- Do not plan to migrate to a PNNI hierarchy in the future

If you plan to migrate your flat network topology to a PNNI hierarchical topology, proceed to the next section "Configuring the Lowest Level of the PNNI Hierarchy."

Configuring the Lowest Level of the PNNI Hierarchy

This section describes how to configure the lowest level of the PNNI hierarchy. The lowest-level nodes comprise the lowest level of the PNNI hierarchy. When only the lowest-level nodes are configured, there is no hierarchical structure. If your network is relatively small and you want the benefits of PNNI, but do not need the benefits of a hierarchical structure, follow the procedures in this section to configure the lowest level of the PNNI hierarchy.

To implement multiple levels of PNNI hierarchy, first complete the procedures in this section and then proceed to the "Configuring Higher Levels of the PNNI Hierarchy" section on page 10-16.

Configuring an ATM Address and PNNI Node Level

The ATM switch router is preconfigured as a single lowest-level PNNI node (locally identified as node 1) with a level of 56. The node ID and peer group ID are calculated based on the current active ATM address.

Note If you are planning to implement only a flat topology network (and have no future plans to migrate to PNNI hierarchy), you can skip this section and use the preconfigured ATM address.

To configure a node in a higher level of the PNNI hierarchy, the value of the node level must be a smaller number. For example, a three-level hierarchical network could progress from level 72 to level 64 to level 56. Notice that the level numbers graduate from largest at the lowest level (72) to smallest at the highest level (56).

To change the active ATM address you must create a new address, verify that it exists, and then delete the current active address. After you have entered the new ATM address, disable node 1 and then reenable it. At the same time, you can change the node level if required for your configuration. The identifiers for all higher level nodes are recalculated based on the new ATM address.

Caution Node IDs and peer group IDs are not recalculated until the node is disabled and then reenabled.

To change the active ATM address, perform these steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **atm address** *new-address-template*	Configures the new ATM address for the switch.
Step 2	Switch(config)# **end** Switch#	Returns to privileged EXEC mode.
Step 3	Switch# **show atm addresses**	Verifies the new address.
Step 4	Switch# **configure terminal** Switch(config)#	Enters configuration mode from the terminal.
Step 5	Switch(config)# **no atm address** *old-address-template*	Removes the old ATM address from the switch.
Step 6	Switch(config)# **atm router pnni** Switch(config-atm-router)#	Enters ATM router PNNI mode from the terminal.
Step 7	Switch(config-atm-router)# **node 1 disable** Switch(config-pnni-node)#	Disables the PNNI node.
Step 8	Switch(config-pnni-node)# **node 1 level** *number* **enable**	Reenables the node. You can also change the node level if required for your configuration.

Example

The following example changes the ATM address of the switch from the autoconfigured address 47.0091.8100.0000.0041.0b0a.1081.0041.0b0a.1081.00 to the new address prefix 47.0091.8100.5670.0000.0000.1122.0041.0b0a.1081.00, and causes the node identifier and peer group identifier to be recalculated:

```
Switch(config)# atm address 47.0091.8100.5670.0000.0000.1122...
Switch(config)# no atm address 47.0091.8100.0000.0041.0b0a.1081...
Switch(config)# atm router pnni
Switch(config-atm-router)# node 1 disable
Switch(config-pnni-node)# node 1 enable
```

Displaying the PNNI Node Configuration

To display the ATM PNNI node configuration, use the following privileged EXEC command:

Command	Purpose
show atm pnni local-node	Displays the ATM PNNI node configuration.

Example

The following example shows the PNNI node configuration using the show **atm pnni local-node** privileged EXEC command:

```
Switch# show atm pnni local-node

PNNI node 1 is enabled and running
  Node name: eng_1
  System address          47.0091810000000002EB1FFE00.0002EB1FFE00.01
  Node ID         56:160:47.0091810000000002EB1FFE00.0002EB1FFE00.00
  Peer group ID       56:160:47.0000.0000.0000.0000.0000
  Level 56, Priority 0 0, No. of interfaces 1, No. of neighbors 0
  Parent Node Index: 2
  Node Allows Transit Calls
  Node Representation: simple

  Hello interval 15 sec, inactivity factor 5,
  Hello hold-down 10 tenths of sec
  Ack-delay 10 tenths of sec, retransmit interval 5 sec,
  Resource poll interval 5 sec
  SVCC integrity times: calling 35 sec, called 50 sec,
  Horizontal Link inactivity time 120 sec,
  PTSE refresh interval 1800 sec, lifetime factor 200 percent,
  Min PTSE interval 10 tenths of sec
  Auto summarization: on, Supported PNNI versions: newest 1, oldest 1
  Default administrative weight mode: uniform
  Max admin weight percentage: -1
  Next resource poll in 3 seconds
  Max PTSEs requested per PTSE request packet: 32
  Redistributing static routes: Yes
```

Configuring Static Routes

Because PNNI is a dynamic routing protocol, static routes are not necessary between nodes that support PNNI. However, you can extend the routing capability of PNNI beyond nodes that support PNNI to:

- Connect to nodes outside of a peer group that do not support PNNI
- Define routes to end systems that do not support Integrated Local Management Interface (ILMI)

Use the **atm route** command to configure a static route. A static route attached to an interface allows all ATM addresses matching the configured address prefix to be reached through that interface.

Note Two PNNI peer groups can be connected using the IISP protocol. Connecting PNNI peer groups requires that a static route be configured on the IISP interfaces, allowing connections to be set up across the IISP link(s).

To configure a static route connection, use the following global configuration command:

Command	Purpose
atm route *addr-prefix* **atm** *card/subcard/port* [**e164-address** *address-string* [**number-type** *numtype*]] [**internal**] [**scope** *org-scope*]	Specifies a static route to a reachable address prefix.

Examples

The following example uses the **atm route** command to configure a static route to the 13-byte switch prefix 47.00918100000000410B0A1081 to ATM interface 0/0/0:

```
Switch(config)# atm route 47.0091.8100.0000.0041.0B0A.1081 atm 0/0/0
```

The following example uses the **atm route** command to configure a static route to the 13-byte switch prefix 47.00918100000000410B0A1081 to ATM interface 0/0/0 configured with a scope 1 associated:

```
Switch(config)# atm route 47.0091.8100.0000.0041.0B0A.1081 atm 0/0/0 scope 1
```

Displaying the Static Route Configuration

To display the ATM static route configuration, use the following EXEC command:

Command	Purpose
show atm route	Displays the static route configuration.

Example

The following example shows the ATM static route configuration using the **show atm route** EXEC command:

```
Switch# show atm route

Codes: P - installing Protocol (S - Static, P - PNNI, R - Routing control),
       T - Type (I - Internal prefix, E - Exterior prefix, SE -
                 Summary Exterior prefix, SI - Summary Internal prefix,
                 ZE - Suppress Summary Exterior, ZI - Suppress Summary Internal)

P  T Node/Port        St Lev Prefix
~ ~~ ~~~~~~~~~~~~~~~~ ~~ ~~~ ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
S  E 1   ATM0/0/0     DN 56  47.0091.8100.0000/56
S  E 1   ATM0/0/0     DN 0   47.0091.8100.0000.00/64
                             (E164 Address 1234567)
R SI 1   0            UP 0   47.0091.8100.0000.0041.0b0a.1081/104
R  I 1   ATM0         UP 0   47.0091.8100.0000.0041.0b0a.1081.0041.0b0a.1081/152
R  I 1   ATM0         UP 0   47.0091.8100.0000.0041.0b0a.1081.4000.0c/128
R SI 1   0            UP 0   47.0091.8100.5670.0000.0000.0000/104
R  I 1   ATM0         UP 0   47.0091.8100.5670.0000.0000.0000.0040.0b0a.1081/152
R  I 1   ATM0         UP 0   47.0091.8100.5670.0000.0000.0000.4000.0c/128
```

Configuring a Summary Address

You can configure summary addresses to reduce the amount of information advertised by a PNNI node and contribute to scalability in large networks. Each summary address consists of a single reachable address prefix that represents a collection of end system or node addresses. We recommend that you use summary addresses when all end system addresses that match the summary address are directly reachable from the node. However, this is not always required because routes are always selected by nodes advertising the longest matching prefix to a destination address.

By default, each lowest-level node has a summary address equal to the 13-byte address prefix of the ATM address of the switch. This address prefix is advertised into its peer group.

You can configure multiple addresses for a single switch which are used during ATM address migration. ILMI registers end systems with multiple prefixes during this period until an old address is removed. PNNI automatically creates 13-byte summary address prefixes from all of its ATM addresses.

You must configure summary addresses (other than the defaults) on each node. Each node can have multiple summary address prefixes. Use the **summary-address** command to manually configure summary address prefixes.

Note The **no auto-summary** command removes the default summary address(es). Use the **no auto-summary** command when systems that match the first 13-bytes of the ATM address(es) of your switch are attached to different switches. You can also use this command for security purposes.

To configure a summary address, perform these steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **atm router pnni** Switch(config-atm-router)#	Enters ATM router PNNI mode.
Step 2	Switch(config-atm-router)# **node** *node-index* Switch(config-pnni-node)#	Enters node configuration mode.
Step 3	Switch(config-pnni-node)# **no auto-summary**	Removes the default summary address(es).
Step 4	Switch(config-pnni-node)# **summary-address** *address-prefix*	Configures the ATM PNNI summary address prefix.

Example

The following example shows how to remove the default summary address(es) and add summary address 47.009181005670:

```
Switch(config)# atm router pnni
Switch(config-atm-router)# node 1
Switch(config-pnni-node)# no auto-summary
Switch(config-pnni-node)# summary-address 47.009181005670
```

Displaying the Summary Address Configuration

To display the ATM PNNI summary address configuration, use the following privileged EXEC command:

Command	Purpose
show atm pnni summary	Displays a summary of the PNNI hierarchy.

Example

The following example shows the ATM PNNI summary address configuration using the **show atm pnni summary** privileged EXEC command:

```
Switch# show atm pnni summary

Codes: Node - Node index advertising this summary
       Type - Summary type (INT - internal, EXT - exterior)
       Sup  - Suppressed flag (Y - Yes, N - No)
       Auto - Auto Summary flag (Y - Yes, N - No)
       Adv  - Advertised flag (Y - Yes, N - No)

 Node Type Sup Auto Adv  Summary Prefix
 ~~~~ ~~~~ ~~~ ~~~~ ~~~  ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
  1   Int   N   Y    Y   47.0091.8100.0000.0040.0b0a.2a81/104
  2   Int   N   Y    N   47.01b1.0000.0000.0000.00/80
```

Configuring Scope Mapping

The PNNI address scope allows you to restrict advertised reachability information within configurable boundaries.

Note On UNI and IISP interfaces, the scope is specified in terms of organizational scope values ranging from 1 (local) to 15 (global). (Refer to the ATM Forum UNI Signalling 4.0 specification for more information.)

In PNNI networks, the scope is specified in terms of PNNI levels. The mapping from organizational scope values used at UNI and IISP interfaces to PNNI levels is configured on the lowest-level node. The mapping can be determined automatically (which is the default setting) or manually, depending on the configuration of the **scope mode** command.

In manual mode, whenever the level of node 1 is modified, the scope map should be reconfigured to avoid unintended suppression of reachability advertisements. Misconfiguration of the scope map might cause addresses to remain unadvertised.

In automatic mode, the UNI to PNNI level mapping is automatically reconfigured whenever the level of the node 1 is modified. The automatic reconfiguration avoids misconfigurations caused by node level modifications. Automatic adjustment of scope mapping uses the values shown in Table 10-1.

Table 10-1 Scope Mapping Table

Organizational Scope	ATM Forum PNNI 1.0 Default Level	Automatic Mode PNNI Level
1 to 3	96	Minimum (l,96)
4 to 5	80	Minimum (l,80)
6 to 7	72	Minimum (l,72)
8 to 10	64	Minimum (l,64)
11 to 12	48	Minimum (l,48)
13 to 14	32	Minimum (l,32)
15 (global)	0	0

Entering the **scope mode automatic** command ensures that all organizational scope values cover an area at least as wide as the current node's peer group. Configuring the scope mode to **manual** disables this feature and no changes can be made without explicit configuration.

To configure the PNNI scope mapping, perform these steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **atm router pnni** Switch(config-atm-router)#	Enters ATM router PNNI mode.
Step 2	Switch(config-atm-router)# **node** *node-index* Switch(config-pnni-node)#	Enters node configuration mode.
Step 3	Switch(config-pnni-node)# **scope mode manual**	Configures scope mode as manual.[1]
Step 4	Switch(config-pnni-node)# **scope map** *low-org-scope* [*high-org-scope*] **level** *number*	Configures node scope mapping.

1. You must enter the **scope mode manual** command to allow scope mapping configuration.

Example

The following example shows how to configure PNNI scope mapping manually so that organizational scope values 1 through 8 map to PNNI level 72:

```
Switch(config)# atm router pnni
Switch(config-atm-router)# node 1
Switch(config-pnni-node)# scope mode manual
Switch(config-pnni-node)# scope map 1 8 level 72
```

Displaying the Scope Mapping Configuration

To display the PNNI scope mapping configuration, use the following privileged EXEC command:

Command	Purpose
show atm pnni scope	Displays the node PNNI scope mapping configuration.

Example

The following example shows the ATM PNNI scope mapping configuration using the **show atm pnni scope** privileged EXEC command:

```
Switch# show atm pnni scope

UNI scope   PNNI Level
~~~~~~~~~   ~~~~~~~~~~
(1  - 10)      56
(11 - 12)      48
(13 - 14)      32
(15 - 15)      0

Scope mode: manual
```

Configuring Higher Levels of the PNNI Hierarchy

Once you have configured the lowest level of the PNNI hierarchy, you can configure the higher levels. To do so, you must configure peer group leaders (PGLs) and logical group nodes (LGNs).

For an explanation of PGLs and LGNs, as well as guidelines for creating a PNNI hierarchy, refer to the *Guide to ATM Technology*.

Configuring a Logical Group Node and Peer Group Identifier

The LGN is created only when the child node in the same switch (that is, the node whose parent configuration points to this node) is elected PGL of the child peer group.

The peer group identifier defaults to a value created from the first part of the child peer group identifier, and does not need to be specified. If you want a nondefault peer group identifier, you must configure all logical nodes within a peer group with the same peer group identifier.

Higher level nodes are only active if:

- A lower-level node specifies the higher-level node as a parent.
- The election leadership priority of the child node is configured with a non-zero value and is elected as the PGL.

To configure a LGN and peer group identifier, perform these steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **atm router pnni** Switch(config-atm-router)#	Enters ATM router PNNI mode.
Step 2	Switch(config-atm-router)# **node** *node-index* **level** *number* [**lowest**] [**peer-group-identifier** *dd:xxx*] [**enable** | **disable**]	Configures the logical node and optionally its peer group identifier. Configures each logical node in the peer group with the same peer group identifier. When you have more than one logical node on the same switch, you must specify a different index number to distinguish it from node 1.

Examples

The following example shows how to create a new node 2 with a level of 56 and a peer group identifier of 56:47009111223344:

```
Switch(config)# atm router pnni
Switch(config-atm-router)# node 2 level 56 peer-group-identifier 56:47009111223344 enable
Switch(config-pnni-node)# end
```

Notice that the PNNI level and the first two digits of the peer group identifier are the same.

Displaying the Logical Group Node Configuration

To display the LGN configuration, use the following privileged EXEC command:

Command	Purpose
show atm pnni local-node	Displays the PNNI node information.

Example

The following example shows the PNNI node information using the **show atm pnni local-node** privileged EXEC command:

```
Switch# show atm pnni local-node 2

PNNI node 2 is enabled and not running
  Node name: Switch.2.56
  System address          47.00918100000000000000001.000000000001.02
  Node ID            56:0:00.000000000000000000000000.000000000001.00
  Peer group ID        56:47.0091.1122.3344.0000.0000.0000
  Level 56, Priority 0 0, No. of interfaces 0, No. of neighbors 0
  Parent Node Index: NONE
  Node Allows Transit Calls
  Node Representation: simple

  Hello interval 15 sec, inactivity factor 5,
  Hello hold-down 10 tenths of sec
  Ack-delay 10 tenths of sec, retransmit interval 5 sec,
  Resource poll interval 5 sec
  SVCC integrity times: calling 35 sec, called 50 sec,
  Horizontal Link inactivity time 120 sec,
  PTSE refresh interval 1800 sec, lifetime factor 200 percent,
  Min PTSE interval 10 tenths of sec
  Auto summarization: on, Supported PNNI versions: newest 1, oldest 1
  Default administrative weight mode: uniform
  Max admin weight percentage: -1
  Max PTSEs requested per PTSE request packet: 32
  Redistributing static routes: No
```

Configuring the Node Name

PNNI node names default to names based on the host name. However, you can change the default node name to more accurately reflect the peer group. We recommend you chose a node name of 12 characters or less so that your screen displays remain nicely formatted and easy to read.

After a node name has been configured, it is distributed to all other nodes by PNNI flooding. This allows the node to be identified by its node name in PNNI **show** commands.

Note See section "Configuring System Information" section on page 3-19 for information about configuring host names.

To configure the PNNI node name, perform these steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **atm router pnni** Switch(config-atm-router)#	Enters ATM router PNNI mode.
Step 2	Switch(config-atm-router)# **node** *node-index* Switch(config-pnni-node)#	Enters node configuration mode.
Step 3	Switch(config-pnni-node)# **name** *name*	Configures the node name.

Example

Configure the name of the node as eng_1 using the **name** command, as in the following example:

```
Switch(config)# atm router pnni
Switch(config-atm-router)# node 1
Switch(config-pnni-node)# name eng_1
```

Displaying the Node Name Configuration

To display the ATM PNNI node name configuration, use the following privileged EXEC command:

Command	Purpose
show atm pnni local-node	Displays the ATM PNNI router configuration.

Example

This example shows how to display the ATM node name configuration using the **show atm pnni local-node** command from user EXEC mode:

```
Switch# show atm pnni local-node
PNNI node 1 is enabled and running
➔ Node name: eng_1
  System address          47.0091810000000002EB1FFE00.0002EB1FFE00.01
  Node ID          56:160:47.0091810000000002EB1FFE00.0002EB1FFE00.00
  Peer group ID        56:16.0347.0000.0000.0000.0000.0000
  Level 56, Priority 0 0, No. of interfaces 1, No. of neighbors 0
  Parent Node Index: 2
  Node Allows Transit Calls
  Node Representation: simple

  Hello interval 15 sec, inactivity factor 5,
  Hello hold-down 10 tenths of sec
  Ack-delay 10 tenths of sec, retransmit interval 5 sec,
  Resource poll interval 5 sec
  SVCC integrity times: calling 35 sec, called 50 sec,
  Horizontal Link inactivity time 120 sec,
  PTSE refresh interval 1800 sec, lifetime factor 200 percent,
  Min PTSE interval 10 tenths of sec
  Auto summarization: on, Supported PNNI versions: newest 1, oldest 1
  Default administrative weight mode: uniform
  Max admin weight percentage: -1
  Next resource poll in 3 seconds
  Max PTSEs requested per PTSE request packet: 32
  Redistributing static routes: Yes
```

Configuring a Parent Node

For a node to be eligible to become a PGL within its own peer group, you must configure a parent node and a nonzero election leadership level (described in the following section, "Configuring the Node Election Leadership Priority"). If the node is elected a PGL, the node specified by the **parent** command becomes the parent node and represents the peer group at the next hierarchical level.

To configure a parent node, perform these steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **atm router pnni** Switch(config-atm-router)#	Enters ATM router PNNI mode.
Step 2	Switch(config-atm-router)# **node** *node-index*	Enters node configuration mode.
Step 3	Switch(config-pnni-node)# **parent** *node-index*	Configures the parent node index.

Example

The following example shows how to create a parent node for node 1:

```
Switch(config)# atm router pnni
Switch(config-pnni-node)# node 1
Switch(config-pnni-node)# parent 2
```

Displaying the Parent Node Configuration

To display the parent node configuration, use the following privileged EXEC command:

Command	Purpose
show atm pnni hierarchy	Displays the PNNI hierarchy.

Example

The following example shows the ATM parent node information using the **show atm pnni hierarchy** privileged EXEC command:

```
Switch# show atm pnni hierarchy
Locally configured parent nodes:
  Node          Parent
  Index  Level  Index   Local-node Status     Node Name
  ~~~~~  ~~~~~  ~~~~~~  ~~~~~~~~~~~~~~~~~~~~  ~~~~~~~~~~~~~~~~~~~~~~
  1      80     2       Enabled/ Running      Switch
  2      72     N/A     Enabled/ Running      Switch.2.72
```

Configuring the Node Election Leadership Priority

Normally the node with the highest election leadership priority is elected PGL. If two nodes share the same election priority, the node with the highest node identifier becomes the PGL. To be eligible for election the configured priority must be greater than zero. You can configure multiple nodes in a peer group with nonzero leadership priority so that if one PGL becomes unreachable, the node configured with the next highest election leadership priority becomes the new PGL.

Note The choice of PGL does not directly affect the selection of routes across the peer group.

The control for election is done through the assignment of leadership priorities. We recommend that the leadership priority space be divided into three tiers:

- First tier: 1 to 49
- Second tier: 100 to 149
- Third tier: 200 to 205

This subdivision is used because when a node becomes PGL, it increases the advertised leadership priority by a value of 50. This avoids instabilities after election.

The following guidelines apply when configuring the node election leadership priority:

- Nodes that you do not want to become PGLs should remain with the default leadership priority value of 0.
- Unless you want to force one of the PGL candidates to be the PGL, you should assign all leadership priority values within the first tier. After a node is elected PGL, it remains PGL until it goes down or is configured to step down.
- If certain nodes should take precedence over nodes in the first tier, even if one is already PGL, leadership priority values can be assigned from the second tier. We recommend that you configure more than one node with a leadership priority value from this tier. This prevents one unstable node with a larger leadership priority value from repeatedly destabilizing the peer group.
- If you need a strict master leader, use the third tier.

Note The **election leadership-priority** command does not take effect unless a parent node has already been configured using the **node** and **parent** commands.

To configure the election leadership priority, perform these steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **atm router pnni** Switch(config-atm-router)#	Enters ATM router PNNI mode from the terminal.
Step 2	Switch(config-atm-router)# **node** *node-index* Switch(config-pnni-node)#	Enters node configuration mode.
Step 3	Switch(config-pnni-node)# **election leadership-priority** *number*	Configures the election leadership priority. The configurable range is from 0 to 205.

Example

The following example shows how to change the election leadership priority for node 1 to 100:

```
Switch(config)# atm router pnni
Switch(config-pnni-node)# node 1
Switch(config-pnni-node)# election leadership-priority 100
```

Displaying Node Election Leadership Priority

To display the node election leadership priority, use one of the following privileged EXEC commands:

Command	Purpose
show atm pnni election	Displays the node election leadership priority.
show atm pnni election peers	Displays all nodes in the peer group.

Examples

The following example shows the election leadership priority using the **show atm pnni election** privileged EXEC command:

```
Switch# show atm pnni election

PGL Status.............: PGL
Preferred PGL..........: (1) Switch
Preferred PGL Priority.: 255
Active PGL.............: (1) Switch
Active PGL Priority....: 255
Active PGL For.........: 00:01:07
Current FSM State......: PGLE Operating: PGL
Last FSM State.........: PGLE Awaiting Unanimity
Last FSM Event.........: Unanimous Vote

Configured Priority....: 205
Advertised Priority....: 255
Conf. Parent Node Index: 2
PGL Init Interval......: 15 secs
Search Peer Interval...: 75 secs
Re-election Interval...: 15 secs
Override Delay.........: 30 secs
```

The following example shows all nodes in the peer group using the **show atm pnni election peers** command:

```
Switch# show atm pnni election peers

  Node No.   Priority   Connected   Preferred PGL
  ~~~~~~~~   ~~~~~~~~   ~~~~~~~~~   ~~~~~~~~~~~~~
  1          255        Yes         Switch
  9          0          Yes         Switch
  10         0          Yes         Switch
  11         0          Yes         Switch
  12         0          Yes         Switch
```

Configuring a Summary Address

Summary addresses can be used to decrease the amount of information advertised by a PNNI node. Summary addresses should only be used when all end system addresses that match the summary address are directly reachable from this node. However, this is not always required because routes are always selected to nodes advertising the longest matching prefix to a destination address.

A single default summary address is configured for each logical group node (LGN) in the PNNI hierarchy. The length of that summary for any LGN equals the level of the child peer group, and its value is equal to the first level bits of the child peer group identifier. This address prefix is advertised into the LGN's peer group.

Summary addresses other than defaults must be explicitly configured on each node. A node can have multiple summary address prefixes. Note also that every node in a peer group that has a potential to become a peer group leader (PGL) should have the same summary address lists in its parent node configuration.

Note The **no auto-summary** command removes the default summary address(es). Use the **no auto-summary** command when systems that match the first 13-bytes of the ATM address(es) of your switch are attached to different switches.

To configure the ATM PNNI summary address prefix, perform these steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **atm router pnni** Switch(config-atm-router)#	Enters ATM router PNNI mode.
Step 2	Switch(config-atm-router)# **node** *node-index* Switch(config-pnni-node)#	Enters node configuration mode.
Step 3	Switch(config-pnni-node)# **no auto-summary**	Removes the default summary address(es).
Step 4	Switch(config-pnni-node)# **summary-address** *address-prefix*	Configures the ATM PNNI summary address prefix.

Example

The following example shows how to remove the default summary address(es) and add summary address 47.009181005670:

```
Switch(config)# atm router pnni
Switch(config-atm-router)# node 1
Switch(config-pnni-node)# no auto-summary
Switch(config-pnni-node)# summary-address 47.009181005670
```

Displaying the Summary Address Configuration

To display the ATM PNNI summary address configuration, use the following privileged EXEC command:

Command	Purpose
show atm pnni summary	Displays the ATM PNNI summary address configuration.

Example

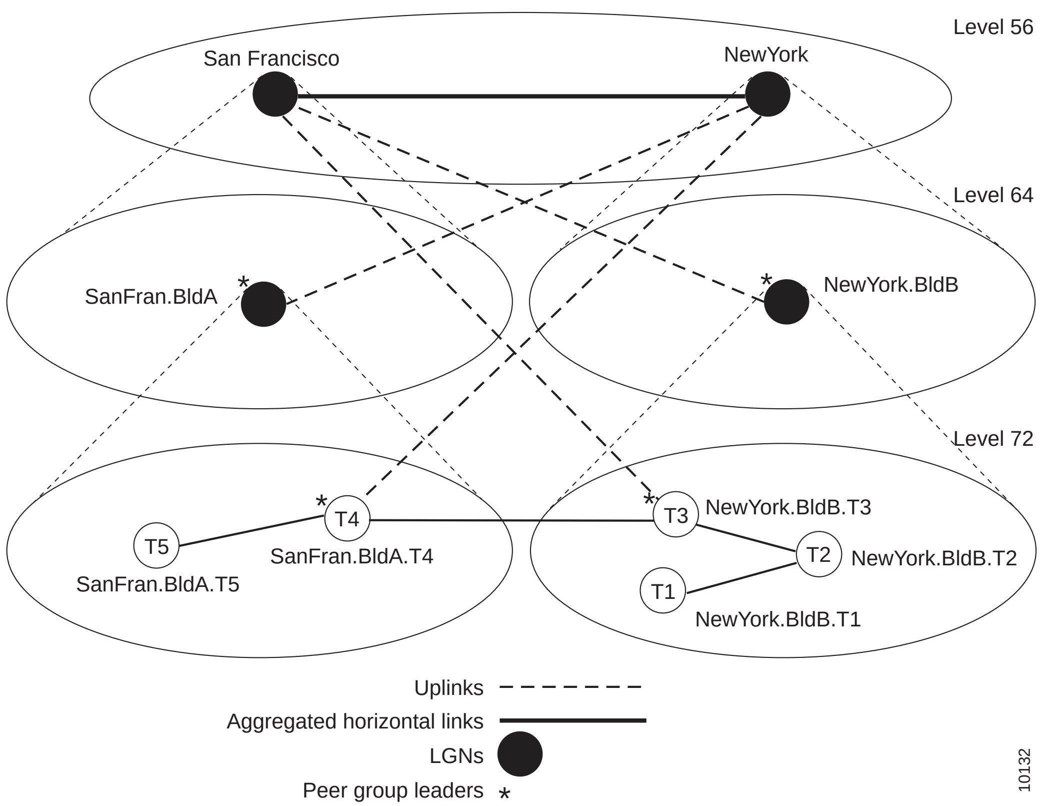
The following example shows the ATM PNNI summary address configuration using the **show atm pnni summary** privileged EXEC command:

```
Switch# show atm pnni summary

Codes: Node - Node index advertising this summary
       Type - Summary type (INT - internal, EXT - exterior)
       Sup  - Suppressed flag (Y - Yes, N - No)
       Auto - Auto Summary flag (Y - Yes, N - No)
       Adv  - Advertised flag (Y - Yes, N - No)

 Node Type Sup Auto Adv  Summary Prefix
 ~~~~ ~~~~ ~~~ ~~~~ ~~~  ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
  1   Int   N   Y    Y   47.0091.8100.0000.0040.0b0a.2a81/104
  2   Int   N   Y    N   47.01b1.0000.0000.0000.00/80
```

PNNI Hierarchy Configuration Example

An example configuration for a three-level hierarchical topology is shown in Figure 10-1. The example shows the configuration of only five switches, although there can be many other switches in each peer group.

Figure 10-1 Example Three-Level Hierarchical Topology



At the lowest level (level 72), the hierarchy represents two separate peer groups. Each of the four switches named T2 to T5 are eligible to become a peer group leader (PGL) at two levels, and each has two configured ancestor nodes (a parent node or a parent node’s parent). Switch T1 has no configured ancestor nodes and is not eligible to become a PGL. As a result of the peer group leader election at the lowest level, switches T4 and T3 become leaders of their peer groups. Therefore, each switch creates an LGN at the second level (level 64) of the hierarchy. As a result of the election at the second level of the hierarchy, logical group nodes (LGNs) SanFran.BldA and NewYork.BldB are elected as PGLs, creating LGNs at the highest level of the hierarchy (level 56). At that level, the uplinks that have been induced through level 64 form an aggregated horizontal link within the common peer group at level 56.

Examples

The sections that follow show the configurations for each switch and the outputs of the **show atm pnni local-node** command. Some of the output text has been suppressed because it is not relevant to the example.

Switch NewYork.BldB.T1 Configuration

```
hostname NewYork.BldB.T1
atm address 47.0091.4455.6677.1144.1011.1233.0060.3e7b.3a01.00
atm router pnni
  node 1 level 72 lowest
  redistribute atm-static

NewYork.BldB.T1# show atm pnni local-node

PNNI node 1 is enabled and running
  Node name: NewYork.BldB.T1
  System address          47.009144556677114410111233.00603E7B3A01.01
  Node ID          72:160:47.009144556677114410111233.00603E7B3A01.00
  Peer group ID        72:47.0091.4455.6677.1144.0000.0000
  Level 72, Priority 0 0, No. of interfaces 3, No. of neighbors 2
  Parent Node Index: NONE

<information deleted>
```

Switch NewYork.BldB.T2 Configuration

```
hostname NewYork.BldB.T2
atm address 47.0091.4455.6677.1144.1011.1244.0060.3e5b.bc01.00
atm router pnni
node 1 level 72 lowest
  parent 2
  redistribute atm-static
  election leadership-priority 40
 node 2 level 64
  parent 3
  election leadership-priority 40
  name NewYork.BldB
 node 3 level 56
  name NewYork

NewYork.BldB.T2# show atm pnni local-node

PNNI node 1 is enabled and running
  Node name: NewYork.BldB.T2
  System address          47.009144556677114410111244.00603E5BBC01.01
  Node ID          72:160:47.009144556677114410111244.00603E5BBC01.00
  Peer group ID        72:47.0091.4455.6677.1144.0000.0000
  Level 72, Priority 40 40, No. of interfaces 3, No. of neighbors 1
  Parent Node Index: 2

<information deleted>
```

```
PNNI node 2 is enabled and not running
  Node name: NewYork.BldB
  System address          47.00914455667711441011124 4.00603E5BBC01.02
  Node ID           64:72:47.00914455667711440000000 0.00603E5BBC01.00
  Peer group ID        64:47.0091.4455.6677.1100.0000.0000
  Level 64, Priority 40 40, No. of interfaces 0, No. of neighbors 0
  Parent Node Index: 3

<information deleted>

PNNI node 3 is enabled and not running
  Node name: NewYork
  System address          47.009144556677114410111244.00603E5BBC01.03
  Node ID           56:64:47.009144556677110000000000.00603E5BBC01.00
  Peer group ID        56:47.0091.4455.6677.0000.0000.0000
  Level 56, Priority 0 0, No. of interfaces 0, No. of neighbors 0
  Parent Node Index: NONE

<information deleted>
```

Switch NewYork.BldB.T3 Configuration

```
hostname NewYork.BldB.T3
atm address 47.0091.4455.6677.1144.1011.1255.0060.3e5b.c401.00
atm router pnni
 node 1 level 72 lowest
  parent 2
  redistribute atm-static
  election leadership-priority 45
 node 2 level 64
  parent 3
  election leadership-priority 45
  name NewYork.BldB
 node 3 level 56
  name NewYork

NewYork.BldB.T3# show atm pnni local-node

PNNI node 1 is enabled and running
  Node name: NewYork.BldB.T3
  System address          47.009144556677114410111255.00603E5BC401.01
  Node ID          72:160:47.009144556677114410111255.00603E5BC401.00
  Peer group ID        72:47.0091.4455.6677.1144.0000.0000
  Level 72, Priority 45 95, No. of interfaces 4, No. of neighbors 1
  Parent Node Index: 2

<information deleted>

PNNI node 2 is enabled and running
  Node name: NewYork.BldB
  System address          47.009144556677114410111255.00603E5BC401.02
  Node ID           64:72:47.009144556677114400000000.00603E5BC401.00
  Peer group ID        64:47.0091.4455.6677.1100.0000.0000
  Level 64, Priority 45 95, No. of interfaces 0, No. of neighbors 0
  Parent Node Index: 3

<information deleted>

PNNI node 3 is enabled and running
  Node name: NewYork
  System address          47.009144556677114410111255.00603E5BC401.03
  Node ID           56:64:47.009144556677110000000000.00603E5BC401.00
```

```
  Peer group ID        56:47.0091.4455.6677.0000.0000.0000
  Level 56, Priority 0 0, No. of interfaces 0, No. of neighbors 1
  Parent Node Index: NONE

<information deleted>
```

Switch SanFran.BldA.T4 Configuration

```
hostname SanFran.BldA.T4
atm address 47.0091.4455.6677.2233.1011.1266.0060.3e7b.2001.00
atm router pnni
 node 1 level 72 lowest
  parent 2
  redistribute atm-static
  election leadership-priority 45
 node 2 level 64
  parent 3
  election leadership-priority 45
  name SanFran.BldA
 node 3 level 56
  name SanFran

SanFran.BldA.T4# show atm pnni local-node

PNNI node 1 is enabled and running
  Node name: SanFran.BldA.T4
  System address          47.009144556677223310111266.00603E7B2001.01
  Node ID          72:160:47.009144556677223310111266.00603E7B2001.00
  Peer group ID        72:47.0091.4455.6677.2233.0000.0000
  Level 72, Priority 45 95, No. of interfaces 4, No. of neighbors 1
  Parent Node Index: 2

<information deleted>

PNNI node 2 is enabled and running
  Node name: SanFran.BldA
  System address          47.009144556677223310111266.00603E7B2001.02
  Node ID           64:72:47.009144556677223300000000.00603E7B2001.00
  Peer group ID        64:47.0091.4455.6677.2200.0000.0000
  Level 64, Priority 45 95, No. of interfaces 0, No. of neighbors 0
  Parent Node Index: 3
<information deleted>

PNNI node 3 is enabled and running
  Node name: SanFran
  System address          47.009144556677223310111266.00603E7B2001.03
  Node ID           56:64:47.009144556677220000000000.00603E7B2001.00
  Peer group ID        56:47.0091.4455.6677.0000.0000.0000
  Level 56, Priority 0 0, No. of interfaces 0, No. of neighbors 1
  Parent Node Index: NONE

<information deleted>
```

Switch SanFran.BldA.T5 Configuration

```
hostname SanFran.BldA.T5
atm address 47.0091.4455.6677.2233.1011.1244.0060.3e7b.2401.00
atm router pnni
 node 1 level 72 lowest
  parent 2
  redistribute atm-static
  election leadership-priority 10
 node 2 level 64
  parent 3
  election leadership-priority 40
  name SanFran.BldA
 node 3 level 56
  name SanFran

SanFran.BldA.T5# show atm pnni local-node

PNNI node 1 is enabled and running
  Node name: SanFran.BldA.T5
  System address          47.00914455667722331011124 4.00603E7B2401.01
  Node ID          72:160:47.009144556677223310111244.00603E7B2401.00
  Peer group ID        72:47.0091.4455.6677.2233.0000.0000
  Level 72, Priority 10 10, No. of interfaces 2, No. of neighbors 1
  Parent Node Index: 2

<information deleted>

PNNI node 2 is enabled and not running
  Node name: SanFran.BldA
  System address          47.009144556677223310111244.00603E7B2401.02
  Node ID           64:72:47.009144556677223300000000.00603E7B2401.00
  Peer group ID        64:47.0091.4455.6677.2200.0000.0000
  Level 64, Priority 40 40, No. of interfaces 0, No. of neighbors 0
  Parent Node Index: 3

<information deleted>

PNNI node 3 is enabled and not running
  Node name: SanFran
  System address          47.009144556677223310111244.00603E7B2401.03
  Node ID           56:64:47.009144556677220000000000.00603E7B2401.00
  Peer group ID        56:47.0091.4455.6677.0000.0000.0000
  Level 56, Priority 0 0, No. of interfaces 0, No. of neighbors 0
  Parent Node Index: NONE

<information deleted>
```

Advanced PNNI Configuration

This section describes how to configure advanced PNNI features. The advanced features described in this section are not required to enable PNNI, but are provided to tune your network performance.

For additional information about the features described in this section, refer to the *Guide to ATM Technology*.

This section includes the following subsections:

- Tuning Route Selection, page 10-29
- Tuning Topology Attributes, page 10-39

Tuning Route Selection

The tasks described in the following subsections are used to tune the mechanisms by which routes are selected in your PNNI network.

Configuring Background Route Computation

The ATM switch router supports the following two route selection modes:

- On-demand—A separate route computation is performed each time a SETUP or ADD PARTY message is received over a User-Network Interface (UNI) or Interim Interswitch Signaling Protocol (IISP) interface. In this mode, the most recent topology information received by this node is always used for each setup request.
- Background routes—Call setups are routed using precomputed routing trees. In this mode, multiple background trees are precomputed for several service categories and quality of service (QoS) metrics. If no route can be found in the multiple background trees that satisfies the QoS requirements of a particular call, route selection reverts to on-demand route computation.

The background routes mode should be enabled in large networks where it usually exhibits less stringent processing requirements and better scalability. Route computation is performed at almost every poll interval when a significant change in the topology of the network is reported or when significant threshold changes have occurred since the last route computation.

To configure the background route computation, perform these steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **atm router pnni** Switch(config-atm-router)#	Enters ATM router PNNI mode.
Step 2	Switch(config-atm-router)# **background-routes-enable** [**insignificant-threshold number**] [**poll-interval** *seconds*]	Enables background routes and configures background route parameters.

Example

The following example shows how to enable background routes and configures the background routes poll interval to 30 seconds:

```
Switch(config)# atm router pnni
Switch(config-atm-router)# background-routes-enable poll-interval 30
```

Displaying the Background Route Computation Configuration

To display the background route configuration, use the following privileged EXEC commands:

Command	Purpose
show atm pnni background status	Displays the background route configuration.
show atm pnni background routes	Displays background routing tables.

Examples

The following example shows the ATM PNNI background route configuration using the **show atm pnni background status** privileged EXEC command:

```
Switch# show atm pnni background status

Background Route Computation is Enabled
Background Interval is set at 10 seconds
Background Insignificant Threshold is set at 32
```

The following example shows the ATM PNNI background route tables for constant bit rate (CBR) using the **show atm pnni background routes** privileged EXEC command:

```
Switch# show atm pnni background routes cbr
  Background Routes From CBR/AW Table
  ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
  2 Routes To Node 2
      1. Hops 1. 1:ATM0/1/2 -> 2
          ->: aw 5040  cdv 138  ctd 154 acr 147743 clr0 10 clr01 10
          <-: aw 5040  cdv 138  ctd 154 acr 147743 clr0 10 clr01 10
      2. Hops 1. 1:ATM0/1/1 -> 2
          ->: aw 5040  cdv 138  ctd 154 acr 147743 clr0 10 clr01 10
          <-: aw 5040  cdv 138  ctd 154 acr 147743 clr0 10 clr01 10

  1 Routes To Node 5
      1. Hops 1. 1:ATM0/1/0 -> 5
          ->: aw 5040  cdv 138  ctd 154 acr 147743 clr0 10 clr01 10
          <-: aw 5040  cdv 138  ctd 154 acr 147743 clr0 10 clr01 10

  Background Routes From CBR/CDV Table
  ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
  2 Routes To Node 2
      1. Hops 1. 1:ATM0/1/2 -> 2
          ->: aw 5040  cdv 138  ctd 154 acr 147743 clr0 10 clr01 10
          <-: aw 5040  cdv 138  ctd 154 acr 147743 clr0 10 clr01 10
      2. Hops 1. 1:ATM0/1/1 -> 2
          ->: aw 5040  cdv 138  ctd 154 acr 147743 clr0 10 clr01 10
          <-: aw 5040  cdv 138  ctd 154 acr 147743 clr0 10 clr01 10

  1 Routes To Node 5
      1. Hops 1. 1:ATM0/1/0 -> 5
          ->: aw 5040  cdv 138  ctd 154 acr 147743 clr0 10 clr01 10
          <-: aw 5040  cdv 138  ctd 154 acr 147743 clr0 10 clr01 10

  Background Routes From CBR/CTD Table
  ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
2 Routes To Node 2
      1. Hops 1. 1:ATM0/1/2 -> 2
          ->: aw 5040  cdv 138  ctd 154 acr 147743 clr0 10 clr01 10
          <-: aw 5040  cdv 138  ctd 154 acr 147743 clr0 10 clr01 10
      2. Hops 1. 1:ATM0/1/1 -> 2
```

```
        ->: aw 5040  cdv 138  ctd 154 acr 147743 clr0 10 clr01 10
        <-: aw 5040  cdv 138  ctd 154 acr 147743 clr0 10 clr01 10

1 Routes To Node 5
    1. Hops 1. 1:ATM0/1/0 -> 5
        ->: aw 5040  cdv 138  ctd 154 acr 147743 clr0 10 clr01 10
        <-: aw 5040  cdv 138  ctd 154 acr 147743 clr0 10 clr01 10

Background Routes From CBR/CTD Table
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
2 Routes To Node 2
    1. Hops 1. 1:ATM0/1/2 -> 2
        ->: aw 5040  cdv 138  ctd 154 acr 147743 clr0 10 clr01 10
        <-: aw 5040  cdv 138  ctd 154 acr 147743 clr0 10 clr01 10
    2. Hops 1. 1:ATM0/1/1 -> 2
        ->: aw 5040  cdv 138  ctd 154 acr 147743 clr0 10 clr01 10
        <-: aw 5040  cdv 138  ctd 154 acr 147743 clr0 10 clr01 10

1 Routes To Node 5
    1. Hops 1. 1:ATM0/1/0 -> 5
        ->: aw 5040  cdv 138  ctd 154 acr 147743 clr0 10 clr01 10
        <-: aw 5040  cdv 138  ctd 154 acr 147743 clr0 10 clr01 10
```

Configuring Link Selection

Link selection applies to parallel PNNI links between two switches. Link selection allows you to choose the method the switch uses during call setup for selecting one link among multiple parallel links to forward the call.

Note Calls always use the load balance method over parallel IISP links between two switches.

Table 10-2 lists the PNNI link selection methods from which you can choose.

Table 10-2 PNNI Link Selection Methods

Precedence Order	Method	Description	Service Category Availability
1	admin-weight-minimize	Places the call on the link with the lowest administrative weight.	CBR[1], VBR-RT[2], VBR-NRT[3]
2	blocking-minimize	Places the call on the link so that higher bandwidth is available for subsequent calls, thus minimizing call blocking.	CBR, VBR-RT, VBR-NRT
3	transmit-speed-maximize	Places the call on the highest speed link.	CBR, VBR-RT, VBR-NRT
4	load-balance	Places the call on the link so that the load is balanced among parallel links for a group.	ABR[4], UBR[5]

1. CBR = constant bit rate
2. VBR-RT = variable bit rate real time
3. VBR-NRT = variable bit rate non-real time
4. ABR = available bit rate
5. UBR = unspecified bit rate

The switch applies a single link selection method for a group of parallel links connected to a neighbor switch. If multiple links within this group are configured with a different link selection method, then the switch selects a method according to the order of precedence as shown in Table 10-2.

The link selection feature allows you to specify one or more links among the parallel links as an alternate (or backup) link. An alternate link is a link that is used only when all other non-alternate links are either down or full. Alternate links are not considered part of the parallel link group targeted for link selection. Calls are always load balanced over multiple parallel alternate links by default.

To configure the PNNI link selection feature, perform these steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **interface atm** *card*/*subcard*/*port* Switch(config-if)#	Specifies an ATM interface and enter interface configuration mode.
Step 2	Switch(config-if)# **atm pnni link-selection** {**cbr** | **vbr-rt** | **vbr-nrt** | **abr** | **ubr** | **all**} {**admin-weight-minimize** | **alternate** | **blocking-minimize** | **load-balance** | **transmit-speed-maximize**}	Configures ATM PNNI link selection for a specific link.

Examples

The following example shows how to configure link selection on ATM interface 0/0/0 with a VBR-NRT service category and transmit-speed-maximize mode:

```
Switch(config)# interface atm 0/0/0
Switch(config-if)# atm pnni link-selection vbr-nrt transmit-speed-maximize
```

The following example shows how to configure link selection on ATM interface 0/0/0 with a CBR service category and then designate the link as an alternate:

```
Switch(config)# interface atm 0/0/0
Switch(config-if)# atm pnni link-selection cbr alternate
```

Displaying the Link Selection Configuration

To display the ATM PNNI link selection configuration, use the following EXEC command:

Command	Purpose
show atm pnni neighbor	Displays the ATM PNNI link selection configuration.

Example

The following example shows the detailed PNNI link selection configuration using the **show atm pnni neighbor** EXEC command:

```
Switch# show atm pnni neighbor

Neighbors For Node (Index 1, Level 56)

  Neighbor Name: XXXXXX, Node number: 9
  Neighbor Node Id: 56:160:47.00918100000000E04FACB401.00E04FACB401.00
  Neighboring Peer State: Full
  Link Selection For CBR    : minimize blocking of future calls
  Link Selection For VBR-RT : minimize blocking of future calls
  Link Selection For VBR-NRT: minimize blocking of future calls
  Link Selection For ABR    : balance load
  Link Selection For UBR    : balance load
   Port                      Remote Port Id            Hello state
   ATM4/0/0                  ATM3/1/1                  2way_in  (Flood Port)
Switch#
```

Configuring the Maximum Administrative Weight Percentage

The maximum administrative weight percentage feature, a generalized form of a hop count limit, allows you to prevent the use of alternate routes that consume too many network resources. The maximum acceptable administrative weight is equal to the specified percentage of the least administrative weight of any route to the destination (from the background routing tables).

To configure the maximum AW percentage, perform these steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **atm router pnni** Switch(config-atm-router)#	Enters ATM router PNNI mode.
Step 2	Switch(config-atm-router)# **max-admin-weight-percentage** *percent*	Configures the maximum AW percentage. The value can range from 100 to 2000.

Note The **max-admin-weight-percentage** command only takes effect if background route computation is enabled. See the "Configuring Background Route Computation" section on page 10-29.

Example

The following example shows how to configure the node maximum AW percentage value as 300:

```
Switch(config)# atm router pnni
Switch(config-atm-router)# max-admin-weight-percentage 300
```

Displaying the Maximum Administrative Weight Percentage Configuration

To display the node ATM PNNI maximum AW percentage configuration, use the following privileged EXEC command:

Command	Purpose
show atm pnni local-node	Displays the node ATM PNNI maximum AW configuration.

Example

The following example shows the maximum AW percentage configuration using the **show atm pnni local-node** privileged EXEC command:

```
Switch# show atm pnni local-node
PNNI node 1 is enabled and running
  Node name: eng_1
  System address 47.00918100000000000001212.121212121212.00
  Node ID 56:160:47.00918100000000000001212.121212121212.00
  Peer group ID 56:47.0091.8100.0000.0000.0000.0000
  Level 56, Priority 0, No. of interface 4, No. of neighbor 1

  Hello interval 15 sec, inactivity factor 5, Hello hold-down 10 tenths of sec
  Ack-delay 2 sec, retransmit interval 10 sec, rm-poll interval 10 sec
  PTSE refresh interval 90 sec, lifetime factor 7, minPTSEinterval 1000 msec
  Auto summarization: on, Supported PNNI versions: newest 1, oldest 1
  Default administrative weight mode: linespeed
➔ Max admin weight percentage: 300
  Next RM poll in 3 seconds
```

Configuring the Precedence

The route selection algorithm chooses routes to particular destinations using the longest match reachable address prefixes known to the switch. When there are multiple longest match reachable address prefixes known to the switch, the route selection algorithm first attempts to find routes to reachable addresses with types of greatest precedence. Among multiple longest match reachable address prefixes of the same type, routes with the least total administrative weight are chosen first.

Local internal reachable addresses, whether learned via Integrated Local Management Interface (ILMI) or as static routes, are given highest precedence or a precedence value of one. The precedence of other reachable address types is configurable.

To configure the precedence of reachable addresses, perform these steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **atm router pnni** Switch(config-atm-router)#	Enters ATM router PNNI mode.
Step 2	Switch(config-atm-router)# **precedence** [**pnni-remote-exterior** *value* | **pnni-remote-exterior-metrics** *value* | **pnni-remote-internal** *value* | **pnni-remote-internal-metrics** *value* | **static-local-exterior** *value* | **static-local-exterior-metrics** *value* | **static-local-internal-metrics** *value*]	Enters PNNI precedence and configure the PNNI node.

Example

The following example shows how to configure all PNNI remote exterior routes with a precedence value of 4:

```
Switch(config)# atm router pnni
Switch(config-atm-router)# precedence pnni-remote-exterior 4
```

Displaying Precedence Configuration

To display the ATM PNNI route determination precedence configuration, use the following privileged EXEC command:

Command	Purpose
show atm pnni precedence	Displays the node ATM PNNI route determination precedence configuration.

Example

The following example shows the ATM PNNI route determination precedence configuration using the **show atm pnni precedence** privileged EXEC command:

```
Switch# show atm pnni precedence
                                 Working   Default
  Prefix Poa Type                Priority  Priority
  ----------------------------   --------  --------
  local-internal                    1         1
  static-local-internal-metrics     2         2
  static-local-exterior             3         3
  static-local-exterior-metrics     2         2
  pnni-remote-internal              2         2
  pnni-remote-internal-metrics      2         2
➔ pnni-remote-exterior              4         4
  pnni-remote-exterior-metrics      2         2
```

Configuring Explicit Paths

The explicit path feature enables you to manually configure either a fully specified or partially specified path for routing soft permanent virtual channels (soft PVC) and soft permanent virtual path (soft PVP) connections. Once these routes are configured, up to three explicit paths might be applied to these connections.

A fully specified path includes all adjacent nodes (and optionally the corresponding exit port) for all segments of the path. A partially specified path consists of one or more segment target nodes that should appear in their proper order in the explicit path. The standard routing algorithm is used to determine all unspecified parts of the partially specified path.

You can specify a path name for an explicit path and the switch assigns the next available unused *path-id* value, or you can choose the *path-id* value and assign or modify its name.

To enter the PNNI explicit path configuration mode, use the following global configuration command:

Command	Purpose
atm pnni explicit-path {**identifier** *path-id-number* [**name** *path-name*] | **name** *path-name*} [**enable** | **disable**]	Enters the PNNI explicit path configuration mode.

The **disable** option can be used to prevent an explicit path from being used for routing while it is being configured, if any soft connections already reference it. If the explicit path has not been created, the initial default is to enable the explicit path upon configuration.

Example

The following example shows how to enter the PNNI explicit path configuration mode for a path named boston_2.path1:

```
Switch(config)# atm pnni explicit-path name boston_2.path1
Switch(cfg-pnni-expl-path)#
```

Adding Entries to the Explicit Path

Once in PNNI explicit path configuration mode, you can use the following subcommands repeatedly to build up the ordered list that specifies the explicit path:

Command	Purpose
next-node {*name-string* | *node-id* | *node-id-prefix*} [**port** *hex-port-id* | **agg-token** *hex-agg-token-id*]	The **next-node** keyword specifies the next adjacent node for fully specified paths. Add next PNNI explicit path entry with this command.
segment-target {*name-string* | *node-id* | *node-id-prefix*} [**port** *hex-port-id* | **agg-token** *hex-agg-token-id*]	The **segment-target** keyword specifies the target node for cases where the path through intermediate nodes should be automatically routed.
exclude-node {*name-string* | *node-id* | *node-id-prefix*} [**port** *hex-port-id* | **agg-token** *hex-agg-token-id*]	The **exclude-node** keyword specifies nodes or ports that are excluded from all partial path segments.

Node IDs can be entered either with the full 22-byte length address or as a Node ID prefix with a length of 15 or more bytes. To specify routes that include higher level nodes (parent LGNs) for other peer groups, we recommend that you enter exactly 15 bytes so that the address remains valid in the event of a PGL update.

Node IDs appear in the following format:

dec : *dec* : *13-20 hex digits*

Node names can be entered instead of Node IDs. If names are used to identify higher level LGNs, the resulting explicit paths are not guaranteed to remain valid if the PGL changes in the neighboring peer group. To prevent invalid paths, configure all parent LGNs (for all potential PGL nodes) with the same node name.

Optionally, an exit port can be specified for any entry. The port should be specified as a *hex-port-id* rather than a *port-name*. For excluded entries, only this port is excluded from the path.

Since the port ID could change if the following neighbor peer group changes PGL leaders, the aggregation token is used in place of the port ID for nodes with higher level LGNs. The LGN aggregation token can only identify the port uniquely if the following entry is a next-node entry. Aggregation tokens are not allowed for excluded nodes.

Example

The following example shows how to configure an explicit path list consisting of four entries. The first two are adjacent nodes and, in one case, an exit port is specified. Next, a partially-specified segment to the node chicago_2 is configured, several hops away. Finally, a higher level LGN node adjacent to chicago_2 is configured, which is specified by its 15-byte Node ID prefix.

```
Switch(cfg-pnni-expl-path)# next-node dallas_2
Switch(cfg-pnni-expl-path)# next-node dallas_4 port 80003004
Switch(cfg-pnni-expl-path)# segment-target chicago_2
Switch(cfg-pnni-expl-path)# next-node 40:72:47.00918100000106000000000
```

Displaying Node IDs

To display the node IDs that correspond to named nodes in a network, use either of the following EXEC commands:

Command	Purpose
show atm pnni identifier	Displays the node IDs.
show atm pnni topology node *name-or-number*	Displays the node IDs.

Displaying Hex-Port-IDs

Since the explicit path subcommands require a *hex-port-id* rather than a *port name*, use either of the following EXEC commands to display the corresponding *hex-port-ids* for a node:

Command	Purpose
show atm pnni identifiers *node-number* **port**	Displays hex-port-ids for a node.
show atm pnni topology node *node-number* **hex-port-id**	Displays hex-port-ids for a node.

Editing Entries within the Explicit Path

Each entry has an index that gives its relative position within the list. Indices are used as an aid to edit an explicit path. The entire current list showing the entry index displays after each entry is added, or it is redisplayed when you use the **list** keyword.

The optional **index** keyword allows the exact index to be specified for an entry. If no index is specified for a new entry, it always defaults to one higher than the last path entry. If the index matches the index of an existing entry, the index is overwritten with new information. The **no** form deletes an existing entry for a given index.

Example

The following example shows the original path:

```
Explicit_path name new_york.path1 (id 5) from node dallas_1:
1 next-node dallas_2
2 next-node dallas_4 port 80003004
3 segment   chicago_2
4 next-node 40:72:47.00918100000010600000000.
```

You can modify the first entry to add an exit port for the original path. As shown in the following example, use the **index** keyword to specify the index of the entry to modify:

```
dallas_1 (cfg-pnni-expl-path)# index 1 next-node dallas_2 port 80000000
Explicit_path name new_york.path1 (id 5) from node dallas_1:
1 next-node dallas_2 port 80000000
2 next-node dallas_4 port 80003004
3 segment   chicago_2
4 next-node 40:72:47.00918100000010600000000.
```

The **append-after** keyword adds a path entry after the specified index. Renumbering the following path entries, if necessary, to make room for the new entry.

Example

If there are four **next-node** entries labelled as index 1 through 4, you can squeeze a new entry in after index 2 (using the **append-after** keyword), resulting in index 3. The following two entries are automatically renumbered to indexes 4 and 5 in order to make room for index 3.

```
dallas_1(cfg-pnni-expl-path)# append 2 next-node st_louis
Explicit_path name new_york.path1 (id 5) from node dallas_1:
1 next-node dallas_2 port 80000000
2 next-node dallas_4 port 80003004
3 next-node st_louis
4 segment   chicago_2
5 next-node 40:72:47.00918100000010600000000.
```

Displaying Explicit Path Configuration

To display the PNNI explicit path configuration, use the following EXEC command:

Command	Purpose
s**how atm pnni explicit-path** [{**name** *path-name* | **identifier** *path-id*} [**upto** *index*]] [**detail**]	Displays the PNNI explicit path configuration.

Example

The following example shows a summary of explicit paths:

```
Switch# show atm pnni explicit-paths
Summary of configured Explicit Paths:
 PathId Status       UpTo  Routable AdminWt Explicit Path Name
 ~~~~~~ ~~~~~~~~~~~  ~~~~~ ~~~~~~~~ ~~~~~~~ ~~~~~~~~~~~~~~~~~~~~
 1      enabled      3     yes      10040   dallas_4.path1
 2      enabled      6     yes      15120   chicago_2.path1
 3      enabled      2     yes      10080   chicago_2.path2
 4      enabled      2     yes      20595   new_york.path1
```

The following example shows the detailed configuration including any known warnings and error messages for a non-routable explicit path named new_york.path2:

```
Switch# show atm pnni explicit-paths name new_york.path2 detail
 PathId Status       UpTo  Routable AdminWt Explicit Path Name
 ~~~~~~ ~~~~~~~~~~~  ~~~~~ ~~~~~~~~ ~~~~~~~ ~~~~~~~~~~~~~~~~~~~~
 1      enabled      4     no       0       new_york.path2
  PNNI routing err_code for UBR call = 6 (PNNI_DEST_UNREACHABLE)

  Entry Type      Node [Port] specifier
  ~~~~~ ~~~~~~~~~ ~~~~~~~~~~~~~~~~~~~~~~
  1     next-node dallas_2
  2     next-node dallas_4 port 80000004
                  Warning:Entry index 2 specifies a non-routable port
  3     next-node wash_dc_1
                  Warning:Entry index 3 has no connectivity from prior node
  4     segment   new_york.2.40
```

Note The **upto** keyword can be used for troubleshooting explicit paths that are shown as non-routable. Routable status is only calculated up to the specified path entry index which allows the first failing path entry to be isolated.

Tuning Topology Attributes

The tasks in the following subsections describe how to configure attributes that affect the network topology.

Configuring the Global Administrative Weight Mode

Administrative weight is the primary routing metric for minimizing use of network resources. You can configure the administrative weight to indicate the relative desirability of using a link. For example, assigning equal administrative weight to all links in the network minimizes the number of hops used by each connection.

To configure the administrative weight mode, perform these steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **atm router pnni** Switch(config-atm-router)#	Enters ATM router PNNI mode.
Step 2	Switch(config-atm-router)# **administrative-weight** {**linespeed** | **uniform**}	Configures the administrative weight for all node connections.

Example

The following example shows how to configure the administrative weight for the node as line speed:

```
Switch(config)# atm router pnni
Switch(config-atm-router)# administrative-weight linespeed
```

Displaying the Administrative Weight Mode Configuration

To display the administrative weight configuration, use the following privileged EXEC command:

Command	Purpose
show atm pnni local-node	Displays the AW configuration for the node.

Example

The following example shows the AW configuration for the node using the **show atm pnni local-node** privileged EXEC command:

```
Switch# show atm pnni local-node
PNNI node 1 is enabled and running
  Node name: switch
  System address 47.00918100000000000001212.121212121212.00
  Node ID 56:160:47.00918100000000000001212.121212121212.00
  Peer group ID 56:47.0091.8100.0000.0000.0000.0000
  Level 56, Priority 0, No. of interface 4, No. of neighbor 1

  Hello interval 15 sec, inactivity factor 5, Hello hold-down 10 tenths of sec
  Ack-delay 2 sec, retransmit interval 10 sec, rm-poll interval 10 sec
  PTSE refresh interval 90 sec, lifetime factor 7, minPTSEinterval 1000 msec
  Auto summarization: on, Supported PNNI versions: newest 1, oldest 1
  Default administrative weight mode: linespeed
  Max admin weight percentage: 300
  Next RM poll in 3 seconds
```

Configuring Administrative Weight Per Interface

In addition to the global administrative weight (AW), you can also configure the administrative weight for an interface. To configure the administrative weight on an interface, perform these steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **interface atm** *card*/*subcard*/*port* Switch(config-if)#	Specifies an ATM interface and enters interface configuration mode.
Step 2	Switch(config-if)# **atm pnni admin-weight** *number service-category*	Configures the ATM AW for this link.

Example

The following example shows how to configure ATM interface 0/0/0 with ATM PNNI AW of 7560 for traffic class ABR:

```
Switch(config)# interface atm 0/0/0
Switch(config-if)# atm pnni admin-weight 7560 abr
```

Displaying the Administrative Weight Per Interface Configuration

To display the ATM PNNI interface AW configuration, use the following EXEC command:

Command	Purpose
show atm pnni [**interface atm** *card*/*subcard*/*port*] [**detail**]	Displays the interface ATM PNNI AW configuration.

Example

The following example shows the AW configuration for interface 0/0/0 using the **show atm pnni interface** EXEC command:

```
Switch# show atm pnni interface atm 0/0/0 detail

Port ATM0/0/0 is up  , Hello state 2way_in with node eng_18
    Next hello occurs in 11 seconds, Dead timer fires in 73 seconds
    CBR    : AW 5040 MCR 155519 ACR 147743 CTD 154 CDV 138 CLR0 10 CLR01 10
    VBR-RT : AW 5040 MCR 155519 ACR 155519 CTD 707 CDV 691 CLR0 8 CLR01 8
    VBR-NRT: AW 5040 MCR 155519 ACR 155519 CLR0 8 CLR01 8
    ABR    : AW 5040 MCR 155519 ACR 0
    UBR    : AW 5040 MCR 155519
    Remote node ID 56:160:47.00918100000000613E7B2F01.00613E7B2F99.00
    Remote node address 47.00918100000000613E7B2F01.00613E7B2F99.00
    Remote port ID  ATM0/1/2 (80102000) (0)
```

Configuring Transit Restriction

Transit calls originate from another ATM switch and pass through the switch. Some edge switches might want to eliminate this transit traffic and only allow traffic originating or terminating at the switch.

To configure a transit restriction, perform these steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **atm router pnni** Switch(config-atm-router)#	Enters ATM router PNNI mode.
Step 2	Switch(config-atm-router)# **node** *node-index* Switch(config-pnni-node)#	Enters node configuration mode.
Step 3	Switch(config-pnni-node)# **transit-restricted**	Enables transit restricted on this node.

Example

The following example shows how to enable the transit-restricted feature:

```
Switch(config)# atm router pnni
Switch(config-atm-router)# node 1
Switch(config-pnni-node)# transit-restricted
```

Displaying the Transit Restriction Configuration

To display the ATM PNNI transit-restriction configuration, use the following privileged EXEC command:

Command	Purpose
show atm pnni local-node	Displays the ATM configuration.

Example

The following example shows the ATM PNNI transit-restriction configuration using the **show atm pnni local-node** privileged EXEC command:

```
Switch# show atm pnni local-node
  PNNI node 1 is enabled and running
    Node name: Switch
    System address 47.00918100000000400B0A3081.00400B0A3081.00
    Node ID 56:160:47.00918100000000400B0A3081.00400B0A3081.00
    Peer group ID 56:47.0091.8100.0000.0000.0000.0000
    Level 56, Priority 0, No. of interfaces 4, No. of neighbors 2
➔   Node Does Not Allow Transit Calls

    Hello interval 15 sec, inactivity factor 5,
    Hello hold-down 10 tenths of sec
    Ack-delay 10 tenths of sec, retransmit interval 5 sec,
    Resource poll interval 5 sec
    PTSE refresh interval 1800 sec, lifetime factor 200 percent,
    Min PTSE interval 10 tenths of sec
    Auto summarization: on, Supported PNNI versions: newest 1, oldest 1
    Default administrative weight mode: uniform
    Max admin weight percentage: -1
    Next resource poll in 3 seconds
    Max PTSEs requested per PTSE request packet: 32
    Redistributing static routes: Yes
```

Configuring Redistribution

Redistribution instructs PNNI to distribute reachability information from non-PNNI sources throughout the PNNI routing domain. The ATM switch router supports redistribution of static routes, such as those configured on Interim Interswitch Signaling Protocol (IISP) interfaces.

Note By default, redistribution of static routes is enabled.

To enable redistribution of static routes, perform these steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **atm router pnni** Switch(config-atm-router)#	Enters ATM router PNNI mode.
Step 2	Switch(config-atm-router)# **node** *node-index* Switch(config-pnni-node)#	Enters node configuration mode.
Step 3	Switch(config-pnni-node)# **redistribute atm-static**	Enables redistribution of static routes.

Example

The following example shows how to enable redistribution of static routes:

```
Switch(config)# atm router pnni
Switch(config-atm-router)# node 1
Switch(config-pnni-node)# redistribute atm-static
```

Displaying the Redistribution Configuration

To display the node redistribution configuration, use the following privileged EXEC command:

Command	Purpose
show atm pnni local-node	Displays the node redistribution configuration.

Example

The following example shows the node redistribution configuration using the **show atm pnni local-node** privileged EXEC command:

```
Switch# show atm pnni local-node
  PNNI node 1 is enabled and running
    Node name: Switch
    System address 47.00918100000000400B0A3081.00400B0A3081.00
    Node ID 56:160:47.00918100000000400B0A3081.00400B0A3081.00
    Peer group ID 56:47.0091.8100.0000.0000.0000.0000
    Level 56, Priority 0, No. of interfaces 4, No. of neighbors 2
    Node Allows Transit Calls

    Hello interval 15 sec, inactivity factor 5,
    Hello hold-down 10 tenths of sec
    Ack-delay 10 tenths of sec, retransmit interval 5 sec,
    Resource poll interval 5 sec
    PTSE refresh interval 1800 sec, lifetime factor 200 percent,
    Min PTSE interval 10 tenths of sec
    Auto summarization: on, Supported PNNI versions: newest 1, oldest 1
    Default administrative weight mode: uniform
    Max admin weight percentage: -1
    Next resource poll in 3 seconds
    Max PTSEs requested per PTSE request packet: 32
→   Redistributing static routes: Yes
```

Configuring Aggregation Token

The aggregation token controls the grouping of multiple physical links into logical links. Uplinks to the same higher level node, or upnode, with the same aggregation token value, are represented at a higher level as horizontal aggregated links. Resource Availability Information Groups (RAIGs) are computed according to the aggregation algorithm.

To specify an aggregation token value, perform these steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **interface atm** *card*/*subcard*/*port* Switch(config-if)#	Specifies the ATM interface.
Step 2	Switch(config-if)# **atm pnni aggregation-token** *value*	Enters a value for the aggregation-token on the ATM interface.

Example

The following example shows how to configure an aggregation token on ATM interface 1/0/1:

```
Switch(config)# interface atm 1/0/1
Switch(config-if)# atm pnni aggregation-token 100
```

Displaying the Aggregation Token Configuration

To display the aggregation token configuration, use the following EXEC command:

Command	Purpose
show atm pnni interface atm *card*/*subcard*/*port* [**detail**]	Displays the interface PNNI configuration.

Examples

The following example shows the aggregation token value for all interfaces using the **show atm pnni interface** EXEC command:

```
NewYork.BldB.T3# show atm pnni interface

PNNI Interface(s) for local-node 1 (level=56):
  Local Port    Type  RCC Hello St Deriv Agg  Remote Port   Rem Node(No./Name)
  ~~~~~~~~~~~~~ ~~~~~ ~~~ ~~~~~~~~ ~~~~~~~~~~ ~~~~~~~~~~~~~ ~~~~~~~~~~~~~~~~~~
  ATM0/0/2      Phy   UP  comm_out 2          ATM0/0/3      - SanFran.BldA.T4
  ATM0/1/2      Phy   DN  down     35
  ATM0/1/3      Phy   UP  2way_in  0          ATM1/1/3      10 NewYork.BldB.T1
NewYork.BldB.T3#
```

The following example shows the aggregation token value details for a specific interface using the **show atm pnni interface** EXEC command with the **detail** keyword:

```
NewYork.BldB.T3# show atm pnni interface atm 0/0/2 detail

PNNI Interface(s) for local-node 1 (level=56):

Port ATM0/0/2 RCC is up  , Hello state common_out with node SanFran.BldA.T4
  Next hello occurs in 4 seconds, Dead timer fires in 72 seconds
  CBR    : AW 5040 MCR 155519 ACR 147743 CTD 154 CDV 138 CLR0 10 CLR01 10
  VBR-RT : AW 5040 MCR 155519 ACR 155519 CTD 707 CDV 691 CLR0 8 CLR01 8
  VBR-NRT: AW 5040 MCR 155519 ACR 155519 CLR0 8 CLR01 8
  ABR    : AW 5040 MCR 155519 ACR 0
  UBR    : AW 5040 MCR 155519
  Aggregation Token: configured 0 , derived 2, remote 2
  Tx ULIA seq# 1, Rx ULIA seq# 1, Tx NHL seq# 1, Rx NHL seq# 2
  Remote node ID        72:160:47.00914455667722331011266.00603E7B2001.00
  Remote node address          47.00914455667722331011266.00603E7B2001.01
  Remote port ID         ATM0/0/3 (80003000) (0)
  Common peer group ID      56:47.0091.4455.6677.0000.0000.0000
  Upnode ID              56:72:47.00914455667722330000000.00603E7B2001.00
  Upnode Address               47.00914455667722331011266.00603E7B2001.02
  Upnode number: 11       Upnode Name: SanFran
NewYork.BldB.T3#
```

Configuring Aggregation Mode

You configure the aggregation mode for calculating metrics and attributes for aggregated PNNI links and nodes advertised to higher PNNI levels. The ATM switch router has two algorithms to perform link and node aggregation: best link and aggressive.

To configure link or node aggregation, perform the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **atm router pnni** Switch(config-atm-router)#	Enters ATM router PNNI mode.
Step 2	Switch(config-atm-router)# **node** *node-index* Switch(config-pnni-node)#	Enters node configuration mode and specify the local node you want to configure.
Step 3	Switch(config-pnni-node)# **aggregation-mode** {**link** | **node**} {**abr** | **cbr** | **ubr** | **vbr-rt** | **vbr-nrt** | **all**} {**best-link** | **aggressive**}	Configures the service category and aggregation mode for a link or a complex node.

Examples

The following example shows how to configure aggressive link aggregation mode for constant bit rate (CBR) traffic:

```
Switch(config)# atm router pnni
Switch(config-pnni-node)# node 2
Switch(config-pnni-node)# aggregation-mode link cbr aggressive
```

The following example shows how to configure best link aggregation mode for variable bit rate real time (VBR-RT) traffic on node 2:

```
Switch(config)# atm router pnni
Switch(config-pnni-node)# node 2
Switch(config-pnni-node)# aggregation-mode node vbr-rt best-link
```

Displaying the Aggregation Mode Configuration

To display the aggregation mode configuration, enter the following commands in EXEC mode:

Command	Purpose
show atm pnni aggregation link	Displays the link aggregation mode.
show atm pnni aggregation node	Displays the node aggregation mode.

Examples

The following example shows the link aggregation mode:

```
Switch# show atm pnni aggregation link

PNNI PGL link aggregation for local-node 2 (level=72, name=Switch.2.72)

  Configured aggregation modes (per service class):
     CBR         VBR-RT       VBR-NRT       ABR           UBR
  ~~~~~~~~~~~  ~~~~~~~~~~~  ~~~~~~~~~~~  ~~~~~~~~~~~  ~~~~~~~~~~~
  aggressive   best-link    best-link    best-link    best-link

 No Aggregated links for this node.
Switch#
```

The following example shows how to display the node aggregation mode:

```
Switch# show atm pnni aggregation node
PNNI nodal aggregation for local-node 2 (level=56, child PG level=60)
  Complex node representation, exception threshold: 60%

  Configured nodal aggregation modes (per service class):
    CBR          VBR-RT       VBR-NRT      ABR          UBR
  ~~~~~~~~~~~  ~~~~~~~~~~~  ~~~~~~~~~~~  ~~~~~~~~~~~  ~~~~~~~~~~~
  best-link    best-link    best-link    best-link    aggressive

Summary Complex Node Port List:
  Port ID  Rem Inn Agg-Token  Border Cnt In-Spoke  Out-Spoke Agg-Accur
  ~~~~~~~~ ~~~~~~~ ~~~~~~~~~~ ~~~~~~~~~~ ~~~~~~~~~ ~~~~~~~~~ ~~~~~~~~~~
  21FB000  12      0          1          default   default   ok
  2371000  13      0          1          default   default   ok

Summary Complex Node Bypass Pairs  List (exception bypass pairs only)
  /~~~~~~~~ LOWER PORT ID ~~~~~~~~\ /~~~~~~~~ HIGHER PORT ID ~~~~~~~\
  Port ID  Rem Inn Agg-Token  Inacc Port ID  Rem Inn Agg-Token  Inacc Exceptns
  ~~~~~~~~ ~~~~~~~ ~~~~~~~~~~ ~~~~~ ~~~~~~~~ ~~~~~~~ ~~~~~~~~~~ ~~~~~ ~~~~~~~~
  21FB000  12      0          no    2371000  13      0          no    fwd rev
```

Configuring Significant Change Thresholds

PNNI topology state elements (PTSEs) would overwhelm the network if they were transmitted every time any parameter in the network changed. To avoid this problem, PNNI uses significant change thresholds that control the origination of PTSEs.

Note Any change in administrative weight (AW) and cell loss ratio (CLR) is considered significant and triggers a new PTSE.

To configure the PTSE significant change threshold, take these steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **atm router pnni** Switch(config-atm-router)#	Enters ATM router PNNI mode.
Step 2	Switch(config-atm-router)# **node** *node-index* Switch(config-pnni-node)#	Enters node configuration mode.
Step 3	Switch(config-pnni-node)# **ptse significant-change** {**acr-mt** *percent* | **acr-pm** *percent* | **cdv-pm** *percent* | **ctd-pm** *percent*}	Configures a PTSE significant change percentage.

For an example of other **ptse** command keywords, see the "Configuring PNNI Hello, Database Synchronization, and Flooding Parameters" section on page 10-50.

Example

The following example shows how to configure a PTSE being sent only if the available cell rate changes 30 percent from the current metric:

```
Switch(config)# atm router pnni
Switch(config-atm-router)# node 1
Switch(config-pnni-node)# ptse significant-change acr-pm 30
```

Displaying the Significant Change Thresholds Configuration

To display the PTSE configuration, use the following EXEC command:

Command	Purpose
show atm pnni resource-info	Displays the PTSE identifier.

Example

The following example shows the significant change threshold configuration using the **show atm pnni resource-info** EXEC command:

```
Switch# show atm pnni resource-info
PNNI:80.1 Insignificant change parameters
acr pm 50,  acr mt 3, cdv pm 25, ctd pm 50, resource poll interval 5 sec
Interface insignificant change bounds:
Interface ATM1/0/0
  CBR    : MCR 155519, ACR 147743 [73871,366792], CTD 50 [25,75],CDV 34 [26,42],
 CLR0 10, CLR01 10,
  VBR-RT : MCR 155519, ACR 155519 [77759,366792], CTD 359 [180,538],CDV 342 [257
,427], CLR0 8, CLR01 8,
  VBR-NRT: MCR 155519, ACR 155519 [77759,155519], CLR0 8, CLR01, 8
  ABR    : MCR 155519 ACR 147743 [73871,155519]
  UBR    : MCR 155519
<information deleted>
```

Configuring the Complex Node Representation for LGNs

By default, higher-level logical group nodes (LGNs) represent their child peer groups (PGs) in the simple node representation. With simple node representation, the entire peer group is represented as a single node. When there are many nodes in the child peer group, you can use complex node representation to present a more accurate model of the PG. With complex node representation, the PG is represented by a nucleus, or center, and border ports.

For a detailed description of complex node representation and implementation guidelines, refer to the *Guide to ATM Technology*.

To configure complex node representation, perform the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **atm router pnni** Switch(config-atm-router)#	Enters ATM router PNNI mode.
Step 2	Switch(config-atm-router)# **node** *local-node-index* Switch(config-pnni-node)#	Enters node configuration mode and specifies the local node you want to configure.
Step 3	Switch(config-pnni-node)# **nodal-representation** {**simple** | **complex** [**threshold** *threshold-value* | **radius-only**]}	Configures complex nodal representation and specifies how to handle exceptions.

Example

The following example shows how to configure a PNNI complex node:

```
Switch(config)# atm router pnni
Switch(config-atm-router)# node 2
Switch(config-pnni-node)# nodal-representation complex
```

Displaying the PNNI Complex Node Configuration

To display the PNNI complex node configuration, perform the following task in privileged EXEC mode:

Command	Purpose
show atm pnni aggregation node	Displays the PNNI complex node configuration.

Example

The following example shows the PNNI complex node configuration:

```
Switch# show atm pnni aggregation node
PNNI nodal aggregation for local-node 2 (level=56, child PG level=60)
  Complex node representation, exception threshold: 60%

  Configured nodal aggregation modes (per service class):
    CBR          VBR-RT       VBR-NRT      ABR          UBR
  ~~~~~~~~~~~  ~~~~~~~~~~~  ~~~~~~~~~~~  ~~~~~~~~~~~  ~~~~~~~~~~~
  best-link    best-link    best-link    best-link    aggressive

Summary Complex Node Port List:
  Port ID  Rem Inn Agg-Token  Border Cnt In-Spoke  Out-Spoke Agg-Accur
  ~~~~~~~~ ~~~~~~~ ~~~~~~~~~~ ~~~~~~~~~~ ~~~~~~~~~ ~~~~~~~~~ ~~~~~~~~~~
  21FB000  12      0          1          default   default   ok
  2371000  13      0          1          default   default   ok

Summary Complex Node Bypass Pairs  List (exception bypass pairs only)
  /~~~~~~~~ LOWER PORT ID ~~~~~~~~\ /~~~~~~~~ HIGHER PORT ID ~~~~~~~\
  Port ID  Rem Inn Agg-Token  Inacc Port ID  Rem Inn Agg-Token  Inacc Exceptns
  ~~~~~~~~ ~~~~~~~ ~~~~~~~~~~ ~~~~~ ~~~~~~~~ ~~~~~~~ ~~~~~~~~~~ ~~~~~ ~~~~~~~~
  21FB000  12      0          no    2371000  13      0          no    fwd rev
```

Tuning Protocol Parameters

The tasks in the following subsections describe how to tune the PNNI protocol parameters that can affect the performance of your network.

Configuring PNNI Hello, Database Synchronization, and Flooding Parameters

PNNI uses the Hello protocol to determine the status of neighbor nodes and PNNI topology state elements (PTSEs) to disseminate topology database information in the ATM network.

To configure the Hello protocol parameters and PTSE significant change, perform these steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **atm router pnni** Switch(config-atm-router)#	Enters ATM router PNNI mode.
Step 2	Switch(config-atm-router)# **node** *node-index* Switch(config-pnni-node)#	Enters node configuration mode.

	Command	Purpose
Step 3	Switch(config-pnni-node)# **timer** [**ack-delay** *tenths-of-second*] [**hello-holddown** *tenths-of-second*] [**hello-interval** *seconds*] [**inactivity-factor** *number*] [**retransmit-interval** *seconds*]	Configures Hello database synchronization and flooding parameters.
Step 4	Switch(config-pnni-node)# **ptse** [**lifetime-factor** *percentage-factor*] [**min-ptse-interval** *tenths-of-second*] [**refresh-interval** *seconds*] [**request** *number*] [**significant-change acr-mt** *percent*] [**significant-change acr-pm** *percent*] [**significant-change cdv-pm** *percent*] [**significant-change ctd-pm** *percent*]	Configure PTSE significant change percent number.

Example

The following example shows how to configure the PTSE refresh interval to 600 seconds:

```
Switch(config-pnni-node)# ptse refresh-interval 600
```

The following example shows how to configure the retransmission of the Hello timer to 60 seconds:

```
Switch(config-pnni-node)# timer hello-interval 60
```

Displaying the PNNI Hello, Database Synchronization, and Flooding Configuration

To display the ATM PNNI Hello, database synchronization, and flooding configuration, use the following privileged EXEC command:

Command	Purpose
show atm pnni local-node	Displays the ATM PNNI Hello, database synchronization, and flooding configuration.

Example

The following example shows the ATM PNNI Hello, database synchronization, and flooding configuration using the **show atm pnni local-node** privileged EXEC command:

```
Switch# show atm pnni local-node
PNNI node 1 is enabled and running
    Node name: Switch
    System address 47.00918100000000400B0A3081.00400B0A3081.00
    Node ID 56:160:47.00918100000000400B0A3081.00400B0A3081.00
    Peer group ID 56:47.0091.8100.0000.0000.0000.0000
    Level 56, Priority 0, No. of interfaces 4, No. of neighbors 2
    Node Allows Transit Calls

    Hello interval 15 sec, inactivity factor 5,
    Hello hold-down 10 tenths of sec
    Ack-delay 10 tenths of sec, retransmit interval 5 sec,
    Resource poll interval 5 sec
    PTSE refresh interval 1800 sec, lifetime factor 200 percent,
    Min PTSE interval 10 tenths of sec
    Auto summarization: on, Supported PNNI versions: newest 1, oldest 1
    Default administrative weight mode: uniform
    Max admin weight percentage: -1
    Next resource poll in 3 seconds
    Max PTSEs requested per PTSE request packet: 32
    Redistributing static routes: Yes
```

Configuring the Resource Management Poll Interval

The resource management poll interval specifies how often PNNI polls resource management to update the values of link metrics and attributes. You can configure the resource poll interval to control the tradeoff between the processing load and the accuracy of PNNI information. A larger value usually generates a smaller number of PTSE updates. A smaller value results in greater accuracy in tracking resource information.

To configure the resource management poll interval, perform these steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **atm router pnni** Switch(config-atm-router)#	Enters ATM router PNNI mode.
Step 2	Switch(config-atm-router)# **resource-poll-interval** *seconds*	Configures the resource management poll interval.

Example

The following example shows how to configure the resource management poll interval to 10 seconds:

```
Switch(config)# atm router pnni
Switch(config-atm-router)# resource-poll-interval 10
```

Displaying the Resource Management Poll Interval Configuration

To display the resource management poll interval configuration, use the following EXEC command:

Command	Purpose
show atm pnni resource-info	Displays the resource management poll interval configuration.

Example

The following example shows the resource management poll interval configuration using the **show atm pnni resource-info** EXEC command:

```
Switch# show atm pnni resource-info
PNNI:80.1 Insignificant change parameters
acr pm 50,  acr mt 3, cdv pm 25, ctd pm 50, resource poll interval 5 sec
Interface insignificant change bounds:
Interface ATM1/0/0
  CBR    : MCR 155519, ACR 147743 [73871,366792], CTD 50 [25,75],CDV 34 [26,42],
 CLR0 10, CLR01 10,
  VBR-RT : MCR 155519, ACR 155519 [77759,366792], CTD 359 [180,538],CDV 342 [257
,427], CLR0 8, CLR01 8,
  VBR-NRT: MCR 155519, ACR 155519 [77759,155519], CLR0 8, CLR01, 8
  ABR    : MCR 155519 ACR 147743 [73871,155519]
  UBR    : MCR 155519
Interface ATM1/0/3
  CBR    : MCR 155519, ACR 147743 [73871,366792], CTD 50 [25,75],CDV 34 [26,42],
 CLR0 10, CLR01 10,
  VBR-RT : MCR 155519, ACR 155519 [77759,366792], CTD 359 [180,538],CDV 342 [257
,427], CLR0 8, CLR01 8,
  VBR-NRT: MCR 155519, ACR 155519 [77759,155519], CLR0 8, CLR01, 8
  ABR    : MCR 155519 ACR 147743 [73871,155519]
  UBR    : MCR 155519
<information deleted>
```

Configuring ATM PNNI Statistics Collection

You can collect the following statistics about the routing of ATM connections:

- Number of source route requests
- Number of micro-seconds spent in dijkstra algorithm
- Number of crankback source route requests
- Number of next port requests
- Number of background route lookups
- Number of on-demand route computations

To enable statistics collection, perform these steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **atm router pnni** Switch(config-atm-router)#	Enters ATM router PNNI mode.
Step 2	Switch(config-atm-router)# **statistics call**	Enables ATM PNNI statistics gathering.

Example

The following example shows how to enable PNNI ATM statistics gathering:

```
Switch(config)# atm router pnni
Switch(config-atm-router)# statistics call
```

Displaying ATM PNNI Statistics

To display the ATM PNNI statistics, use the following privileged EXEC command:

Command	Purpose
show atm pnni statistics call	Displays the ATM PNNI statistics.

Example

The following example shows the ATM PNNI statistics using the **show atm pnni statistics** privileged EXEC command:

```
Switch# show atm pnni statistics call

pnni call statistics since 22:19:29

                   total     cbr       rtvbr     nrtvbr    abr       ubr
source route reqs  1346      0         0         0         0         0
successful         1342      1342      0         0         0         0
unsuccessful       4         4         0         0         0         0
crankback reqs     0         0         0         0         0         0
successful         0         0         0         0         0         0
unsuccessful       0         0         0         0         0         0
on-demand attempts 0         0         0         0         0         0
successful         0         0         0         0         0         0
unsuccessful       0         0         0         0         0         0
background lookups 0         0         0         0         0         0
successful         0         0         0         0         0         0
unsuccessful       0         0         0         0         0         0
next port requests 0         0         0         0         0         0
successful         0         0         0         0         0         0
unsuccessful       0         0         0         0         0         0

                   total     average
usecs in queue     2513166   1867
usecs in dijkstra  0         0
usecs in routing   132703    98
```

CHAPTER 11

Using Access Control

This chapter describes how to configure and maintain access control lists, which are used to permit or deny incoming calls or outgoing calls on an interface of the ATM switch router.

Note This chapter provides advanced configuration instructions for the Catalyst 8540 MSR, Catalyst 8510 MSR, and LightStream 1010 ATM switch routers. For complete descriptions of the commands mentioned in this chapter, refer to the *ATM Switch Router Command Reference* publication.

This chapter includes the following sections:

Access Control Overview

The ATM signalling software uses the access control list to filter setup messages on an interface based on destination, source, or a combination of both. Access lists can be used to deny connections known to be security risks and permit all other connections, or to permit only those connections considered acceptable and deny all the rest. For firewall implementation, denying access to security risks offers more control.

During initial configuration, perform the following steps to use access control to filter setup messages:

Step 1 Create a template alias allowing you to use real names instead of ATM addresses in your ATM filter expressions.

Step 2 Create the ATM filter set or filter expression based on your requirements.

Step 3 Associate the filter set or filter expression to an interface using the **atm access-group** command.

Step 4 Confirm the configuration.

Configuring a Template Alias

To configure an ATM template alias, use the following command in global configuration mode:

Command	Purpose
atm template-alias *name template*	Configures a global ATM address template alias.

Examples

The following example creates a template alias named *training* using the ATM address template 47.1328 and the ellipses (...) to fill in the trailing 4-bit hexadecimal digits in the address:

```
Switch(config)# atm template-alias training 47.1328...
```

The following example creates a template alias named *bit_set* with the ATM address template 47.9f9.(1*0*).88ab... that matches the four addresses that begin with the following:

- 47.9F9(1*000*).88AB... = 47.9F9*8*.88AB...
- 47.9F9(1*001*).88AB... = 47.9F9*9*.88AB...
- 47.9F9(1*100*).88AB... = 47.9F9*C*.88AB...
- 47.9F9(1*101*).88AB... = 47.9F9*D*.88AB...

```
Switch(config)# atm template-alias bit_set 47.9f9(1*0*).88ab...
```

The following example creates a template alias named *byte_wise* with the ATM address template 47.9*F8.33... that matches all ATM addresses beginning with the following sixteen prefixes:

- 47.9*0*F8.33...

  through
- 47.9*F*F8.33...

```
Switch(config)# atm template-alias byte_wise 47.9*F8.33...
```

Displaying the Template Alias Configuration

To display template alias configuration, use the following privileged EXEC command:

Command	Purpose
more system:running-config	Displays the current configuration.

Example

The following example shows the template aliases configured in the previous examples using the **more system:running-config** privileged EXEC command:

```
Switch# more system:running-config
Building configuration...

Current configuration:
!
version XX.X
no service pad
service udp-small-servers
service tcp-small-servers
!
hostname Switch
!
!
username dtate
ip rcmd remote-username dplatz
atm template-alias training 47.1328...
atm template-alias bit_set 47.9f9(1*0*).88ab...
atm template-alias byte_wise 47.9*f8.33...
!
<information deleted>
```

Configuring ATM Filter Sets

To create an ATM address filter or time-of-day filter, use the following command in global configuration mode:

Command	Purpose
atm filter-set *name* [**index** *number*] [**permit** | **deny**] {*template* | **time-of-day** {**anytime** | *start-time end-time*}}	Configures a global ATM address filter set.

Examples

The following example creates a filter named *filter_1* that permits access to the specific ATM address 47.0000.8100.1234.0003.c386.b301.0003.c386.b301.00:

```
Switch(config)# atm filter-set filter_1 permit 47.0000.8100.1234.0003.c386.b301.0003.c386.b301.00
```

The following example creates a filter named *filter_2* that denies access to the specific ATM address 47.000.8100.5678.0003.c386.b301.0003.c386.b301.00, but allows access to all other ATM addresses:

```
Switch(config)# atm filter-set filter_2 deny 47.0000.8100.5678.0003.c386.b301.0003.c386.b301.00
Switch(config)# atm filter-set filter_2 permit default
```

The following example creates a filter named *filter_3* that denies access to all ATM addresses that begin with the prefix 47.840F, but permits all other calls:

```
Switch(config)# atm filter-set filter_3 deny 47.840F...
Switch(config)# atm filter-set filter_3 permit default
```

Note The order in which deny and permit filters are configured is very important. See the following example.

In the following example, the first filter set, *filter_4*, has its first filter configured to permit all addresses and its second filter configured to deny access to all addressees that begin with the prefix 47.840F. Since the default filter matches all addresses, the second filter is never used. Addresses that begin with prefix 47.840F are also permitted.

```
Switch(config)# atm filter-set filter_4 permit default
Switch(config)# atm filter-set filter_4 deny 47.840F...
```

The following example creates a filter named *filter_5* that denies access to all ATM addresses described by the ATM template alias bad_users:

```
Switch(config)# atm filter-set filter_5 deny bad_users
Switch(config)# atm filter-set filter_5 permit default
```

The following example shows how to configure a filter set named *tod1*, with an index of 2, to deny calls between 11:15 a.m. and 10:45 p.m.:

```
Switch(config)# atm filter-set tod1 index 2 deny time-of-day 11:15 22:45
Switch(config)# atm filter-set tod1 index 3 permit time-of-day anytime
```

The following example shows how to configure a filter set named *tod1*, with an index of 4, to permit calls any time:

```
Switch(config)# atm filter-set tod1 index 4 permit time-of-day anytime
```

The following example shows how to configure a filter set named *tod2* to deny calls between 8:00 p.m. and 6:00 a.m.:

```
Switch(config)# atm filter-set tod2 deny time-of-day 20:00 06:00
Switch(config)# atm filter-set tod2 permit time-of-day anytime
```

The following example shows how to configure a filter set named *tod2* to permit calls at any time:

```
Switch(config)# atm filter-set tod2 permit time-of-day 3:30 3:30
```

Once you create a filter set using the previous configuration commands, it must be associated with an interface as an access group to actually filter any calls. See the "Configuring ATM Interface Access Control" section on page 11-6 to configure an individual interface with an access group.

Deleting Filter Sets

To delete an ATM filter set, use the following command in global configuration mode:

Command	Purpose
no atm filter-set *name* [**index** *number*]	Deletes a global ATM address filter set.

Example

The following example shows how to display and delete filter sets:

```
Switch# show atm filter-set
ATM filter set tod1
  deny From 11:15 Hrs Till 22:45 Hrs  index 2
  permit From 0:0 Hrs Till 0:0 Hrs  index 4
ATM filter set tod2
  deny From 20:0 Hrs Till 6:0 Hrs  index 1
  permit From 3:30 Hrs Till 3:30 Hrs  index 2
Switch# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)# no atm filter-set tod1 index 2
Switch(config)# no atm filter-set tod2
Switch(config)# end
Switch#
%SYS-5-CONFIG_I: Configured from console by console
Switch# show atm filter-set
ATM filter set tod1
  permit From 0:0 Hrs Till 0:0 Hrs  index 4
```

Configuring an ATM Filter Expression

To create global ATM filter expressions, perform the following steps in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **atm filter-expr** *name term*	Defines a simple filter expression with only one term and no operators.
Step 2	Switch(config)# **atm filter-expr** *name* [**destination** | **source** | **src**] *term1* **and** [**destination** | **source** | **src**] *term2*	Defines a filter expression using the operator **and**.
Step 3	Switch(config)# **atm filter-expr** *name* **not** [**destination** | **source** | **src**] *term*	Defines a filter expression using the operator **not**.
Step 4	Switch(config)# **atm filter-expr** *name* [**destination** | **source** | **src**] *term1* **or** [**destination** | **source** | **src**] *term2*	Defines a filter expression using the operator **or**.
Step 5	Switch(config)# **atm filter-expr** *name* [**destination** | **source** | **src**] *term1* **xor** [**destination** | **source** | **src**] *term2*	Defines a filter expression using the operator **xor**.
Step 6	Switch(config)# **no atm filter-expr** *name*	Deletes a filter.

Examples

The following example defines a simple filter expression that has only one term and no operators:

```
Switch(config)# atm filter-expr training filter_1
```

The following example defines a filter expression using the operator **not**:

```
Switch(config)# atm filter-expr training not filter_1
```

The following example defines a filter expression using the operator **or**:

```
Switch(config)# atm filter-expr training filter_2 or filter_1
```

The following example defines a filter expression using the operator **and**:

```
Switch(config)# atm filter-expr training filter_1 and source filter_2
```

The following example defines a filter expression using the operator **xor**:

```
Switch(config)# atm filter-expr training filter_2 xor filter_1
```

Configuring ATM Interface Access Control

To subscribe an ATM interface or subinterface to an existing ATM filter set or filter expression, perform the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **interface atm** *card*/*subcard*/*port*[**.***vpt#*] Switch(config-if)#	Selects the interface or subinterface to be configured.
Step 2	Switch(config-if)# **atm access-group** *name* [**in** | **out**]	Configures an existing ATM address pattern matching the filter expression.

Examples

The following example shows how to configure access control for outgoing calls on ATM interface 3/0/0:

```
Switch(config)# interface atm 3/0/0
Switch(config-if)# atm access-group training out
```

The following example shows how to configure access control for both outgoing and incoming calls on ATM interface 3/0/0:

```
Switch(config)# interface atm 3/0/0
Switch(config-if)# atm access-group training out
Switch(config-if)# atm access-group marketing in
```

Displaying ATM Filter Configuration

To display access control configuration, use the following EXEC commands:

Command	Purpose
show atm filter-set [*name*]	Displays a specific or a summary of ATM filter set.
show atm filter-expr [**detail**] *name*	Displays a specific or a summary of ATM filter expression.

Examples

The following command displays the configured ATM filters:

```
Switch# show atm filter-set
ATM filter set tod1
  deny From 11:15 Hrs Till 22:45 Hrs  index 2
  permit From 0:0 Hrs Till 0:0 Hrs  index 4
ATM filter set tod2
  deny From 20:0 Hrs Till 6:0 Hrs  index 1
  permit From 3:30 Hrs Till 3:30 Hrs  index 2
```

The following command displays the configured ATM filter expressions:

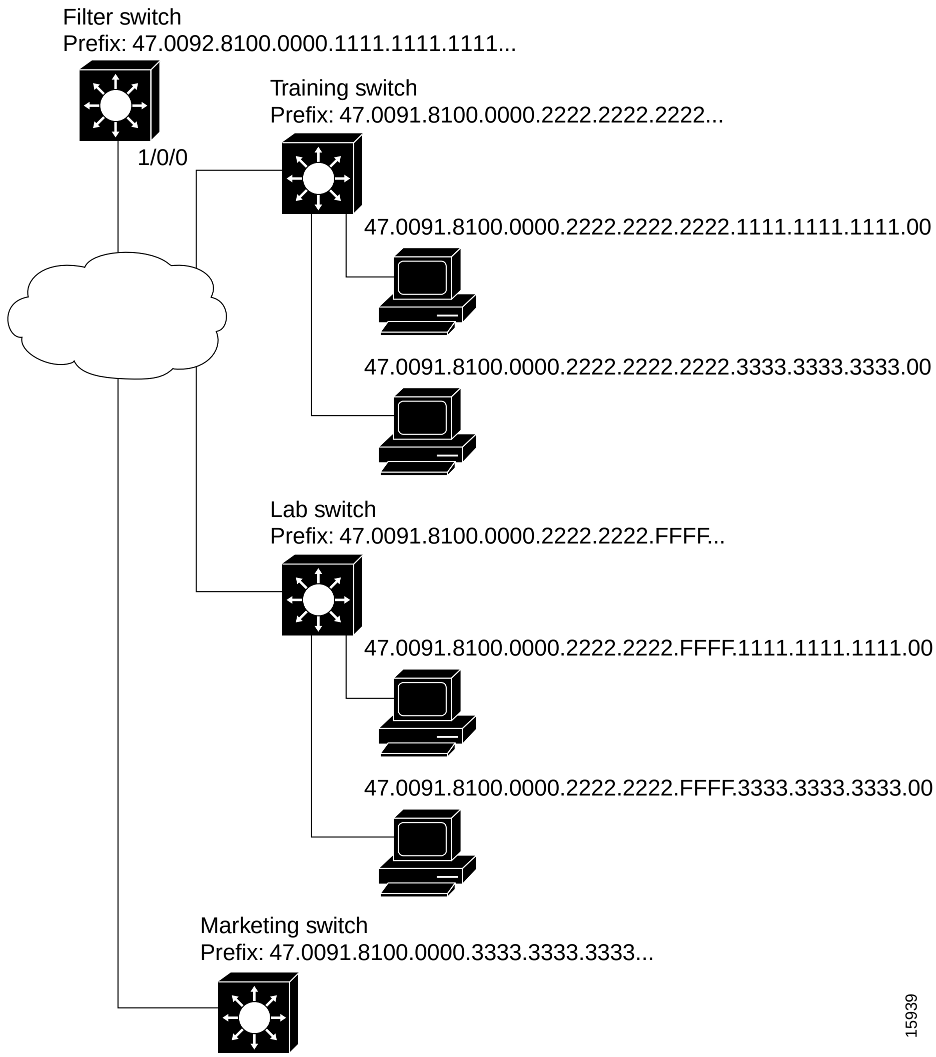
```
Switch# show atm filter-expr
training = dest filter_1
```

ATM Filter Configuration Scenario

This section provides a complete access filter configuration example using the information described in the preceding sections.

The example network configuration used in the following filter set configuration scenario is shown in Figure 11-1.

Figure 11-1 ATM Access Filter Configuration Example



Example

The following example shows how to configure the Filter Switch, shown in Figure 11-1, to deny access to all calls received on ATM interface 1/0/0 from the workstations directly attached to the Lab Switch, but to allow all other calls. The Filter Switch denies all calls if the calling party address begins with the prefix 47.0091.8100.0000.2222.2222.FFFF:

```
Filter Switch(config)# atm template-alias lab-sw 47.0091.8100.0000.2222.2222.FFFF...
Filter Switch(config)# atm filter-set filter_1 deny lab-sw
Filter Switch(config)# atm filter-set filter_1 permit default
Filter Switch(config)# atm filter-expr exp1 src filter_1
Filter Switch(config)#
Filter Switch(config)# interface atm 1/0/0
Filter Switch(config-if)# atm access-group exp1 in
Filter Switch(config-if)# end
Filter Switch# show atm filter-set
ATM filter set filter_1
  deny   47.0091.8100.0000.2222.2222.ffff...  index 1
  permit default  index 2
Filter Switch# show atm filter-expr
exp1 = src filter_1
```

Filtering IP Packets at the IP Interfaces

IP packet filtering helps control packet movement through the network. Such control can help limit network traffic and restrict network use by certain users or devices. To permit or deny packets from crossing specified IP interfaces, Cisco provides access lists.

You can use access lists for the following reasons:

- Control the transmission of packets on an IP interface
- Control virtual terminal line access
- Restrict contents of routing updates

This section summarizes how to create IP access lists and how to apply them.

Note This section applies to the IP interfaces only.

An access list is a sequential collection of permit and deny conditions that apply to IP addresses. The ATM switch router software tests addresses against the conditions in an access list one by one. The first match determines whether the software accepts or rejects the address. Because the software stops testing conditions after the first match, the order of the conditions is critical. If no conditions match, the software rejects the address.

The two steps involved in using access lists follow:

Step 1 Create an access list by specifying an access list number and access conditions.

Step 2 Apply the access list to interfaces or terminal lines.

These steps are described in the following sections:

- "Creating Standard and Extended IP Access Lists" section on page 11-10
- "Applying an IP Access List to an Interface or Terminal Line" section on page 11-11

Creating Standard and Extended IP Access Lists

The ATM switch router software supports three styles of access lists for IP interfaces:

- Standard IP access lists use source addresses for matching operations.
- Extended IP access lists use source and destination addresses for matching operations, as well as optional protocol type information for increased control.
- Dynamic extended IP access lists grant access per user to a specific source or destination host through a user authentication process. In essence, you can allow user access through a firewall dynamically, without compromising security restrictions.

To create a standard access list, use one of the following commands in global configuration mode:

Command	Purpose
access-list *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*]	Defines a standard IP access list using a source address and wildcard.
access-list *access-list-number* {**deny** | **permit**} **any**	Defines a standard IP access list using an abbreviation for the source and source mask of 0.0.0.0 255.255.255.255.

To create an extended access list, use one of the following commands in global configuration mode:

Command	Purpose
access-list *access-list-number* {**deny** | **permit**} *protocol source source-wildcard destination destination-wildcard* [**precedence** *precedence*] [**tos** *tos*] [**established**] [**log**]	Defines an extended IP access list number and the access conditions. Use the **log** keyword to get access list logging messages, including violations.
access-list *access-list-number* {**deny** | **permit**} *protocol* **any**	Defines an extended IP access list using an abbreviation for a source and source wildcard of 0.0.0.0 255.255.255.255, and an abbreviation for a destination and destination wildcard of 0.0.0.0 255.255.255.255.
access-list *access-list-number* {**deny** | **permit**} *protocol* **host** *source* **host** *destination*	Defines an extended IP access list using an abbreviation for a source and source wildcard of *source* 0.0.0.0, and an abbreviation for a destination and destination wildcard of *destination* 0.0.0.0.
access-list *access-list-number* **dynamic** *dynamic-name* [**timeout** *minutes*] {**deny** | **permit**} *protocol source source-wildcard destination destination-wildcard* [**precedence** *precedence*] [**tos** *tos*] [**established**] [**log**]	Defines a dynamic access list.

After you create an access list, any subsequent additions (possibly entered from the terminal) are placed at the end of the list. In other words, you cannot selectively add or remove access list command lines from a specific access list.

Note When making the standard and extended access list, by default, the end of the access list contains an implicit deny statement for everything if it does not find a match before reaching the end. Further, with standard access lists, if you omit the mask from an associated IP host address access list specification, 0.0.0.0 is assumed to be the mask.

Applying an IP Access List to an Interface or Terminal Line

After you create an access list, you can apply it to one or more interfaces. Access lists can be applied on *either* outbound or inbound interfaces. The following two tables show how this task is accomplished for both terminal lines and network interfaces.

To apply an access list to a terminal line, perform the following tasks, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **line** [**aux** | **console** | **vty**] *line-number* Switch(config-line)#	Selects the line to be configured.
Step 2	Switch(config-line)# **access-class** *access-list-number* {**in** | **out**}	Restricts incoming and outgoing connections between a particular virtual terminal line (into a device) and the addresses in an access list.

To apply an access list to a network interface, perform the following tasks, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **interface atm** *card*/*subcard*/*port* Switch(config-if)#	Selects the interface or subinterface to be configured.
Step 2	Switch(config-if)# **ip access-group** *access-list-number* {**in** | **out**}	Controls access to an interface.

For inbound access lists, after receiving a packet, the ATM switch router software checks the source address of the packet against the access list. If the access list permits the address, the software continues to process the packet. If the access list rejects the address, the software discards the packet and returns an Internet Control Message Protocol (ICMP) host unreachable message.

For outbound access lists, after receiving and routing a packet to a controlled interface, the software checks the source address of the packet against the access list. If the access list permits the address, the software transmits the packet. If the access list rejects the address, the software discards the packet and returns an ICMP host unreachable message.

If you apply an access list (standard or extended) that has not yet been defined to an interface, the software acts as if the access list has not been applied to the interface and accepts all packets. You must define the access list to the interface if you use it as a means of security in your network.

Note Set identical restrictions on all the virtual terminal lines, because a user can attempt to connect to any of them.

IP Access List Examples

In the following example, network 36.0.0.0 is a Class A network whose second octet specifies a subnet; that is, its subnet mask is 255.255.0.0. The third and fourth octets of a network 36.0.0.0 address specify a particular host.

Using access list 2, the ATM switch router software accepts one address on subnet 48 and rejects all others on that subnet. The last line of the list shows that the software accepts addresses on all other network 36.0.0.0 subnets.

```
Switch(config)# access-list 2 permit 36.48.0.3
Switch(config)# access-list 2 deny 36.48.0.0 0.0.255.255
Switch(config)# access-list 2 permit 36.0.0.0  0.255.255.255
Switch(config)# interface ethernet0
Switch(config-if)# ip access-group 2 in
```

Examples of Implicit Masks in IP Access Lists

IP access lists contain *implicit* masks. For example, if you omit the mask from an associated IP host address access list specification, 0.0.0.0 is assumed to be the mask. Consider the following example configuration:

```
Switch(config)# access-list 1 permit 0.0.0.0
Switch(config)# access-list 1 permit 131.108.0.0
Switch(config)# access-list 1 deny 0.0.0.0 255.255.255.255
```

For this example, the following masks are implied in the first two lines:

```
Switch(config)# access-list 1 permit 0.0.0.0 0.0.0.0
Switch(config)# access-list 1 permit 131.108.0.0 0.0.0.0
```

The last line in the configuration (using the **deny** keyword) can be omitted, because IP access lists implicitly *deny* all other access, which is equivalent to finishing the access list with the following command statement:

```
Switch(config)# access-list 1 deny 0.0.0.0 255.255.255.255
```

The following access list only allows access for those hosts on the three specified networks. It assumes that subnetting is not used; the masks apply to the host portions of the network addresses. Any hosts with a source address that does not match the access list statements is rejected.

```
Switch(config)# access-list 1 permit 192.5.34.0 0.0.0.255
Switch(config)# access-list 1 permit 128.88.0.0 0.0.255.255
Switch(config)# access-list 1 permit 36.0.0.0 0.255.255.255
! (Note: all other access implicitly denied)
```

To specify a large number of individual addresses more easily, you can omit the address mask that is all zeros from the **access-list** global configuration command. Thus, the following two configuration commands are identical in effect:

```
Switch(config)# access-list 2 permit 36.48.0.3
Switch(config)# access-list 2 permit 36.48.0.3 0.0.0.0
```

Examples of Configuring Extended IP Access Lists

In the following example, the first line permits any incoming Transmission Control Protocol (TCP) connections with destination ports greater than 1023. The second line permits incoming TCP connections to the simple mail transfer protocol (SMTP) port of host 128.88.1.2. The last line permits incoming ICMP messages for error feedback.

```
Switch(config)# access-list 102 permit tcp 0.0.0.0 255.255.255.255 128.88.0.0 0.0.255.255 gt 1023
Switch(config)# access-list 102 permit tcp 0.0.0.0 255.255.255.255 128.88.1.2 0.0.0.0 eq 25
Switch(config)# access-list 102 permit icmp 0.0.0.0 255.255.255.255 128.88.0.0 255.255.255.255
Switch(config)# interface ethernet0
Switch(config-if)# ip access-group 102 in
```

As another example, suppose you have a network connected to the Internet, and you want any host on an Ethernet to be able to form TCP connections to any host on the Internet. However, you do not want IP hosts to be able to form TCP connections to hosts on the Ethernet except to the mail (SMTP) port of a dedicated mail host.

SMTP uses TCP port 25 on one end of the connection and a random port number on the other end. The same two port numbers are used throughout the life of the connection. Mail packets coming in from the Internet have a destination port of 25. Outbound packets will have the port numbers reversed. The fact that the secure system behind the switch always accepts mail connections on port 25 is what makes it possible to separately control incoming and outgoing services. The access list can be configured on either the outbound or inbound interface.

In the following example, the Ethernet network is a Class B network with the address 128.88.0.0, and the mail host's address is 128.88.1.2. The keyword **established** is used only for the TCP protocol to indicate an established connection. A match occurs if the TCP datagram has the acknowledgment (ACK) or RST bits set, indicating that the packet belongs to an existing connection.

```
Switch(config)# access-list 102 permit tcp 0.0.0.0 255.255.255.255 128.88.0.0 0.0.255.255 established
Switch(config)# access-list 102 permit tcp 0.0.0.0 255.255.255.255 128.88.1.2 0.0.0.0 eq 25
Switch(config)# interface ethernet0
Switch(config-if)# ip access-group 102 in
```

Configuring Per-Interface Address Registration with Optional Access Filters

The ATM switch router allows configuration of per-interface access filters for Integrated Local Management Interface (ILMI) address registration to override the global default of access filters.

To configure ILMI address registration and the optional access filters for a specified interface, perform the following tasks, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **interface atm** *card*/*subcard*/*port* Switch(config-if)#	Specifies an ATM interface and enters interface configuration mode.
Step 2	Switch(config-if)# **atm address-registration permit** {**all** | **matching-prefix** [**all-groups** | **wellknown-groups**]}	Configures ILMI address registration and the optional access filters for a specified interface.

Example

The following example shows how to configure ILMI address registration on an individual interface to permit all groups with a matching ATM address prefix:

```
Switch(config)# interface atm 3/0/0
Switch(config-if)# atm address-registration permit matching-prefix all-groups
%ATM-5-ILMIACCFILTER: New access filter setting will be applied to registration
of new addresses on ATM3/0/0.
Switch(config-if)#
```

Displaying the ILMI Access Filter Configuration

To display the interface ILMI address registration access filter configuration, use the following EXEC command:

Command	Purpose
more system:running-config	Displays the interface ILMI address registration access filter configuration.

Example

The following example displays address registration access filter configuration for ATM interface 3/0/0:

```
Switch# more system:running-config
Building configuration...
Current configuration:
!
version XX.X
no service pad

<Information Deleted>

interface ATM0
 no ip address
 atm maxvp-number 0
!
interface Ethernet0
 ip address 172.20.41.110 255.255.255.0
 ip access-group 102 out
!
interface ATM3/0/0
 no atm auto-configuration
 atm address-registration permit matching-prefix all-groups
 atm iisp side user
 atm pvc 100 200
 atm signalling cug access permit-unknown-cugs both-direction permanent
 atm accounting
!
interface ATM3/0/1
!

<information deleted>
```

CHAPTER 12

Configuring IP over ATM

This chapter describes how to configure IP over ATM on the ATM switch router. The primary use of IP over ATM is for inband management of the ATM switch router.

Note This chapter provides advanced configuration instructions for the Catalyst 8540 MSR, Catalyst 8510 MSR, and LightStream 1010 ATM switch routers. For further information about Layer 3 protocols over ATM, refer to the *Guide to ATM Technology*. For complete descriptions of the commands mentioned in this chapter, refer to the *ATM Switch Router Command Reference* publication.

This chapter includes the following sections:

Configuring Classical IP over ATM

This section describes configuring a port on a ATM switch router to allow a classical IP-over-ATM connection to the ATM switch router's route processor and optional ATM router module.

The following sections describe configuring the ATM switch router for classical IP over ATM in either a switched virtual channel (SVC) or permanent virtual channel (PVC) environment.

Configuring Classical IP over ATM in an SVC Environment

This section describes classical IP over ATM in an SVC environment. It requires configuring only the device's own ATM address and that of a single ATM Address Resolution Protocol (ARP) server into each client device.

For a detailed description of the role and operation of the ATM ARP server, refer to the *Guide to ATM Technology*.

The ATM switch router can be configured as an ATM ARP client to work with any ATM ARP server conforming to RFC 1577. Alternatively, one of the ATM switch routers in a logical IP subnet (LIS) can be configured to act as the ATM ARP server itself. In that case, it automatically acts as a client as well. The following sections describe configuring the ATM switch router in an SVC environment as either an ATM ARP client or an ATM ARP server.

Configuring as an ATM ARP Client

In an SVC environment, configure the ATM ARP mechanism on the interface by performing the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **interface atm 0** Switch(config-if)# or Switch(config)# **interface atm** *card*/*subcard*/*port* Switch(config-if)#	Selects the route processor interface. or If you are using the optional Catalyst 8540 MSR enhanced ATM router module, specifies the ATM interface number.
Step 2	Switch(config-if)# **atm nsap-address** *nsap-address* or Switch(config-if)# **atm esi-address** *esi***.***selector*	Specifies the network service access point (NSAP) ATM address of the interface. or Specifies the end-system-identifier (ESI) address of the interface.
Step 3	Switch(config-if)# **ip address** *ip-address mask*	Specifies the IP address of the interface.
Step 4	Switch(config-if)# **atm arp-server nsap** *nsap-address*	Specifies the ATM address of the ATM ARP server.
Step 5	Switch(config-if)# **exit** Switch(config)#	Exits interface configuration mode.
Step 6	Switch(config)# **atm route** *addr-prefix*[1] {**atm 0** | **atm** *card*/*subcard*/*port*} **internal**	Configures a static route through the ATM switch router to the route processor interface, or the optional Catalyst 8540 MSR enhanced ATM router module interface. See the following note.

1. Address prefix is first 19 bytes of the NSAP address.

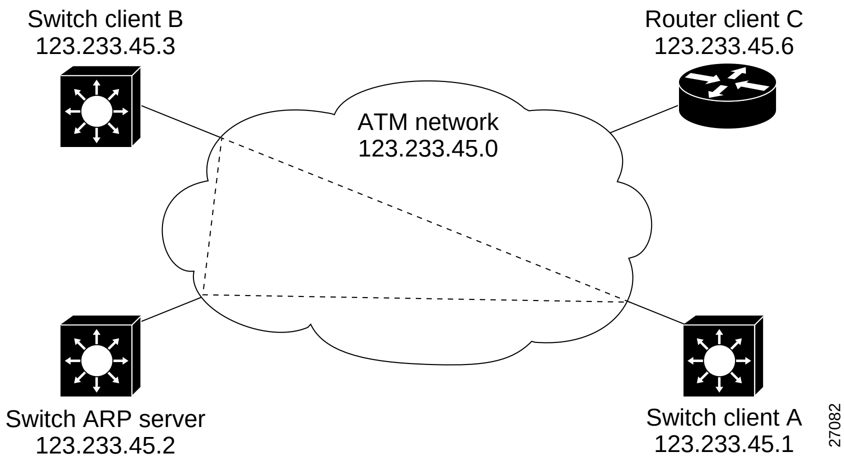
Note The end system identifier (ESI) address form is preferred in that it automatically handles the advertising of the address. Use the network service access point (NSAP) form of the command when you need to define a full 20-byte unique address with a prefix unrelated to the network prefix on that interface. You only need to specify a static route when configuring an ARP client using an NSAP address.

Note Since the 12.0(1a)W5(5b) release of the system software, addressing the interface on the processor card has changed. The ATM interface is now called atm0, and the Ethernet interface is now called ethernet0. The old formats (atm 2/0/0 and ethernet 2/0/0) are still supported.

NSAP Address Example

Figure 12-1 shows three ATM switch routers and a router connected using classical IP over ATM.

Figure 12-1 Classical IP over ATM Connection Setup



The following example shows how to configure the route processor interface ATM 0 of client A in Figure 12-1, using the NSAP address:

```
Client A(config)# interface atm 0
Client A(config-if)# atm nsap-address 47.0091.8100.0000.1111.1111.1111.1111.1111.1111.00
Client A(config-if)# ip address 123.233.45.1 255.255.255.0
Client A(config-if)# atm arp-server nsap 47.0091.8100.0000.1111.1111.1111.2222.2222.2222.00
Client A(config-if)# exit
Client A(config)# atm route 47.0091.8100.0000.1111.1111.1111.1111.1111.1111 atm 0 internal
```

ESI Example

The following example shows how to configure route processor interface ATM 0 of client A in Figure 12-1 using the ESI:

```
Client A(config)# interface atm 0
Client A(config-if)# atm esi-address 0041.0b0a.1081.40
Client A(config-if)# ip address 123.233.45.1 255.255.255.0
Client A(config-if)# atm arp-server nsap 47.0091.8100.0000.1111.1111.1111.2222.2222.2222.00
Client A(config-if)# exit
Client A(config)# atm route 47.0091.8100.0000.1111.1111.1111.1111.1111.1111 atm 0 internal
```

Configuring as an ATM ARP Server

Cisco's implementation of the ATM ARP server supports a single, nonredundant server per LIS and one ATM ARP server per subinterface. Thus, a single ATM switch router can support multiple ARP servers by using multiple interfaces.

To configure the ATM ARP server, perform the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **interface atm 0**[**.***subinterface#*] Switch(config-if)# or Switch(config)# **interface atm** *card***/***subcard***/***port*[**.***subinterface#*] Switch(config-if)#	Selects the route processor interface. or If you are using the optional Catalyst 8540 MSR enhanced ATM router module, specifies the ATM interface number.
Step 2	Switch(config-if)# **atm nsap-address** *nsap-address* or Switch(config-if)# **atm esi-address** *esi***.***selector*	Specifies the NSAP ATM address of the interface. or Specifies the end-system-identifier address of the interface.
Step 3	Switch(config-if)# **ip address** *ip-address mask*	Specifies the IP address of the interface.
Step 4	Switch(config-if)# **atm arp-server self** [**time-out** *minutes*][1]	Configures this interface as the ATM ARP server for the logical IP network.
Step 5	Switch(config-if)# **atm route** *addr-prefix*[2] {**atm 0** | **atm** *card***/***subcard***/***port*} **internal**	Configures a static route through the ATM switch router to the route processor interface, or the optional Catalyst 8540 MSR enhanced ATM router module interface. See the following note.

1. This form of the **atm arp-server** command indicates that this interface performs the ATM ARP server functions. When you configure the ATM ARP client (described earlier), the **atm arp-server** command is used—with a different keyword and argument—to identify a different ATM ARP server to the client.
2. Address prefix is first 19 bytes of the NSAP address.

Note The ESI address form is preferred in that it automatically handles the advertising of the address. Use the NSAP form of the command when you need to define a full 20-byte unique address with a prefix unrelated to the network prefix on that interface. You only need to specify a static route when configuring an ARP server using an NSAP address.

The idle timer interval is the number of minutes a destination entry listed in the ATM ARP server ARP table can be idle before the server takes any action to timeout the entry.

Example

The following example configures the route processor interface ATM 0 as an ARP server (shown in Figure 12-1):

```
ARP_Server(config)# interface atm 0
ARP_Server(config-if)# atm esi-address 0041.0b0a.1081.00
ARP_Server(config-if)# atm arp-server self
ARP_Server(config-if)# ip address 123.233.45.2 255.255.255.0
Client A(config)# atm route 47.0091.8100.0000.1111.1111.1111.1111.1111.1111 atm 0 internal
```

Displaying the IP-over-ATM Interface Configuration

To show the IP-over-ATM interface configuration, use the following EXEC commands:

Command	Purpose
show atm arp-server	Shows the ATM interface ARP configuration.
show atm map	Shows the ATM map list configuration.

Examples

In the following example, the **show atm arp-server** command displays the configuration of the interface ATM 0:

```
Switch# show atm arp-server

Note that a '*' next to an IP address indicates an active call

      IP Address        TTL     ATM Address
ATM2/0/0:
    * 10.0.0.5          19:21   4700918100567000000000112200410b0a108140
```

The following example displays the map-list configuration of the static map and IP-over-ATM interfaces:

```
Switch# show atm map
Map list ATM2/0/0_ATM_ARP : DYNAMIC
arp maps to NSAP 36.0091810000000003D5607900.0003D5607900.00
        , connection up, VPI=0 VCI=73, ATM2/0/0
ip 5.1.1.98 maps to s 36.0091810000000003D5607900.0003D5607900.00
        , broadcast, connection up, VPI=0 VCI=77, ATM2/0/0

Map list ip : PERMANENT
ip 5.1.1.99 maps to VPI=0 VCI=200
```

Configuring Classical IP over ATM in a PVC Environment

This section describes how you configure classical IP over ATM in a permanent virtual channel (PVC) environment. The ATM Inverse ARP (InARP) mechanism is applicable to networks that use PVCs, where connections are established but the network addresses of the remote ends are not known. A server function is *not* used in this mode of operation.

In a PVC environment, configure the ATM InARP mechanism by performing the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **interface atm 0** Switch(config-if)# or Switch(config)# **interface atm** *card*/*subcard*/*port* Switch(config-if)#	Selects the route processor interface. If you are using the optional ATM router module, specifies the ATM interface number.
Step 2	Switch(config-if)# **ip address** *ip-address mask*	Specifies the IP address of the interface.
Step 3	Switch(config-if)# **atm pvc** [**0** | **2**] *vci* **interface atm** *card*/*subcard*/*port vpi vci* **encap** [**aal5mux** | **aal5snap**] [**inarp** *minutes*]	Creates a PVC and enables Inverse ARP. The VPI value on interface ATM 0 is 0. The VPI value on an ATM router module interface is 2.

Repeat these tasks for each PVC you want to create.

The **inarp** *minutes* interval specifies how often Inverse ARP datagrams are sent on this virtual circuit. The default value is 15 minutes.

Note The ATM ARP and ATM InARP mechanisms work with IP only. All other protocols require **map-list** command entries to operate.

Example

The following example shows how to configure an IP-over-ATM interface on interface ATM 0, using a PVC with AAL5SNAP encapsulation, inverse ARP set to ten minutes, VPI = 0, and VCI = 100:

```
Switch(config)# interface atm 0
Switch(config-if)# ip address 11.11.11.11 255.255.255.0
Switch(config-if)# atm pvc 0 100 interface atm 0/0/0 50 100 encap aal5snap inarp 10
```

Displaying the IP-over-ATM Interface Configuration

To show the IP-over-ATM interface configuration, use the following EXEC command:

Command	Purpose
show atm map	Shows the ATM interface ARP configuration.

Example

The following example displays the map-list configuration of the static map and IP-over-ATM interfaces:

```
Switch# show atm map
Map list yyy : PERMANENT
ip 1.1.1.2 maps to VPI=0 VCI=200

Map list zzz : PERMANENT

Map list a : PERMANENT

Map list 1 : PERMANENT

Map list ATM2/0/0_ATM_ARP : DYNAMIC
arp maps to NSAP 47.00918100567000000001122.00410B0A1081.40
        , connection up, VPI=0 VCI=85, ATM2/0/0
ip 10.0.0.5 maps to NSAP 47.00918100567000000001122.00410B0A1081.40
        , broadcast, ATM2/0/0
```

Mapping a Protocol Address to a PVC Using Static Map Lists

The ATM interface supports a static mapping scheme that identifies the ATM address of remote hosts or ATM switch routers. This IP address is specified as a permanent virtual channel (PVC) or as a network service access point (NSAP) address for switch virtual channel (SVC) operation.

The following sections describe configuring both PVC-based and SVC-based map lists on the ATM switch router. For a more detailed discussion of static map lists, refer to the *Guide to ATM Technology*.

Configurations for both PVC and SVC map lists are described in the following sections:

- Configuring a PVC-Based Map List, page 12-7
- Configuring an SVC-Based Map List, page 12-9

Configuring a PVC-Based Map List

This section describes how to map a PVC to an address, which is a required task if you are configuring a PVC.

You enter mapping commands as groups. You first create a map list and then associate it with an interface. Perform the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config-if)# **interface atm** *card*/*subcard*/*port*[**.***subinterface*#]	Specifies an ATM interface and enters interface configuration mode.
Step 2	Switch(config-if)# **ip address** *ip-address mask*	Enters the IP address and subnet mask associated with this interface.
Step 3	Switch(config-if)# **map-group** *name*	Enters the map group name associated with this PVC.

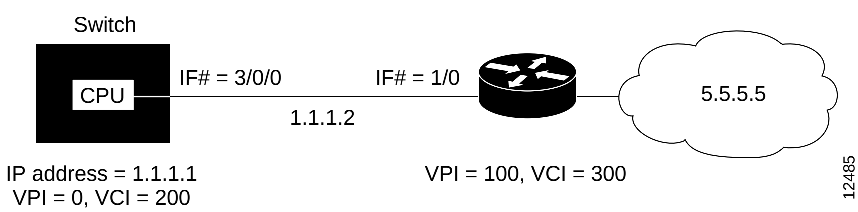
	Command	Purpose
Step 4	Switch(config-if)# **atm pvc** *vpi-a vci-a* [**upc** *upc*] [**pd** *pd*] [**rx-cttr** *index*] [**tx-cttr** *index*] **interface atm** *card*/*subcard*/*port*[**.***vpt#*] *vpi-b vci-b* [**upc** *upc*] [**encap** *aal-encap*]	Configures the PVC.
Step 5	Switch(config-if)# **exit** Switch(config)#	Exits interface configuration mode.
Step 6	Switch(config)# **ip route** *ip-address mask forward-ip address*	Configures an IP route to the router.
Step 7	Switch(config)# **map-list** *name* Switch(config-map-list)#	Creates a map list by naming it, and enters map-list configuration mode.
Step 8	Switch(config-map-list)# **ip** *ip-address* {**atm-nsap** *address* | **atm-vc** *vci*} [**aal5mux** *encapsulation*] [**broadcast** *pseudo-broadcast*] [**class** *class-name*]	Associates a protocol and address to a specific virtual circuit.

You can create multiple map lists, but only one map list can be associated with an interface. Different map lists can be associated with different interfaces.

Example

Figure 12-2 illustrates a connection configured with a PVC map list.

Figure 12-2 PVC Map List Configuration Example



The following example shows the commands used to configure the connection in Figure 12-2.

```
Switch(config)# interface atm 0
Switch(config-if)# ip address 1.1.1.1 255.0.0.0
Switch(config-if)# map-group yyy
Switch(config-if)# atm pvc 0 200 interface atm 3/0/0 100 300 encap aal5snap
Switch(config-if)# exit
Switch(config)# ip route 1.1.1.1 255.0.0.0 1.1.1.2
Switch(config)# map-list yyy
Switch(config-map-list)# ip 1.1.1.2 atm-vc 200
```

Displaying the Map-List Interface Configuration

To show the map-list interface configuration, use the following EXEC command:

Command	Purpose
show atm map	Shows the ATM interface map-list configuration.

Example

The following example displays the map-list configuration at interface ATM 0:

```
Switch# show atm map
Map list yyy : PERMANENT
ip 1.1.1.2 maps to VPI=0 VCI=200
```

Configuring an SVC-Based Map List

This section describes how to map an SVC to an NSAP address. This is a required task if you are configuring an SVC.

You enter mapping commands as groups. You first create a map list and then associate it with an interface. Perform the following steps, beginning in global configuration mode:

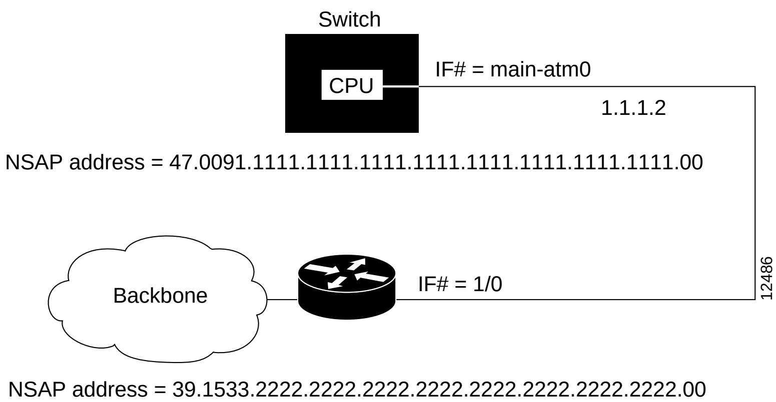
	Command	Purpose
Step 1	Switch(config)# **interface atm** *card*/*subcard*/*port*[**.***subinterface#*] Switch(config-if)#	Specifies an ATM interface and enters interface configuration mode.
Step 2	Switch(config-if)# **ip address** *ip-address mask*	Enters the IP address and subnet mask associated with this interface.
Step 3	Switch(config-if)# **atm nsap-address** *nsap-address*	Configures the interface NSAP address.
Step 4	Switch(config-if)# **map-group** *name*	Enters the map-group name associated with this PVC.
Step 5	Switch(config-if)# **exit** Switch(config)#	Exits interface configuration mode.
Step 6	Switch(config)# **map-list** name Switch(config-map-list)#	Creates a map list by naming it, and enters map-list configuration mode.
Step 7	Switch(config-map-list)# **ip** *ip-address* {**atm-nsap** *address* | **atm-vc** *vci*} [**aal5mux** *encapsulation*] [**broadcast** *pseudo-broadcast*] [**class** *class-name*]	Associates a protocol and address to a specific virtual circuit.

You can create multiple map lists, but only one map list can be associated with an interface. Different map lists can be associated with different interfaces.

Examples

Figure 12-3 illustrates an SVC connection configured with a map list.

Figure 12-3 SVC Map-List Configuration Example



The following example shows the commands used to configure the connection in Figure 12-3:

```
Switch(config)# interface atm 0
Switch(config-if)# ip address 1.1.1.1 255.0.0.0
Switch(config-if)# atm nsap-address 47.0091.1111.1111.1111.1111.1111.1111.1111.1111.00
Switch(config-if)# map-group zzz
Switch(config-if)# exit
Switch(config)# map-list zzz
Switch(config-map-list)# ip 1.1.1.2 atm-nsap 39.1533.2222.2222.2222.2222.2222.2222.2222.2222.00
```

Displaying the Map-List Interface Configuration

To show the map-list interface configuration, use the following EXEC command:

Command	Purpose
show atm map	Shows the ATM interface map-list configuration.

Example

The following example displays the map-list configuration at interface ATM 0:

```
Switch# show atm map

Map list zzz : PERMANENT
ip 1.1.1.2 maps to NSAP AC.1533222222222222222222222.222222222222.00
```

CHAPTER 13

Configuring LAN Emulation

This chapter describes LAN emulation (LANE) and how to configure it on the ATM switch router.

Note This chapter provides advanced configuration instructions for the Catalyst 8540 MSR, Catalyst 8510 MSR, and LightStream 1010 ATM switch routers. For an overview of LANE architecture and operation, refer to the *Guide to ATM Technology*. For complete descriptions of the commands mentioned in this chapter, refer to the *ATM Switch Router Command Reference* publication. For a detailed description of LANE and its components, refer to *Cisco IOS Switching Services Configuration Guide: Virtual LANs*.

This chapter contains the following sections:

LANE Functionality and Requirements

LANE uses ATM as a backbone to interconnect existing legacy LANs. In doing so, LANE allows legacy LAN users to take advantage of ATM's benefits without requiring modifications to end station hardware or software.

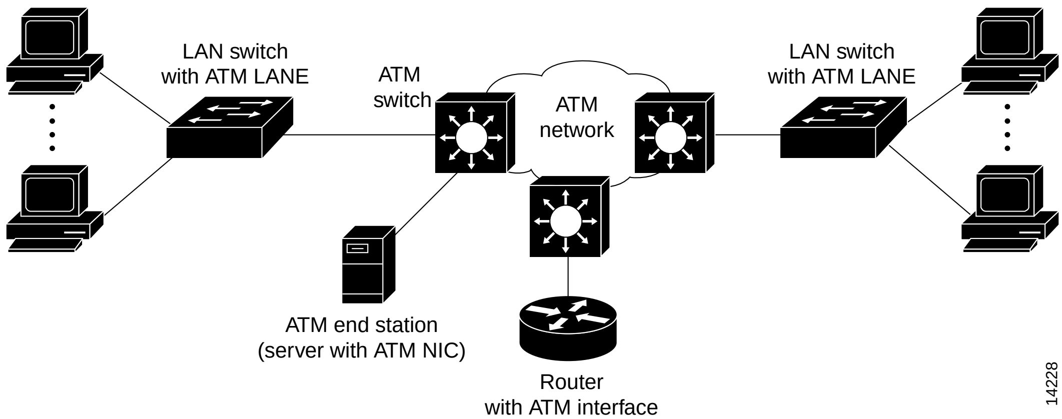
Multiple emulated LANs (ELANs), which are logically separated, can share the same physical ATM network and the same physical ATM interface. LANE makes an ATM interface look like one or more separate Ethernet or Token Ring interfaces.

LANE services provide connectivity between ATM-attached devices and LAN-attached devices. Two primary applications for the LANE protocol are as follows:

- Connectivity between LAN-attached stations across an ATM network, effectively extending LANs over a high-speed ATM transport backbone.
- Connectivity between ATM-attached hosts and LAN-attached hosts. Centralized hosts with high-speed ATM port adapters provide services, such as Domain Name System (DNS), to traditional LAN-attached devices.

Figure 13-1 illustrates the various connections LANE provides.

Figure 13-1 LANE Concept



Refer to the *Guide to ATM Technology* for the following background topics on LANE:

- How LANE works—the operation of LANE and the function of ATM network devices in LANE
- LANE components—the function of the server and client components that are required for LANE
- LANE virtual circuit connection (VCC) types—the role of each VCC type in establishing, maintaining, and tearing down LANE connections
- Addressing—the scheme used in automatically assigning ATM addresses to LANE components
- LANE examples—step-by-step process of joining an emulated LAN and building a LANE connection from a PC

LANE Router and Switch Router Requirements

You must manually configure Q.2931 over Signaling ATM Adaptation Layer (QSAAL) and ILMI signalling PVCs on routers and edge LAN switch routers to run LANE. However, these signalling permanent virtual channels (PVCs) are automatically configured on the ATM switch router.

Note The Catalyst 8510 MSR and LightStream 1010 processor and port adapters can be installed in slots 9 through 13 of the Catalyst 5500 switch. In this case, no physical connection is required between the ATM port adapter and the LANE card if the ATM Fabric Integration Module is used.

At least one ATM switch router is required to run LANE. For example, you cannot run LANE on routers connected back-to-back.

LANE Configuration Tasks

Before you begin to configure LANE, you must decide whether you want to set up one or multiple emulated LANs. If you set up multiple emulated LANs, you must also decide where the servers and clients will be located, and whether to restrict the clients that can belong to each emulated LAN. The procedure for configuring bridged emulated LANs is the same as for any other LAN.

To configure LANE, complete the tasks in the following sections:

- Creating a LANE Plan and Worksheet, page 13-3
- Displaying LANE Default Addresses, page 13-6
- Entering the ATM Address of the Configuration Server, page 13-7
- Setting Up the Configuration Server Database, page 13-7

Note For fault tolerance, multiple LANE services and servers can be assigned to the emulated LAN. This requires the use of our ATM switch routers and our ATM edge devices end-to-end.

- Enabling the Configuration Server, page 13-10

  An ATM cloud can contain multiple configuration servers.
- Setting Up LESs and Clients, page 13-11

  Every ELAN must have at least a LAN emulation server/broadcast-and unknown server (LES/BUS) pair, the maximum is 10. Every LANE cloud (one or multiple ELANs) must have at least one LAN emulation configuration server (LECS).

You can configure some emulated LANs with unrestricted membership and some emulated LANs with restricted membership. You can also configure a default emulated LAN, which must have unrestricted membership.

After LANE is configured, you can monitor and maintain the components, as described in the "Monitoring and Maintaining the LANE Components" section on page 13-16.

Creating a LANE Plan and Worksheet

Draw up a plan and a worksheet for your LANE scenario, containing the following information and leaving spaces for the ATM address of each LANE component on each subinterface of each participating router or switch router:

- The component and interface where the LECS will be located.
- The component, interface, and subinterface where the LES and BUS for each emulated LAN will be located. Each emulated LAN has multiple servers for fault-tolerant operation.
- The component, interfaces, and subinterfaces where the clients for each emulated LAN will be located.
- The component and database name of the default database.
- The name of the default emulated LAN (optional).
- The names of the emulated LANs that have unrestricted membership.
- The names of the emulated LANs that have restricted membership.

The last three items in this list are very important; they determine how you set up each emulated LAN in the configuration server database.

Automatic ATM Addressing and Address Templates for LANE Components

The ATM switch router automatically assigns ATM addresses to LANE components using the scheme described in the *Guide to ATM Technology*. You can also override the automatic address assignments using an ATM address template.

You can use ATM address templates in many LANE commands that assign ATM addresses to LANE components or that link client ATM addresses to emulated LANs. Using templates can greatly simplify the use of these commands.

Note E.164-format ATM addresses do not support the use of LANE ATM address templates.

LANE ATM address templates can use two types of wildcards: an asterisk (*) to match any single character, and an ellipsis (...) to match any number of leading or trailing characters.

In LANE, a *prefix template* explicitly matches the prefix but uses wildcards for the end station interface (ESI) and selector fields. An *ESI template* explicitly matches the ESI field but uses wildcards for the prefix and selector fields. Table 13-1 shows how the values of unspecified digits are determined when an ATM address template is used.

Table 13-1 Values of Unspecified Digits in ATM Address Templates

Unspecified Digits In	Value Is
Prefix (first 13 bytes)	Obtained from ATM switch router via Integrated Local Management Interface (ILMI)
ESI (next 6 bytes)	Filled with the slot MAC address[1] plus • 0—LANE Client (LEC) • 1—LANE Server (LES) • 2—LANE broadcast-and-unknown server (BUS) • 3—LANE Configuration Server (LECS)
Selector field (last 1 byte)	Subinterface number, in the range 0 through 255

1. The lowest MAC addresses in the pool addresses assigned to the ATM interface plus a value that indicates the LANE component.

Rules for Assigning Components to Interfaces and Subinterfaces

The following rules apply to assigning LANE components to the major ATM interface and its subinterfaces:

- The LECS always runs on the major interface.

  The assignment of any other component to the major interface is identical to assigning that component to the 0 subinterface.

- The server and the client of the *same* emulated LAN can be configured on the same subinterface.
- Clients of two *different* emulated LANs cannot be configured on the same subinterface.
- Servers of two *different* emulated LANs cannot be configured on the same subinterface.

Note On the ATM switch router, LANE components can be configured only on the multiservice route processor interface or on one of its subinterfaces.

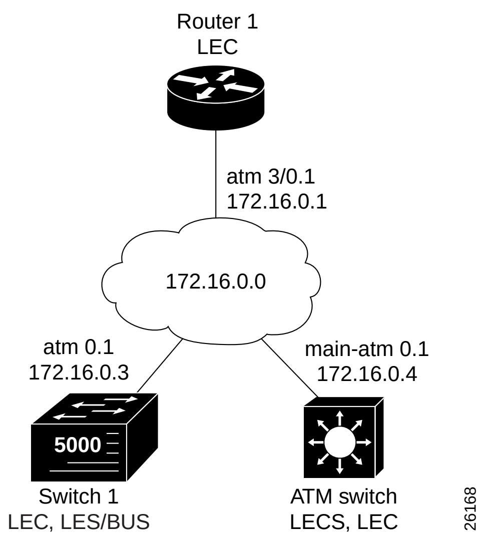
Example LANE Plan and Worksheet

This section is an example of the LANE plan and worksheet that would be created for the example network configuration described in the "Default Configuration for a Single Emulated LAN" section on page 13-17.

Note This example configures LANE on the route processor interface (ATM 0), rather than an ATM router module interface. For LANE client configuration examples on ATM router module interfaces, see "Configuring LECs on ATM Router Module Interfaces (Catalyst 8540 MSR)" section on page 21-11.

Figure 13-2 shows the single emulated LAN example network.

Figure 13-2 LANE Plan Example Network



The following information describes the LANE plan in Figure 13-2:

- LECS:
  —Location: ATM_Switch
  —Interface: atm 0
  —ATM address: 47.00918100000000E04FACB401.00E04FACB405.00
- LES:
  —Location: Switch_1
  —Interface/Subinterface: atm 0.1
  —Type: Ethernet
  —ATM address: 47.00918100000000E04FACB401.00E04FACB403.01
- BUS:
  —Location: Switch_1
  —Interface/Subinterface: atm 0.1
  —Type: Ethernet
  —ATM address: "use default"
- Database:
  —Location: ATM_Switch
  —Name: eng_dbase

—ELAN name: eng_elan
—Default ELAN name: eng_elan
—ATM address: 47.00918100000000E04FACB401.00E04FACB403.01

- LANE Client:
—Location: ATM_Switch
—Interface/Subinterface: atm 0.1
—Server/BUS name: eng_elan
—IP Address/Subnet mask: 172.16.0.4 255.255.0.0
—Type: Ethernet
- LANE Client:
—Location: Switch_1
—Interface/Subinterface: atm 0.1
—Server/BUS name: eng_elan
—Type: Ethernet
- LANE Client:
—Location: Router_1
—Interface/Subinterface: atm 3/0.1
—Server/BUS name: eng_elan
—IP Address/Subnet mask: 172.16.0.1 255.255.0.0
—Type: Ethernet

Note Virtual LANs (VLANs) need to be configured on the LAN edge switches. These VLANs must be mapped to the appropriate ELANs.

Continue with the following sections to start configuring LANE on your ATM network.

Displaying LANE Default Addresses

To make configuration easier, you should display the LANE default addresses for each router or switch router that is running any of the LESs or services and write down the displayed addresses on your worksheet.

To display the default LANE addresses, use the following EXEC command:

Command	Purpose
show lane default-atm-addresses	Displays the LANE default addresses for all ATM interfaces present on the router or switch router.

Example

The following example displays the default LANE addresses:

```
Switch# show lane default-atm-addresses
interface ATM13/0/0:
LANE Client:        47.00918100000000E04FACB401.00E04FACB402.**
LANE Server:        47.00918100000000E04FACB401.00E04FACB403.**
LANE Bus:           47.00918100000000E04FACB401.00E04FACB404.**
LANE Config Server: 47.00918100000000E04FACB401.00E04FACB405.00
note: ** is the subinterface number byte in hex
```

Entering the ATM Address of the Configuration Server

You must enter the configuration server ATM address into the ATM switch routers and save it permanently, so that the value is not lost when the device is reset or powered off. The configuration server address can be specified for all of the ATM switch routers, or per port.

To enter the configuration server addresses for all of the ATM switch routers, perform the following steps in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **atm lecs-address-default** *lecsaddress*	Specifies the LECS ATM address for all of the ATM switch routers.
Step 2	Switch(config)# **end** Switch#	Exits configuration mode.
Step 3	Switch# **copy system:running-config nvram:startup-config**	Saves the configuration.

For examples of these commands, see the "LANE Configuration Examples" section on page 13-17.

Setting Up the Configuration Server Database

After you have determined all LESs, BUSs, and LECS on all ATM subinterfaces on all routers and switch routers that will participate in LANE, and have displayed their ATM addresses, you can use the information to populate the configuration server's database.

You can set up a default emulated LAN, whether or not you set up any other emulated LANs. You can also set up some emulated LANs with restricted membership and others with unrestricted membership.

To set up the LANE database, complete the tasks in the following subsections as appropriate for your emulated LAN plan and scenario. To set up fault-tolerant operation, see the "Configuring Fault-Tolerant Operation" section on page 13-15.

Setting Up the Database for the Default Emulated LAN Only

When you configure a router as the LECS for one default emulated LAN, you provide the following information:

- A name for the database
- The ATM address of the server for the emulated LAN
- The ring number of the emulated LAN for Token Ring (Catalyst 8510 MSR and LightStream 1010)
- A default name for the emulated LAN

When you set up a database of only a default unrestricted emulated LAN, you do not have to specify where the LANE *clients* are located. That is, when you set up the configuration servers database for a single default emulated LAN, you do not have to provide any database entries that link the ATM addresses of any clients with the emulated LAN name.

To set up the LECS for the default emulated LAN, perform the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **lane database** *database-name* Switch(lane-config-database)#	Creates a named database for the LECS.
Step 2	Switch(lane-config-database)# **name** *elan-name* **server-atm-address** *atm-address* [**index** *n*]	In the configuration database, binds the name of the emulated LAN to the ATM address of the LES.
Step 3	Switch(lane-config-database)# **name** *elan-name* **local-seg-id** *seg-num*	(Token Ring only.) In the configuration database, specifies the ring number for the emulated LAN. (Catalyst 8510 MSR and LightStream 1010)
Step 4	Switch(lane-config-database)# **default-name** *elan-name*	In the configuration database, assigns an emulated LAN to the LECS trying to join without specifying an ELAN name.

In Step 2, enter the ATM address of the server for the specified emulated LAN, as noted in your worksheet and obtained in the “Displaying LANE Default Addresses” section on page 13-6. You can have any number of servers per emulated LAN for fault tolerance. Entry order determines priority: the first entry has the highest priority unless you override it with the index option.

If you are setting up only a default emulated LAN, the *elan-name* value in Step 2 is the same as the default emulated LAN name you provide in Step 4.

To set up fault-tolerant operation, see the “Configuring Fault-Tolerant Operation” section on page 13-15.

For examples of these commands, see the “LANE Configuration Examples” section on page 13-17.

Setting Up the Database for Unrestricted-Membership Emulated LANs

When you set up a database for unrestricted emulated LANs, you create database entries that link the name of each emulated LAN to the ATM address of its *server.*

However, you can choose *not* to specify the locations of the LANE clients. That is, when you set up the configuration server database, you do not have to provide any database entries that link the ATM addresses or media access control (MAC) addresses of any *clients* with the emulated LAN name.

To configure a router or switch router as the LECS for multiple emulated LANs with unrestricted membership, perform the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **lane database** *database-name* Switch(lane-config-database)#	Creates a named database for the LECS.
Step 2	Switch(lane-config-database)# **name** *elan-name1* **server-atm-address** *atm-address* [**index** *n*]	In the configuration database, binds the name of the first emulated LAN to the ATM address of the LES for that emulated LAN.

	Command	Purpose
Step 3	Switch(lane-config-database)# **name** *elan-name1* **local-seg-id** *seg-num*	(Token Ring only.) In the configuration database, specifies the ring number for the first emulated LAN. (Catalyst 8510 MSR and LightStream 1010)
Step 4	Switch(lane-config-database)# **name** *elan-name2* **server-atm-address** *atm-address* [**index** *n*]	In the configuration database, binds the name of the second emulated LAN to the ATM address of the LES. Repeat this step, providing a different emulated LAN name and an ATM address, for each additional emulated LAN in this switch cloud.
Step 5	Switch(lane-config-database)# **name** *elan-name2* **local-seg-id** *seg-num*	(Token Ring only) In the configuration database, specifies the ring number for the second emulated LAN. (Catalyst 8510 MSR and LightStream 1010) Repeat this step for each additional Token Ring emulated LAN.
Step 6	Switch(lane-config-database)# **default name** *elan-name1*	Specifies a default emulated LAN for LANE clients not explicitly bound to an emulated LAN. (Optional)

In Steps 2 and 4, enter the ATM address of the server for the specified emulated LAN, as noted in your worksheet and obtained in the "Displaying LANE Default Addresses" section on page 13-6.

To set up fault-tolerant operation, see the "Configuring Fault-Tolerant Operation" section on page 13-15.

For examples of these commands, see the "LANE Configuration Examples" section on page 13-17.

Setting Up the Database for Restricted-Membership Emulated LANs

When you set up the database for restricted-membership emulated LANs, you create database entries that link the name of each emulated LAN to the ATM address of its *server.* However, you also must specify where the LANE clients are located. That is, for each restricted-membership emulated LAN, you provide a database entry that explicitly links the ATM address or MAC address of each *client* of that emulated LAN with the name of that emulated LAN.

When clients for the same restricted-membership emulated LAN are located in multiple routers, each client's ATM address or MAC address must be linked explicitly with the name of the emulated LAN. As a result, you must configure as many client entries (See Step 7 in the following procedure) as you have clients for emulated LANs in all the routers. Each client will have a different ATM address in the database entries.

To set up the configuration server for emulated LANs with restricted membership, perform the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **lane database** *database-name* Switch(lane-config-database)#	Creates a named database for the LECS.
Step 2	Switch(lane-config-database)# **name** *elan-name1* **server-atm-address** *atm-address* [**index** *n*]	In the configuration database, binds the name of the first emulated LAN to the ATM address of the LES for that emulated LAN.
Step 3	Switch(lane-config-database)# **name** *elan-name1* **local-seg-id** *seg-num*	(Token Ring only) In the configuration database, specifies the ring number for the first emulated LAN. (Catalyst 8510 MSR and LightStream 1010)
Step 4	Switch(lane-config-database)# **name** *elan-name2* **server-atm-address** *atm-address* [**index** *n*]	In the configuration database, binds the name of the second emulated LAN to the ATM address of the LES. Repeat this step, providing a different name and a different ATM address for each additional emulated LAN.
Step 5	Switch(lane-config-database)# **name** *elan-name2* **local-seg-id** *seg-num*	(Token Ring only.) In the configuration database, specifies the ring number for the second emulated LAN. (Catalyst 8510 MSR and LightStream 1010) Repeat this step for each additional Token Ring emulated LAN.
Step 6	Switch(lane-config-database)# **default-name** *elan-name1*	(Optional.) Specifies a default emulated LAN for LANE clients not explicitly bound to an emulated LAN.
Step 7	Switch(lane-config-database)# **client-atm-address** *atm-address-template* **name** *elan-name*	Adds a database entry associating a specific client's ATM address with a specific restricted-membership emulated LAN. Repeat this step for each client of each restricted-membership emulated LANs on this switch cloud, in each case specifying that client's ATM address and the name of the emulated LAN with which it is linked.

To set up fault-tolerant operation, see the "Configuring Fault-Tolerant Operation" section on page 13-15.

Enabling the Configuration Server

After you create the database entries appropriate to the type and to the membership conditions of the emulated LANs, you enable the configuration server on the selected ATM interface, router, or switch router, and specify that the configuration server's ATM address is to be computed automatically.

To enable the configuration server, perform the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **interface atm 0**[**.***subinterface#* [**multipoint**]] Switch(config-if)#	If you are not currently configuring the interface, specifies the major ATM interface where the configuration server is located.
Step 2	Switch(config-if)# **lane config database** *database-name*	Links the configuration server's database name to the specified major interface, and enables the configuration server.
Step 3	Switch(config-if)# **lane config auto-config-atm-address**	Specifies that the configuration server's ATM address will be computed by our automatic method.

For examples of these commands, see the "LANE Configuration Examples" section on page 13-17.

Note Since the 12.0(1a)W5(5b) release of the system software, addressing the interface on the Catalyst 8510 MSR and LightStream 1010 processor card has changed. The ATM interface is now called atm0, and the Ethernet interface is now called ethernet0. The old formats (atm 2/0/0 and ethernet 2/0/0) are still supported.

Setting Up LESs and Clients

For each device that participates in LANE, set up the necessary servers and clients for each emulated LAN; then display and record the server and client ATM addresses. Be sure to keep track of the router or switch router interface where the LECS will be located.

For one default emulated LAN, you must set up one set of servers: one as a primary server and the rest as backup servers for the same emulated LAN. For multiple emulated LANs, you can set up servers for another emulated LAN on a different subinterface or on the same interface of this router or switch router, or you can place the servers on a different router.

When you set up a server and BUS on a router, you can combine them with a client on the same subinterface, a client on a different subinterface, or no client at all on the router.

Where you put the clients is important, because any router with clients for multiple emulated LANs can route frames between those emulated LANs.

Note For Token Ring LANE environments that source-route bridge IP traffic to the ATM switch routers, multiring must be configured to enable Routing Information Field (RIF) packets. For an example, see the "Default Configuration for a Token Ring ELAN with IP Source Routing (Catalyst 8510 MSR and LightStream 1010)" section on page 13-31.

Setting Up the Server, BUS, and a Client on a Subinterface

To set up the server, BUS, and (optionally) clients for an emulated LAN, perform the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **interface atm 0.***subinterface#* [**multipoint**] Switch(config-subif)#	Specifies the subinterface for the first emulated LAN on this router.
Step 2	Switch(config-subif)# **lane server-bus** {**ethernet** | **tokenring**} *elan-name1*	Enables a LES and a LANE BUS for the first emulated LAN. (The **tokenring** option is not supported on the Catalyst 8540 MSR.)
Step 3	Switch(config-subif)# **lane client** {**ethernet** | **tokenring**} [*elan-name1*]	(Optional.) Enables a LANE client for the first emulated LAN. (The **tokenring** option is not supported on the Catalyst 8540 MSR.)
Step 4	Switch(config-subif)# **ip address** *ip-address mask*	Provides a protocol address for the client.

If the emulated LAN in Step 2 will have *restricted membership*, consider carefully whether you want to specify its name here. You will specify the name in the LECS's database when it is set up. However, if you link the client to an emulated LAN, and by some mistake it does not match the database entry linking the client to an emulated LAN, this client will not be allowed to join this or any other emulated LAN.

If you do decide to include the name of the emulated LAN linked to the client in Step 3 and later want to associate that client with a different emulated LAN, make the change in the configuration server's database before you make the change for the client on this subinterface.

Each emulated LAN is a separate subnetwork. In Step 4, make sure that the clients of the same emulated LAN are assigned protocol addresses on the same subnetwork, and that clients of different emulated LANs are assigned protocol addresses on different subnetworks.

For examples of these commands, see the "LANE Configuration Examples" section on page 13-17.

Setting Up a Client on a Subinterface

On any given router or switch router, you can set up one client for one emulated LAN or multiple clients for multiple emulated LANs without a server and BUS. You can set up a client for a given emulated LAN on any routers you select to participate in that emulated LAN. Any router with clients for multiple emulated LANs can route packets among those emulated LANs.

To set up a client for an emulated LAN, perform the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **interface atm 0.***subinterface#* [**multipoint**] Switch(config-subif)# or Switch(config)# **interface atm** *card*/*subcard*/*port***.***subinterface#* [**multipoint**] Switch(config-subif)#	Specifies the route processor subinterface number for an emulated LAN on this router. If you are using the optional ATM router module, specifies the ATM subinterface number. (Catalyst 8540 MSR)
Step 2	Switch(config-subif)# **ip address** *ip-address*	Provides a protocol address for the client on this subinterface.
Step 3	Switch(config-subif)# **lane client** {**ethernet** | **tokenring**} *elan-name1*	Enables a LANE client for the first emulated LAN. (The **tokenring** option is not supported on the Catalyst 8540 MSR.)

Note To route traffic between an emulated LAN and a Fast Ethernet (FE) or Gigabit Ethernet (GE) interface, you must configure the LANE client on an ATM router module interface rather than a route processor interface.

Each emulated LAN is a separate subnetwork. In Step 2, make sure that the clients of the same emulated LAN are assigned protocol addresses on the same subnetwork, and that clients of different emulated LANs are assigned protocol addresses on different subnetworks.

Note For Token Ring LANE environments that source-route bridge IP traffic to the ATM switch routers, multiring must be configured to enable Routing Information Field (RIF) packets. For an example, see the "Default Configuration for a Token Ring ELAN with IP Source Routing (Catalyst 8510 MSR and LightStream 1010)" section on page 13-31.

Example (Catalyst 8540 MSR)

The following example shows how to configure a client for emulated LAN on an ATM router module subinterface:

```
Switch(config)# interface atm 10/0/1.1
Switch(config-if)# ip address 172.16.4.0 255.255.0.0
Switch(config-if)# lane client ethernet elan_1205
```

For additional examples of these commands, see the "LANE Configuration Examples" section on page 13-17.

Configuring a LAN Emulation Client on the ATM Switch Router

This section explains how to configure a LANE client connection from the ATM switch router in the headquarters building to the route processor interface (or optional ATM router module interface on the Catalyst 8540 MSR) of the ATM switch router.

Note This connection can be used for switch router management only.

A route processor (or optional ATM router module interface) configured as a LANE client allows you to configure the ATM switch router from a remote host.

Configuring an Ethernet LANE Client

To configure the route processor interface (or optional ATM router module interface on the Catalyst 8540 MSR) as an Ethernet LANE client on the ATM switch router, perform the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **atm lecs-address** *lecsaddress*	Specifies the address to the LECS.
Step 2	Switch(config)# **interface atm 0**[**.***subinterface#* [**multipoint**]] Switch(config-if)#	Specifies the route processor interface.
	or	
	Switch(config)# **interface atm** *card/subcard/port*[**.***subinterface#* [**multipoint**]] Switch(config-if)#	If you are using the optional ATM router module, specifies the ATM interface number. (Catalyst 8540 MSR)
Step 3	Switch(config-if)# **lane client-atm-address** *atm-address-template*	Specifies an ATM address, and overrides the automatic ATM address assignment for the LANE client.
Step 4	Switch(config-if)# **lane client ethernet** [*elan-name*]	Configures a LANE client on the specified subinterface.

Note To route traffic between an emulated LAN and a Fast Ethernet (FE) or Gigabit Ethernet (GE) interface, you must configure the LANE client on an ATM router module interface rather than a route processor interface.

Example

The following example shows how to specify the LANE configuration server (LECS) address and configure a LANE client on the route processor interface to emulate an Ethernet connection using the automatic ATM address assignment:

```
Switch(config)# atm lecs-address 47.0091.0000.0000.0000.0000.0000.0000.00
Switch(config)# interface atm 0
Switch(config-if)# lane client ethernet eng_elan
```

For additional examples of these commands, see the "LANE Configuration Examples" section on page 13-17. For LANE client configuration examples on ATM router module interfaces, see "Configuring LECs on ATM Router Module Interfaces (Catalyst 8540 MSR)" section on page 21-11.

Configuring Fault-Tolerant Operation

The LANE simple server redundancy feature creates fault tolerance using standard LANE protocols and mechanisms. If a failure occurs on the LECS or on the LES/BUS, the emulated LAN can continue to operate using the services of a backup LES. This protocol is called the Simple Server Redundancy Protocol (SSRP).

For a detailed description of SSRP for LANE, refer to the *Guide to ATM Technology*.

Enabling Redundant LECSs and LES/BUSs

To enable fault tolerance, you enable multiple, redundant, and standby LECSs and multiple, redundant, and standby LES/BUSs. This allows the connected LANE components to obtain the global list of LECS addresses. Our LANE continues to operate seamlessly with other vendors' LANE components, but fault tolerance is not effective when other vendors' LANE components are present.

To configure multiple LES/BUSs for emulated LANs on the routers or switch routers, perform the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **lane database** *database-name* Switch(lane-config-database)#	Creates a named database for the LECS.
Step 2	Switch(lane-config-database)# **name** *elan-name* **server-atm-address** *address* **index** *n*	Specifies redundant LES/BUSs, or simple server replication. Enter the command for each LES address for the same emulated LAN. The index determines the priority. The 0 is the highest priority.
Step 3	Switch(lane-config-database)# **lane client** {**ethernet** | **tokenring**} *elan-name*	Enables a LANE client for the first emulated LAN. (The **tokenring** option is not supported on the Catalyst 8540 MSR.)

Server redundancy guards against the failure of the hardware on which LES components are running. This includes all the ATM interface cards in our routers and Catalyst switches. Fault tolerance is not effective for ATM network or switch router failures.

Caution For server redundancy to work correctly, all ATM switch routers must have identical lists of the global LECS addresses, in the identical priority order. The operating LECSs must use exactly the same configuration database.

Load the configuration table data using the **configure network** command. This method minimizes errors and enables the database to be maintained centrally in one place.

For examples of these commands, see the "LANE Configuration Examples" section on page 13-17.

Implementation Considerations

For important considerations when implementing SSRP, refer to the LANE discussion in the *Guide to ATM Technology*.

Caution You can override the LECS address on any subinterface by using the **lane auto-config-atm-address**, **lane fixed-config-atm-address**, and **lane config-atm-address** commands. When you perform an override using one of these commands, however, fault-tolerant operation cannot be guaranteed. To avoid affecting the fault-tolerant operation, do not override any LECS, LES, or BUS addresses.

Monitoring and Maintaining the LANE Components

After configuring LANE components on an interface or any of its subinterfaces, on a specified subinterface, or on an emulated LAN, you can display their status. To show LANE information, use the following EXEC commands:

Command	Purpose
show lane [**interface atm** *card*/*subcard*/*port*[**.***subinterface#*] | **name** *elan-name*] [**brief**]	Displays the global and per-virtual channel connection LANE information for all the LANE components and emulated LANs configured on an interface or any of its subinterfaces.
show lane bus [**interface atm** *card*/*subcard*/*port*[**.***subinterface#*] | **name** *elan-name*] [**brief**]	Displays the global and per-VCC LANE information for the BUS configured on any subinterface or emulated LAN.
show lane client [**interface atm** *card*/*subcard*/*port*[**.***subinterface#*] | **name** *elan-name*] [**brief**]	Displays the global and per-VCC LANE information for all LANE clients configured on any subinterface or emulated LAN.
show lane config [**interface atm** *card*/*subcard*/*port*[**.***subinterface#*]]	Displays the global and per-VCC LANE information for the configuration server configured on any interface.
show lane database [*name*]	Displays the LECS's database.
show lane le-arp [**interface atm** *card*/*subcard*/*port*[**.***subinterface#*] | **name** *elan-name*]	Displays the LANE ARP table of the LANE client configured on the specified subinterface or emulated LAN.
show lane server [**interface atm** *card*/*subcard*/*port*[**.***subinterface#*] | **name** *elan-name*] [**brief**]	Displays the global and per-VCC LANE information for the LES configured on a specified subinterface or emulated LAN.

LANE Configuration Examples

The examples in the following sections illustrate how to configure LANE for the following cases:

- Default configuration for a single emulated LAN with a LANE client on the ATM switch router
- Default configuration for a single emulated LAN with a backup LECS and LES on the ATM switch router
- Default configuration for a single emulated Token Ring LAN using IP source routing across a source-route bridged network with a LANE client on the ATM switch router

All examples use the automatic ATM address assignment method described in the "Automatic ATM Addressing and Address Templates for LANE Components" section on page 13-3.

These examples show the LANE configurations, not the process of determining the ATM addresses and entering them.

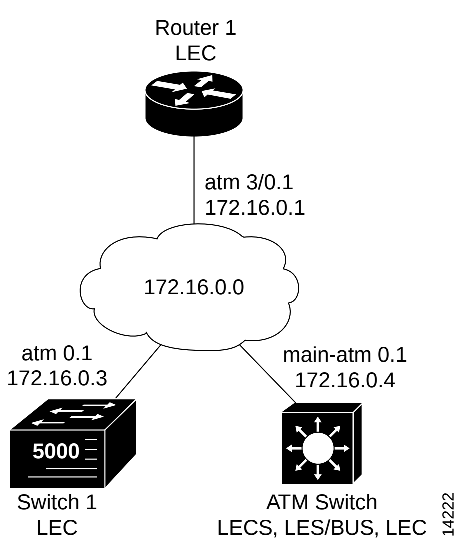
Note For LANE client configuration examples on ATM router module interfaces, see "Configuring LECs on ATM Router Module Interfaces (Catalyst 8540 MSR)" section on page 21-11.

Default Configuration for a Single Emulated LAN

The following examples show how to configure one Cisco 7505 router, one ATM switch, and one Catalyst 5500 switch for a single emulated LAN. Configurations for both Ethernet and Token Ring emulated LANs are shown.

The ATM switch contains the LECS, LES, BUS, and an LEC. The router and Catalyst 5500 switch each contain an LEC for the emulated LAN. This example uses all LANE default settings. For example, it does not explicitly set ATM addresses for the different LANE components that are colocated on the ATM switch. Membership in this emulated LAN is not restricted (see Figure 13-3).

Figure 13-3 Single Emulated LAN Example Network



Ethernet Example

ATM Switch

```
ATM_Switch# show lane default-atm-addresses
interface ATM13/0/0:
LANE Client:        47.00918100000000E04FACB401.00E04FACB402.**
LANE Server:        47.00918100000000E04FACB401.00E04FACB403.**
LANE Bus:           47.00918100000000E04FACB401.00E04FACB404.**
LANE Config Server: 47.00918100000000E04FACB401.00E04FACB405.00
note: ** is the subinterface number byte in hex

ATM_Switch# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
ATM_Switch(config)# atm lecs-address-default 47.00918100000000E04FACB401.00E04FACB405.00
ATM_Switch(config)# end
ATM_Switch#
ATM_Switch# copy system:running-config nvram:startup-config
Building configuration...
[OK]
ATM_Switch# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
ATM_Switch(config)# lane database eng_dbase
ATM_Switch(lane-config-database)# name eng_elan server-atm-address
47.00918100000000E04FACB401.00E04FACB403.01
ATM_Switch(lane-config-database)# default-name eng_elan
ATM_Switch(lane-config-database)# end
ATM_Switch# show lane database

LANE Config Server database table 'eng_dbase'
default elan: eng_elan
elan 'eng_elan': un-restricted
  server 47.00918100000000E04FACB401.00E04FACB403.01 (prio 0)

ATM_Switch# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
ATM_Switch(config)# interface atm 0
ATM_Switch(config-if)# lane config database eng_dbase
ATM_Switch(config-if)# lane config auto-config-atm-address
ATM_Switch(config-if)# exit
ATM_Switch(config)# end
ATM_Switch# show lane config
LE Config Server ATM13/0/0 config table: eng_dbase
Admin: up  State: operational
LECS Mastership State: active master
list of global LECS addresses (42 seconds to update):
47.00918100000000E04FACB401.00E04FACB405.00  <-------- me
ATM Address of this LECS: 47.00918100000000E04FACB401.00E04FACB405.00 (auto)
cumulative total number of unrecognized packets received so far: 0
cumulative total number of config requests received so far: 0
cumulative total number of config failures so far: 0

ATM_Switch# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
ATM_Switch(config)# interface atm 0.1 multipoint
ATM_Switch(config-subif)# lane server-bus ethernet eng_elan
ATM_Switch(config-subif)# ip address 172.16.0.4 255.255.0.0
ATM_Switch(config-subif)# end
```

```
ATM_Switch# show lane
LE Config Server ATM13/0/0 config table: eng_dbase
Admin: up  State: operational
LECS Mastership State: active master
list of global LECS addresses (46 seconds to update):
47.00918100000000E04FACB401.00E04FACB405.00  <-------- me
ATM Address of this LECS: 47.00918100000000E04FACB401.00E04FACB405.00 (auto)
 vcd  rxCnt  txCnt  callingParty
  82      0      0  47.00918100000000E04FACB401.00E04FACB403.01 LES eng_elan 0 active
cumulative total number of unrecognized packets received so far: 0
cumulative total number of config requests received so far: 0
cumulative total number of config failures so far: 0

LE Server ATM13/0/0.1  ELAN name: eng_elan  Admin: up  State: operational
type: ethernet         Max Frame Size: 1516
ATM address: 47.00918100000000E04FACB401.00E04FACB403.01
LECS used: 47.00918100000000E04FACB401.00E04FACB405.00 connected, vcd 81

LE BUS ATM13/0/0.1  ELAN name: eng_elan  Admin: up  State: operational
type: ethernet         Max Frame Size: 1516
ATM address: 47.00918100000000E04FACB401.00E04FACB404.01

ATM_Switch# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
ATM_Switch(config)# interface atm 0.1 multipoint
ATM_Switch(config-subif)# lane client ethernet eng_elan
ATM_Switch(config-subif)# end
ATM_Switch# show lane client
LE Client ATM13/0/0.1  ELAN name: eng_elan  Admin: up  State: operational
Client ID: 1                 LEC up for 30 seconds
ELAN ID: 0
Join Attempt: 1
HW Address: 00e0.4fac.b402   Type: ethernetMax Frame Size: 1516
ATM Address: 47.00918100000000E04FACB401.00E04FACB402.01

 VCD  rxFrames  txFrames  Type       ATM Address
   0         0         0  configure  47.00918100000000E04FACB401.00E04FACB405.00
  87         1         2  direct     47.00918100000000E04FACB401.00E04FACB403.01
  90         1         0  distribute 47.00918100000000E04FACB401.00E04FACB403.01
  91         0         1  send       47.00918100000000E04FACB401.00E04FACB404.01
  94         0         0  forward    47.00918100000000E04FACB401.00E04FACB404.01

ATM_Switch# copy system:running-config nvram:startup-config
Building configuration...
[OK]
ATM_Switch#
```

Note The ELAN ID shown in the above **show lane client** command display is relevant only for LANE version 2-capable clients. The ELAN ID is configured with either the **name** *elan-name* command in database configuration mode, or the **lane server-bus** command in subinterface configuration mode.

Router 1

```
router1# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
router1(config)# interface atm 3/0
router1(config-if)# atm pvc 1 0 5 qsaal
router1(config-if)# atm pvc 2 0 16 ilmi
router1(config-if)# interface atm 3/0.1
router1(config-subif)# ip address 172.16.0.1 255.255.0.0
router1(config-subif)# lane client ethernet eng_elan
router1(config-subif)# end
router1# more system:running-config
Building configuration...

Current configuration:
!
version 11.1

<Information deleted>

!
interface ATM3/0
 no ip address
 atm pvc 1 0 5 qsaal
 atm pvc 2 0 16 ilmi
!
interface ATM3/0.1 midpoint
 lane client ethernet eng_elan
!

<information deleted>

!
end

router1# show interfaces atm 3/0.1
ATM3/0.1 is up, line protocol is up
 Hardware is Caxias ATM
  MTU 1500 bytes, BW 156250 Kbit, DLY 80 usec, rely 255/255, load 1/255
  Encapsulation ATM-LANE
  ARP type: ARPA, ARP Timeout 04:00:00
router1#
```

Catalyst 5500 Switch 1

```
Switch1> session 4
Trying ATM-4...
Connected to ATM-4.
Escape character is '^]'.
ATM> enable
ATM# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
ATM(config)# interface atm 0
ATM(config-if)# lane server-bus ethernet eng_elan
ATM(config-if)# end
ATM# copy system:running-config nvram:startup-config
Building configuration...
[OK]
ATM# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
ATM(config)# interface atm 0
ATM(config-if)# atm pvc 1 0 5 qsaal
ATM(config-if)# atm pvc 2 0 16 ilmi
ATM(config-if)# end
ATM#
ATM# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
ATM(config)# interface atm 0.1 multipoint
ATM(config-subif)# lane client ethernet 1 eng_elan
ATM(config-subif)# end
ATM# show lane client
LE Client ATM0.1  ELAN name: eng_elan  Admin: up  State: operational
Client ID: 3                 LEC up for 24 seconds
Join Attempt: 11
HW Address: 00e0.4fac.b030  Type: ethernetMax Frame Size: 1516        VLANID: 1

ATM Address: 47.00918100000000E04FACB401.00E04FACB030.01

 VCD  rxFrames  txFrames  Type       ATM Address
   0         0         0  configure  47.00918100000000E04FACB401.00E04FACB405.00
  27         1        14  direct     47.00918100000000E04FACB401.00E04FACB403.01
  29        13         0  distribute 47.00918100000000E04FACB401.00E04FACB403.01
  30         0        15  send       47.00918100000000E04FACB401.00E04FACB404.01
  31         0         0  forward    47.00918100000000E04FACB401.00E04FACB404.01

ATM# copy system:running-config nvram:startup-config
Building configuration...
[OK]
ATM#
```

Confirming Connectivity between the ATM Switch and Other LANE Members

The following example shows how to use the **show lane** and **ping** commands to confirm the connection between the ATM switch, routers, and LAN switches.

ATM Switch

```
Switch# show lane
LE Config Server ATM13/0/0 config table: eng_dbase
Admin: up  State: operational
LECS Mastership State: active master
list of global LECS addresses (31 seconds to update):
47.00918100000000E04FACB401.00E04FACB405.00  <-------- me
ATM Address of this LECS: 47.00918100000000E04FACB401.00E04FACB405.00 (auto)
 vcd  rxCnt  txCnt  callingParty
  82      2      2  47.00918100000000E04FACB401.00E04FACB403.01 LES eng_elan 0 active
cumulative total number of unrecognized packets received so far: 0
cumulative total number of config requests received so far: 4
cumulative total number of config failures so far: 0

LE Server ATM13/0/0.1  ELAN name: eng_elan  Admin: up  State: operational
type: ethernet         Max Frame Size: 1516
ATM address: 47.00918100000000E04FACB401.00E04FACB403.01
LECS used: 47.00918100000000E04FACB401.00E04FACB405.00 connected, vcd 81
control distribute: vcd 89, 2 members, 2 packets

proxy/ (ST: Init, Conn, Waiting, Adding, Joined, Operational, Reject, Term)
lecid ST vcd    pkts Hardware Addr  ATM Address
   1  O   88       2 00e0.4fac.b402 47.00918100000000E04FACB401.00E04FACB402.01
   2  O   96       2 0080.1c93.8060 47.00918100000000E04FACB401.00801C938060.01

LE BUS ATM13/0/0.1  ELAN name: eng_elan  Admin: up  State: operational
type: ethernet         Max Frame Size: 1516
ATM address: 47.00918100000000E04FACB401.00E04FACB404.01
data forward: vcd 93, 2 members, 95 packets, 0 unicasts

lecid  vcd     pkts   ATM Address
    1   92       95 47.00918100000000E04FACB401.00E04FACB402.01
    2   97       42 47.00918100000000E04FACB401.00801C938060.01

LE Client ATM13/0/0.1  ELAN name: eng_elan  Admin: up  State: operational
Client ID: 1                 LEC up for 1 hour 34 minutes 46 seconds
ELAN ID: 0
Join Attempt: 1
HW Address: 00e0.4fac.b402   Type: ethernetMax Frame Size: 1516
ATM Address: 47.00918100000000E04FACB401.00E04FACB402.01

 VCD  rxFrames  txFrames  Type       ATM Address
   0         0         0  configure  47.00918100000000E04FACB401.00E04FACB405.00
  87         1         2  direct     47.00918100000000E04FACB401.00E04FACB403.01
  90         2         0  distribute 47.00918100000000E04FACB401.00E04FACB403.01
  91         0        95  send       47.00918100000000E04FACB401.00E04FACB404.01
  94        42         0  forward    47.00918100000000E04FACB401.00E04FACB404.01

ATM_Switch# ping 172.16.0.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.0.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/202/1000 ms
ATM_Switch# ping 172.16.0.3
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.0.2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/202/1000 ms
```

Token Ring Example (Catalyst 8510 MSR and LightStream 1010)

In this Token Ring example, the Cisco 7505 router contains the LECS, LES, BUS, and an LEC. The ATM switch router and Catalyst 5500 Fast Ethernet switch each contain an LEC for the emulated LAN. This example uses all LANE default settings. For example, it does not explicitly set ATM addresses for the different LANE components that are co-located on the router. Membership in this emulated LAN is not restricted.

Router 1

```
router1# show lane default-atm-addresses
interface ATM3/0:
LANE Client:        47.00918100000000603E7B2001.00000C407572.**
LANE Server:        47.00918100000000603E7B2001.00000C407573.**
LANE Bus:           47.00918100000000603E7B2001.00000C407574.**
LANE Config Server: 47.00918100000000603E7B2001.00000C407575.00
note: ** is the subinterface number byte in hex
```

ATM Switch

```
Switch# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)# atm lecs-address-default 47.00918100000000603E7B2001.00000C407575.00
Switch(config)# end
Switch#
```

Router 1

```
router1# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
router1(config)# lane database eng_dbase
router1(lane-config-database)# name eng_elan server-atm-address
47.00918100000000603E7B2001.00000C407573.01
router1(lane-config-database)# name eng_elan local-seg-id 2048
router1(lane-config-database)# default-name eng_elan
router1(lane-config-database)# exit
router1(config)# interface atm0
router1(config-if)# atm pvc 1 0 5 qsaal
router1(config-if)# atm pvc 2 0 16 ilmi
router1(config-if)# lane config auto-config-atm-address
router1(config-if)# lane config database eng_dbase
router1(config-if)#
%LANE-5-UPDOWN: ATM0 database example1: LE Config Server (LECS) changed state to up
router1(config-if)# interface atm3/0.1
router1(config-subif)# ip address 172.16.0.1 255.255.0.0
router1(config-subif)# lane server-bus tokenring eng_elan
router1(config-subif)# lane client tokenring eng_elan
router1(config-subif)#
%LANE-5-UPDOWN: ATM0.1 elan eng: LE Client changed state to up
router1(config-subif)# end
router1#
```

Catalyst 5000 Switch 1

```
Switch1> session 4
Trying ATM-4...
Connected to ATM-4.
Escape character is '^]'.
ATM> enable
ATM# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
ATM(config)# interface atm 0
ATM(config-if)# lane server-bus tokenring eng_elan
ATM(config-if)# end
ATM# copy system:running-config nvram:startup-config
Building configuration...
[OK]
ATM# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
ATM(config)# interface atm 0
ATM(config-if)# atm pvc 1 0 5 qsaal
ATM(config-if)# atm pvc 2 0 16 ilmi
ATM(config-if)# end
ATM#
ATM# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
ATM(config)# interface atm 0.1 multipoint
ATM(config-subif)# lane client tokenring 1 eng_elan
ATM(config-subif)# end
ATM#
```

ATM Switch

```
Switch# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)# interface atm 0.1 multipoint
Switch(config-subif)# ip address 172.16.0.4 255.255.0.0
Switch(config-subif)# lane client tokenring eng_elan
Switch(config-subif)#
%LANE-5-UPDOWN: ATM13/0/0.1 elan : LE Client changed state to up
Switch(config-subif)# end
Switch#
```

Confirming Connectivity between the ATM switch and the Routers

The following example shows how to use the **ping** command to confirm the connection between the ATM switch and routers:

```
ATM_Switch# ping 172.16.0.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.0.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/202/1000 ms
ATM_Switch# ping 172.16.0.3

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.0.3, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/202/1000 ms
```

Displaying the LANE Client Configuration on the ATM switch

The following example shows the **show lane client** command display for the Ethernet LANE client in the ATM switch:

```
ATM_Switch# show lane client
LE Client ATM13/0/0.1  ELAN name: eng  Admin: up  State: operational
Client ID: 3                 LEC up for 4 minutes 58 seconds
Join Attempt: 1
HW Address: 0060.3e7b.2002   Type: ethernet             Max Frame Size: 1516
ATM Address: 47.00918100000000603E7B2001.00603E7B2002.01

 VCD  rxFrames  txFrames  Type       ATM Address
   0         0         0  configure  47.00918100000000603E7B2001.00000C407575.00
  52         1         4  direct     47.00918100000000603E7B2001.00000C407573.01
  53         9         0  distribute 47.00918100000000603E7B2001.00000C407573.01
  54         0        13  send       47.00918100000000603E7B2001.00000C407574.01
  55        19         0  forward    47.00918100000000603E7B2001.00000C407574.01
  56        11        10  data       47.00918100000000603E7B2001.00000C407572.01
  57         6         5  data       47.00918100000000603E7B2001.00000C407C02.02
```

The following example shows the **show lane client** command display for the Token Ring LANE client in the ATM switch router:

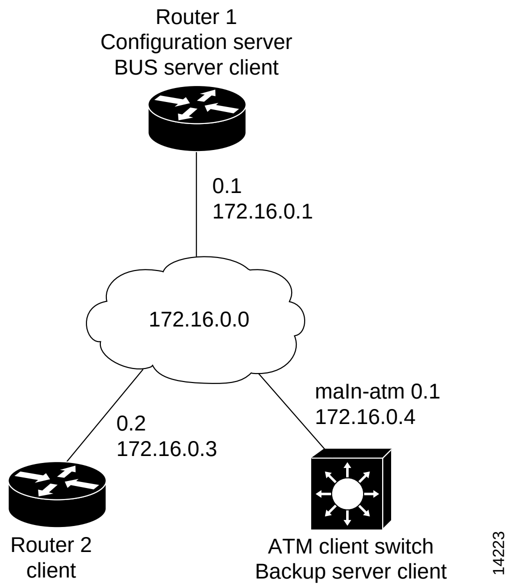
```
ATM_Switch# show lane client
LE Client ATM13/0/0.1  ELAN name: eng  Admin: up  State: operational
Client ID: 3                 LEC up for 4 minutes 58 seconds
Join Attempt: 1
HW Address: 0060.3e7b.2002   Type: token ring           Max Frame Size: 4544
ATM Address: 47.00918100000000603E7B2001.00603E7B2002.01

 VCD  rxFrames  txFrames  Type       ATM Address
   0         0         0  configure  47.00918100000000603E7B2001.00000C407575.00
  52         1         4  direct     47.00918100000000603E7B2001.00000C407573.01
  53         9         0  distribute 47.00918100000000603E7B2001.00000C407573.01
  54         0        13  send       47.00918100000000603E7B2001.00000C407574.01
  55        19         0  forward    47.00918100000000603E7B2001.00000C407574.01
  56        11        10  data       47.00918100000000603E7B2001.00000C407572.01
  57         6         5  data       47.00918100000000603E7B2001.00000C407C02.02
```

Default Configuration for a Single Emulated LAN with Backup LECS and LES on the ATM Switch Router

The following examples show how to configure two Cisco 4500 routers and one ATM switch router for one emulated LAN with fault tolerance. Configurations for both Ethernet and Token Ring emulated LANs are shown.

Router 1 contains the LECS, LES, BUS, and an LEC. Router 2 contains only an LEC. The ATM switch router contains the backup LECS and the backup LES for this emulated LAN, along with another LEC (see Figure 13-4).

Figure 13-4 Single Emulated LAN with Backup LANE Example Network



This example shows how to accept all default settings provided. For example, it does not explicitly set ATM addresses for the different LANE components that are also on the router. Membership in this emulated LAN is not restricted.

Ethernet Example

Router 1

```
router1# show lane default-atm-addresses
interface ATM0:
LANE Client:        47.00918100000000603E7B2001.00000C407572.**
LANE Server:        47.00918100000000603E7B2001.00000C407573.**
LANE Bus:           47.00918100000000603E7B2001.00000C407574.**
LANE Config Server: 47.00918100000000603E7B2001.00000C407575.00
note: ** is the subinterface number byte in hex
```

ATM Switch Router

```
Switch# show lane default-atm-address
interface ATM2/0/0:
LANE Client:        47.00918100000000603E7B2001.00603E7B2002.**
LANE Server:        47.00918100000000603E7B2001.00603E7B2003.**
LANE Bus:           47.00918100000000603E7B2001.00603E7B2004.**
LANE Config Server: 47.00918100000000603E7B2001.00603E7B2005.00
note: ** is the subinterface number byte in hex

Switch# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)# atm lecs-address-default 47.00918100000000603E7B2001.00000C407575.00
Switch(config)# atm lecs-address-default 47.00918100000000603E7B2001.00603E7B2005.00
Switch(config)# end
Switch#
```

Router 1

```
router1# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
router1(config)# lane database example1
router1(lane-config-database)# name eng server-atm-address
47.00918100000000603E7B2001.00000C407573.01
router1(lane-config-database)# name eng server-atm-address
47.00918100000000603E7B2001.00603E7B2003.01
router1(lane-config-database)# default-name eng
router1(lane-config-database)# exit
router1(config)# interface atm 3/0
router1(config-if)# atm pvc 1 0 5 qsaal
router1(config-if)# atm pvc 2 0 16 ilmi
router1(config-if)# lane config auto-config-atm-address
router1(config-if)# lane config database example1
router1(config-if)#
%LANE-5-UPDOWN: ATM0 database example1: LE Config Server (LECS) changed state to up
router1(config-if)# interface atm 3/0.1
router1(config-subif)# ip address 172.16.0.1 255.255.0.0
router1(config-subif)# lane server-bus ethernet eng
router1(config-subif)# lane client ethernet eng
router1(config-subif)#
%LANE-5-UPDOWN: ATM0.1 elan eng: LE Client changed state to up
router1(config-subif)# end
router1#
```

ATM Switch Router

```
Switch# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)# lane database example1_backup
Switch(lane-config-database)# name eng server-atm-address
47.00918100000000603E7B2001.00000C407573.01
Switch(lane-config-database)# name eng server-atm-address
47.00918100000000603E7B2001.00603E7B2003.01
Switch(lane-config-database)# default-name eng
Switch(lane-config-database)# exit
Switch(config)# interface atm 0
Switch(config-if)# lane config auto-config-atm-address
Switch(config-if)# lane config database example1_backup
Switch(config-if)#
%LANE-5-UPDOWN: ATM2/0/0 database example1_backup: LE Config Server (LECS) changed state
to up
%LANE-6-LECS_INFO: ATM2/0/0: started listening on the well known LECS address
%LANE-6-LECS_INFO: LECS on interface ATM2/0/0 became a BACKUP
%LANE-6-LECS_INFO: ATM2/0/0: stopped listening on the well known LECS address
Switch(config-if)# interface atm 0.1 multipoint
Switch(config-subif)# ip address 172.16.0.4 255.255.0.0
Switch(config-subif)# lane server-bus ethernet eng
Switch(config-subif)#
%LANE-5-UPDOWN: ATM2/0/0.1 elan eng: LE Server/BUS changed state to up
Switch(config-subif)# lane client ethernet eng
Switch(config-subif)#
%LANE-5-UPDOWN: ATM2/0/0.1 elan eng: LE Client changed state to up
Switch(config-subif)# end
Switch#
```

Router 2

```
router2# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
router2(config)# interface atm 3/0
router2(config-if)# atm pvc 1 0 5 qsaal
router2(config-if)# atm pvc 2 0 16 ilmi
router2(config-if)# interface atm 3/0.2
router2(config-subif)# ip address 172.16.0.3 255.255.0.0
router2(config-subif)# lane client ethernet eng
router2(config-subif)#
%LANE-5-UPDOWN: ATM0.2 elan : LE Client changed state to up
router2(config-subif)# end
router2#
```

Token Ring Example (Catalyst 8510 MSR and LightStream 1010)

Router 1

```
router1# show lane default-atm-addresses
interface ATM3/0:
LANE Client:        47.00918100000000603E7B2001.00000C407572.**
LANE Server:        47.00918100000000603E7B2001.00000C407573.**
LANE Bus:           47.00918100000000603E7B2001.00000C407574.**
LANE Config Server: 47.00918100000000603E7B2001.00000C407575.00
note: ** is the subinterface number byte in hex
```

ATM Switch

```
Switch# show lane default-atm-address
interface ATM2/0/0:
LANE Client:        47.00918100000000603E7B2001.00603E7B2002.**
LANE Server:        47.00918100000000603E7B2001.00603E7B2003.**
LANE Bus:           47.00918100000000603E7B2001.00603E7B2004.**
LANE Config Server: 47.00918100000000603E7B2001.00603E7B2005.00
note: ** is the subinterface number byte in hex

Switch# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)# atm lecs-address-default 47.00918100000000603E7B2001.00000C407575.00
Switch(config)# atm lecs-address-default 47.00918100000000603E7B2001.00603E7B2005.00
Switch(config)# end
Switch#
```

Router 1

```
router1# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
router1(config)# lane database example1
router1(lane-config-database)# name eng server-atm-address
47.00918100000000603E7B2001.00000C407573.01
router1(lane-config-database)# name eng server-atm-address
47.00918100000000603E7B2001.00603E7B2003.01
router1(lane-config-database)# name eng local-seg-id 2048
router1(lane-config-database)# default-name eng
router1(lane-config-database)# exit
router1(config)# interface atm 3/0
router1(config-if)# atm pvc 1 0 5 qsaal
router1(config-if)# atm pvc 2 0 16 ilmi
router1(config-if)# lane config auto-config-atm-address
router1(config-if)# lane config database example1
router1(config-if)#
%LANE-5-UPDOWN: ATM0 database example1: LE Config Server (LECS) changed state to up
router1(config-if)# interface atm 3/0.1
router1(config-subif)# ip address 172.16.0.1 255.255.0.0
router1(config-subif)# lane server-bus tokenring eng
router1(config-subif)# lane client tokenring eng
router1(config-subif)#
%LANE-5-UPDOWN: ATM0.1 elan eng: LE Client changed state to up
router1(config-subif)# end
router1#
```

ATM Switch

```
Switch# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)# lane database example1_backup
Switch(lane-config-database)# name eng server-atm-address
47.00918100000000603E7B2001.00000C407573.01
Switch(lane-config-database)# name eng server-atm-address
47.00918100000000603E7B2001.00603E7B2003.01
Switch(lane-config-database)# name eng local-seg-id 2048
Switch(lane-config-database)# default-name eng
Switch(lane-config-database)# exit
Switch(config)# interface atm 0
Switch(config-if)# lane config auto-config-atm-address
Switch(config-if)# lane config database example1_backup
Switch(config-if)#
%LANE-5-UPDOWN: ATM2/0/0 database example1_backup: LE Config Server (LECS) changed state
to up
%LANE-6-LECS_INFO: ATM2/0/0: started listening on the well known LECS address
%LANE-6-LECS_INFO: LECS on interface ATM2/0/0 became a BACKUP
%LANE-6-LECS_INFO: ATM2/0/0: stopped listening on the well known LECS address
Switch(config-if)# interface atm 0.1 multipoint
Switch(config-subif)# ip address 172.16.0.4 255.255.0.0
Switch(config-subif)# lane server-bus tokenring eng
Switch(config-subif)#
%LANE-5-UPDOWN: ATM2/0/0.1 elan eng: LE Server/BUS changed state to up
Switch(config-subif)# lane client tokenring eng
Switch(config-subif)#
%LANE-5-UPDOWN: ATM2/0/0.1 elan eng: LE Client changed state to up
Switch(config-subif)# end
Switch#
```

Router 2

```
router2# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
router2(config)# interface atm 3/0
router2(config-if)# atm pvc 1 0 5 qsaal
router2(config-if)# atm pvc 2 0 16 ilmi
router2(config-if)# interface atm 3/0.2
router2(config-subif)# ip address 172.16.0.3 255.255.0.0
router2(config-subif)# lane client tokenring eng
router2(config-subif)#
%LANE-5-UPDOWN: ATM0.2 elan : LE Client changed state to up
router2(config-subif)# end
router2#
```

Displaying the LECS Configuration on the ATM Switch Router

The following example shows the **show lane config** command display for the LECS (Ethernet and Token Ring):

```
Switch# show lane config
LE Config Server ATM2/0/0 config table: example1_backup
Admin: up  State: operational
LECS Mastership State: backup
list of global LECS addresses (45 seconds to update):
47.00918100000000603E7B2001.00000C407575.00 incoming call (vcd 88)
47.00918100000000603E7B2001.00603E7B2005.00  <-------- me
ATM Address of this LECS: 47.00918100000000603E7B2001.00603E7B2005.00 (auto)
 vcd  rxCnt  txCnt  callingParty
  88      0      0  47.00918100000000603E7B2001.00000C407575.00 LECS
cumulative total number of unrecognized packets received so far: 0
cumulative total number of config requests received so far: 0
cumulative total number of config failures so far: 0
```

Displaying the LES Configuration on the ATM Switch Router

The following example shows the **show lane server** command display for the Ethernet LES:

```
Switch# show lane server
LE Server ATM2/0/0.1  ELAN name: eng  Admin: up  State: operational
type: ethernet         Max Frame Size: 1516
ATM address: 47.00918100000000603E7B2001.00603E7B2003.01
LECS used: 47.00918100000000603E7B2001.00000C407575.00 connected, vcd 95
```

The following example shows the **show lane server** command display for the Token Ring LANE server:

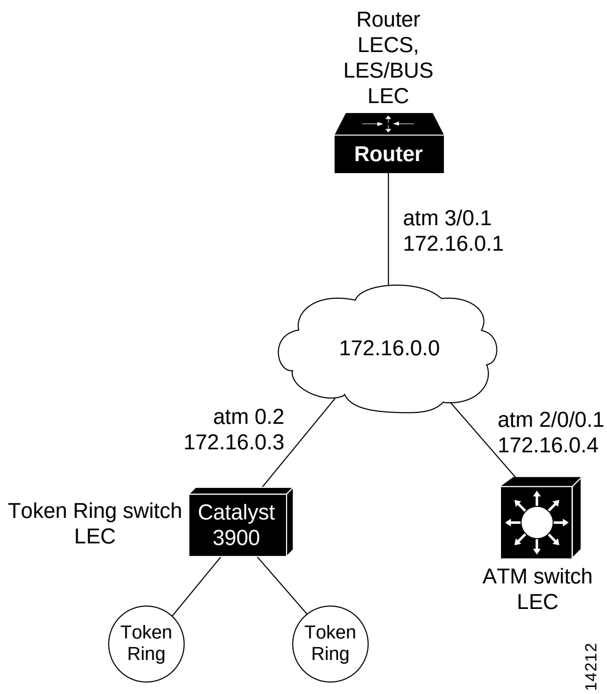
```
Switch# show lane server
LE Server ATM2/0/0.1  ELAN name: eng  Admin: up  State: operational
type: token ring         Max Frame Size: 4544      Segment ID: 2048
ATM address: 47.00918100000000603E7B2001.00603E7B2003.01
LECS used: 47.00918100000000603E7B2001.00000C407575.00 connected, vcd 95
```

Default Configuration for a Token Ring ELAN with IP Source Routing (Catalyst 8510 MSR and LightStream 1010)

The following example shows how to configure a single emulated Token Ring LAN using a Cisco 4500 router and an ATM switch with IP source routing across a source-route bridged network. In this example, the emulated Token Ring LAN is source-route bridged to two physical Token Rings.

The router contains the LECS, LES, BUS, and an LEC. Both the ATM switch and Token Ring switch contain an LEC for the emulated LAN. This example uses all LANE default settings. For example, it does not explicitly set ATM addresses for the different LANE components that are colocated on the router. Membership in this emulated LAN is not restricted (see Figure 13-5).

Figure 13-5 Single Emulated Token Ring LAN with Token Ring Switch



Router

```
router# show lane default-atm-addresses
interface ATM0:
LANE Client:        47.00918100000000603E7B2001.00000C407572.**
LANE Server:        47.00918100000000603E7B2001.00000C407573.**
LANE Bus:           47.00918100000000603E7B2001.00000C407574.**
LANE Config Server: 47.00918100000000603E7B2001.00000C407575.00
note: ** is the subinterface number byte in hex
```

ATM Switch

```
Switch# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)# atm lecs-address-default 47.00918100000000603E7B2001.00000C407575.00
Switch(config)# end
Switch#
```

Router

```
router# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
router(config)# lane database example1
router(lane-config-database)# name eng server-atm-address
47.00918100000000603E7B2001.00000C407573.01
router(lane-config-database)# name eng local-seg-id 2048
router(lane-config-database)# default-name eng
router(lane-config-database)# exit
router(config)# interface atm 3/0
router(config-if)# atm pvc 1 0 5 qsaal
router(config-if)# atm pvc 2 0 16 ilmi
router(config-if)# lane config auto-config-atm-address
router(config-if)# lane config database example1
router(config-if)#
%LANE-5-UPDOWN: ATM0 database example1: LE Config Server (LECS) changed state to up
router(config-if)# interface atm 3/0.1
router(config-subif)# ip address 172.16.0.1 255.255.0.0
router(config-subif)# lane server-bus tokenring eng
router(config-subif)# lane client tokenring eng
router(config-subif)#
%LANE-5-UPDOWN: ATM0.1 elan eng: LE Client changed state to up
router(config-subif)# end
router#
```

ATM Switch

```
Switch# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)# interface atm 0.1 multipoint
Switch(config-subif)# ip address 172.16.0.4 255.255.0.0
Switch(config-subif)# lane client tokenring eng
Switch(config-subif)# multiring ip
Switch(config-subif)#
%LANE-5-UPDOWN: ATM2/0/0.1 elan : LE Client changed state to up
Switch(config-subif)# end
Switch#
```

CHAPTER 14

Configuring ATM Accounting and ATM RMON

This chapter describes the ATM accounting and Remote Monitoring (RMON) features used with the ATM switch router.

Note This chapter provides advanced configuration instructions for the Catalyst 8540 MSR, Catalyst 8510 MSR, and LightStream 1010 ATM switch routers. For complete descriptions of the commands mentioned in this chapter, refer to the *ATM Switch Router Command Reference* publication.

This chapter includes the following sections:

Note The ATM accounting and ATM RMON features both require a minimum of 32 MB of dynamic random access memory (DRAM) installed on the multiservice route processor. If you want to run both ATM accounting and ATM RMON features together, you must have 64 MB of DRAM.

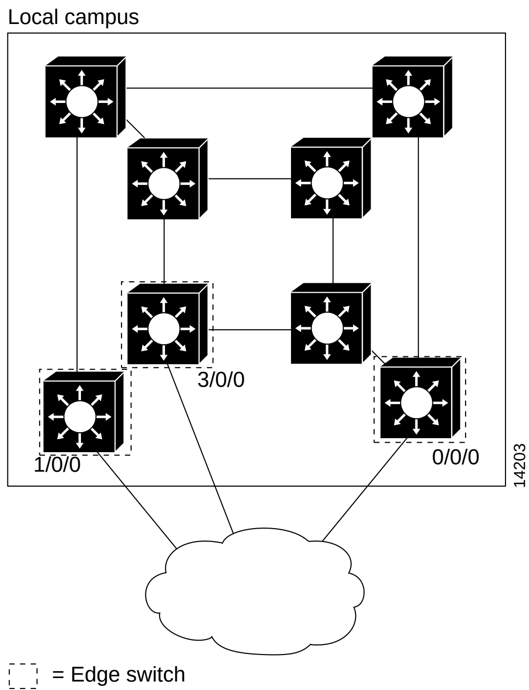
Configuring ATM Accounting

The following sections describe the process used to enable and configure the ATM accounting feature on the ATM switch router:

ATM Accounting Overview

The ATM accounting feature provides accounting and billing services for virtual circuits (VCs) used on the ATM switch router. You enable ATM accounting on an edge switch to monitor call setup and traffic activity. A specific interface can be configured to monitor either incoming or outgoing or incoming and outgoing VC use. Figure 14-1 shows a typical ATM accounting environment.

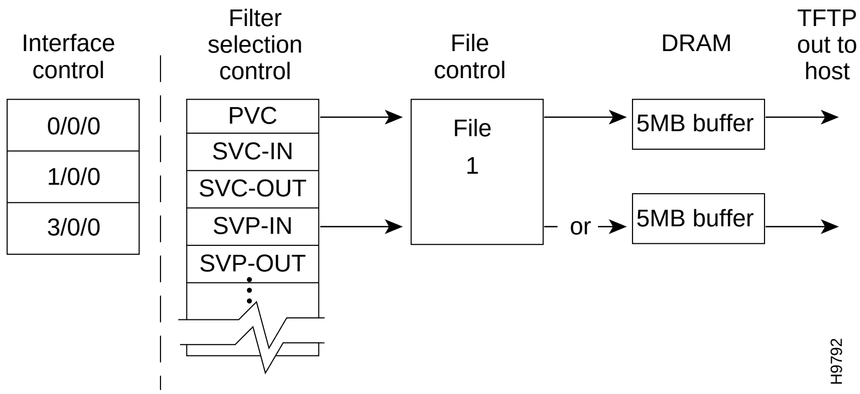
Figure 14-1 ATM Accounting Environment



The edge switches, connected to the exterior Internet, are connections that require monitoring for accounting and billing purposes.

Switching speeds and number of VCs supported by the ATM switch router while monitoring virtual circuit use for accounting purposes can cause the amount of data to be gathered to reach the megabyte range. With such a large amount of data in the ATM accounting files, using traditional Simple Network Management Protocol (SNMP) methods of data retrieval is not feasible. You can store the collected accounting information in a file that you can retrieve using a file transfer protocol. SNMP provides management control of the selection and collection of accounting data. Figure 14-2 shows an interface, filtering, and file configuration example.

Figure 14-2 Interface and File Management for ATM Accounting



A file used for data collection actually corresponds to two memory buffers on the multiservice route processor. One buffer is actively saving data, while the second is passive and ready to have its data either retrieved using Trivial File Transport Protocol (TFTP) or overwritten when the currently active file reaches its maximum capacity. Alternatively, the file can be written to a remotely connected PC over a TCP connection.

Configuring Global ATM Accounting

The ATM accounting feature must be enabled to start gathering ATM accounting virtual circuit call setup and use data. The ATM accounting feature runs in the background and captures configured accounting data for VC changes such as calling party, called party, or start time and connection type information for specific interfaces to a file.

Caution Enabling ATM accounting could slow the basic operation of the ATM switch router.

Note Even when ATM accounting is disabled globally, other ATM accounting commands, both global and for individual interfaces, remain in the configuration file.

To enable the ATM accounting feature, use the following command in global configuration mode:

Command	Purpose
atm accounting enable	Enables ATM accounting for the ATM switch router.

Displaying the ATM Accounting Configuration

To display the ATM accounting status, use the following privileged EXEC command:

Command	Purpose
more system:running-config	Displays the ATM accounting status.

Enabling ATM Accounting on an Interface

After you enable ATM accounting, you must configure specific ingress or egress interfaces, usually on edge switches connected to the external network, to start gathering the ATM accounting data.

To enable ATM accounting on a specific interface, perform the following tasks, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **interface atm** *card*/*subcard*/*port* Switch(config-if)#	Selects the interface to be configured.
Step 2	Switch(config-line)# **privilege level** *number*	Configures the default privilege level.

Example

The following example shows how to enable ATM accounting on ATM interface 1/0/3:

```
Switch(config)# interface atm 1/0/3
Switch(config-if)# atm accounting
```

Displaying the ATM Accounting Interface Configuration

To display the ATM accounting status, use the following privileged EXEC command:

Command	Purpose
more system:running-config	Displays the ATM accounting status.

Example

The following display shows that ATM accounting is enabled on ATM interface 1/0/3:

```
Switch# more system:running-config
Building configuration...

Current configuration:
!

<information deleted>

!
interface ATM1/0/3
 no keepalive
 atm accounting
!
<information deleted>
```

Configuring the ATM Accounting Selection Table

The ATM accounting selection table determines the connection data to be gathered from the ATM switch router. To configure the ATM accounting selection entries, perform the following tasks, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **atm accounting selection** *index* Switch(config-acct-sel)#	Specifies the ATM accounting selection index number and changes to accounting selection mode.
Step 2	Switch(config-acct-sel)# **default** [**connection-type** | **list**]	Resets the ATM accounting selection table configuration to the default.
Step 3	Switch(config-acct-sel)# **connection-types** [**pvc** | **pvp** | **spvc-originator** | **spvc-target** | **spvp-originator** | **spvp-target** | **svc-in** | **svc-out** | **svp-in** | **svp-out**]	Specifies the connection type(s) for which you want to collect accounting records.
Step 4	Switch(config-acct-sel)# **list** *hex-bitmap*	Configures the list of ATM accounting MIB objects to collect.[1]

1. The MIB objects are listed in the *ATM Accounting Information MIB* publication.

The **atm accounting selection** command creates or modifies an entry in the selection table by specifying the fields of the entry.

Note A default selection entry is automatically configured during initial startup and cannot be deleted.

Some features of the ATM accounting selection table configuration include:

- An entry in the selection table points to a data collection file.
- A selection entry cannot be deleted when data collection is active.
- A selection entry can point to a nonexistent file, in which case the entry is considered inactive.
- One selection entry can apply to more than one type of VC (or example, SVC and PVC).
- If you modify a selection entry list, the new value is used the next time the data collection cycle begins, (for example, the next time the ATM accounting collection file swap occurs).

Note The following ATM accounting MIB objects are not supported:
• atmAcctngTransmittedClp0Cells (object number 16)
• atmAcctngReceivedClp0Cells (object number 18)
• atmAcctngCallingPartySubAddress (object number 31)
• atmAcctngCalledPartySubAddress (object number 32)
• atmAcctngRecordCrc16 (object number 33)

Examples

The following example shows how to change to ATM accounting selection configuration mode and add the SPVC originator connection type entry to selection entry 1:

```
Switch(config)# atm accounting selection 1
Switch(config-acct-sel)# connection-types spvc-originator
```

The following example shows how to change to ATM accounting selection configuration mode and reset the connection types for selection entry 1:

```
Switch(config)# atm accounting selection 1
Switch(config-acct-sel)# default connection-types
```

The following example shows how to change to ATM accounting selection configuration mode and configure the selection list to include all objects:

```
Switch(config)# atm accounting selection 1
Switch(config-acct-sel)# default list
```

The following example shows how to change to ATM accounting selection configuration mode and configure the selection list to include object number 20 (atmAcctngTransmitTrafficDescriptorParam1):

```
Switch(config)# atm accounting selection 1
Switch(config-acct-sel)# list 00001000
```

Displaying ATM Accounting Selection Configuration

To display the ATM accounting status, use the following EXEC command:

Command	Purpose
show atm accounting	Displays the ATM accounting selection configuration.

Example

The following example shows the ATM accounting status using the **show atm accounting** EXEC command:

```
Switch# show atm accounting
ATM Accounting Info:     AdminStatus - UP;      OperStatus : UP
Trap Threshold - 90 percent (4500000 bytes)
Interfaces:
File Entry 1: Name acctng_file1
    Descr: atm accounting data
    Min-age (seconds): 3600
    Failed_attempt : C0
    Sizes: Active 69 bytes (#records 0); Ready 73 bytes (#records 0)
➔ selection Entry -
➔     Selection entry 1, subtree - 1.3.6.1.4.1.9.10.18.1.1
➔     Selection entry 1, list - 00.00.10.00
➔     Selection entry 1, connType - F0.00
Active selection -
    Selection entry 1, subtree - 1.3.6.1.4.1.9.10.18.1.1
    Selection entry 1, list - FF.FE.BF.FC
    Selection entry 1, connType - F0.00

Debug output
<information deleted>
```

Configuring ATM Accounting Files

Direct the ATM accounting data being gathered from the configured selection control table to a specific ATM accounting file. To configure the ATM accounting files and change to ATM accounting file configuration mode, perform the following tasks, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **atm accounting file acctng_file1** Switch(config-acct-file)#	Specifies the ATM accounting file and enters accounting file configuration mode.
Step 2	Switch(config-acct-file)# **collection-modes** [**on-release**] [**periodic**]	Configures when to write to the accounting file.
Step 3	Switch(config-acct-file)# **default** [**min-age**]	Resets the ATM accounting file configuration to the default.
Step 4	Switch(config-acct-file)# **description** *string*	Configures a short description for the ATM accounting file.
Step 5	Switch(config-acct-file)# **enable**	Enables ATM accounting for a specific file.
Step 6	Switch(config-acct-file)# **failed-attempts** [**none**] [**regular**] [**soft**]	Configures whether to record failed connection attempts.
Step 7	Switch(config-acct-file)# **interval** *seconds*	Configures the interval for periodic collection, in seconds.
Step 8	Switch(config-acct-file)# **min-age** *seconds*	Configures the ATM accounting file minimum age of the VC.

Note Only one ATM accounting file can be configured and that file cannot be deleted.

Examples

The following example shows how to enable ATM accounting file configuration mode for acctng_file1 and reconfigure the collection mode on release of a connection:

```
Switch(config)# atm accounting file acctng_file1
Switch(config-acct-file)# collection-mode on-release
```

The following example shows how to enable ATM accounting file configuration mode for acctng_file1 and reconfigure the minimum age to the default value:

```
Switch(config)# atm accounting file acctng_file1
Switch(config-acct-file)# default min-age
```

The following example shows how to enable ATM accounting file configuration mode for acctng_file1 and configure a short description to be displayed in the **show atm accounting file** display and the file header:

```
Switch(config)# atm accounting file acctng_file1
Switch(config-acct-file)# description Main accounting file for engineering
```

The following example shows how to enable ATM accounting file configuration mode for acctng_file1:

```
Switch(config)# atm accounting file acctng_file1
Switch(config-acct-file)# enable
```

The following example shows how to enable ATM accounting file configuration mode for acctng_file1 to collect connection data every hour:

```
Switch(config)# atm accounting file acctng_file1
Switch(config-acct-file)# interval 3600
```

Displaying the ATM Accounting File Configuration

To display the ATM accounting status, use the following EXEC command:

Command	Purpose
show atm accounting	Displays the ATM accounting.

Example

The following example shows the ATM accounting file status using the **show atm accounting** EXEC command:

```
Switch# show atm accounting
ATM Accounting Info:     AdminStatus - UP;      OperStatus : UP
Trap Threshold - 90 percent (4500000 bytes)
Interfaces:
➔ File Entry 1: Name acctng_file1
➔     Descr: atm accounting data
➔     Min-age (seconds): 3600
➔     Failed_attempt : C0
➔     Sizes: Active 69 bytes (#records 0); Ready 73 bytes (#records 0)
selection Entry -
    Selection entry 1, subtree - 1.3.6.1.4.1.9.10.18.1.1
    Selection entry 1, list - FF.FE.BF.FC
    Selection entry 1, connType - F0.00
Active selection -
    Selection entry 1, subtree - 1.3.6.1.4.1.9.10.18.1.1
    Selection entry 1, list - FF.FE.BF.FC
    Selection entry 1, connType - F0.00

Debug output
Sig API: Err - 0
New_Conn: OK - 0; Err - 0
Rel_Conn: OK - 0; Err - 0
New_Leg: OK - 0; Err - 0
Rel_Leg: OK - 0; Err - 0
New_Party: OK - 0; Err - 0
Rel_Party: OK - 0; Err - 0
```

Controlling ATM Accounting Data Collection

To configure the behavior of the buffers used for ATM accounting collection, use the following command in privileged EXEC mode:

Command	Purpose
atm accounting collection {**collect-now** | **swap**} *filename*	Configures the ATM accounting data collection.

Examples

The following example specifies that all VCs that meet the minimum age requirement should be collected:

```
Switch# atm accounting collection collect-now accntg_file1
```

The following example swaps the buffers used to store accounting records; the old buffer is now ready to download:

```
Switch# atm accounting collection swap acctng_file1
```

Displaying the ATM Accounting Data Collection Configuration and Status

To display the ATM accounting file configuration status, use the following EXEC command:

Command	Purpose
show atm accounting	Displays the ATM accounting status.

Example

The following example shows the ATM accounting status using the **show atm accounting files** EXEC command:

```
Switch# show atm accounting
ATM Accounting Info:     AdminStatus - UP;      OperStatus : DOWN
Trap Threshold - 90 percent (4500000 bytes)
Interfaces:
File Entry 1: Name acctng_file1
    Descr: atm accounting data
    Min-age (seconds): 3600
    Failed_attempt : C0
➔ No file buffers initialized
selection Entry -
    Selection entry 1, subtree - 1.3.6.1.4.1.9.10.18.1.1
    Selection entry 1, list - FF.FE.BF.FC
    Selection entry 1, connType - F0.00
Active selection -
    Selection entry 1, subtree - 1.3.6.1.4.1.9.10.18.1.1
    Selection entry 1, list - FF.FE.BF.FC
    Selection entry 1, connType - F0.00
<Information deleted>
```

Configuring ATM Accounting SNMP Traps

You can configure SNMP traps to be generated when the ATM accounting file reaches a specified threshold. You can use these traps to alert you when a file is full and needs to be downloaded.

Configuring ATM Accounting Trap Generation

To configure ATM accounting SNMP traps, use the following command in global configuration mode:

Command	Purpose
atm accounting trap threshold *percent-value*	Configures the ATM accounting file threshold to generate an SNMP trap when it reaches a percentage of the maximum size.

Example

The following example shows how to configure ATM accounting SNMP traps to be sent when the file size reaches 85 percent full:

```
Switch(config)# atm accounting trap threshold 85
```

Displaying ATM Accounting Trap Threshold Configuration

To display the ATM accounting trap threshold configuration, use the following EXEC command:

Command	Purpose
show atm accounting	Displays the ATM accounting trap configuration.

Example

The following example shows the ATM accounting trap threshold configuration using the **show atm accounting** command:

```
Switch# show atm accounting
ATM Accounting Info:     AdminStatus - UP;      OperStatus : UP
Trap Threshold - 90 percent (4500000 bytes)
Interfaces:
File Entry 1: Name acctng_file1
    Descr: atm accounting data
    Min-age (seconds): 3600
    Failed_attempt : C0
    Sizes: Active 69 bytes (#records 0); Ready 73 bytes (#records 0)
selection Entry -
    Selection entry 1, subtree - 1.3.6.1.4.1.9.10.18.1.1
    Selection entry 1, list - FF.FE.BF.FC
    Selection entry 1, connType - F0.00
Active selection -
    Selection entry 1, subtree - 1.3.6.1.4.1.9.10.18.1.1
    Selection entry 1, list - FF.FE.BF.FC
    Selection entry 1, connType - F0.00

<information deleted>
```

Configuring SNMP Server for ATM Accounting

To enable SNMP ATM accounting trap generation and specify an SNMP server, perform the following steps in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **snmp-server enable traps atm-accounting**	Enables SNMP server ATM accounting trap generation.
Step 2	Switch(config)# **snmp-server host** *host community-string* **atm-accounting**	Configures SNMP server host IP address and community string for ATM accounting.

Example

The following example shows how to enable SNMP server ATM accounting traps and configure the SNMP server host at IP address 1.2.3.4 with community string *public* for ATM accounting:

```
Switch(config)# snmp-server enable traps atm-accounting
Switch(config)# snmp-server host 1.2.3.4 public atm-accounting
```

Displaying SNMP Server ATM Accounting Configuration

To display the SNMP server ATM accounting configuration, use the following privileged EXEC command:

Command	Purpose
more system:running-config	Displays the SNMP server ATM accounting configuration.

Example

The following example shows the SNMP server ATM accounting configuration using the **more system:running-config** privileged EXEC command:

```
Switch# more system:running-config
Building configuration...

Current configuration:
!
<information deleted>
!
ip rcmd rcp-enable
ip rcmd remote-host dplatz 171.69.194.9 dplatz
ip rcmd remote-username dplatz
atm template-alias byte_wise 47.9*f8.33...
atm template-alias bit_set 47.9f9(1*0*)88ab...
atm template-alias training 47.1328...
atm accounting enable
atm accounting trap threshold 85
!
<information deleted>

no ip classless
atm route 47.0091.8100.0000.0000.0ca7.ce01... ATM3/0/0
snmp-server enable traps chassis-fail
snmp-server enable traps chassis-change
snmp-server enable traps atm-accounting
snmp-server host 1.2.3.4 public atm-accounting
!
<information deleted>
```

Using TFTP to Copy the ATM Accounting File

After the ATM accounting file is written to DRAM, you must configure TFTP to allow network requests to copy the accounting information to a host for processing. To do this, use the following command in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **access-list** *access-list-number* {**deny** | **permit**} {*source* [*source-wildcard*] | **any**}	Defines a standard IP access list using a source address and wildcard or the **any** option default source 0.0.0.0 and source mask 255.255.255.255.
Step 2	Switch(config)# **tftp-server** {**atm-acct-active:acctng_file1** | **atm-acct-ready:acctng_file1**} *ip-access-list*	Allows TFTP to copy the ATM accounting file to an IP host in response to a read request.

Example

The following example shows how to allow TFTP service to copy the ATM accounting file *acctng_file1* to the IP access list of requesting host number 1:

```
Switch(config)# access-list 1 permit 10.1.1.1
Switch(config)# tftp-server atm-acct-ready:acctng_file1 1
```

For more information about access lists, see the “Filtering IP Packets at the IP Interfaces” section on page 11-9.

Configuring Remote Logging of ATM Accounting Records

You can collect ATM accounting records to a remotely connected PC or UNIX workstation. You can use this method in place of, or in addition to, collecting ATM accounting records as a file into the switch's memory.

The remote logging method requires a server daemon to be running on a PC or a UNIX workstation that is reachable from the switch using IP. The server daemon listens to the TCP port specified in the switch side remote logging configuration. When the ATM accounting process on the switch sends a TCP connect request, the daemon accepts the connection. After connection has been established, the switch side ATM accounting process sends accounting records, as they are created, to the remote host. The remote host then receives the records and stores them in a local file. The collected ATM accounting records are in ASN1 format. The first record contains the format of the following records.

To configure remote logging, perform the following steps in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **atm accounting file acctng_file1**	Configures the ATM accounting file and changes to accounting file configuration mode.
Step 2	Switch(config)# **remote-log** [**only**] **primary-host** *hostname1 tcp-port1* [**alternate-host** *hostname2 tcp-port2*]	Specifies the main and optional backup hostname or IP address and TCP port number.

The PC or workstation configured as backup takes over collection of ATM accounting records if the primary fails. Using the keyword **only** causes only remote logging to be performed, freeing the ATM switch router's memory for other purposes.

Example

The following example shows how to configure remote logging to a PC named eagle on port 2001, with port 2002 as a backup:

```
Switch(config)# atm accounting file acctng_file1
Switch(config-acct-file)# remote-log primary-host eagle 2001 alternate-host eagle 2002
```

Displaying the Remote Logging Configuration

To display the remote logging configuration, use the following privileged EXEC command:

Command	Purpose
show atm accounting	Displays the remote logging configuration.

The following example shows the remote logging configuration using the **show atm accounting** EXEC command:

```
Switch# show atm accounting
ATM Accounting Info:     AdminStatus - UP;      OperStatus : UP
Trap Threshold - 90 percent (4500000 bytes)
Interfaces:
    AT1/0/0
    AT2/0/0
File Entry 1 -
    Name: acctng_file1
    Descr: atm accounting data
    Min-age (seconds): 0
    Failed_attempt : soft regular
    Interval (seconds) : 60
    Collect Mode : on-release periodic
    Sizes: Active 68 bytes (#records 0); Ready 74 bytes (#records 0)
    Remote Log and local storage are enabled.
    Primary Log Host: eagle, TCP listen port: 2001, OperStatus: DOWN
    Alternate Log Host: eagle, TCP listen port: 2002, OperStatus: DOWN
Selection Entry 1 -
    Subtree OID : 1.3.6.1.4.1.9.10.18.1.1
    List Bitmap : FF.FE.BF.FC
    Conn Type : svc-in svc-out pvc pvp spvc-originator spvc-target
  Active List Bitmap - FF.FE.BF.FC
```

Configuring ATM RMON

This section describes the process you use to configure ATM RMON on the ATM switch router. The following sections describe the process:

RMON Overview

The ATM RMON feature allows you to monitor network traffic for reasons such as fault monitoring or capacity planning. The ATM RMON feature is an extension of an existing, well-known RMON standard and provides high-level per-host and per-conversation statistics in a standards-track MIB similar to the following RMON MIBs:

- RMON-1 MIB—RFC 1757
- RMON-2 MIB—RFC 2021 and 2074

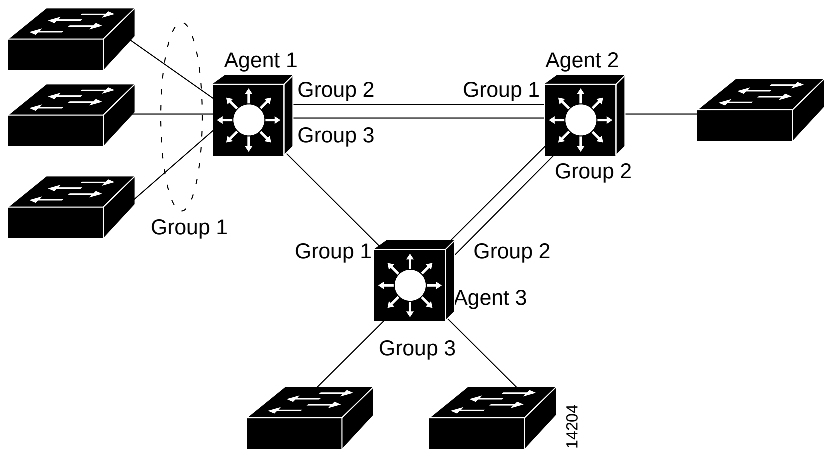
The ATM-RMON counter uses the per-VC counters already maintained in the hardware and polled by the software. The ATM RMON agent can report cell traffic statistics by monitoring connection management activity. At connection setup and release time, some ATM-RMON bookkeeping code is executed. The amount of information varies, depending on the ATM RMON configuration. The ATM-RMON bookkeeping capability significantly reduces the processing requirements for ATM-RMON, and allows collecting statistics on many or all the of ATM switch router ports at once.

The ATM-RMON agent uses the 64-bit version of each cell counter if 64-bit counter support is present in the SNMP master-agent library.

Configuring Port Select Groups

Previously, RMON allowed collection of connection information on a per-interface basis only. ATM RMON allows a group of ports to be configured as an aggregate. The port select group defines this *collection unit* used by the ATM RMON agent to gather host and matrix connection data. For example, in Figure 14-3, agent 1 has a port selection group 1 made up of ports.

Figure 14-3 ATM RMON Port Select Group Examples



An active port select group must be defined before any data collection can begin. You can use the command-line interface (CLI) and Simple Network Management Protocol (SNMP) modules to configure and access port select group structures.

To configure an RMON port selection group, use the following command in global configuration mode:

Command	Purpose
atm rmon portselgrp *number* [**descr** *string* | **host-prio** *number* | **host-scope** *number* | **matrix-prio** *number* | **matrix-scope** *number* | **maxhost** number | **maxmatrix** | **nostats** | **owner** *string*]	Configures the ATM RMON port selection group.

Example

The following example shows how to configure port selection group 7 with the a maximum host count of 500, maximum matrix count of 2000, host priority of 1, and owner name “nms 3”.

```
Switch(config)# atm rmon portselgrp 7 maxhost 500 maxmatrix 2000 host-prio 1 owner “nms 3”
```

Displaying the ATM RMON Port Select Group

To display the ATM RMON port select group statistics, use the following EXEC command:

Command	Purpose
show atm rmon stats *number*	Displays the ATM RMON port select group statistics.

Example

The following example shows how to display the configuration of port selection group 3 using the **show atm rmon stats** command from EXEC mode:

```
Switch# show atm rmon stats 3
PortSelGrp: 3   Collection: Enabled     Drops: 0
  CBR/VBR: calls: 0/0  cells: 0 connTime: 0 days 00:00:00
  ABR/UBR: calls: 0/0  cells: 0 connTime: 0 days 00:00:00
```

Configuring Interfaces into a Port Select Group

Before the port selection group can begin gathering host and matrix connection information, an interface or group of interfaces must be added to the port selection group.

To configure an interface to an ATM RMON port selection group, perform the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **interface atm** *card*/*subcard*/*port* Switch(config-if)#	Selects the interface to be configured.
Step 2	Switch(config-if)# **atm rmon collect** *port_sel_group*	Configures the interface to an ATM RMON port selection group.

Example

The following example shows how to configure ATM interface 0/1/3 to ATM RMON port selection group 6:

```
Switch(config)# interface atm 0/1/3
Switch(config-if)# atm rmon collect 6
```

Displaying the Interface Port Selection Group Configuration

To display the ATM RMON port configuration status, use the following EXEC command:

Command	Purpose
show atm rmon {**host** *number* | **matrix** *number* | **stats** *number* | **status**}	Displays the interface port selection group configuration.

Examples

The following example shows how to display the ATM RMON host configuration for port selection group 6 using the **show atm rmon host** command from user EXEC mode:

```
Switch# show atm rmon host 6
PortSelGrp: 6   Collection: Enabled     Drops: 0
```

The following example shows how to display the ATM RMON matrix configuration for port selection group 6 using the **show atm rmon matrix** command from user EXEC mode:

```
Switch# show atm rmon matrix 6
PortSelGrp: 6   Collection: Enabled     Drops: 0
```

The following example shows how to display the ATM RMON statistics configuration for port selection group 6 using the **show atm rmon stats** command from user EXEC mode:

```
Switch# show atm rmon stats 6
PortSelGrp: 6   Collection: Enabled     Drops: 0
  CBR/VBR: calls: 0/0  cells: 0 connTime: 0 days 00:00:00
  ABR/UBR: calls: 0/0  cells: 0 connTime: 0 days 00:00:00
```

The following example shows how to display the ATM RMON status for all port selection groups using the **show atm rmon status** command from user EXEC mode:

```
Switch# show atm rmon status
PortSelGrp: 1 Status: Enabled  Hosts:  4/no-max  Matrix:  4/no-max
        ATM0/0/0        ATM0/0/2
PortSelGrp: 2 Status: Enabled  Hosts:  0/no-max  Matrix:  0/no-max
        ATM0/0/3
PortSelGrp: 3 Status: Enabled  Hosts:  0/no-max  Matrix:  0/no-max
        ATM0/1/0        ATM0/1/1
PortSelGrp: 4 Status: Enabled  Hosts:  0/1  Matrix:  0/5
        ATM0/0/1
PortSelGrp: 5 Status: Enabled  Hosts:  0/no-max  Matrix:  0/no-max
        ATM0/1/2
PortSelGrp: 6 Status: Enabled  Hosts:  0/no-max  Matrix:  0/no-max
        ATM0/1/3
PortSelGrp: 7 Status: Enabled  Hosts:  0/no-max  Matrix:  0/no-max
        ATM2/0/0
PortSelGrp: 8 Status: Enabled  Hosts:  0/no-max  Matrix:  0/no-max

PortSelGrp: 9 Status: Enabled  Hosts:  0/no-max  Matrix:  0/no-max
```

Enabling ATM RMON Data Collection

Use the **atm rmon enable** command to start ATM RMON data collection.

Note If you disable ATM RMON the configuration remains but becomes inactive (similar to using the **shutdown** command on an interface).

To enable ATM RMON data collection, use the following command in global configuration mode:

Command	Purpose
atm rmon enable	Enables ATM RMON.

Displaying the ATM RMON Configuration

To display the ATM RMON configuration, use the following privileged EXEC command:

Command	Purpose
more system:running-config	Displays the ATM RMON configuration.

Example

The following example shows the ATM RMON configuration using the **more system:running-config** privileged EXEC command:

```
Switch# more system:running-config
Building configuration...

Current configuration:
!

<information deleted>

ip default-gateway 172.20.53.206
no ip classless
snmp-server community public RW
snmp-server location racka-cs:2016
snmp-server contact abierman
atm rmon portselgrp 1 host-scope 3 matrix-scope 3
atm rmon portselgrp 2 host-scope 3 matrix-scope 3 descr "router port 2" owner
rubble"
atm rmon portselgrp 3 host-scope 3 matrix-scope 3 descr "test" owner "bam_bam"
atm rmon portselgrp 4 maxhost 1 maxmatrix 5 host-scope 1 descr "no active ports" owner "wilma"
atm rmon portselgrp 5
atm rmon portselgrp 6 matrix-prio 1
atm rmon portselgrp 7 host-scope 3 matrix-scope 3 descr "CPU port" owner "pebbles"
atm rmon portselgrp 8
atm rmon portselgrp 9
atm rmon enable
!
<information deleted>
```

Configuring an RMON Event

To configure an RMON event being generated, use the following command in global configuration mode:

Command	Purpose
rmon event *number* [**log**] [**trap** *community*] [**description** *string*] [**owner** *string*]	Configures an RMON event.

Example

The following example shows how to configure a generated RMON event with an assigned name, description string, owner, and SNMP trap with community string:

```
Switch(config)# rmon event 1 description test owner nms_3 trap test
```

Displaying the Generated RMON Events

To display the generated RMON events, use the following EXEC command:

Command	Purpose
show rmon events	Displays generated RMON events.

Example

The following example shows the RMON events generated using the **show rmon events** EXEC command:

```
Switch# show rmon events
Event 1 is active, owned by nms_3
 Description is test
 Event firing causes trap to community test, last fired 00:00:00
```

Configuring an RMON Alarm

You can configure RMON alarm generation if any of the configured parameters are met.

Note Refer to the *Configuration Fundamentals Configuration Guide* for general SNMP RMON configuration information.

To configure RMON alarms, use the following command in global configuration mode:

Command	Purpose
rmon alarm *number variable interval* {**delta** | **absolute**} **rising-threshold** *value* [*event-number*] **falling-threshold** *value* [*event-number*] [**owner** *string*]	Configures the ATM RMON alarm.

Example

The following example shows how to configure RMON alarm number 1 to generate an alarm under the following conditions:

- If the MIB atmHostHCCells exceed 500
- If each sample, in absolute mode, shows:
  - Rising threshold exceeding 10,000
  - Falling threshold falling below 1000
- The RMON alarm number 1 sends the alarm to the owner “nms 3”

```
Switch(config)# rmon alarm 1 atmHostInHCCells 500 absolute rising-threshold 10000 falling-threshold 1000
owner “nms 3”
```

Displaying the Generated RMON Alarms

To display the RMON alarm event, use the following EXEC command:

Command	Purpose
show rmon alarms events	Displays RMON alarms.

Example

The following example shows the RMON alarms and events using the **show rmon alarms events** EXEC command:

```
Switch# show rmon alarms events
Event 1 is active, owned by nms 3
 Description is test
 Event firing causes trap to community test, last fired 00:00:00
Alarm table is empty
```

CHAPTER 15

Configuring Tag Switching

This chapter describes tag switching, a high-performance packet-forwarding technology that assigns tags to mulitprotocol frames for transport across packet- or cell-based networks.

Note This chapter provides advanced configuration instructions for the Catalyst 8540 MSR, Catalyst 8510 MSR, and LightStream 1010 ATM switch routers. For an overview of tag switching, refer to the *Guide to ATM Technology*. For complete descriptions of the commands mentioned in this chapter, refer to the *ATM Switch Router Command Reference* publication.

This chapter includes the following sections:

Tag Switching Overview

In conventional Layer 3 forwarding, as a packet traverses the network, each router extracts forwarding information from the Layer 3 header. Header analysis is repeated at each router (hop) through which the packet passes.

In a tag switching network, the Layer 3 header is analyzed just once. It is then mapped into a short fixed-length tag. At each hop, the forwarding decision is made by looking only at the value of the tag. There is no need to reanalyze the Layer 3 header. Because the tag is a fixed-length, unstructured value, lookup is fast and simple.

For an overview of how tag switching works and its benefits, refer to the *Guide to ATM Technology*.

Hardware and Software Requirements and Restrictions (Catalyst 8540 MSR)

The Catalyst 8540 MSR hardware requirements for tag switching include the following:

- The ATM switch router (used as a tag switch)
- A tag edged router such as a Cisco 7000 Route Switch Processor (RSP) with an Optical Carrier 3 (OC-3) ATM interface processor (AIP) installed

Tag switching has the following software restrictions:

- Open Shortest Path First (OSPF) is the only routing protocol currently supported.
- IP is the only network layer protocol supported.
- Hierarchical VP tunnels cannot co-exist on a physical interface with tag switching.

Hardware and Software Requirements and Restrictions (Catalyst 8510 MSR and LightStream 1010)

The Catalyst 8510 MSR and LightStream 1010 ATM switch router hardware requirements for tag switching include the following:

- The ATM switch router (used as a tag switch).
- A switch processor feature card installed on the route processor, if you want to enable VC merge (multipoint-to-point connection). Note that FC-PFQ requires 64 MB of DRAM.
- A tag edged router such as a Cisco 7000 RSP with an OC-3 AIP installed.

Tag switching has the following software restrictions:

- Open Shortest Path First (OSPF) is the only routing protocol currently supported.
- IP is the only network layer protocol supported.
- Hierarchical VP tunnels cannot co-exist on a physical interface with tag switching.

Configuring Tag Switching

This section describes how to configure tag switching on ATM switch routers, and includes the following procedures:

- Configuring a Loopback Interface, page 15-3
- Enabling Tag Switching on the ATM Interface, page 15-4
- Configuring OSPF, page 15-5
- Configuring a VPI Range (Optional), page 15-7
- Configuring TDP Control Channels (Optional), page 15-8
- Configuring Tag Switching on VP Tunnels, page 15-10
- Connecting the VP Tunnels, page 15-12
- Configuring VC Merge, page 15-12

Configuring a Loopback Interface

You should configure a loopback interface on every ATM switch router configured for tag switching. The loopback interface, a virtual interface, is always active. The IP address of the loopback interface is used as the Tag Distribution Protocol (TDP) identifier for the ATM switch router. If a loopback interface does not exist, the TDP identifier is the highest IP address configured on the ATM switch router. If that IP address is administratively shut down, all TDP sessions through the ATM switch router restart. Therefore, we recommend that you configure a loopback interface.

To configure the loopback interface, perform the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **interface loopback** *number* Switch(config-if)#	Enters interface configuration mode and assigns a number to the loopback interface.
Step 2	Switch(config-if)# **ip address** *ip-address mask*	Assigns an IP address and subnet mask to the loopback interface. **Note** We recommend a 32-bit subnet mask (255.255.255.255) for the loopback interface. If you do not use a 32-bit subnet mask, two TVCs[1] terminate for the same address—one for a 32-bit subnet mask and the other for the mask you entered. Entering a 32-bit subnet mask reduces the number of TVCs to one.

1. TVCs = tag virtual channels.

Example

In the following example, loopback interface 0 is created with an IP address of 1.0.1.11 and a subnet mask of 255.255.255.255:

```
Switch(config)# interface loopback 0
Switch(config-if)# ip address 1.0.1.11 255.255.255.255
Switch(config-if)# exit
```

Displaying Loopback Interface Configuration

The following example shows the loopback 0 configuration using the **show interfaces** privileged EXEC command:

```
Switch# show interfaces loopback 0
Loopback0 is up, line protocol is up
  Hardware is Loopback
  Internet address is 1.0.1.11/24
  MTU 1500 bytes, BW 8000000 Kbit, DLY 5000 usec, rely 255/255, load 1/255
  Encapsulation LOOPBACK, loopback not set, keepalive set (10 sec)
  Last input 00:00:03, output never, output hang never
  Last clearing of "show interface" counters never
  Queueing strategy: fifo
  Output queue 0/0, 0 drops; input queue 0/75, 0 drops
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
     0 packets input, 0 bytes, 0 no buffer
     Received 0 broadcasts, 0 runts, 0 giants, 0 throttles
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
     73 packets output, 0 bytes, 0 underruns
     0 output errors, 0 collisions, 0 interface resets
     0 output buffer failures, 0 output buffers swapped out
```

Enabling Tag Switching on the ATM Interface

Note Configure all parallel interfaces between ATM switch routers for either IP unnumbered or with a specific IP address. Unnumbering some parallel interfaces and assigning specific IP addresses to others might cause TDP sessions to restart on some parallel interfaces when another parallel interface is shut down. Therefore, we highly recommend that you unnumber all parallel interfaces to loopback.

To enable tag switching on the ATM interface, perform the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **interface atm** *card*/*subcard*/*port* Switch(config-if)#	Enters interface configuration mode on the specified ATM interface.
Step 2	Switch(config-if)# **ip unnumbered** *type number*	Enables IP unnumbered on the ATM interface and assigns the unnumbered interface to an interface that has an IP address. We recommend enabling IP unnumbered because it allows you to conserve IP addresses and it reduces the number of TVCs terminating on the switch.
	or	or
	Switch(config-if)# **ip address** *ip-address mask*	Assigns an IP address and subnet mask to the ATM interface.
Step 3	Switch(config-if)# **tag-switching ip**	Enables tag switching of IPv4 packets.

Examples

In the following example, ATM interface 1/0/1 is configured for IP unnumbered to loopback interface 0:

```
Switch(config-if)# interface atm 1/0/1
Switch(config-if)# ip unnumbered loopback 0
Switch(config-if)# tag-switching ip
Switch(config-if)# exit
```

In the following example, ATM interface 0/0/3 is configured with a specific IP address and subnet mask (1.3.11.3 255.255.0.0):

```
Switch(config)# interface atm 0/0/3
Switch(config-if)# ip address 1.3.11.3 255.255.0.0
Switch(config-if)# tag-switching ip
Switch(config-if)# exit
```

Displaying the ATM Interface Configuration

To display the ATM interface configuration, use the following EXEC command:

Command	Purpose
show tag-switching interfaces	Displays the tag switching configuration on the ATM interface.

The following example shows that tag switching is configured on ATM interfaces 0/0/3 and 1/0/1:

```
Switch# show tag-switching interfaces
Interface              IP    Tunnel   Operational
ATM0/0/3               Yes   No       Yes          (ATM tagging)
ATM1/0/1               Yes   No       Yes          (ATM tagging)
```

Configuring OSPF

Enable OSPF on the ATM switch router so that it can create routing tables, which identify routes through the network. Then add the addresses and associated routing areas to the OSPF process so that it can propagate the addresses to other ATM switch routers:

	Command	Purpose
Step 1	Switch(config)# **router ospf** *process_number* Switch(config-router)#	Enables OSPF and assigns it a process number. The process number can be any positive integer.
Step 2	Switch(config-router)# **network** *address wildcard-mask* **area** *area-id*	Defines the network prefix, a wildcard subnet mask, and the associated area number on which to run OSPF. An area number is an identification number for an OSPF address range. Repeat this command for each additional area you want to add to the OSPF process. Caution Ethernet0 is used for system management only. Do not add this interface to the routing protocol process.

Note Since the 12.0(1a)W5(5b) release of the system software, addressing the interface on the route processor (CPU) has changed. The ATM interface is now called atm0, and the Ethernet interface is now called ethernet0. Old formats (atm 2/0/0 and ethernet 2/0/0) are still supported.

Example

The following is an example of OSPF enabled and assigned process number 10000. All addresses are in area 0:

Note An IP address of 1.1.1.1 with a subnet mask of 255.255.255.0 is entered as an IP network prefix of 1.1.1.0 with a subnet mask of 0.0.0.255. Likewise, an IP address of 1.2.1.1 with a subnet mask of 255.255.255.0 is entered as an IP network prefix of 1.2.1.0 with a subnet mask of 0.0.0.255.

```
Switch(config)# router ospf 10000
Switch(config-router)# network 1.1.1.0 0.0.0.255 area 0
Switch(config-router)# network 1.2.1.0 0.0.0.255 area 0
Switch(config-router)# network 1.3.0.0 0.0.255.255 area 0
Switch(config-router)# network 200.2.2.0 0.0.0.255 area 0
Switch(config-router)# network 1.0.1.0 0.0.0.255 area 0
Switch(config-router)# network 1.18.0.0 0.0.255.255 area 0
```

Displaying the OSPF Configuration

To display the OSPF configuration, use the following privileged EXEC command:

Command	Purpose
show ip ospf	Displays the OSPF configuration.

The following example shows the OSPF configuration using the **show ip ospf** privileged EXEC command:

```
Switch# show ip ospf
 Routing Process "ospf 10000" with ID 1.0.1.11
 Supports only single TOS(TOS0) routes
 SPF schedule delay 5 secs, Hold time between two SPFs 10 secs
 Number of DCbitless external LSA 0
 Number of DoNotAge external LSA 0
 Number of areas in this router is 1. 1 normal 0 stub 0 nssa
    Area BACKBONE(0) (Inactive)
        Number of interfaces in this area is 4
        Area has no authentication
        SPF algorithm executed 2 times
        Area ranges are
        Link State Update Interval is 00:30:00 and due in 00:14:42
        Link State Age Interval is 00:20:00 and due in 00:14:10
        Number of DCbitless LSA 0
        Number of indication LSA 0
        Number of DoNotAge LSA 0
```

Configuring a VPI Range (Optional)

Although not necessary for most configurations, you might need to change the default tag virtual path identifier (VPI) range on the switch if:

- It is an administrative policy to use a VPI value other than 1, the default VPI.
- There are a large number of tag virtual channels (TVCs) on an interface.

Note You cannot enter a VPI range on a VP tunnel. On VP tunnels, the VPI is the permanent virtual path (PVP) number of the tunnel.

To change the default tag VPI range, perform the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **interface atm** *card*/*subcard*/*port* Switch(config-if)#	Enters interface configuration mode on the specified ATM interface.
Step 2	Switch(config-if)# **tag-switching atm vpi** *vpi* [– *vpi*]	Enters the VPI range. **Note** If the TDP neighbor is a router, the VPI range can be no larger than two. For example, from 5 to 6 (a range of two), not 5 to 7 (a range of three). If the TDP neighbor is a switch, the maximum VPI range is 0 to 255.

Examples

The following example shows how to select a VPI range from 5 to 6 (a range of two), an acceptable range if the TDP neighbor is a router:

```
Switch(config)# interface atm 3/0/1
Switch(config-if)# tag-switching ip
Switch(config-if)# tag-switching atm vpi 5 - 6
```

The following example shows how to select a VPI range from 5 to 7 (a range of three), an acceptable range if the TDP neighbor is a switch:

```
Switch(config)# interface atm 3/0/1
Switch(config-if)# tag-switching ip
Switch(config-if)# tag-switching atm vpi 5 - 7
```

Note Although the example shows a VPI range of three, you are not limited to a range of three if the TDP neighbor is a switch. The maximum VPI range is 0 to 255 if the TDP neighbor is a switch.

Displaying the Tag Switching VPI Range

To display the tag switching VPI range, use the following EXEC command:

Command	Purpose
show tag-switching interfaces detail	Displays the tag switching VPI range on an interface.

Example

The following example shows the tag switching VPI range on ATM interface 1/0/1:

```
Switch# show tag-switching interfaces detail
Interface ATM0/0/3:
        IP tagging enabled
        TSP Tunnel tagging not enabled
        Tagging operational
        MTU = 4470
        ATM tagging: Tag VPI = 1, Control VC = 0/32
Interface ATM1/0/1:
        IP tagging enabled
        TSP Tunnel tagging not enabled
        Tagging operational
        MTU = 4470
        ATM tagging: Tag VPI range = 5 - 6, Control VC = 6/32
<information deleted>
```

Configuring TDP Control Channels (Optional)

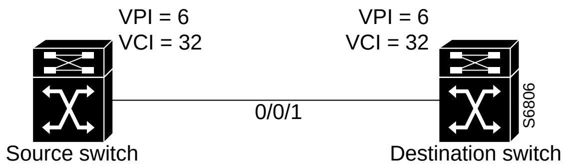
Although not necessary for most configurations, you can change the default Tag Distribution Protocol (TDP) control channel VPI and virtual channel identifier (VCI) if you want to use a nondefault value. The default TDP control channel is on VPI 0 and VCI 32. TDP control channels exchange TDP HELLOs and Protocol Information Elements (PIEs) to establish two-way TDP sessions. TVCs are created by the exchange of PIEs through TDP control channels.

To change the TDP control channel, perform the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **interface atm** *card*/*subcard*/*port* Switch(config-if)#	Enters interface configuration mode on the specified ATM interface.
Step 2	Switch(config-if)# **ip address** *ip-address mask*	Assigns an IP address and subnet mask to the ATM interface.
Step 3	Switch(config-if)# **tag-switching ip**	Enables tag switching of IPv4 packets.
Step 4	Switch(config-if)# **tag-switching atm control-vc** *vpi vci*	Changes the TDP control channel.

Figure 15-1 shows an example TDP control channel configuration between a source switch and destination switch on ATM interface 0/0/1. Note that the VPI and VCI values match on the source switch and destination switch.

Figure 15-1 Configuring TDP Control Channels



Examples

In the following example, a TDP control channel is configured on the source switch:

```
Switch(config)# interface atm 0/0/1
Switch(config-if)# ip address 1.2.0.11 255.255.255.0
Switch(config-if)# tag-switching ip
Switch(config-if)# tag-switching atm control-vc 6 32
Switch(config-if)# exit
```

In the following example, a TDP control channel is configured on the destination switch:

```
Switch(config)# interface atm 0/0/1
Switch(config-if)# ip address 1.2.0.12 255.255.255.0
Switch(config-if)# tag-switching ip
Switch(config-if)# tag-switching atm control-vc 6 32
Switch(config-if)# exit
```

If you are having trouble establishing a TDP session, verify that the VPI and VCI values match on the TDP control channels of the source switch and destination switch.

Displaying the TDP Control Channels

To display the TDP control channel configuration, use the following EXEC command:

Command	Purpose
show tag-switching interfaces detail	Displays the TDP control channel configuration on an interface.

The following example shows the TDP control channel configuration on interface ATM 0/0/3:

```
Switch# show tag-switching interfaces detail
Interface ATM0/0/3:
        IP tagging enabled
        TSP Tunnel tagging not enabled
        Tagging operational
        MTU = 4470
➔       ATM tagging: Tag VPI = 1, Control VC = 0/32
<information deleted>
```

Configuring Tag Switching on VP Tunnels

If you want to configure tag switching on virtual path (VP) tunnels, perform the following steps, beginning in global configuration mode:

Note This procedure is optional.

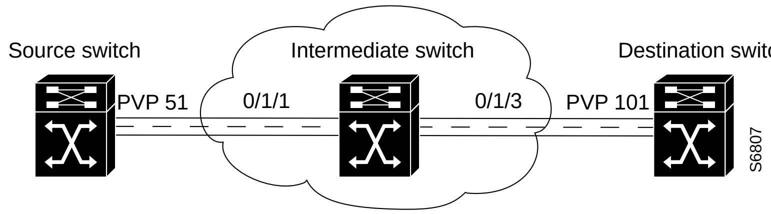
	Command	Purpose
Step 1	Switch(config)# **interface atm** *card/subcard/port* Switch(config-if)#	Enters interface configuration mode on the specified ATM interface.
Step 2	Switch(config-if)# **atm pvp** *vpi*	Creates a PVP. When configuring PVP connections, configure the lowest VPI numbers first.
Step 3	Switch(config-if)# **exit** Switch(config)#	Returns to global configuration mode.
Step 4	Switch(config)# **interface atm** *card/subcard/port***.***subinterface#* Switch(config-subif)#	Enters subinterface configuration mode.
Step 5	Switch(config-subif)# **ip unnumbered** *type number* or Switch(config-subif)# **ip address** *ip-address mask*	Enables IP unnumbered on the ATM interface and assigns the unnumbered interface to an interface that has an IP address. We recommend enabling IP unnumbered because it allows you to conserve IP addresses and reduces the number of TVCs terminating on the switch. or Assigns an IP address and subnet mask to the ATM interface.
Step 6	Switch(config-subif)# **tag-switching ip**	Enables tag switching of IPv4 packets.

Because a VP tunnel runs between switches, you must also configure a VP tunnel on the connecting ATM interface on the destination switch. The examples that follow show how to configure VP tunnels between switches.

Note The intermediate switch configuration follows in the next section, "Connecting the VP Tunnels."

Figure 15-2 shows an example VP tunnel between a source switch and destination switch.

Figure 15-2 Configuring VP Tunnels



Examples

In the following example, ATM interface 0/1/1 on the source switch has no IP address and PVP 51 is configured for IP unnumbered to loopback interface 0:

```
Switch(config-if)# interface atm 0/1/1
Switch(config-if)# atm pvp 51
Switch(config-if)# exit
Switch(config-if)# interface atm 0/1/1.51
Switch(config-subif)# ip unnumbered loopback 0
Switch(config-subif)# tag-switching ip
Switch(config-subif)# exit
```

In the following example, ATM interface 0/1/3 on the destination switch has no IP address and PVP 101 is configured for IP unnumbered to loopback interface 0:

```
Switch(config)# interface atm 0/1/3
Switch(config-if)# atm pvp 101
Switch(config-if)# exit
Switch(config)# interface atm 0/1/3.101
Switch(config-subif)# ip unnumbered loopback 0
Switch(config-subif)# tag-switching ip
Switch(config-subif)# exit
```

To connect the source and destination switch VP tunnels, proceed to the next section, "Connecting the VP Tunnels."

Displaying the VP Tunnel Configuration

To display the VP tunnel configuration, use the following EXEC command:

Command	Purpose
show atm vp	Displays the VP tunnel configuration on an interface.

The following example shows PVP 51 configured on ATM interface 0/1/1:

```
Switch# show atm vp
Interface    VPI    Type  X-Interface     X-VPI     Status
ATM0/1/1     51      PVP     TUNNEL
```

Connecting the VP Tunnels

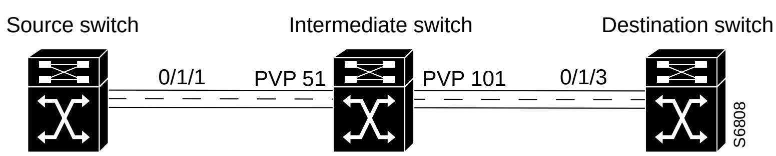
To complete the VP tunnel, you must configure the ATM ports on the intermediate switch to designate where to send packets coming from the source switch and going to the destination switch.

To connect the permanent virtual path (PVP), perform the following steps, beginning in interface configuration mode:

	Command	Purpose
Step 1	Switch(config)# **interface atm** *card*/*subcard*/*port* Switch(config-if)#	Enters interface configuration mode on the specified ATM interface.
Step 2	Switch(config-if)# **atm pvp** *vpi* **interface atm** *card*/*subcard*/*port vpi-B*	Connects the PVP from the source switch to the destination switch.

Figure 15-3 shows an example configuration on an intermediate switch.

Figure 15-3 Connecting the VP Tunnels



Example

In the following example, PVP 51 on ATM interface 0/1/1 is connected to PVP 101 on ATM interface 0/1/3:

```
Switch(config)# interface atm 0/1/1
Switch(config-if)# atm pvp 51 interface atm 0/1/3 101
Switch(config-if)# exit
```

Displaying the VP Tunnel Configuration

The following example shows PVP 51 on ATM interface 0/1/1 connected to PVP 101 on ATM interface 0/1/3:

```
Switch# show atm vp
Interface    VPI    Type  X-Interface     X-VPI     Status
ATM0/1/1     51      PVP     ATM0/1/3     101       DOWN
ATM0/1/3     101     PVP     ATM0/1/1     51        DOWN
```

Configuring VC Merge

VC merge allows the switch to aggregate multiple incoming flows with the same destination address into a single outgoing flow. Where VC merge occurs, several incoming tags are mapped to one single outgoing tag. Cells from different VCIs going to the same destination are transmitted to the same outgoing VC using multipoint-to-point connections. This sharing of tags reduces the total number of virtual circuits required for tag switching. Without VC merge, each source-destination prefix pair consumes one tag VC on each interface along the path. VC merge reduces the tag space shortage by sharing tags for different flows with the same destination.

Note VC merge support requires FC-PFQ on the route processor. If you do not have FC-PFQ, and you try to enable VC merge, the TVCs remain point-to-point. (Catalyst 8510 MSR and LightStream 1010)

VC merge is enabled by default. To disable VC merge, enter the following command in global configuration mode:

Command	Purpose
no tag-switching atm vc-merge	Enables VC merge.

Displaying the VC Merge Configuration

To display the VC merge configuration, use the following EXEC command:

Command	Purpose
show tag-switching atm-tdp capability	Displays the TDP control channel configuration on an interface.

The following example shows that VC merge configuration is enabled on ATM interface 0/1/0:

```
Switch# show tag-switching atm-tdp capability

                Control    VPI          VCI           Alloc   VC Merge
ATM0/1/0       VP    VC    Range        Range         Scheme  IN   OUT
  Negotiated   0     32    [7 - 8]      [33 - 1023]   UNIDIR  -    -
  Local        -     -     [7 - 8]      [33 - 16383]  UNIDIR  Yes  Yes
  Peer         -     -     [7 - 8]      [33 - 1023]   UNIDIR  -    -
```

Configuring Tag Switching CoS

Quality of service (QoS) allows ATM to meet the transmission *quality* and *service* availability of many different types of data. The need for delay-sensitive data, such as voice, can be given a higher priority than data that is not delay-sensitive, such as e-mail. The following service categories were created for ATM Forum VCs to meet the transmission needs of various types of data: VBR-RT, VBR-NRT, ABR, and UBR. See Chapter 8, "Configuring Resource Management," for more information about the standard ATM Forum implementation of QoS. This section describes tag switching class of service (CoS).

Up to eight QoS classes (0 to 7) can be allocated to each physical interface port. Each port has an independent logical rate scheduler (RS) and a weighted round-robin (WRR) scheduler. The RS guarantees minimum bandwidth and has first priority on supplying an eligible cell for transmission. Second priority is given to the service classes, which have been assigned relative weights that are based on the ratio of the total leftover bandwidth. The service class relative weights are configurable so you can change the priority of the default values. The VCs within a service class also have relative weights. The service classes and VCs within a service class are scheduled by their relative weights.

With tag switching CoS, tag switching can dynamically set up to four tag virtual channels (TVCs) with different service categories between a source and destination. TVCs do not share the same QoS classes reserved for ATM Forum VCs (VBR-RT, VBR-NRT, ABR, and UBR). The following four new service classes were created for TVCs: TBR_1 (WRR_1), TBR_2 (WRR_2), TBR_3 (WRR_3), and TBR_4 (WRR_4). These new service classes are called Tag Bit Rate (TBR) classes. TVCs and ATM Forum VCs can only coexist on the same physical interface, but they operate in ships in the night (SIN) mode and are unaware of each other.

TBR classes support only best-effort VCs (similar to the ATM Forum service category UBR); therefore, there is no bandwidth guarantee from the RS, which is not used for TVCs. All of the TVCs fall into one of the four TBR classes, each carrying a different default relative weight. The default values of the relative weights for the four TBR classes are configurable, so you can change the priority of the default values.

Table 15-1 lists the TBR classes and ATM Forum class mappings into the service classes for physical ports.

Table 15-1 Service Class to Weight Mapping for Physical Ports

TBR Class	Service Class	Relative Weight
TBR_1 (WRR_1)	1	1
TBR_2 (WRR_2)	6	2
TBR_3 (WRR_3)	7	3
TBR_4 (WRR_4)	8	4

ATM Forum Service Category	Service Class	Relative Weight
CBR[1]	2	8
VBR-RT	2	8
VBR-NRT	3	1
ABR	4	1
UBR	5	1

1. Even though the CBR service category is mapped to service class 2, all of the CBR VCs are rate scheduled only, and therefore they are not WRR scheduled.

When tag switching is enabled on a hierarchical VP tunnel, the tunnel can only be used for tag switching. Because hierarchical VP tunnels support only four service classes, both TVCs and ATM Forum VCs map to the same service classes. Therefore, both ATM Forum VCs and TVCs cannot coexist in a hierarchical VP tunnel. The relative weights assigned to the service classes depend on which is active (either tag switching or ATM Forum). The class weights change whenever a hierarchical VP tunnel is toggled between ATM Forum and tag switching. By default, a hierarchical VP tunnel comes up as an ATM Forum port.

Table 15-2 lists the TBR classes and ATM Forum service category mappings for hierarchical VP tunnels.

Table 15-2 Service Class to Weight Mapping for Hierarchical VP Tunnels

TBR Class	Service Class	Relative Weight
TBR_1 (WRR_1)	1	1
TBR_2 (WRR_2)	2	2
TBR_3 (WRR_3)	3	3
TBR_4 (WRR_4)	4	4

ATM Forum Service Category	Service Class	Relative Weight
VBR-RT	1	8
VBR-NRT	2	1
ABR	3	1
UBR	4	1

Configuring the Service Class and Relative Weight

Each service class is assigned a relative weight. These weights are configurable and range from 1 to 15.

To configure the service class and relative weight on a specific interface, perform the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **interface atm** *card*/*subcard*/*port*[**.***vpt#*] Switch(config-if)#	Specifies an ATM interface and enters interface configuration mode.
Step 2	Switch(config-if)# **atm service-class** {**1** | **6** | **7** | **8**} **wrr-weight** *weight* or Switch(config-if)# **atm service-class** {**1** | **2** | **3** | **4**} **wrr-weight** *weight*	Enters the service class and relative weight for a physical interface. or Enters the service class and relative weight for a hierarchical interface.

Example

In the following example, ATM interface 0/0/3 is configured with service class 1 and a WRR weight of 3:

```
Switch(config)# interface atm 0/0/3
Switch(config-if)# atm service-class 1 wrr-weight 3
```

Displaying the TVC Configuration

To display the TVC configuration, perform the following task in EXEC mode:

Command	Purpose
show atm vc interface atm *card*/*subcard*/*port* [*vpi vci*]	Displays the ATM layer connection information about the virtual connection.

The following example shows the service category of the TVC:

```
Switch# show atm vc interface atm 0/0/3 1 35
Interface: ATM0/0/3, Type: oc3suni
VPI = 1  VCI = 35
Status: UP
Time-since-last-status-change: 1d00h
Connection-type: TVC(I)
Cast-type: multipoint-to-point-input
Packet-discard-option: enabled
Usage-Parameter-Control (UPC): pass
Wrr weight: 2
Number of OAM-configured connections: 0
OAM-configuration: disabled
OAM-states:  Not-applicable
Cross-connect-interface: ATM0/1/3.10, Type: oc3suni
Cross-connect-VPI = 10
Cross-connect-VCI = 34
Cross-connect-UPC: pass
Cross-connect OAM-configuration: disabled
Cross-connect OAM-state:  Not-applicable
Threshold Group: 7, Cells queued: 0
Rx cells: 0, Tx cells: 0
Tx Clp0:0,  Tx Clp1: 0
Rx Clp0:0,  Rx Clp1: 0
Rx Upc Violations:0, Rx cell drops:0
Rx pkts:0, Rx pkt drops:0
Rx connection-traffic-table-index: 63998
➔ Rx service-category: WRR_1 (WRR Bit Rate)
Rx pcr-clp01: none
Rx scr-clp01: none
Rx mcr-clp01: none
Rx      cdvt: 1616833580 (from default for interface)
Rx       mbs: none
Tx connection-traffic-table-index: 63998
➔ Tx service-category: WRR_1 (WRR Bit Rate)
Tx pcr-clp01: none
Tx scr-clp01: none
Tx mcr-clp01: none
Tx      cdvt: none
Tx       mbs: none
```

Threshold Group for TBR Classes

A threshold group utilizes the memory efficiently among VCs of a particular traffic type. Each threshold group is programmed with a dynamic memory allocation profile that maps into the needs of the connections of a particular service class. There are 16 threshold groups (0 to 15) available on the ATM switch router. Each threshold group has a set of eight regions, and each region has a set of thresholds. When these thresholds are exceeded, cells are dropped to maintain the integrity of the shared memory resource.

Each ATM Forum service category is mapped into a distinct threshold group. All the connections in a particular service category map into one threshold group. Similarly, all the Tag Bit Rate (TBR) classes have best effort traffic and the service differentiation comes mainly by giving different weights. Each of the TBR classes map into four different threshold groups whose parameters are the same as the unspecified bit rate (UBR) threshold group.

Table 15-3 shows the threshold group parameters mapped to the connections in all of the TBR classes for the Catalyst 8540 MSR.

Table 15-3 Threshold Group Parameters for TVCs (Catalyst 8540 MSR)

Group	Maximum Cells	Maximum Queue Limit	Minimum Queue Limit	Mark Threshold	Discard Threshold	Use
7	131,071	511	31	25%	87%	TBR_1
8	131,071	511	31	25%	87%	TBR_2
9	131,071	511	31	25%	87%	TBR_3
10	131,071	511	31	25%	87%	TBR_3

Table 15-4 shows the threshold group parameters mapped to the connections in all of the TBR classes for the Catalyst 8510 MSR and LightStream 1010 ATM switch routers.

Table 15-4 Threshold Group Parameters for TVCs (Catalyst 8510 MSR and LightStream 1010)

Group	Maximum Cells	Maximum Queue Limit	Minimum Queue Limit	Mark Threshold	Discard Threshold	Use
7	65,535	511	31	25%	87%	TBR_1
8	65,535	511	31	25%	87%	TBR_2
9	65,535	511	31	25%	87%	TBR_3
10	65,535	511	31	25%	87%	TBR_3

Each threshold group is divided into eight regions. Each region has a set of thresholds that are calculated from the corresponding threshold group parameters given in Table 15-3. The threshold group might be in any one of the regions depending on the fill level (cell occupancy) of that group. And that region is used to derive the set of thresholds which apply to all the connections in that group.

Table 15-5 gives the eight thresholds for threshold groups 6, 7, 8, and 9.

Table 15-5 Region Thresholds for Threshold Groups

Region	Lower Limit	Upper Limit	Queue Limit	Marking Threshold	Discard Threshold
0	0	8191	511	127	447
1	8128	16,383	255	63	223
2	16,320	24,575	127	31	111
3	24,512	32,767	63	15	63
4	32,704	40,959	31	15	31
5	40,896	49,151	31	15	31
6	49,088	57,343	31	15	31
7	57,280	65,535	31	15	31

For more information about threshold groups and configuration parameters, see the “Overview of Threshold Groups” section on page 8-15 and the *Guide to ATM Technology*.

CTT Row

A row in the connection traffic table (CTT) is created for each unique combination of traffic parameters. When a TVC is set up in response to a request by tag switching, a CTT row is obtained from the resource manager by passing the traffic parameters that include the service category (TBR_*x* [WRR_*x*], where *x* is 1, 2, 3, or 4). If a match is found for the same set of traffic parameters, the row index is returned; otherwise a new table is created and the row index of that CTT row is returned. Since all data TVCs use the same traffic parameters, the same CTT row can be used for all TVCs of a particular service category once it is created.

Note There are no user configurable parameters for the CTT with TVCs.

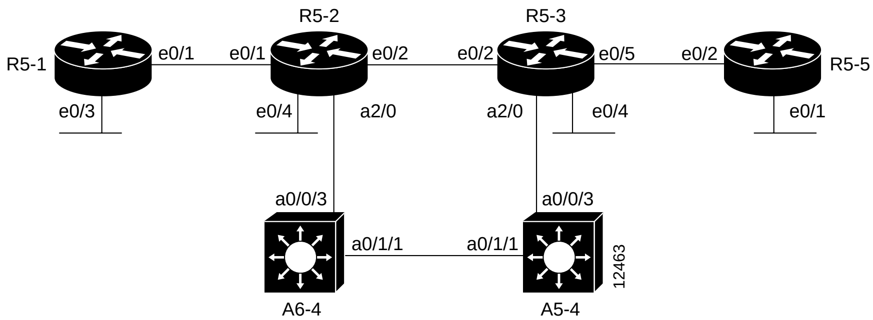
RM CAC Support

Connection admission control (CAC) is not supported for tag virtual channels (TVCs). All TVCs are best effort connections; therefore, no bandwidth is guaranteed by the RS. Only the WRR scheduler is used. So, all of the traffic parameters (PCR, MCR, MBS, CDVT, and SCR) are unspecified. There is no best effort limit like there is with ATM Forum UBR and ABR connections. CAC is bypassed for TVCs.

Tag Switching Configuration Example

Figure 15-4 shows an example tag switching network.

Figure 15-4 Example Network for Tag Switching



Router 5-1 Configuration

The configuration of router R5-1, interface e0/1, follows:

```
router_R5-1# configure terminal
router_R5-2(config)# ip cef switch
router_R5-1(config)# tag-switching advertise-tags
router_R5-1(config)# interface e0/1
router_R5-1(config-if)# tag-switching ip
router_R5-1(config-if)# exit
router_R5-1(config)#
```

Router 5-2 Configuration

The configuration between router R5-1, interface e0/1, and R5-2, interface e0/1, follows:

```
router_R5-2# configure terminal
router_R5-2(config)# ip cef switch
router_R5-2(config)# tag-switching advertise-tags
router_R5-2(config)# interface e0/1
router_R5-2(config-if)# tag-switching ip
router_R5-2(config-if)# exit
router_R5-2(config)#
```

The configuration between router R5-2, interface e0/2, and R5-3, interface e0/2, follows:

```
route_R5-2(config)# interface e0/2
route_R5-2(config-if)# tag-switching ip
route_R5-2(config-if)# exit
```

The configuration of router R5-2, interface a2/0.1, follows:

```
router_R5-2(config-if)# interface a2/0.1
router_R5-2(config-subif)# ip address 189.26.11.15 255.255.0.0
router_R5-2(config-subif)# tag-switching ip
router_R5-2(config-subif)# no shutdown
router_R5-2(config-subif)# exit
router_R5-2(config)# interface a2/0
router_R5-2(config)# no shutdown
```

Router 5-3 Configuration

The configuration of router R5-3, interface e0/2, follows:

```
router_R5-3# configure terminal
router_R5-3(config)# ip cef switch
router_R5-3(config)# tag-switching advertise-tags
router_R5-3(config)# interface e0/2
router_R5-3(config-if)# tag-switching ip
router_R5-3(config-if)# exit
```

The configuration of router R5-3, interface e0/5 follows:

```
router_R5-3(config)# interface e0/5
router_R5-3(config-if)# tag-switching ip
router_R5-3(config-if)# exit
```

The configuration of router R5-3, interface atm 2/0.1, follows:

```
router_R5-3# configure terminal
router_R5-3(config)# interface atm 2/0.1
router_R5-3(config-if)# ip address 189.25.12.13 255.255.0.0
router_R5-3(config-if)# tag-switching ip
router_R5-3(config-if)# no shutdown
router_R5-3(config-if)# exit
router_R5-3(config)# interface a2/0
router_R5-3(config-if)# no shutdown
```

ATM Switch Router A5-4 Configuration

The configuration of ATM switch router A5-4, interfaces atm 0/1/1 and atm 0/0/3, follows:

```
atm_A5-4# configure terminal
atm_A5-4(config)# interface atm 0/1/1
atm_A5-4(config-if)# no shutdown
atm_A5-4(config-if)# ip address 189.24.15.12 255.255.0.0
atm_A5-4(config-if)# tag-switching ip
atm_A5-4(config-if)# exit
atm_A5-4(config)# tag-switching ip
atm_A5-4(config)# interface atm 0/0/3
atm_A5-4(config-if)# no shutdown
atm_A5-4(config-if)# ip address 189.25.15.11 255.255.0.0
atm_A5-4(config-if)# tag-switching ip
atm_A5-4(config-if)# exit
atm_A5-4(config)# tag-switching ip
```

Router 5-5 Configuration

The configuration of router R5-5, interface e0/2, follows:

```
router_R5-5# configure terminal
router_R5-5(config)# ip cef switch
router_R5-5(config)# tag-switching advertise-tags
router_R5-5(config)# interface e0/2
router_R5-5(config-if)# tag-switching ip
router_R5-5(config-if)# exit
```

ATM Switch Router A6-4 Configuration

The configuration of ATM switch router A6-4, interface atm 0/1/1, follows:

```
atm_A6-4# configure terminal
atm_A6-4(config)# interface atm 0/1/1
atm_A6-4(config-if)# no shutdown
```

```
atm_A6-4(config-if)# ip address 189.24.14.12 255.255.0.0
atm_A6-4(config-if)# tag-switching ip
atm_A6-4(config-if)# exit
```

The configuration of ATM switch router A6-4, interface atm 0/0/3, follows:

```
atm_A6-4# configure terminal
atm_A6-4(config)# interface atm 0/0/3
atm_A6-4(config-if)# no shutdown
atm_A6-4(config-if)# ip address 189.26.14.11 255.255.0.0
atm_A6-4(config-if)# tag-switching ip
atm_A6-4(config-if)# exit
```

CHAPTER 16

Configuring Signalling Features

This chapter describes signalling-related features and their configuration for the ATM switch router.

Note This chapter provides advanced configuration instructions for the Catalyst 8540 MSR, Catalyst 8510 MSR, and LightStream 1010 ATM switch routers. For general information about ATM signalling protocols, refer to the *Guide to ATM Technology*. For complete descriptions of the commands mentioned in this chapter, refer to the *ATM Switch Router Command Reference* publication.

This chapter includes the following sections:

Configuring Signalling IE Forwarding

You enable signalling information element (IE) forwarding of the specified IE from the calling party to the called party.

Note The default is to transfer all the information elements in the signalling message.

To configure interface signalling IE transfer, perform the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **interface atm** *card/subcard/port* Switch(config-if)#	Selects the interface to be configured.
Step 2	Switch(config-if)# **atm signalling ie forward** {**aal-info** | **all** | **bli-repeat-ind** | **called-subaddress** | **calling-number** | **higher-layer-info** | **lower-layer-info** | **unknown-ie**}	Configures the signalling information element forwarding.

Example

The following example shows how to disable signalling of all forwarded IEs on ATM interface 0/0/0:

```
Switch(config)# interface atm 0/0/0
Switch(config-if)# no atm signalling ie forward all
```

Displaying the Interface Signalling IE Forwarding Configuration

To display the interface signalling IE forwarding configuration, use the following privileged EXEC command:

Command	Purpose
more system:running-config	Displays the interface signalling IE forwarding configuration.

Example

The following example displays the modified configuration of the signalling IE forwarding:

```
Switch# more system:running-config
Building configuration...

Current configuration:
!
version XX.X
no service pad
service udp-small-servers
service tcp-small-servers
!
hostname Switch
!
<information deleted>
!
interface ATM0/0/0
 no atm signallling ie forward calling-number
 no atm signallling ie forward calling-subaddress
 no atm signallling ie forward called-subaddress
 no atm signallling ie forward higher-layer-info
 no atm signallling ie forward lower-layer-info
 no atm signallling ie forward blli-repeat-ind
 no atm signallling ie forward aal-info
!
<information deleted>
```

Configuring ATM SVC Frame Discard

You can select the criteria used to install frame discard on switched virtual channels (SVCs). The default is to install packet discard based on the presence of the ATM adaptation layer 5 (AAL5) information element in the SETUP message.

Note The term *frame discard* is referred to as *packet discard* on ATM switch router virtual circuits.

You can use this global configuration function to modify frame discard for all connections.

To configure frame discard, use the following command in global configuration mode:

Command	Purpose
atm svc-frame-discard-on-aal5ie	Configures the SVC frame discard.

This command changes the information that the ATM switch router uses to decide whether or not to install frame discard on SVCs. User-Network Interface (UNI) 4.0 signalling allows for explicit signalling of frame discard. Pre-UNI 4.0 versions use the presence of the AAL5 information elements to determine whether or not to install frame discard. If the AAL5 information element is present, frame discard is installed; otherwise it is not, as shown in the following example.

- When you configure **atm svc-frame-discard-on-aal5ie**, frame discard is installed if the AAL5 information element is present.
- When you configure **no atm svc-frame-discard-on-aal5ie**, frame discard is installed on UNI 4 or PNNI interfaces if explicitly requested by the SETUP and CONNECT messages.

Example

In the following example, the ATM switch router behavior is set to not use the AAL5 information element to dictate frame discard.

```
Switch(config)# no atm svc-frame-discard-on-aal5ie
```

Displaying the ATM Frame Discard Configuration

To display the ATM frame discard configuration, use the following privileged EXEC command:

Command	Purpose
more system:running-config	Displays the frame discard configuration.

Example

The following example shows how to display the frame discard configuration:

```
Switch# more system:running-config
Building configuration...

Current configuration:
!
version XX.X
no service pad
service udp-small-servers
service tcp-small-servers
!
hostname Switch
!
network-clock-select 1 ATM0/0/0
network-clock-select 4 ATM0/0/0
ip host-routing
no atm svc-frame-discard-on-aal5ie
!
<information deleted>
```

Configuring E.164 Addresses

E.164 support allows networks that use network service access point (NSAP) ATM addresses formats (for example, 45.000001234567777F00000000.000000000000.00) to work with networks that use E.164 address formats (for example, 1–123–456–7777). For an overview of address types and E.164 subtypes, refer to the *Guide to ATM Technology*.

The following sections describe configuring E.164 support:

- E.164 Conversion Methods, page 16-5
- Configuring E.164 Gateway, page 16-5
- Configuring E.164 Address Autoconversion, page 16-8
- Configuring E.164 Address One-to-One Translation Table, page 16-9

E.164 Conversion Methods

There are three features you can configure on the ATM switch router for E.164 address conversion. The feature you chose depends on the address format you are using. The features are as follows:

- E.164 gateway—Use this feature when addresses are in international code designator (ICD) or data country code (DCC) format and a call must traverse an E.164 network.
- E.164 address autoconversion—Use this feature when addresses are in E164_ZDSP or E.164_AESA format and a call must traverse an E.164 network. An E.164_AESA uses the ATM end system address (AESA) format with the E.164 number embedded; an E164_ZDSP is an E164_AESA address with all zeros after the embedded E.164 number; for example, 45.000001234567777F00000000.000000000000.00.
- E.164 address one-to-one translation table—Use this feature when you want to create an E.164 to AESA address translation table manually. This feature is not recommended for most networks.

Caution Manually creating the E.164 to AESA address translation table is a time consuming and error prone process. We strongly recommend that you use either the E.164 gateway or E.164 autoconversion feature instead of the E.164 one-to-one address translation feature.

Configuring E.164 Gateway

The E.164 gateway feature allows calls with AESAs to be forwarded, based on prefix matching, on interfaces that are statically mapped to E.164 addresses. To configure the E.164 gateway feature, you must first configure a static ATM route with an E.164 address, then configure the E.164 address to use on the interface.

When a static route is configured on an interface, all ATM addresses that match the configured address prefix are routed through that interface to an E.164 address.

Signalling uses E.164 addresses in the called and calling party IEs, and uses AESAs in the called and calling party subaddress IEs. For a detailed description of how the E.164 gateway feature works, refer to the *Guide to ATM Technology*.

Note Enter access lists for E.164 addresses in the E164_AESA format, not native E.164 format. For example, if the E.164 address is 7654321, then the E164_AESA format is 45.000000007654321F00000000.000000000000.00. To filter prefix “765”, enter the prefix 45.00000000765..., not just 765.... Access lists operate on the called and calling party IEs. See Chapter 11, “Using Access Control.”

Configuring an E.164 Address Static Route

To configure an E.164 address static route, use the following command in global configuration mode:

Command	Purpose
atm route *address-prefix* **atm** *card*/*subcard*/*port* [**e164-address** *address-string* [**number-type** {**international** | **local** | **national** | **subscriber**}]] [**internal**] [**scope** *org-scope*]	At the configure prompt, configures the static route prefix with the E.164 address.

Example

The following example uses the **atm route** command to configure a static route using the 13-byte switch prefix 47.00918100000000410B0A1081 to ATM interface 0/0/0 with the E.164 address 1234567:

```
Switch(config)# atm route 47.00918100000000410B0A1081 atm 0/0/0 e164-address 7654321
```

To complete the E.164 address static route configuration, proceed to the "Configuring an ATM E.164 Address on an Interface" section on page 16-7.

Displaying the E.164 Static Route Configuration

To display the E.164 address configuration, use the following privileged EXEC command:

Command	Purpose
show atm route	Displays the static route E.164 address configuration.

Example

The following example displays the E.164 address configuration using the **show atm route** privileged EXEC command:

```
Switch# show atm route
Codes: P - installing Protocol (S - Static, P - PNNI, R - Routing control),
       T - Type (I - Internal prefix, E - Exterior prefix, SE -
                 Summary Exterior prefix, SI - Summary Internal prefix,
                 ZE - Suppress Summary Exterior, ZI - Suppress Summary Internal)
P  T Node/Port        St Lev Prefix
~ ~~ ~~~~~~~~~~~~~~~~ ~~ ~~~ ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
S  E 1   ATM0/1/0     DN 0   47.0091.8100.0000.0001/72
P SI 1   0            UP 0   47.0091.8100.0000.0002.eb1f.fe00/104
R  I 1   ATM2/0/0     UP 0   47.0091.8100.0000.0002.eb1f.fe00.0002.eb1f.fe00/152
R  I 1   ATM2/0/0     UP 0   47.0091.8100.0000.0002.eb1f.fe00.4000.0c/128
P SI 1   0            UP 0   47.0091.8100.0000.0040.0b0a.2b81/104
S  E 1   ATM0/0/0     DN 0   47.0091.8100.0000.0040.0b0a.2b81/104
                             (E164 Address 1234567)
R  I 1   ATM2/0/0     UP 0   47.0091.8100.0000.0040.0b0a.2b81.0040.0b0a.2b81/152
R  I 1   ATM2/0/0     UP 0   47.0091.8100.0000.0040.0b0a.2b81.4000.0c/128
```

Configuring an ATM E.164 Address on an Interface

One E.164 address can be configured per ATM port. Signalling uses E.164 addresses in the called and calling party IEs, and uses AESA addresses in the called and calling party subaddress IEs.

To configure an E.164 address on a per-interface basis, perform the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **interface atm** *card*/*subcard*/*port* Switch(config-if)#	Selects an interface port.
Step 2	Switch(config-if)# **atm e164 address** *e164-address*	Associates the E.164 address to the interface.

Example

The following example shows how to configure the E.164 address 7654321 on ATM interface 0/0/1:

```
Switch(config)# interface atm 0/0/1
Switch(config-if)# atm e164 address 7654321
```

Displaying the E.164 Address Association to Interface Configuration

To display the E.164 configuration, use the following EXEC command:

Command	Purpose
show atm interface atm *card*/*subcard*/*port*	Shows the E.164 address configuration on a per-port basis.

Example

The following example shows how to display the E.164 address configuration for ATM interface 0/0/1:

```
Switch# show atm interface atm 0/0/1

Interface:      ATM0/0/1        Port-type:    oc3suni
IF Status:      UP              Admin Status: up
Auto-config:    enabled         AutoCfgState: completed
IF-Side:        Network         IF-type:      NNI
Uni-type:       not applicable  Uni-version:  not applicable
Max-VPI-bits:   8               Max-VCI-bits: 14
Max-VP:         255             Max-VC:       16383
ConfMaxSvpcVpi: 255             CurrMaxSvpcVpi: 255
ConfMaxSvccVpi: 255             CurrMaxSvccVpi: 255
ConfMinSvccVci: 35              CurrMinSvccVci: 35
Svc Upc Intent: pass            Signalling:   Enabled
ATM Address for Soft VC: 47.0091.8100.0000.0041.0b0a.1081.4000.0c80.0010.00
➔ ATM E164 Address: 7654321
<information deleted>
```

When the E.164 gateway feature is configured, the switch first attempts to make a connection using the E.164 gateway feature. If that connection fails, the switch attempts to make the connection using the E.164 address autoconversion feature, described in the following section.

Configuring E.164 Address Autoconversion

If your network uses E164_ZDSP or E164_AESA addresses, you can configure E.164 address autoconversion. The E164_ZDSP and E164_AESA addresses include an embedded E.164 number in the E.164 portion of an E.164 ATM address. This embedded E.164 number is used in the autoconversion process.

For a detailed description of the E.164 autoconversion feature and differences in the autoconversion process between the E164_ZDSP and E164_AESA address formats, refer to the *Guide to ATM Technology*.

Note Enter access lists for E.164 addresses in the E164_AESA format, not the native E.164 format. For example, if the E.164 address is 7654321, then the E164_AESA format is 45.000000007654321F00000000.000000000000.00. To filter prefix "765," enter the prefix 45.00000000765..., not just 765.... Access lists operate on the called and calling party IEs. See Chapter 11, "Using Access Control.".

E.164 address autoconversion configuration is the same, regardless of which type of address (E164_ZDSP or E164_AESA) your network uses. To configure E.164 address autoconversion, perform the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **atm route** *address-prefix* **atm** *card/subcard/port* [**e164-address** *address-string* [**number-type** {**international** | **local** | **national** | **subscriber**}]] [**internal**] [**scope** *org-scope*]	At the configure prompt, configures the static route prefix with the E.164 address.
Step 2	Switch(config-if)# **interface atm** *card/subcard/port* Switch(config-if)#	Selects the ATM interface.
Step 3	Switch(config-if)# **atm e164 auto-conversion**	Configures E.164 autoconversion.
Step 4	Switch(config-if)# **exit** Switch(config)#	Returns to global configuration mode.

Examples

In the following example a static route is configured on interface 0/0/1 using the ATM address of the ATM switch router on the opposite side of the E.164 public network; E.164 autoconversion is also enabled:

```
Switch(config)# atm route 45.000007654321111F atm 0/0/1
Switch(config)# interface atm 0/0/1
Switch(config-if)# atm e164 auto-conversion
```

The converse configuration is done at the ATM switch router across the E.164 network; a static route is configured to the ATM address of the above switch, and E.164 autoconversion is enabled:

```
Switch(config)# atm route 45.000001234567777F atm 0/0/1
Switch(config)# interface atm 0/0/1
Switch(config-if)# atm e164 auto-conversion
```

Displaying the E.164 Address Autoconversion

To display the E.164 configuration on an interface, use the following EXEC command:

Command	Purpose
show atm interface atm *card/subcard/port*	Shows the E.164 address configuration on a per-port basis.

Example

The following example shows how to display the E.164 configuration for ATM interface 0/0/1:

```
Switch# show atm interface atm 0/0/1

Interface:      ATM0/0/1        Port-type:      oc3suni
IF Status:      DOWN            Admin Status:   down
Auto-config:    disabled        AutoCfgState:   not applicable
IF-Side:        Network         IF-type:        UNI
Uni-type:       Private         Uni-version:    V3.0
Max-VPI-bits:   8               Max-VCI-bits:   14
Max-VP:         255             Max-VC:         16383
ConfMaxSvpcVpi: 255             CurrMaxSvpcVpi: 255
ConfMaxSvccVpi: 255             CurrMaxSvccVpi: 255
ConfMinSvccVci: 33              CurrMinSvccVci: 33
Svc Upc Intent: pass            Signalling:     Enabled
ATM Address for Soft VC: 47.0091.8100.0000.0002.eb1f.fe00.4000.0c80.0010.00
ATM E164 Auto Conversion Interface
Configured virtual links:
  PVCLs SoftVCLs   SVCLs   TVCLs   PVPLs SoftVPLs   SVPLs Total-Cfgd Inst-Conns
      2        0       0       0       0        0       0          2          0
Logical ports(VP-tunnels):     0
Input cells:    0               Output cells:   0
5 minute input rate:             0 bits/sec,       0 cells/sec
5 minute output rate:            0 bits/sec,       0 cells/sec
Input AAL5 pkts: 0, Output AAL5 pkts: 0, AAL5 crc errors: 0
```

Configuring E.164 Address One-to-One Translation Table

The ATM interface to a public network commonly uses an E.164 address for ATM signalling, with international code designator (ICD) or data country code (DCC) format AESA addresses carried in the subaddress fields of the message. The one-to-one translation table allows signalling to look up the E.164 addresses and the AESA addresses in a database, allowing a one-to-one correspondence between AESA addresses and E.164 addresses.

Caution Manually mapping AESA addresses to E.164 addresses is a time consuming and error prone process. We highly recommend that you use either the E.164 gateway or E.164 autoconversion feature instead of the E.164 one-to-one address translation feature.

For a detailed explanation of how the E.164 translation table feature works, refer to the *Guide to ATM Technology*.

Configuring one-to-one E.164 translation tables requires the following steps:

Step 1 Configure specific ATM interface(s) to connect to E.164 public networks to use the translation table.

Step 2 Configure the translation table.

Step 3 Add entries to the translation table for both the called and calling parties.

To configure E.164 translation on the interface, perform the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **interface atm** *card*/*subcard*/*port* Switch(config-if)#	Selects an interface port.
Step 2	Switch(config-if)# **atm e164 translation**	Configures the ATM E.164 interface.
Step 3	Switch(config-if)# **exit** Switch(config)#	Returns to EXEC configuration mode.
Step 4	Switch(config)# **atm e164 translation-table** Switch(config-atm-e164)#	Changes to E.164 ATM configuration mode.
Step 5	Switch(config-atm-e164)# **e164 address** *address* **nsap-address**[1] *nsap-address*	Configures the E.164 translation table.

1. The NSAP address is the same as the ARB_AESA address.

Example

The following example shows how to configure the ATM interface 0/0/1 to use the one-to-one E.164 translation table and specifies three table entries:

```
Switch(config)# interface atm 0/0/1
Switch(config-if)# atm e164 translation
Switch(config-if)# exit
Switch(config)# atm e164 translation-table
Switch(config-atm-e164)# e164 address 1111111 nsap-address 11.1111111111111111111111111.112233445566.11
Switch(config-atm-e164)# e164 address 2222222 nsap-address 22.2222222222222222222222222.112233445566.22
Switch(config-atm-e164)# e164 address 3333333 nsap-address 33.3333333333333333333333333.112233445566.33
```

Displaying the ATM E.164 Translation Table Configuration

To display the ATM E.164 translation table configuration, use the following privileged EXEC commands:

Command	Purpose
more system:running-config	Displays the E.164 translation table configuration.
show atm interface atm *card*/*subcard*/*port*	Displays the E.164 address configuration on a per-port basis.

Example

The following example shows how to display the E.164 translation table configuration:

```
Switch# more system:running-config
Building configuration...

Current configuration:
!
version XX.X
no service pad
service udp-small-servers
service tcp-small-servers
!
hostname Switch
!
atm e164 translation-table
 e164 address 1111111 nsap-address 11.111111111111111111111111.112233445566.11
 e164 address 2222222 nsap-address 22.222222222222222222222222.112233445566.22
 e164 address 3333333 nsap-address 33.333333333333333333333333.112233445566.33
!
atm service-category-limit cbr 64544
atm service-category-limit vbr-rt 64544
atm service-category-limit vbr-nrt 64544
atm service-category-limit abr-ubr 64544
atm address 47.0091.8100.0000.0040.0b0a.2b81.0040.0b0a.2b81.00
!
<information deleted>
```

Example

The following example shows how to display the E.164 configuration for ATM interface 0/0/1:

```
Switch# show atm interface atm 0/0/1

Interface:      ATM0/0/1        Port-type:    oc3suni
IF Status:      DOWN            Admin Status: administratively down
Auto-config:    enabled         AutoCfgState: waiting for response from peer
IF-Side:        Network         IF-type:      UNI
Uni-type:       Private         Uni-version:  V3.0
Max-VPI-bits:   8               Max-VCI-bits: 14
Max-VP:         255             Max-VC:       16383
ConfMaxSvpcVpi: 255             CurrMaxSvpcVpi: 255
ConfMaxSvccVpi: 255             CurrMaxSvccVpi: 255
ConfMinSvccVci: 35              CurrMinSvccVci: 35
Svc Upc Intent: pass            Signalling:   Enabled
ATM Address for Soft VC: 47.9999.9999.0000.0000.0000.0216.4000.0c80.0010.00
➔ ATM E164 Translation Interface
Configured virtual links:
  PVCLs SoftVCLs   SVCLs   PVPLs SoftVPLs   SVPLs  Total-Cfgd  Installed-Conns
      2        0       0       0        0       0           2                0
Logical ports(VP-tunnels):     0
Input cells:    0               Output cells: 0
5 minute input rate:             0 bits/sec,       0 cells/sec
5 minute output rate:            0 bits/sec,       0 cells/sec
Input AAL5 pkts: 0, Output AAL5 pkts: 0, AAL5 crc errors: 0
```

Configuring Signalling Diagnostics Tables

Signalling diagnostics enable you to diagnose a specific call failure in your network and pinpoint the location of the call failure along with the reason for the failure. To do this, you must configure a signalling diagnostics table that stores the filtering criteria and a filter index, an integer value between 1 and 50, used to uniquely identify each set of filtering criteria you select. Each filtering criteria occupies one entry in the signalling diagnostics table. Each entry in the filter table is entered using command-line interface (CLI) commands or Simple Network Management Protocol (SNMP). Then the diagnostics software module, when enabled, filters rejected calls based on the entries in your filter table. A successful match in the filter table causes the rejected call information to be stored for analysis.

Note Signalling diagnostics is a tool for troubleshooting failed calls and should not be enabled during normal operation of the ATM switch router.

To configure the signalling diagnostics table entries, perform the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **atm signalling diagnostics enable**	Enables ATM signalling diagnostics.
Step 2	Switch(config)# **atm signalling diagnostics** *index* Switch(config-atmsig-diag)#	Changes to ATM signalling diagnostics configuration mode.
Step 3	Switch(config-atmsig-diag)# **age-timer** *seconds*	Configures the timeout value for the entry, in seconds.
Step 4	Switch(config-atmsig-diag)# **called-nsap-address** *nsap-address*	Configures a filtering criteria based on the called NSAP address of the rejected call.
Step 5	Switch(config-atmsig-diag)# **called-address-mask** *nsap-address-mask*[1]	Configures a filtering criteria based on the called address mask value used to identify the valid bits of the calling NSAP address of the rejected call.
Step 6	Switch(config-atmsig-diag)# **calling-nsap-address** *nsap-address*	Configures a filtering criteria based on the calling NSAP address of the rejected call.
Step 7	Switch(config-atmsig-diag)# **atm signalling diagnostics enable**	Enables ATM signalling diagnostics.
Step 8	Switch(config-atmsig-diag)# **clear-cause** *clear-cause-code*[2]	Configures a filtering criteria based on the cleared cause code of the rejected call.
Step 9	Switch(config-atmsig-diag)# **connection-category** {**soft-vc** | **soft-vp** | **reg-vc** | **all**}	Configures a filtering criteria based on the VC connection category of the rejected call.
Step 10	Switch(config-atmsig-diag)# **incoming-port atm** *card/subcard/port*	Configures a filtering criteria based on the incoming port of the rejected call.
Step 11	Switch(config-atmsig-diag)# **outgoing-port atm** *card/subcard/port*	Configures a filtering criteria based on the outgoing port of the rejected call.
Step 12	Switch(config-atmsig-diag)# **max-records** *max-num-records*	Configures the maximum number of entries to be stored in the display table for each of the entries in the filter table.

	Command	Purpose
Step 13	Switch(config-atmsig-diag)# **purge**	Purges all the filtered records in the filter table.
Step 14	Switch(config-atmsig-diag)# **scope** {**internal** | **external**}	Configures a filtering criteria based on the scope of the rejected call which either failed internally in the switch or externally on other switches.
Step 15	Switch(config-atmsig-diag)# **service-category** {**cbr** | **abr** | **vbr-rt** | **vbr-nrt** | **ubr** | **all**}	Configures a filtering criteria based on the service category of the rejected call.
Step 16	Switch(config-atmsig-diag)# **status** [**active** *filter-criteria* | **inactive** *filter-criteria* | **delete** *filter-criteria*]	Configures the status of the entry in the filter table.

1. The combination of the configured *calling_addr_mask* (*called_address_mask*) and the configured *calling_nsap_address* (*called_nsap_address*) are used to filter the rejected call.
2. You can obtain the cause code values from the ATM forum UNI3.1 specification.

The display table contains the records that were collected based on every filtering criteria in the filter table. Each filtering criteria has only a specified number of records that are stored in the table. After that specified number of records is exceeded, the table is overwritten.

Examples

The following example shows how to enable signalling diagnostics on the ATM switch router:

```
Switch(config)# atm signalling diagnostics enable
```

The following example shows how to change to signalling diagnostics mode on the ATM switch router:

```
Switch(config)# atm signalling diagnostics 1
Switch(cfg-atmsig-diag)#
```

The following example shows how to specify the timeout value for the entry in seconds:

```
Switch(config)# atm signalling diagnostics 1
Switch(cfg-atmsig-diag)# age-timer 3600
```

The following example shows how to configure filter criteria for calls rejected based on the called NSAP address of the call:

```
Switch(config)# atm signalling diagnostics 1
Switch(cfg-atmsig-diag)# called-nsap-address 47.0091810000000061705BD901.010203040506.0
```

The following example shows how to configure filter criteria for calls rejected based on the called address mask of the call:

```
Switch(config)# atm signalling diagnostics 1
Switch(cfg-atmsig-diag)# called-address-mask ff.ff.ff.00
```

The following example shows how to configure filter criteria for calls rejected based on the connection type:

```
Switch(config)# atm signalling diagnostics 1
Switch(cfg-atmsig-diag)# cast-type p2p p2mp
```

The following example shows how to configure the filter entry for filtering failed calls based on the clear cause value 3 (destination unreachable):

```
Switch(config)# atm signalling diagnostics 1
Switch(cfg-atmsig-diag)# clearcause 3
```

The following example shows how to configure filter criteria for call failures based on the category of the virtual circuit:

```
Switch(config)# atm signalling diagnostics 1
Switch(cfg-atmsig-diag)# connection-category soft-vc
Switch(cfg-atmsig-diag)# connection-category soft-vc soft-vp
```

The following example shows how to configure the filter entry for filtering failed calls that came in through ATM interface 1/1/1:

```
Switch(config)# atm signalling diagnostics 1
Switch(cfg-atmsig-diag)# incoming-port atM 1/1/1
```

The following example shows how to configure the filter entry for filtering failed calls that went out through ATM interface 1/1/1:

```
Switch(config)# atm signalling diagnostics 1
Switch(cfg-atmsig-diag)# outgoing-port atm 1/1/1
```

The following example shows how to specify the maximum number of entries to be stored in the display table for each of the entries in the filter table:

```
Switch(config)# atm signalling diagnostics 1
Switch(cfg-atmsig-diag)# max-records 40
```

The following example shows how to purge all the filtered records corresponding to this entry in the filter table:

```
Switch(config)# atm signalling diagnostics 1
Switch(cfg-atmsig-diag)# purge
```

The following example shows how to configure filter criteria for calls that failed internally in the switch:

```
Switch(config)# atm signalling diagnostics 1
Switch(cfg-atmsig-diag)# scope internal
```

The following example shows how to configure filter criteria in signalling diagnostics index 1 for call failures based on the service category:

```
Switch(config)# atm signalling diagnostics 1
Switch(cfg-atmsig-diag)# service-category cbr
Switch(cfg-atmsig-diag)# service-category ubr
Switch(cfg-atmsig-diag)# service-category abr ubr
```

The following example shows how to delete an index entry in the filter table:

```
Switch(config)# atm signalling diagnostics 1
Switch(cfg-atmsig-diag)# status delete
```

Displaying the Signalling Diagnostics Table Configuration

To display the signalling diagnostics information, use the following EXEC commands:

Command	Purpose
show atm signalling diagnostics record *filter-index*	Displays the ATM signalling diagnostics for a record.
show atm signalling diagnostics filter [*filter-index*]	Displays the ATM signalling diagnostics for a filter.
show atm signalling diagnostics status	Displays the ATM signalling diagnostic status.

Examples

The following example shows the signalling diagnostic records for index 1:

```
Switch# show atm signalling diagnostics record 1
D I S P L A Y   I N D E X  1
--------------------------------
Scope: internal,  Cast Type: p2p, Conn Indicator: Setup Failure
Connection Kind:   switched-vc
Service Category:  UBR (Unspecified Bit Rate)
Clear Cause: 0x29,  Diagnostics: NULL
Incoming Port: ATM1/0/3,  Outgoing Port:ATM0/1/3
Calling-Address: 47.00918100000000006011000000.470803040506.00
Calling-SubAddr: NULL
Called-Address : 47.009181000000006083C42C01.750203040506.00
Called-SubAddr : NULL
Crankback Type : No Crankback
DTL's :
NodeId:56:160:47.00918100000000006011000000.006083AB9001.00 Port: 0/1/3:2
NodeId:56:160:47.00918100000000603E7B4101.00603E7B4101.00 Port: 0/0/0:2
NodeId:56:160:47.009181000000006083C42C01.006083C42C01.00 Port: 0
```

The following example shows the signalling diagnostics data for filter index 1:

```
Switch# show atm signalling diagnostics filter 1
F I L T E R   I N D E X   1
------------------------------
Scope: internal, Cast Type: p2mp
Connection Kind: soft-vc
Service Category:  CBR (Constant Bit Rate)  UBR (Unspecified Bit Rate)
Clear Cause: 0, Initial TimerValue: 600
Max Records: 20,   NumMatches: 0,   Timer expiry: 600
Incoming Port: ATM0/0/1, Outgoing Port: ATM0/1/1
Calling Nsap Address:47.111122223333444455556666.777788889999.00
Calling Address Mask:FF.FFFFFF000000000000000000.000000000000.00
Called Nsap Address :47.111122223333444455556666.777788889999.01
Called Address Mask :FF.FFFFFF000000000000000000.000000000000.00
Status : active
```

The following example shows the signalling diagnostics status:

```
Switch# show atm signalling diagnostics status
   Signalling diagnostics disabled globally
```

Configuring Closed User Group Signalling

You can configure closed user groups (CUGs) on the ATM switch router to form restricted access groups that function as ATM virtual private networks (VPNs). Access restrictions for users are configured through CUG interlock codes. For a description of how CUGs work using signalling, and an example of CUGs, refer to the *Guide to ATM Technology*.

Configuring a CUG is described in the following sections:

- Configuring Aliases for CUG Interlock Codes, page 16-16
- Configuring CUG on an Interface, page 16-16
- Displaying the CUG, page 16-18

Configuring Aliases for CUG Interlock Codes

You can define an alias for each CUG interlock code used on the ATM switch router. Using an alias can simplify configuration of a CUG on multiple interfaces. When you use an alias, you no longer need to specify the 48-hexadecimal-digit CUG interlock code on each interface attached to a CUG member.

To configure an alias for a CUG interlock code, use the following command in global configuration mode:

Command	Purpose
atm signalling cug alias *alias-name* **interlock-code** *interlock-code*	Configures the alias for the CUG interlock code.

Example

The following example shows how to configure the alias TEST for the CUG interlock code 47009181000000000603E5A790100603E5A790100.12345678:

```
Switch(config)# atm signalling cug alias TEST interlock-code
47009181000000000603E5A790100603E5A790100.12345678
```

Configuring CUG on an Interface

Your first step in CUG configuration is to identify the *access interfaces.* Transmission and reception of CUG interlock codes is not allowed over access interfaces. Configuring all interfaces leading outside of the network as access interfaces ensures that all CUG interlock codes are generated and used only within this network.

You implement CUG procedures only if you configure the interface as an access interface.

Each access interface can be configured to permit or deny calls either *from* users attached to this interface or *to* unknown users who are not members of this interface's CUGs. In International Telecommunications Union Telecommunications Standardization Sector (ITU-T) terminology, this is called *outgoing access*. Similarly, each access interface can be configured to permit or deny calls either *to* the users attached to this interface or *from* unknown users who are not members of this interface's CUGs. In ITU-T terminology, this is called *incoming access*.

> **Note** Interfaces to other networks should be configured as CUG access interfaces, even if no CUGs are configured on the interface. In this case, if you want the ATM switch router to exchange SVCs with the neighbor network, calls *to* and *from* unknown users should be permitted on the interface.

You can configure each access interface to have one or more CUGs associated with it, but only one CUG can be selected as the *preferential* CUG. In this software release, calls received *from* users attached to this interface can only be associated with the preferential CUG. Calls destined *to* users attached to this interface can be accepted based on membership in any of the CUGs configured for the interface.

> **Note** You can configure CUG service without any preferential CUG. If a preferential CUG is not configured on the interface, and calls *from* users attached to this interface *to* unknown users are permitted, the calls will proceed as non-CUG calls, without generating any CUG IEs.

For each CUG configured on the interface, you can specify that calls *to* or *from* other members of the same CUG be denied. In ITU-T terminology, this is called *outgoing-calls-barred* (OCB) and *incoming-calls-barred* (ICB), respectively.

Table 16-1 describes the relationship between the ITU-T CUG terminology and Cisco CUG terminology.

Table 16-1 Cisco CUG and ITU-T CUG Terminology Conversion

ITU-T CUG Terminology	Cisco CUG Terminology
preferential CUG	preferential
incoming access allowed	permit-unknown-cugs to-user
outgoing access allowed	permit-unknown-cugs from-user
incoming calls barred (ICB)	deny-same-cug to-user
outgoing calls barred (OCB)	deny-same-cug from-user

To configure an access interface and the CUGs in which the interface is a member, perform the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **interface atm** *card*/*subcard*/*port* Switch(config-if)#	Specifies an ATM interface and enter interface configuration mode.
Step 2	Switch(config-if)# **atm signalling cug access** [**permit-unknown-cugs** {**to-user** | **from-user permanent** | **both-directions permanent**}]	Configures the interface as a CUG access interface.
Step 3	Switch(config-if)# **atm signalling cug assign** {**alias** *alias-name* | **interlock-code** *interlock-code*} [**deny-same-cug** {**to-user** | **from-user**}] [**preferential**]	Configures the CUG where this interface is a member.

Example

The following example shows how to configure an interface as a CUG access interface and assign a preferential CUG:

```
Switch(config)# interface atm 3/0/0
Switch(config-if)# atm signalling cug access permit-unknown-cugs both-direction permanent
Switch(config-if)# atm signalling cug assign interlock-code
4700918100000000603E5A790100603E5A790100.12345678 preferential
```

Displaying the CUG

To display the global CUG configuration, use the following privileged EXEC commands:

Command	Purpose
show atm signalling cug [**interface atm** *card*/*subcard*/*port*] [**access** | **alias** *alias-name* | **interlock-code** *interlock-code*]	Displays the CUG interface configuration status.
more system:running-config	Displays the CUG global configuration status.

Examples

The following example displays the global CUG configuration using the **show atm signalling cug** EXEC command:

```
Switch# show atm signalling cug
Interface:          ATM3/0/0
Cug Alias Name:
Cug Interlock Code: 4700918100000000603E5A790100603E5A790100.12345678
Non preferential Cug
Permit Network to User Calls
Permit User to Network Calls
```

The following example displays the global CUG access configuration using the **show atm signalling cug access** command:

```
Switch# show atm signalling cug access
Closed User Group Access Interface Parameters:


Interface:          ATM3/0/0
Network To User (incoming) access: Permit calls from unknown CUGs to User
User To Network (outgoing) access: Permit permanent calls to unknown groups
```

The following example displays the CUG global configuration using the **more system:running-config** command:

```
Switch# more system:running-config
Building configuration...
Current configuration:
!
version XX.X
no service pad
service udp-small-servers
service tcp-small-servers
!
```

```
hostname ls1010-2
!
atm signalling cug alias TEST interlock-code
47.0091810000000061705BDA01.0061705BDA01.00.12345678
!
atm address 47.0091.8100.0000.0061.705b.da01.0061.705b.da01.00

<information deleted>
!
interface ATM0/0/0
 atm signalling cug access permit-unknown-cugs both-direction permanent
<information deleted>
```

Displaying the Signalling Statistics

To display the ATM signalling statistics, use the following EXEC command:

Command	Purpose
show atm signalling statistics	Displays the ATM signalling statistics.

Example

The following example displays the ATM signalling statistics:

```
Switch# show atm signalling statistics
Global Statistics:
Calls Throttled: 0
Max Crankback: 3
Max Connections Pending: 255
Max Connections Pending Hi Water Mark: 1
ATM0:0   UP Time 01:06:20  # of int resets: 0
----------------------------------------------------------------
Terminating connections: 0       Soft VCs: 0
Active Transit PTP SVC: 0        Active Transit MTP SVC: 0
Port requests:  0                Source route requests: 0
Conn-Pending: 0                  Conn-Pending High Water Mark: 1
Calls Throttled: 0               Max-Conn-Pending:  40
          Messages:   Incoming  Outgoing
          ---------   --------  --------
PTP Setup Messages:        0         0
MTP Setup Messages:        0         0
  Release Messages:        0         0
  Restart Messages:        0         0
           Message:   Received Transmitted Tx-Reject Rx-Reject
Add Party Messages:          0           0         0         0
     Failure Cause:   Routing     CAC   Access-list    Addr-Reg  Misc-Failure
    Location Local:         0       0             0           0         12334
   Location Remote:         0       0             0           0             0
ATM 0/0/3:0   UP Time 3d21h  # of int resets: 0
----------------------------------------------------------------
Terminating connections: 0       Soft VCs: 0
Active Transit PTP SVC: 0        Active Transit MTP SVC: 0
Port requests:  0                Source route requests: 0
Conn-Pending: 0                  Conn-Pending High Water Mark: 0
Calls Throttled: 0               Max-Conn-Pending:  40

<information deleted>
```

Disabling Signalling on an Interface

If you disable signalling on a Private Network-Network Interface (PNNI) interface, PNNI routing is also disabled and Integrated Local Management Interface (ILMI) is automatically restarted whenever signalling is enabled or disabled.

To disable signalling on an interface, perform the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **interface atm** *card*/*subcard*/*port* Switch(config-if)#	Selects the interface to be configured.
Step 2	Switch(config-if)# **no atm signalling enable**	Disables signalling on the interface.

Example

The following example shows how to shut down signalling on ATM interface 0/1/2:

```
Switch(config)# interface atm 0/1/2
Switch(config-if)# no atm signalling enable
Switch(config-if)#
%ATM-5-ATMSOFTSTART: Restarting ATM signalling and ILMI on ATM0/1/2.
```

Multipoint-to-Point Funnel Signalling

Multipoint-to-point funnel signalling (funneling) merges multiple incoming switched virtual channels (SVCs) into a single outgoing SVC. This feature supports the Microsoft Corporation Proprietary Funnel Join (or Flow Merge) Protocol.

No configuration is necessary to enable this feature. For a complete description, refer to the *Guide to ATM Technology*.

Displaying Multipoint-to-Point Funnel Connections

To display multipoint-to-point funnel connections, use the following EXEC commands:

Command	Purpose
show atm status	Displays the number of active funnels.
show atm vc cast mp2p	Displays the status of the multipoint-to-point messages on the specific interfaces.

Examples

Use the **show atm status** command to display the number of active funnels, point-to-point and point-to-multipoint setup messages. An example of the **show atm status** command output follows:

```
Switch# show atm status
NUMBER OF INSTALLED CONNECTIONS: (P2P=Point to Point, P2MP=Point to MultiPoint,
MP2P=Multipoint to Point)
Type     PVCs SoftPVCs     SVCs     TVCs     PVPs SoftPVPs     SVPs      Total
P2P        26        0        0        0        2        0        0         28
P2MP        1        0        0        0        0        0        0          1
MP2P        0        0        1        0        0        0        0          1
                                      TOTAL INSTALLED CONNECTIONS =         30
PER-INTERFACE STATUS SUMMARY AT 13:34:48 UTC Thu Jan 29 1998:
   Interface      IF         Admin  Auto-Cfg    ILMI Addr     SSCOP    Hello
     Name       Status      Status    Status    Reg State     State    State
------------- -------- ------------ -------- ------------ --------- --------
ATM0/0/0            UP           up     done  UpAndNormal    Active 2way_in
ATM0/0/1          DOWN         down  waiting          n/a      Idle      n/a
ATM0/0/2            UP           up     done  UpAndNormal    Active 2way_in
ATM0/0/3            UP           up     done  UpAndNormal    Active 2way_in
ATM0/0/3.55         UP           up  waiting  WaitDevType      Idle      n/a
ATM0/0/3.60         UP           up  waiting  WaitDevType      Idle      n/a
ATM0/0/3.65         UP           up  waiting  WaitDevType      Idle      n/a
ATM0/1/0            UP           up      n/a  UpAndNormal    Active      n/a
ATM0/1/1            UP           up     done  UpAndNormal    Active      n/a
ATM0/1/2          DOWN     shutdown  waiting          n/a      Idle      n/a
ATM0/1/3          DOWN         down  waiting          n/a      Idle      n/a
```

Use the **show atm vc cast mp2p** command to display the status of the multipoint-to-point messages on the specific interfaces. An example of the **show atm vc cast mp2p** command output follows:

```
Switch# show atm vc cast mp2p
Interface    VPI   VCI   Type    X-Interface  X-VPI X-VCI  Encap Status
ATM0/1/0     0     40     SVC     ATM0/1/1     0     35          UP
                                  ATM0/1/1     0     36          UP
ATM0/1/1     0     35     SVC     ATM0/1/0     0     40          UP
ATM0/1/1     0     36     SVC     ATM0/1/0     0     40          UP
```

CHAPTER 17

Configuring Interfaces

This chapter describes the steps required to configure the physical interfaces on the ATM switch router. Your switch is configured as specified in your order and is ready for installation and startup when it leaves the factory.

Note This chapter provides advanced configuration instructions for the Catalyst 8540 MSR, Catalyst 8510 MSR, and LightStream 1010 ATM switch routers. For complete descriptions of the commands mentioned in this chapter, refer to the *ATM Switch Router Command Reference* publication. For hardware installation and cabling instructions, refer to the *ATM and Layer 3 Port Adapter and Interface Module Installation Guide*.

Each port on the interface module or interface module physical interface can be configured to support the following clocking options:

- Self-timing based on a stratum 4 level clock
- Loop timing from the received data stream—ideal for public network connections
- Timing synchronized to a selected master clock port; required to distribute a single clock across a network

The plug-and-play mechanisms of the ATM switch router allow it to come up automatically. All configuration information for interface modules can be saved between hot swaps and switch router reboots. The switch router automatically discovers interface types and eliminates mandatory manual configuration.

When you upgrade your system, add components, or customize the initial configuration, see the following sections:

- Configuring 25-Mbps Interfaces (Catalyst 8510 MSR and LightStream 1010), page 17-2
- Configuring 155-Mbps SM, MM, and UTP Interfaces, page 17-3
- Configuring OC-3c MMF Interfaces (Catalyst 8540 MSR), page 17-5
- Configuring 622-Mbps SM and MM Interfaces, page 17-7
- Configuring OC-12c SM and MM Interfaces (Catalyst 8540 MSR), page 17-9
- Configuring OC-48c SM and MM Interfaces (Catalyst 8540 MSR), page 17-11
- Configuring DS3 and E3 Interfaces, page 17-13
- Configuring T1/E1 Trunk Interfaces, page 17-15
- Troubleshooting the Interface Configuration, page 17-17

Note For hardware installation and cabling instructions, refer to the *ATM Port Adapter and Interface Module Installation Guide*. For complete descriptions of the commands mentioned in this chapter, refer to the *ATM Switch Router Command Reference* publication.

To configure the circuit emulation service (CES) T1 and E1 port adapters, see Chapter 18, "Configuring Circuit Emulation Services." To configure the Frame Relay E1 port adapters, see Chapter 19, "Configuring Frame Relay to ATM Interworking Port Adapter Interfaces." To configure the T1 and E1 inverse multiplexing over ATM (IMA) port adapters, see Chapter 20, "Configuring IMA Port Adapter Interfaces." To configure the ATM router modules, see Chapter 21, "Configuring ATM Router Module Interfaces."

Configuring 25-Mbps Interfaces (Catalyst 8510 MSR and LightStream 1010)

The ATM switch supports two types of 25-Mbps port adapters: a 4-port version and a 12-port version. The number of ports is determined by the type of cable used with the 25-Mbps port adapters. The cables have a 96-pin Molex connector with a multileg RJ-45 cable assembly. That is, multiple RJ-45 cables branch off from one large 96-pin Molex connector. You can choose either a 4-port version (with four RJ-45 cables) or a 12-port version (with 12 RJ-45 cables). Each 25.6-Mbps ATM port can be used for workgroup links. Each port complies with the ATM Forum PHY standard for 25.6 Mbps over twisted-pair cable.

The plug-and-play mechanisms of the ATM switch allow the switches to come up automatically. All configuration information for the port adapters can be saved between hot swaps and switch reboots, while interface types are automatically discovered by the switch, thereby eliminating mandatory manual configuration.

The ATM switch supports any combination of port adapters. You can configure your switch with up to 32 25-Mbps interface ports with the 4-port 25-Mbps port adapter, or up to 96 25-Mbps interface ports with the 12-port 25-Mbps port adapter.

Default 25-Mbps ATM Interface Configuration without Autoconfiguration (Catalyst 8510 MSR and LightStream 1010)

If ILMI is disabled or if the connecting end node does not support ILMI, the following defaults are assigned to all 25-Mbps interfaces:

- ATM interface type = UNI
- UNI version = 3.0
- Maximum VPI bits = 2
- Maximum VCI bits = 14
- ATM interface side = network
- ATM UNI type = private

For the 12-port 25-Mbps port adapter, the following parameters can be configured on physical ports 0 or 6. Parameters configured on port 0 apply to ports 0 to 5, and parameters configured on port 6 apply to ports 6 to 11. For the 4-port 25-Mbps port adapter, parameters configured on port 0 apply to ports 0 to 4:

- Output-queue
- Output-threshold
- CAC link sharing

Note Pacing might not be configured on any physical port of the 25-Mbps port adapter.

Manual 25-Mbps Interface Configuration (Catalyst 8510 MSR and LightStream 1010)

To manually change any of the default configuration values, perform the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **interface atm** *card*/*subcard*/*port* Switch(config-if)#	Specifies an ATM interface and enters interface configuration mode.
Step 2	Switch(config-if)# **atm uni** [**side network**] [**type private**] [**version** {**3.0** | **3.1** | **4.0**}]	Modifies the ATM interface side, type, or version.
Step 3	Switch(config-if)# **atm maxvpi-bits** *max-vpi-bits*	Modifies the maximum VPI bits configuration.
Step 4	Switch(config-if)# **atm maxvci-bits** *max-vci-bits*	Modifies the maximum VCI bits configuration.

Example

The following example shows how to change the default ATM interface type to private, using the **atm uni type private** command:

```
Switch(config)# interface atm 0/0/0
Switch(config-if)# atm uni type private
```

See the "Troubleshooting the Interface Configuration" section on page 17-17 to confirm your interface configuration.

Configuring 155-Mbps SM, MM, and UTP Interfaces

The 155-Mbps Synchronous Optical Network (SONET) Synchronous Transport Signal level 3/Synchronous Digital Hierarchy (STS3c/SDH) Synchronous Transport Module level 1 (STM1) port adapter, used for intercampus or wide-area links, has four ports.

155-Mbps Interface Configuration

You can configure any number and type of interfaces required, up to 64 155-Mbps interface ports on the Catalyst 8540 MSR and up to 32 155-Mbps interface ports on the Catalyst 8510 MSR and LightStream 1010 ATM switch routers.

Note The 155-Mbps port adapter supports mixed mode. Port 0 is a single-mode interface and ports 1 through 3 are multimode interfaces.

The port adapter supports SC-type and unshielded twisted-pair (UTP) connectors, while receive and transmit LEDs on each port give quick, visual indications of port status and operation.

Traffic pacing allows the aggregate output traffic rate on any port to be set to a rate below the line rate. This feature is useful when communicating with a slow receiver or when connected to public networks with peak-rate tariffs.

Default 155-Mbps ATM Interface Configuration without Autoconfiguration

If Integrated Local Management Interface (ILMI) has been disabled or if the connecting end node does not support ILMI, the following defaults are assigned to all 155-Mbps interfaces:

- ATM interface type = UNI
- UNI version = 3.0
- Maximum virtual path identifier (VPI) bits = 8
- Maximum virtual channel identifier (VCI) bits = 14
- ATM interface side = network
- ATM UNI type = private
- Framing = sts-3c
- Clock source = network-derived
- Synchronous Transfer Signal (STS) stream scrambling = on
- Cell payload scrambling = on

Manual 155-Mbps Interface Configuration

To manually change any of the default configuration values, perform the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **interface atm** *card*/*subcard*/*port* Switch(config-if)#	Specifies an ATM interface and enters interface configuration mode.
Step 2	Switch(config-if)# **atm uni** [**side** {**network** | **user**}] [**type** {**private** | **public**}] [**version** {**3.0** | **3.1** | **4.0**}]	Modifies the ATM interface side, type, or version.
Step 3	Switch(config-if)# **atm maxvpi-bits** *max-vpi-bits*	Modifies the maximum VPI bits configuration.

	Command	Purpose
Step 4	Switch(config-if)# **atm maxvci-bits** *max-vci-bits*	Modifies the maximum VCI bits configuration.
Step 5	Switch(config-if)# **sonet** {**stm-1** | **sts-3c**}	Modifies the framing mode.
Step 6	Switch(config-if)# **clock source** {**free-running** | **loop-timed** | **network-derived**}	Modifies the clock source.
Step 7	Switch(config-if)# **scrambling** {**cell-payload** | **sts-stream**}	Modifies the scrambling mode.

Example

The following example configures ATM interface 3/1/1 as the network side of a private UNI running version 3.1.

```
Switch# interface atm 3/1/1
Switch(config-if)# no atm auto-configuration
Switch(config-if)#
%ATM-6ILMIOAUTOCFG: ILMI(ATM/0/0): Auto-configuration is disabled, current interface
parameters will be used at next interface restart.
Switch(config-if)# atm uni version 3.1
```

See the "Troubleshooting the Interface Configuration" section on page 17-17 to confirm your interface configuration.

Configuring OC-3c MMF Interfaces (Catalyst 8540 MSR)

The 16-port OC-3c MMF interface module provides short-reach intercampus and WAN ATM connections. The OC-3c interface module provides an interface to ATM switching fabrics for transmitting and receiving data bidirectionally at up to 155 Mbps. The OC-3c interface module can support interfaces that connect to the OC-3c MMF STS-3c/STM1 physical layer.

The Catalyst 8540 MSR supports up to eight OC-3c interface modules per chassis, with a maximum of 128 OC-3c interface ports.

Note You can configure traffic pacing on the interfaces to allow the aggregate output traffic rate on any interface to be set to a rate below the line rate. This feature is useful when communicating with a slow receiver or when connected to public networks with peak-rate tariffs.

Default OC-3c MMF Interface Configuration without Autoconfiguration (Catalyst 8540 MSR)

If Integrated Local Management Interface (ILMI) has been disabled or if the connecting end node does not support ILMI, the following defaults are assigned to all OC-3c interfaces:

- ATM interface type = UNI
- UNI version = 3.0
- Maximum virtual path identifier (VPI) bits = 8
- Maximum virtual channel identifier (VCI) bits = 14

- ATM interface side = network
- ATM UNI type = private
- Framing = sts-3c
- Clock source = network-derived
- Synchronous Transfer Signal (STS) stream scrambling = on
- Cell payload scrambling = on

Manual OC-3c MMF Interface Configuration (Catalyst 8540 MSR)

To manually change any of the default configuration values, perform the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **interface atm** *card*/*subcard*/*port* Switch(config-if)#	Specifies an ATM interface and enters interface configuration mode.
Step 2	Switch(config-if)# **atm uni** [**side** {**private** | **public**}] [**type** {**network** | **user**}] [**version** {**3.0** | **3.1** | **4.0**}]	Modifies the ATM interface side, type, or version.
Step 3	Switch(config-if)# **atm maxvpi-bits** *max-vpi-bits*	Modifies the maximum VPI bits configuration.
Step 4	Switch(config-if)# **atm maxvci-bits** *max-vci-bits*	Modifies the maximum VCI bits configuration.
Step 5	Switch(config-if)# **sonet** {**stm-1** | **sts-3c**}	Modifies the framing mode.
Step 6	Switch(config-if)# **clock source** {**free-running** | **loop-timed** | **network-derived**}	Modifies the clock source.
Step 7	Switch(config-if)# **scrambling** {**cell-payload** | **sts-stream**}	Modifies the scrambling mode.

Example

The following example configures ATM interface 3/0/1 as the network side of a private UNI running version 3.1.

```
Switch# interface atm 3/0/1
Switch(config-if)# no atm auto-configuration
Switch(config-if)#
%ATM-6-ILMINOAUTOCFG: ILMI(ATM3/0/1): Auto-configuration is disabled, current interface
parameters will be used at next interface restart.
Switch(config-if)# atm uni version 3.1
```

See the "Troubleshooting the Interface Configuration" section on page 17-17 to confirm your interface configuration.

Configuring 622-Mbps SM and MM Interfaces

These interfaces are used for intercampus or wide-area links.

The 622-Mbps SONET STS12/SDH STM4 port adapter has a single port. You can configure your switch with only the number and type of interfaces required, with up to eight 622-Mbps interface ports.

Note The configuration instructions in this section also apply to the ATM Fabric Integration Module.

The port adapter supports an SC-type connector, and receive and transmit LEDs give quick, visual indications of port status and operation.

Default 622-Mbps ATM Interface Configuration without Autoconfiguration

If ILMI has been disabled or if the connecting end node does not support ILMI, the following defaults are assigned to all 622-Mbps interfaces:

- ATM interface type = UNI
- UNI version = 3.0
- Maximum VPI bits = 8
- Maximum VCI bits = 14
- ATM interface side = network
- ATM UNI type = private
- Framing = sts-12c
- Clock source = network-derived
- STS stream scrambling = on
- Cell payload scrambling = on
- Reporting alarms = SF SLOS SLOF B1-TCA B2-TCA PLOP B3-TCA
- Path trace message = free format 64-byte string containing path information
- Scrambling = On
- BER thresholds: SF = 10e-3 SD = 10e-6
- TCA thresholds: B1 = 10e-6 B2 = 10e-6 B3 = 10e-6

Manual 622-Mbps Interface Configuration

To manually change any of the default configuration values, perform the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **interface atm** *card*/*subcard*/*port*[1] Switch(config-if)#	Specifies the ATM interface and enters interface configuration mode.
Step 2	Switch(config-if)# **atm uni** [**side** {**network** | **user**}] [**type** {**private** | **public**}] [**version** {**3.0** | **3.1** | **4.0**}]	Modifies the ATM interface side, type, or version.
Step 3	Switch(config-if)# **atm maxvpi-bits** *max-vpi-bits*	Modifies the maximum VPI bits configuration.
Step 4	Switch(config-if)# **atm maxvci-bits** *max-vci-bits*	Modifies the maximum VCI bits configuration.
Step 5	Switch(config-if)# **sonet** {**stm-4c** | **sts-12c**} or Switch(config-if)# **framing** {**stm-4c** | **sts-12c**}	Modifies the framing mode.
Step 6	Switch(config-if)# **clock source** {**free-running** | **loop-timed** | **network-derived**}	Modifies the clock source.
Step 7	Switch(config-if)# **sonet overhead** {**c2** *bytes* | **j0** {*bytes* | **msg** *line*} | **j1** {**16byte** {**exp-msg** *line* | **msg** *line*} | **64byte** {**exp-msg** *line* | **msg** *line*}} | **s1s0** *bits*}	Modifies the path trace message.
Step 8	Switch(config-if)# **sonet threshold** {**sd-ber** | **sf-ber** | **b1-tca** | **b2-tca** | **b3-tca**} *ber*	Modifies the bit error rate threshold value from 3 (10e-3) to 9 (10e-9).
Step 9	Switch(config-if)# **sonet report** {**slos** | **slof** | **lais** | **lrdi** | **pais** | **prdi** | **plop** | **sd-be**r | **sf-ber** | **b1-tca** | **b2-tca** | **b3-tca**}	Enables reporting of selected alarms.

1. The subcard for the full-width 622-Mbps interface module is always zero.

Examples

The following example shows how to change the default ATM interface type to **private** using the **atm uni type private** command:

```
Switch# configure terminal
Switch(config)# interface atm 0/0/0
Switch(config-if)# atm uni type private
```

The following example shows how to change the clock source using the **clock source network-derived** command:

```
Switch# configure terminal
Switch(config)# interface atm 0/0/0
Switch(config-if)# clock source network-derived
```

See the "Troubleshooting the Interface Configuration" section on page 17-17 to confirm your interface configuration.

Configuring OC-12c SM and MM Interfaces (Catalyst 8540 MSR)

The 4-port OC-12c SM and MM interface modules provide either single-mode or multimode intermediate reach. The OC-12c interface module provides an interface to ATM switching fabrics for transmitting and receiving data bidirectionally at up to 622 Mbps. The OC-12c interface module can support interfaces that connect to the OC-12c SONET STS12/SDH STM4 physical layer.

These interfaces are used for intercampus or wide-area links.

Note The configuration instructions in this section also apply to the ATM Fabric Integration Module.

OC-12c Interface Configuration (Catalyst 8540 MSR)

The full-width four-port 622-Mbps is available in either a single-mode intermediate reach interface module or a new multimode module.You can configure your Catalyst 8540 MSR with only the number and type of interfaces required, up to 32 622-Mbps interface ports using the full-width interface module.

The interface module supports an SC-type connector, and receive and transmit LEDs give quick, visual indications of port status and operation.

Default OC-12c ATM Interface Configuration without Autoconfiguration (Catalyst 8540 MSR)

If ILMI has been disabled or if the connecting end node does not support ILMI, the following defaults are assigned to all OC-12c interfaces:

- ATM interface type = UNI
- UNI version = 3.0
- Maximum VPI bits = 8
- Maximum VCI bits = 14
- ATM interface side = network
- ATM UNI type = private
- Framing = sts-12c
- Clock source = network-derived
- STS stream scrambling = on
- Cell payload scrambling = on
- Reporting alarms = SF SLOS SLOF B1-TCA B2-TCA PLOP B3-TCA
- Path trace message = free format 64-byte string containing path information
- Scrambling = On
- BER thresholds: SF = 10e-3 SD = 10e-6
- TCA thresholds: B1 = 10e-6 B2 = 10e-6 B3 = 10e-6

Manual OC-12c Interface Configuration (Catalyst 8540 MSR)

To manually change any of the default configuration values, perform the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **interface atm** *card/subcard/port*[1] Switch(config-if)#	Specifies the ATM interface and enters interface configuration mode.
Step 2	Switch(config-if)# **atm uni** [**side** {**network** | **user**}] [**type** {**private** | **public**}] [**version** {**3.0** | **3.1** | **4.0**}]	Modifies the ATM interface side, type, or version.
Step 3	Switch(config-if)# **atm maxvpi-bits** *max-vpi-bits*	Modifies the maximum VPI bits configuration.
Step 4	Switch(config-if)# **atm maxvci-bits** *max-vci-bits*	Modifies the maximum VCI bits configuration.
Step 5	Switch(config-if)# **sonet** {**stm-4c** | **sts-12c**} or Switch(config-if)# **framing** {**stm-4c** | **sts-12c**}	Modifies the framing mode.
Step 6	Switch(config-if)# **clock source** {**free-running** | **loop-timed** | **network-derived**}	Modifies the clock source.
Step 7	Switch(config-if)# **sonet overhead** {**c2** *bytes* | **j0** {*bytes* | **msg** *line*} | **j1** {**16byte** {**exp-msg** *line* | **msg** *line*} | **64byte** {**exp-msg** *line* | **msg** *line*}} | **s1s0** *bits*}	Modifies the path trace message.
Step 8	Switch(config-if)# **sonet threshold** {**sd-ber** | **sf-ber** | **b1-tca** | **b2-tca** | **b3-tca**} *ber*	Modifies the bit error rate threshold value from 3 (10e-3) to 9 (10e-9).
Step 9	Switch(config-if)# **sonet report** {**slos** | **slof** | **lais** | **lrdi** | **pais** | **prdi** | **plop** | **sd-be**r | **sf-ber** | **b1-tca** | **b2-tca** | **b3-tca**}	Enables reporting of selected alarms.

1. The subcard for the full-width 622-Mbps interface module is always zero.

Examples

The following example shows how to change the default ATM interface type to **private** using the **atm uni type private** command:

```
Switch# configure terminal
Switch(config)# interface atm 0/0/0
Switch(config-if)# atm uni type private
```

The following example shows how to change the clock source using the **clock source network-derived** command:

```
Switch# configure terminal
Switch(config)# interface atm 0/0/0
Switch(config-if)# clock source network-derived
```

See the "Troubleshooting the Interface Configuration" section on page 17-17 to confirm your interface configuration.

Configuring OC-48c SM and MM Interfaces (Catalyst 8540 MSR)

The Catalyst 8540 MSR supports the following three OC-48c SM and MM intermediate reach fiber interface modules:

- 1-port OC-48c single-mode intermediate reach *plus* 4-port OC-12 single-mode fiber
- 1-port OC-48c single-mode intermediate reach *plus* 4-port OC-12 multimode fiber
- 2-port OC-48c single-mode intermediate reach
- 1-port OC-48c single-mode long reach *plus* 4-port OC-12 single-mode fiber
- 2-port OC-48c single-mode long reach

Each OC-48c interface module occupies a slot pair. For example, install an OC-48c interface module in slots 0 and 1, 2 and 3, 9 and 10, or 11 and 12. The chassis supports a maximum of four OC-48c interface modules. A maximum configuration provides up to four OC-48c ports and 16 OC-12 ports or up to eight OC-48c ports. The OC-48c interface module supports a dual SC-type connector. Refer to your hardware installation guide for more information.

The OC-48c interface module is used for intercampus or wide-area links. This interface module is functionally similar to the current OC-3c and OC-12c interfaces, but operates at a faster speed. OC-48c supports both UNI and NNI as well as all framing options.

Default OC-48c ATM Interface Configuration Without Autoconfiguration (Catalyst 8540 MSR)

If ILMI is disabled or if the connecting end node does not support ILMI, the following defaults are assigned to all OC-48c interfaces:

- ATM interface type = UNI
- UNI version = 3.0
- Maximum VPI bits = 8
- Maximum VCI bits = 14
- ATM interface side = network
- ATM UNI type = private
- Framing = sts-48c
- Loopback = no loopback
- STS stream scrambling = on
- Cell payload scrambling = on
- Clock source = network-derived
- Reporting alarms enabled = SF SLOS SLOF B1-TCA B2-TCA PLOP B3-TCA
- Path trace message = free format 64-byte string containing path information
- Bit error rate (BER) thresholds: SF = 10e-3, SD = 10e-6
- TCA thresholds: B1 = 10e-6, B2 = 10e-6, B3 = 10e-6

Manual OC-48c Interface Configuration (Catalyst 8540 MSR)

To manually change any of the default configuration values, perform the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **interface atm** *card*/*subcard*/*port* Switch(config-if)#	Specifies the ATM interface and enters interface configuration mode.
Step 2	Switch(config-if)# **atm uni** [**side** {**network** | **user**}] [**type** {**private** | **public**}] [**version** {**3.0** | **3.1** | **4.0**}]	Modifies the ATM interface side, type, or version.
Step 3	Switch(config-if)# **atm maxvpi-bits** *max-vpi-bits*	Modifies the maximum VPI bits configuration.
Step 4	Switch(config-if)# **atm maxvci-bits** *max-vci-bit***s**	Modifies the maximum VCI bits configuration.
Step 5	Switch(config-if)# **sonet** {**stm-16** | **sts-48c**}	Modifies the framing mode.
Step 6	Switch(config-if)# **clock source** {**free-running** | **loop-timed network-derived**}	Modifies the clock source.
Step 7	Switch(config-if)# **sonet overhead** {**c2** *bytes* | **j0** {*bytes* | **msg** *line*} | **j1** {**16byte** {**exp-msg** *line* | **msg** *line*} | **64byte** {**exp-msg** *line* | **msg** *line*}} | **s1s0** *bits*}	Modifies the path trace message.
Step 8	Switch(config-if)# **sonet threshold** {**sd-ber** | **sf-ber** | **b1-tca** | **b2-tca** | **b3-tca**} *ber*	Modifies the BER threshold values.
Step 9	Switch(config-if)# **sonet report** {**slos** | **slof** | **lais** | **lrdi** | **pais** | **prdi** | **plop** | **sd-ber** | **sf-ber** | **b1-tca** | **b2-tca** | **b3-tca**}	Enables reporting of selected alarms.

Example

The following example shows how to change the number of active VCI bits to 12:

```
Switch(config)# interface atm 9/0/0
Switch(config-if)# atm max-vci-bits 12
```

See the "Troubleshooting the Interface Configuration" section on page 17-17 to confirm your interface configuration.

Configuring DS3 and E3 Interfaces

The 45-Mbps DS3 and the 34-Mbps E3 port adapters are used for wide-area connections, to link multiple campuses, or to connect to public networks.

DS3 and E3 Interface Configuration

You can configure your switch router with only the number and type of interfaces required, with up to 64 DS3 or E3 interface ports on the Catalyst 8540 MSR and up to 32 DS3 or E3 interface ports on the Catalyst 8510 MSR and LightStream 1010 ATM switch router.

Traffic-pacing allows the aggregate output traffic rate on any port to be set to a rate below the line rate. This feature is useful when communicating with a slow receiver or when connected to public networks with peak-rate tariffs.

Note Network clocking configuration options are applicable only to DS3 quad interfaces.

Default DS3 and E3 ATM Interface Configuration without Autoconfiguration

If ILMI has been disabled or if the connecting end node does not support ILMI, the following defaults are assigned to all DS3 or E3 interfaces:

- ATM interface type = UNI
- UNI version = 3.0
- Maximum VPI bits = 8
- Maximum VCI bits = 14
- ATM interface side = network
- ATM UNI type = private

The following defaults are assigned to all DS3 port adapter interfaces:

- Framing = cbit-adm
- Cell payload scrambling = off
- Clock source = network-derived
- LBO = short
- Auto-ferf on loss of signal (LOS)= on
- Auto-ferf on out of frame (OOF)= on
- Auto-ferf on red = on
- Auto-ferf on loss of cell delineation (LCD)= on
- Auto-ferf on alarm indication signal (AIS)= on

The following defaults are assigned to all E3 port adapter interfaces:

- Framing = g.832 adm
- Cell payload scrambling = on
- Clock source = network-derived

- Auto-ferf on LOS = on
- Auto-ferf on OOF = on
- Auto-ferf on LCD = on (applicable to nonplcp mode only)
- Auto-ferf on AIS = on

Manual DS3 and E3 Interface Configuration

To manually change any of the default configuration values, perform the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **network-clock-select** *priority* **atm** *card*/*subcard*/*port*	Configures the network-derived clock.
Step 2	Switch(config)# **interface atm** *card*/*subcard*/*port* Switch(config-if)#	Specifies an ATM interface and enters interface configuration mode.
Step 3	Switch(config-if)# **atm uni** [**side** {**private** | **public**} **type** {**network** | **user**} **version** {**3.0** | **3.1** | **4.0**}]	Modifies the ATM interface side, type, or version.
Step 4	Switch(config-if)# **atm maxvpi-bits** *max-vpi-bits*	Modifies the maximum VPI bits configuration.
Step 5	Switch(config-if)# **atm maxvci-bits** *max-vci-bits*	Modifies the maximum VCI bits configuration.
Step 6	Switch(config-if)# **framing** {**cbitadm** | **cbitplcp** | **m23adm** | **m23plcp**}	Modifies the framing mode.
Step 7	Switch(config-if)# **scrambling** {**cell-payload** | **sts-stream**}	Modifies the scrambling mode.
Step 8	Switch(config-if)# **clock source** {**free-running** | **loop-timed** | **network-derived**}	Modifies the clock source.
Step 9	Switch(config-if)# **lbo** {**long** | **short**}	Modifies the line build-out.
Step 10	Switch(config-if)# **auto-ferf** {**ais** | **lcd** | **los** | **oof** | **red**}	Modifies the auto-ferf configuration.

Examples

The following example shows how to change the default ATM interface type to **private** using the **atm uni type private** command:

```
Switch# configure terminal
Switch(config)# interface atm 0/0/0
Switch(config-if)# atm uni type private
```

The following example shows how to change the clock source using the **clock source network-derived** command:

```
Switch# configure terminal
Switch(config)# interface atm 0/0/0
Switch(config-if)# clock source network-derived
```

See the “Troubleshooting the Interface Configuration” section on page 17-17 to confirm your interface configuration.

Configuring T1/E1 Trunk Interfaces

The T1 and E1 trunk port adapters, used for intercampus or wide-area links, have four ports.

T1/E1 Trunk Interface Configuration

The ATM switch router supports any combination of port adapters. You can configure your switch router with only the number and type of interfaces required, with up to 64 T1 or E1 interface ports on the Catalyst 8540 MSR and up to 32 T1 or E1 interface ports on the Catalyst 8510 MSR and LightStream 1010 ATM switch routers.

The port adapter supports SC-type and BNC connectors while receive and transmit LEDs on each port give quick, visual indications of port status and operation.

Traffic-pacing allows the aggregate output traffic rate on any port to be set to a rate below the line rates. This feature is useful when communicating with a slow receiver or when connected to public networks with peak-rate tariffs.

Default T1 and E1 ATM Interface Configuration without Autoconfiguration

If ILMI is disabled or if the connecting end node does not support ILMI, the following defaults are assigned to all T1 and E1 interfaces:

- ATM interface type = UNI
- UNI version = 3.0
- Maximum VPI bits = 8
- Maximum VCI bits = 14
- ATM interface side = network
- ATM UNI type = private

The following port adapter types have specific defaults assigned.

T1 port adapter:

- Framing = ESF
- Line coding = B8ZS
- Cell payload scrambling = off
- Clock source = network-derived
- LBO = 0 to 110 feet
- Auto-ferf on loss of signal (LOS) = on
- Auto-ferf on out of frame (OOF) = on
- Auto-ferf on red = on
- Auto-ferf on loss of cell delineation (LCD) = on
- Auto-ferf on alarm indication signal (AIS) = on

E1 port adapter:

- Framing = g.832 adm
- Line coding = HDB3
- Cell payload scrambling = off
- Clock source = network-derived
- Auto-ferf on LOS = on
- Auto-ferf on OOF = on
- Auto-ferf on red = on
- Auto-ferf on LCD = on
- Auto-ferf on AIS = on

Manual T1 and E1 Interface Configuration

To manually change any of the default configuration values, perform the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **network-clock-select** *priority* **atm** *card*/*subcard*/*port*	Configures the network-derived clock.
Step 2	Switch(config)# **interface atm** *card*/*subcard*/*port* Switch(config-if)#	Specifies an ATM interface and enters interface configuration mode.
Step 3	Switch(config-if)# **atm uni** [**side** {**private** | **public**}] [**type** {**network** | **user**}] [**version** {**3.0** | **3.1** | **4.0**}]	Modifies the ATM interface side, type, or version.
Step 4	Switch(config-if)# **atm maxvpi-bits** *max-vpi-bits*	Modifies the maximum VPI bits configuration.
Step 5	Switch(config-if)# **atm maxvci-bits** *max-vci-bits*	Modifies the maximum VCI bits configuration.
Step 6	Switch(config-if)# **framing** {**esfadm** | **esfplcp** | **sfadm** | **sfplcp**} Switch(config-if)# **framing** {**crc4adm** | **crc4plcp** | **pcm30adm** **pcm30plcp**}	Modifies the T1 framing mode. Modifies the E1 framing mode.
Step 7	Switch(config-if)# **linecode** {**ami** | **b8zs**} Switch(config-if)# **linecode** {**ami** | **hdb3**}	Modifies the T1 line coding. Modifies the E1 line coding.
Step 8	Switch(config-if)# **scrambling** {**cell-payload** | **sts-stream**}	Modifies the scrambling mode.
Step 9	Switch(config-if)# **clock source** {**free-running** | **loop-timed** | **network-derived**}	Modifies the clock source.
Step 10	Switch(config-if)# **lbo** {**0_110** | **110_220** | **220_330** | **330_440** | **440_550** | **550_600** | **gt_600**}	Modifies the line build-out.
Step 11	Switch(config-if)# **auto-ferf** {**ais** | **lcd** | **los** | **oof** | **red**}	Modifies the auto-ferf configuration.

Examples

The following example shows how to change the default ATM interface type to **private** using the **atm uni type private** command:

```
Switch# configure terminal
Switch(config)# interface atm 0/0/0
Switch(config-if)# atm uni type private
```

The following example shows how to change the clock source using the **clock source network-derived** command:

```
Switch# configure terminal
Switch(config)# interface atm 0/0/0
Switch(config-if)# clock source network-derived
```

See the "Troubleshooting the Interface Configuration" section on page 17-17 to confirm your interface configuration.

Troubleshooting the Interface Configuration

Table 17-1 describes commands that you can use to confirm that the hardware, software, and interfaces for the ATM switch router are configured as intended:

Table 17-1 Configuration Testing Commands

Command	Purpose
show version	Confirms the correct version and type of software installed.
show hardware	Confirms the type of hardware installed in the system.
show interfaces	Confirms the type of hardware installed in the system.
show atm addresses	Confirms the correct configuration of the ATM address.
ping atm	Tests for connectivity between the switch and a host.
show {**atm** | **ces**} **interface**	Confirms the correct configuration of the ATM interfaces.
show atm status	Confirms the status of the ATM interfaces.
show atm vc	Confirms the status of ATM virtual interfaces.
show running-config	Confirms the correct configuration.
show startup-config	Confirms the correct configuration saved in NVRAM.
show controllers {**atm** | **ethernet**}	Confirms interface controller memory addressing.

CHAPTER 18

Configuring Circuit Emulation Services

This chapter describes circuit emulation services (CES) and how to configure the CES T1/E1 port adapters in the Catalyst 8540 MSR, Catalyst 8510 MSR, and LightStream 1010 ATM switch routers. You can use CES T1/E1 port adapters for links that require constant bit rate (CBR) services.

Note This chapter provides advanced configuration instructions for the Catalyst 8540 MSR, Catalyst 8510 MSR, and LightStream 1010 ATM switch routers. For an overview of CES applications and operation, refer to the *Guide to ATM Technology*. For complete descriptions of the commands mentioned in this chapter, refer to the *ATM Switch Router Command Reference* publication. For hardware installation and cabling instructions, refer to the *ATM and Layer 3 Port Adapter and Interface Module Installation Guide*.

This chapter includes the following sections:

Overview of CES T1/E1 Interfaces

You can use CES T1/E1 port adapters for links that require CBR services, such as interconnecting PBXs, time-division multiplexers (TDMs), and video conference equipment over campus, public, or private networks.

This section provides an overview of the hardware features and functions supported on the CES T1/E1 port adapters.

Clocking Options

You can configure each interface on the port adapter to support the following clocking options:

- Self-timing based on a stratum 4 level clock
- Loop timing from the received data stream—ideal for public network connections
- Timing synchronized to a selected master clock port—required to distribute a single clock across a network

Interfaces Supported

The number of CES T1/E1 interfaces you can configure is platform dependent:

- Catalyst 8540 MSR—up to 64 CES T1/E1 interfaces
- Catalyst 8510 MSR and LightStream 1010—up to 32 CES T1/E1 interfaces

Connectors Supported

The CES T1 port adapters support UTP connectors and the CES E1 port adapters support UTP, foil twisted-pair, or 75-ohm BNC connectors. Status and carrier-detect LEDs on each port give quick, visual indications of port status and operation. For detailed network management support, comprehensive statistics gathering and alarm monitoring capabilities are provided.

Functions Supported by CES Modules

The functions supported by a CES module include the following:

- Circuit emulation services interworking function (CES-IWF), which enables communication between CBR and ATM UNI interfaces
- T1/E1 CES unstructured services
- T1/E1 CES structured services

Framing Formats and Line Coding Options for CES Modules

The CES modules support the framing formats and line coding options shown in Table 18-1.

Table 18-1 CES Module Framing and Line Coding Options

Module	Framing Options and Description	Line Coding Options
CES T1 port adapter	• Super Frame (SF) • Extended Super Frame (ESF)	ami or b8zs (b8zs is the default)
CES E1 port adapter (120-ohm) and CES E1 port adapter (BNC)	• E1 CRC multiframe (e1_crc_mf_lt). Configures the line type to e1_crc_mf, without channel associated signalling (CAS) enabled. • E1 CRC multiframe (e1_crc_mfCAS_lt). Configures the line type to e1_crc_mf, with CAS enabled. • E1 (e1_lt). Configures the line type to e1_lt. • E1 multiframe (e1_mfCAS_lt). Configures the line type to e1_mf, with CAS enabled.	ami or hdb3 (hdb3 is the default)

Default CES T1/E1 Interface Configuration

The following defaults are assigned to all CES T1/E1 interfaces:

- Loopback = no loopback
- Signalling mode = no signalling
- Transmit clock source = network-derived
- Data format = clear channel
- Line build-out (LBO) = 0 to 110 feet
- Cell delay variation = 2000 microseconds
- Channel associated signalling (CAS) = FALSE
- Partial fill = 47
- AAL1 service type = unstructured
- AAL1 clock mode = synchronous

The following defaults are assigned to CES T1 port adapters:

  – Framing = ESF
  – Line coding = B8ZS

The following defaults are assigned to CES E1 port adapters:

- Framing = E1_LT
- Line coding = HDB3
- International bits = 0x3
- National bits = 0x1f
- Multiframe spare bits = 0xb

Configuring CES T1/E1 Interfaces

To manually change any of the CES T1/E1 default configuration values, enter the **interface cbr** global configuration command to specify a CBR interface, as follows:

interface cbr *card*/*subcard*/*port*

To configure the CES T1/E1 interfaces perform the following commands, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **interface cbr** *card*/*subcard*/*port* Switch(config-if)#	Selects the physical interface to be configured and enters global configuration mode.
Step 2	Switch(config-if)# **shutdown**	Disables the interface.
Step 3	Switch(config-if)# **ces aal1 service** {**structured** | **unstructured**}	Configures the service type. The default is **unstructured**.
Step 4	Switch(config-if)# **ces aal1 clock** {**adaptive** | **srts** | **synchronous**}	Configures the type of clocking. **Note** For structured CES, the default is **synchronous**.
Step 5	Switch(config-if)# **ces circuit** *circuit-id* [**cas**] [**cdv** *max-req*] [**circuit-name** *name*] [**partial-fill** *number*] [**shutdown**] [**timeslots** *number*] [**on-hook-detect** *pattern*]	Configures the following CES connection attributes for the circuit: • Circuit id number. – For unstructured service, use 0. – For CES T1 structured service, use 1 through 24. – For CES E1 structured service, use 1 through 31. • Enables channel-associated signalling for structured service only. The default is no cas. • Enables the peak-to-peak cell delay variation requirement. The default is 2000 milliseconds.

	Command	Purpose
		• Sets the ASCII name for the CES-IWF circuit. The maximum length is 64 characters. The default is CBRx/x/x:0. • Enables the partial AAL1 cell fill service for structured service only. The default is 47. • Disables the circuit. The default is no shutdown. • Configures the time slots for the circuit for structured service only. • Configures on-hook detection.
Step 6	Switch(config-if)# **ces dsx1 clock source** {**loop-timed** | **network-derived**}	Configures the clock source. The default is **network-derived**.
Step 7	Switch(config-if)# **ces dsx1 framing** {**sf | esf**}	Configures CES T1 framing mode. The default is **esf**.
	Switch(config-if)# **ces dsx1 framing** {**e1_crc_mfCAS_lt** | **e1_crc_mf_lt** | **e1_lt** | **e1_mfCAS_lt**}	Configures CES E1 framing mode. The default is **e1_lt**.
Step 8	Switch(config-if)# **ces dsx1 lbo** {**0_110** | **110_220** | **220_330** | **330_440** | **440_550** | **550_660** | **660_above** | **square_pulse**}	Configures the line build-out. The default is **0_110**.
Step 9	Switch(config-if)# **ces dsx1 linecode** {**ami** | **b8zs**}	Configures CES T1 line code type. The default is **b8zs**.
	Switch(config-if)# **ces dsx1 linecode** {**ami | hdb3**}	Configures CES E1 line code type. The default is **hdb3**.
Step 10	Switch(config-if)# **ces dsx1 loopback** {**line** | **noloop** | **payload**}	Configures the loopback test method. The default is **noloop**.
Step 11	Switch(config-if)# **ces dsx1 signalmode robbedbit**	Configures the CES T1 signal mode to robbedbit. The default is **no**.
Step 12	Switch(config-if)# **ces pvc** *circuit-id* **interface atm** *card*/*subcard*/*port* [**vpi** *vpi*] **vci** *vci*	Configures the destination port for the circuit and configures a hard PVC, as follows: • Specifies the circuit identification. – For unstructured service, use 0. – For T1 structured service, use 1 through 24. – For E1 structured service, use 1 through 31. • Specifies the card/subcard/port number of the ATM interface. • Specifies the virtual path identifier of the destination PVC. • Specifies the virtual channel identifier of the destination PVC.

	Command	Purpose
	Switch(config-if)# **ces pvc** *circuit-id* **dest-address** *atm-address* [[**vpi** *vpi*] **vci** *vci*] [**retry-interval** [**first** *retry-interval*] [**maximum** *retry-interval*]] [**follow-ifstate**]	Configures the destination (active) port for the circuit and configures a soft PVC, as follows: • Specifies the circuit identification. – For unstructured service, use 0. – For T1 structured service, use 1 through 24. – For E1 structured service, use 1 through 31. • Specifies the destination address of the soft PVC. • Specifies the virtual path identifier of the destination PVC. • Specifies the virtual channel identifier of the destination PVC. • Configures retry interval timers for a soft PVC, as follows: – Specifies in milliseconds, the retry interval after the first failed attempt. The default is 5,000. – Specifies in seconds, the maximum retry interval between any two attempts. The default is 600. • Configures the source (active) port circuit status to follow the status of the physical interface. The default circuit setting ignores the status of the physical interface.
Step 13	Switch(config-if)# **ces pvc** *circuit-id* **follow-ifstate**	Configures the destination (passive) port circuit status for a soft-PVC to follow the status of the physical interface. The default circuit setting ignores the status of the physical interface.
Step 14	Switch(config-if)# **no shutdown**	Reenables the interface.

Examples

The following example shows how to change the default cell delay variation for circuit 0 to 30,000, using the **ces circuit** command:

```
Switch# configure terminal
Switch(config)# interface cbr 3/0/0
Switch(config-if)# shutdown
Switch(config-if)# ces circuit 0 cdv 3000
Switch(config-if)# no shutdown
```

Note You might have to use the **shutdown** command to shut down the interface before you can modify the circuit. After modifying the circuit, use the **no shutdown** command to reenable the interface.

The following example shows how to change the default CBR interface framing mode to super frame, using the **ces dsx1 framing** command:

```
Switch# configure terminal
Switch(config)# interface cbr 3/0/0
Switch(config-if)# ces dsx1 framing sf
```

The following example shows how to change the default CBR interface line build-out length to range from 330 to 440 feet, using the **ces dsx1 lbo** command:

```
Switch# configure terminal
Switch(config)# interface cbr 3/0/0
Switch(config-if)# ces dsx1 lbo 330_440
```

The following example shows how to change the default CBR interface line code method to binary 8 zero suppression, using the **ces dsx1 linecode** command:

```
Switch# configure terminal
Switch(config)# interface cbr 3/0/0
Switch(config-if)# ces dsx1 linecode b8zs
```

The following example shows how to change the default CBR interface loopback method to payload, using the **ces dsx1 loopback** command:

```
Switch# configure terminal
Switch(config)# interface cbr 3/0/0
Switch(config-if)# ces dsx1 loopback payload
```

See the "Troubleshooting the Interface Configuration" section on page 17-17 to confirm your interface configuration.

General Guidelines for Creating Soft PVCs for Circuit Emulation Services

You can create either hard permanent virtual channels (PVCs) or soft PVCs for unstructured or structured CES, depending on your particular CES application requirements. The main difference between hard and soft PVCs is rerouting in case of failure, as follows:

- A hard PVC on a CES T1/E1 port—Should a failure occur in a midpoint switch, hard PVCs are not automatically rerouted.
- A soft PVC on a CES T1/E1 port—Should a failure occur in a midpoint switch, soft PVCs are rerouted automatically, assuming another route is available.

This section provides general guidelines for configuring soft PVCs for CES modules. For specific instructions for configuring both hard and soft PVCs, see the following sections:

- Configuring T1/E1 Unstructured Circuit Emulation Services, page 18-9
- Configuring T1/E1 Structured (n x 64) Circuit Emulation Services, page 18-19

Note The steps in these guidelines assume that you have already used the **ces circuit** commands to configure circuits on the CES interfaces. If you have not yet configured circuits on the CES interfaces, the **show ces address** command will not display any addresses. For simplicity, the steps in these guidelines describe how to create a soft PVC between interface modules in the same ATM switch router.

To configure soft PVCs for either unstructured or structured circuit emulation services, follow these steps:

Step 1 Determine which CES interfaces are currently configured in your ATM switch router chassis, using the **show ces status** command in privileged EXEC mode.

```
CESwitch# show ces status
   Interface      IF      Admin        Port  Channels in
     Name       Status   Status        Type      use
------------- -------- --------- ----------- -----------
     CBR3/0/0       UP        UP          T1
     CBR3/0/1     DOWN        UP          T1
     CBR3/0/2     DOWN        UP          T1
     CBR3/0/3       UP        UP          T1
```

Step 2 Determine which two ports you want to define as participants in the soft PVC.

Step 3 Decide which of the two ports you want to designate as the destination (or passive) side of the soft PVC.

Note This is an arbitrary decision—you can choose either port as the destination end of the circuit. However, you must decide which port is to function in this capacity and proceed accordingly.

Step 4 Decide whether you want the state of the soft PVC to match the state of the ports.

Step 5 Configure the destination (passive) side of the soft PVC. You must configure the destination end of the soft PVC first, as this end defines a CES-IWF ATM address for that circuit.

Note If the interface is up, you might have to disable it, using the **shutdown** command, before you can configure the circuit. After configuring the circuit, use the **no shutdown** command to reenable the interface.

```
CESwitch# configure terminal
CESwitch(config)# interface cbr 3/0/1
CESwitch(config-if)# shutdown
CESwitch(config-if)# ces circuit 0 circuit-name CBR-PVC-B
CESwitch(config-if)# no shutdown
```

Step 6 Retrieve the CES-IWF ATM address of the soft PVC's destination end, using the **show ces address** command. The following example shows how to display the CES-IWF ATM address and VPI/VCI for a CES circuit:

```
CESwitch# show ces address

CES-IWF ATM Address(es):
47.0091.8100.0000.0060.5c71.1f01.4000.0c80.1030.10  CBR-PVC-A vpi 0 vci 16
47.0091.8100.0000.0060.5c71.1f01.4000.0c80.1030.20  CBR-PVC-AC vpi 0 vci 1056
47.0091.8100.0000.0060.5c71.1f01.4000.0c80.1034.10  CBR-PVC-B vpi 0 vci 1040
47.0091.8100.0000.0060.5c71.1f01.4000.0c80.1038.10  CBR-PVC-CA vp1 0 vci 3088
```

Step 7 Configure the source (active) end of the soft PVC last, using the information derived from Step 6. You must configure the source end of the soft PVC last, because that end not only defines the configuration information for the source port, but also requires you to enter the CES-IWF ATM address and VPI/VCI values for the destination circuit.

> **Note** If the interface is up, you might have to disable it, using the **shutdown** command, before you can configure the circuit. After configuring the circuit, use the **no shutdown** command to reenable the interface.

```
CESwitch(config)# interface cbr 3/0/0
CESwitch(config-if)# shutdown
CESwitch(config-if)# ces circuit 0
CESwitch(config-if)# ces pvc 0 dest-address 47.0091.8100.0000.0060.5c71.1f01.4000.0c80.1034.10 vpi 0 vci 104
CESwitch(config-if)# no shutdown
```

Step 8 To verify that the CES circuits are up on both sides (source and destination), run the **show ces interface** command. To verify that the soft PVC was established between two switches, run the **show atm vc interface** command.

Configuring T1/E1 Unstructured Circuit Emulation Services

This section provides an overview of unstructured (clear channel) circuit emulation services and describes how to configure CES modules for unstructured circuit emulation services.

Overview of Unstructured Circuit Emulation Services

Unstructured circuit emulation services in an ATM switch router network emulate point-to-point connections over T1/E1 leased lines. This service maps the entire bandwidth necessary for a T1/E1 leased line connection across the ATM network, allowing users to interconnect PBXs, TDMs, and video conferencing equipment.

For a detailed description of unstructured circuit emulation services, refer to the *Guide to ATM Technology*.

The circuit you set up on a CBR port for unstructured service is always identified as circuit 0, because you can establish only one unstructured circuit on any given CBR port. An unstructured circuit uses the entire bandwidth of a T1 port (1.544 Mbps) or an E1 port (2.048 Mbps).

The following subsections describe the procedures for configuring CES modules for unstructured circuit emulation services:

- Configuring a Hard PVC for Unstructured CES, page 18-10
- Verifying a Hard PVC for Unstructured CES, page 18-13
- Configuring a Soft PVC for Unstructured CES, page 18-14
- Verifying a Soft PVC for Unstructured CES, page 18-17

Configuring Network Clocking for Unstructured CES

Circuit emulation services require that the network clock be configured properly. Unstructured services can use synchronous, Synchronous Residual Time Stamp (SRTS), or adaptive clocking mode. For instructions on configuring network clocking, see the "Configuring Network Clocking" section on page 3-10. For a discussion of clocking issues and network examples, refer to the network clock synchronization and network clocking for CES topics in the *Guide to ATM Technology*.

Configuring Synchronous Clocking With an OC-12c Interface Module

When synchronous clocking is being used and propagated via an OC-12c interface module, be sure to use the following configurations:

- For the Catalyst 8540 MSR, use the optional clocking module.
- For the Catalyst 8510 MSR and LightStream 1010 ATM switch routers, use feature card per flow queueing (FC-PFQ).

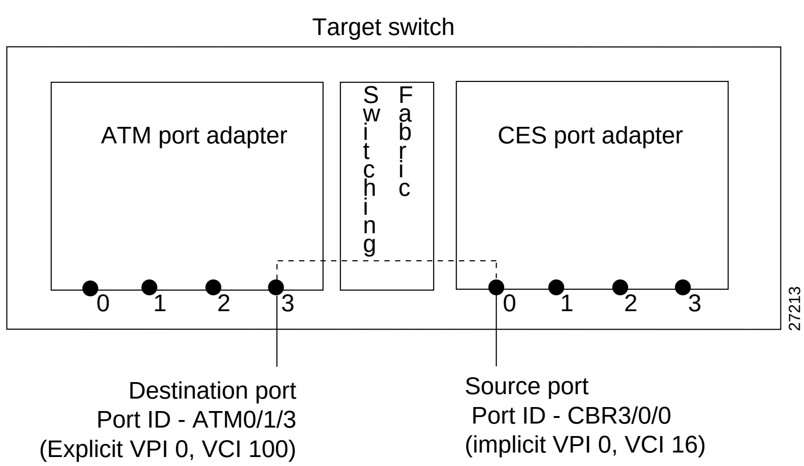
Configuring a Hard PVC for Unstructured CES

A CES module converts CBR traffic into ATM cells for propagation through an ATM network. CBR traffic arriving on a CES module port must first be segmented into ATM cells. This cell stream is then directed to an outgoing ATM or CBR port.

Note As a general rule when configuring a hard PVC, you must interconnect a CBR port and an ATM port in the same ATM switch router chassis.

Figure 18-1 displays unstructured circuit emulation services configured on an ATM switch router, using ATM and CES interface modules to create a hard PVC. In this example, the hard permanent virtual channel (PVC) also uses adaptive clocking, and this CES circuit enables bidirectional, unstructured CBR traffic to flow between these two modules.

Figure 18-1 Hard PVC Configured for Unstructured CES



To configure a hard PVC for unstructured CES, follow these steps, beginning in privileged EXEC mode:

	Command	Purpose
Step 1	Switch# **show ces status**	Displays information about the current CBR interfaces. Use this command to choose the source CBR port.
Step 2	Switch# **show atm status**	Displays information about the current ATM interfaces. Use this command to choose the destination ATM port. **Note** The interface must be up.
Step 3	Switch# **configure terminal** Switch(config)#	At the privileged EXEC prompt, enters global configuration mode.
Step 4	Switch(config)# **interface cbr** *card*/*subcard*/*port* Switch(config-if)#	Selects the physical interface to be configured.
Step 5	Switch(config-if)# **shutdown**	Disables the interface.
Step 6	Switch(config-if)# **ces aal1 service** {**structured** | **unstructured**}	Configures the CES interface AAL1 service type.
Step 7	Switch(config-if)# **ces aal1 clock** {**adaptive** | **srts** | **synchronous**}	Configures the AAL1 clock mode.
Step 8	Switch(config-if)# **ces circuit** *circuit-id* **circuit-name** *name*	Configures the CES interface circuit identifier and circuit name. **Note** For unstructured service, use 0 for the circuit identifier.
Step 9	Switch(config-if)# **ces pvc** *circuit-id* **interface atm** *card*/*subcard*/*port* **vpi** *vpi* **vci** *vci*	Configures the hard PVC to the ATM interface and VPI/VCI. **Note** The VPI/VCI are arbitrary here. They are not fixed, whereas the VPI/VCI described in the "General Guidelines for Creating Soft PVCs for Circuit Emulation Services" section on page 18-7 are fixed.
Step 10	Switch(config-if)# **no shutdown**	Reenables the interface.

Example

The following example shows how to configure the hard PVC for unstructured CES (shown in Figure 18-1):

```
CESwitch# show ces status
Interface      IF      Admin        Port  Channels in
     Name       Status   Status         Type      use
------------- -------- --------- ----------- -----------
     CBR3/0/0       UP        UP          T1
     CBR3/0/1     DOWN        UP          T1
     CBR3/0/2     DOWN        UP          T1
     CBR3/0/3       UP        UP          T1

CESwitch# show atm status
NUMBER OF INSTALLED CONNECTIONS: (P2P=Point to Point, P2MP=Point to MultiPoint,
MP2P=Multipoint to Point)

Type      PVCs SoftPVCs      SVCs      TVCs      PVPs SoftPVPs      SVPs       Total
P2P         27         2        13         0         0         0         0          42
P2MP         0         0         2         0         0         0         0           2
MP2P         0         0         0         0         0         0         0           0
                                      TOTAL INSTALLED CONNECTIONS =          44

PER-INTERFACE STATUS SUMMARY AT 18:12:45 UTC Thu Jul 22 1999:
   Interface      IF         Admin  Auto-Cfg    ILMI Addr     SSCOP    Hello
     Name       Status      Status    Status    Reg State     State    State
------------- -------- ------------ -------- ------------ --------- --------
ATM0/0/1          DOWN         down  waiting          n/a      Idle      n/a
ATM0/0/5          DOWN     shutdown  waiting          n/a      Idle      n/a
ATM0/0/6          DOWN     shutdown  waiting          n/a      Idle      n/a
ATM0/0/7          DOWN     shutdown  waiting          n/a      Idle      n/a
ATM0/0/ima1         UP           up     done  UpAndNormal    Active 2way_in
ATM0/1/0          DOWN     shutdown  waiting          n/a      Idle      n/a
ATM0/1/1          DOWN     shutdown  waiting          n/a      Idle      n/a
ATM0/1/2          DOWN     shutdown  waiting          n/a      Idle      n/a
ATM0/1/3          DOWN     shutdown  waiting          n/a      Idle      n/a
ATM0/1/7          DOWN         down  waiting          n/a      Idle      n/a
ATM0/1/ima2         UP           up     done  UpAndNormal    Active 2way_in
ATM1/0/0          DOWN         down  waiting          n/a      Idle      n/a
ATM1/0/1          DOWN         down  waiting          n/a      Idle      n/a
ATM1/0/2          DOWN         down  waiting          n/a      Idle      n/a
ATM1/0/3            UP           up     done  UpAndNormal    Active      n/a
ATM1/1/0            UP           up     done  UpAndNormal    Active      n/a
ATM1/1/1          DOWN         down  waiting          n/a      Idle      n/a
ATM1/1/2          DOWN         down  waiting          n/a      Idle      n/a
ATM1/1/3          DOWN         down  waiting          n/a      Idle      n/a
ATM2/0/0            UP           up      n/a  UpAndNormal      Idle      n/a
ATM-P3/0/3          UP           up  waiting          n/a      Idle      n/a
ATM3/1/0          DOWN         down  waiting          n/a      Idle      n/a
ATM3/1/1            UP           up     done  UpAndNormal    Active 2way_in
ATM3/1/1.99         UP           up     done  UpAndNormal    Active 2way_in
ATM3/1/2          DOWN         down  waiting          n/a      Idle      n/a
ATM3/1/3          DOWN         down  waiting          n/a      Idle      n/a
ATM-P4/0/0          UP           up  waiting          n/a      Idle      n/a

CESwitch# configure terminal
CESwitch(config)# interface cbr 3/0/0
CESwitch(config-if)# shutdown
CESwitch(config-if)# ces aal1 service unstructured
CESwitch(config-if)# ces aal1 clock adaptive
CESwitch(config-if)# ces circuit 0 circuit-name CBR-PVC-A
CESwitch(config-if)# ces pvc 0 interface atm 0/1/3 vpi 0 vci 100
CESwitch(config-if)# no shutdown
```

Verifying a Hard PVC for Unstructured CES

To verify the hard PVC configuration, use the following privileged EXEC commands:

Command	Purpose
show ces circuit	Shows configuration information for the hard PVC.
show ces circuit interface cbr *card*/*subcard*/*port* *circuit-id*	Shows detailed interface configuration information for the hard PVC.

Examples

The following example shows how to display the basic information about the hard PVC shown in Figure 18-1, using the **show ces circuit** command:

```
CESwitch# show ces circuit
Interface  Circuit  Circuit-Type     X-interface   X-vpi   X-vci Status
 CBR3/0/0    0       HardPVC           ATM0/1/3      0       100  UP
```

The output from this command verifies the source (CBR 3/0/0) and destination (ATM 0/1/3) port IDs of the hard PVC and indicates that the circuit is up.

The following example shows how to display detailed information about the hard PVC shown in Figure 18-1, using the **show ces circuit interface** command:

```
CESwitch# show ces circuit interface cbr 3/0/0 0
Circuit: Name CBR-PVC-A, Circuit-state ADMIN_UP / oper-state UP
Interface CBR3/0/0, Circuit_id 0, Port-Type T1, Port-State UP
Port Clocking network-derived, aal1 Clocking Method CESIWF_AAL1_CLOCK_ADAPT
Channel in use on this port: 1-24
Channels used by this circuit: 1-24
Cell-Rate: 4107, Bit-Rate 1544000
cas OFF, cell_header 0x100 (vci = 16)
Configured CDV 2000 usecs, Measured CDV unavailable
De-jitter: UnderFlow 903952, OverFlow 0
ErrTolerance 8, idleCircuitdetect OFF, onHookIdleCode 0x0
state: VcAlarm, maxQueueDepth      827, startDequeueDepth      437
Partial Fill:       47, Structured Data Transfer 0
HardPVC
src: CBR3/0/0 vpi 0, vci 16
Dst: ATM0/1/3 vpi 0, vci 100
```

The output from this command verifies the following configuration information:

- The circuit named CBR-PVC-A is in an UP state.
- The interface CBR 3/0/0 has a circuit id of 0 (because the entire bandwidth of the port is dedicated to that circuit).
- The AAL1 clocking method is adaptive clocking.
- The source port for the hard PVC is CBR 3/0/0. The destination port is ATM 0/1/3.

Configuring a Soft PVC for Unstructured CES

In a soft PVC, as well as a hard PVC, you configure both ends of the CES circuit. However, a soft PVC typically involves CES modules at opposite edges of an ATM network, so a soft PVC can be set up between any two CES modules anywhere in your network.

The destination address of a soft PVC can point to either of the following:

- Any ATM switch router external ATM port in the network
- A port in any other CES module in the network

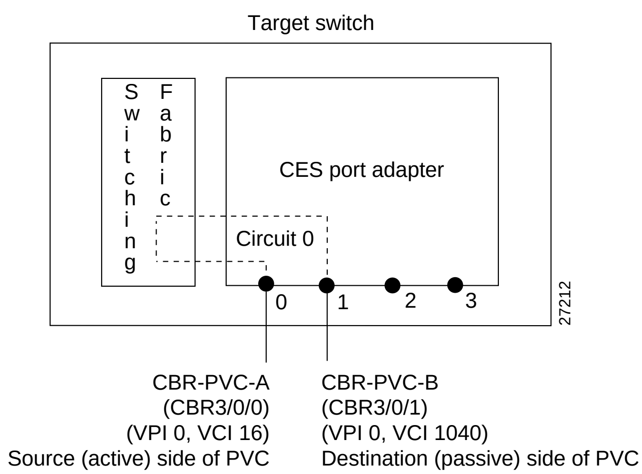
For example, to set up a soft PVC involving a local node and a destination node at the opposite edge of the network, you need to determine the CES-IWF ATM address of the port in the destination node to complete soft PVC setup.

To obtain the destination address (dest-address) for a port already configured in a CES port adapter, log into the remote ATM switch router containing that module. Then use the **show ces address** command to display all the CES-IWF ATM addresses currently configured for that node.

Figure 18-2 displays a soft PVC configured for unstructured CES. The soft PVC uses adaptive clocking and the source clock is network-derived.

Note Typically you will configure a soft PVC between CES modules anywhere in your network. For simplicity, this example and the accompanying procedure describe how to create a soft PVC between modules in the same ATM switch router chassis.

Figure 18-2 Soft PVC Configured for Unstructured CES



Configuring a soft PVC for unstructured CES is a two-phase process:

- Phase 1—Configuring the Destination (Passive) Side of the Soft PVC
- Phase 2—Configuring the Source (Active) Side of the Soft PVC

Phase 1—Configuring the Destination (Passive) Side of the Soft PVC

To configure the destination (passive) side of a soft PVC destination port, follow these steps, beginning in privileged EXEC mode:

	Command	Purpose
Step 1	Switch# **show ces status**	Displays information about current CBR interfaces. Use this command to choose the destination port.
Step 2	Switch# **configure terminal** Switch(config)#	At the privileged EXEC prompt, enters global configuration mode.
Step 3	Switch(config)# **interface cbr** *card*/*subcard*/*port* Switch(config-if)#	Selects the physical interface to be configured.
Step 4	Switch(config-if)# **shutdown**	Disables the interface.
Step 5	Switch(config-if)# **ces aal1 service** {**structured** | **unstructured**}	Configures the CES interface AAL1 service type.
Step 6	Switch(config-if)# **ces aal1 clock** {**adaptive** | **srts** | **synchronous**}	Configures the CES interface AAL1 clock mode.
Step 7	Switch(config-if)# **ces dsx1 clock source** {**loop-timed** | **network-derived**}	Configures the CES interface clock source.
Step 8	Switch(config-if)# **ces circuit** *circuit-id* **circuit-name** *name*	Configures the CES interface circuit identifier and circuit name. **Note** For unstructured service, use 0 for the circuit identifier.
Step 9	Switch(config-if)# **ces pvc** *circuit-id* **passive follow-ifstate**	Configures the destination (passive) port circuit status to follow the status of the physical interface. The default circuit setting ignores the status of the physical interface.
Step 10	Switch(config-if)# **no shutdown**	Reenables the interface.

Example

The following example shows how to configure the destination (passive) side of a soft PVC, as shown in Figure 18-2:

```
CESwitch# show ces status

   Interface      IF      Admin        Port  Channels in
     Name       Status   Status        Type      use
------------- -------- --------- ----------- -----------
     CBR3/0/0       UP        UP          T1
     CBR3/0/1       UP        UP          T1
     CBR3/0/2       UP        UP          T1
     CBR3/0/3       UP        UP          T1

CESwitch# configure terminal
CESwitch(config)# interface cbr 3/0/1
CESwitch(config-if)# shutdown
CESwitch(config-if)# ces aal1 service unstructured
CESwitch(config-if)# ces aal1 clock synchronous
CESwitch(config-if)# ces dsx1 clock source network-derived
CESwitch(config-if)# ces circuit 0 circuit-name CBR-PVC-B
CESwitch(config-if)# no shutdown
```

Note If you do not specify the circuit name and logical name parameters in the command line, the system automatically assigns a unique default name in the form CBR*x/y/z:#* for the circuit being configured. For example, the default name for this particular circuit is CBR3/0/1:0.

Phase 2—Configuring the Source (Active) Side of the Soft PVC

To configure the source (active) side of a soft PVC destination port, follow these steps, beginning in privileged EXEC mode:

	Command	Purpose
Step 1	Switch# **show ces address**	Shows the CES address and VPI/VCI for the destination end of the circuit. Use this command to retrieve the destination's VPI/VCI.
Step 2	Switch# **configure terminal** Switch(config)#	At the privileged EXEC prompt, enters global configuration mode.
Step 3	Switch(config)# **interface cbr** *card*/*subcard*/*port* Switch(config-if)#	Selects the physical interface to be configured.
Step 4	Switch(config-if)# **shutdown**	Disables the interface.
Step 5	Switch(config-if)# **ces aal1 service** {**structured** | **unstructured**}	Configures the CES interface AAL1 service type.
Step 6	Switch(config-if)# **ces aal1 clock** {**adaptive** | **srts** | **synchronous**}	Configures the CES interface AAL1 clock mode.
Step 7	Switch(config-if)# **ces dsx1 clock source** {**loop-timed** | **network-derived**}	Configures the CES interface clock source.

	Command	Purpose
Step 8	Switch(config-if)# **ces circuit** *circuit-id* **circuit-name** *name*	Configures the CES interface circuit identifier and circuit name. **Note** For unstructured service, use 0 for the circuit identifier.
Step 9	Switch(config-if)# **ces pvc** *circuit-id* **dest-address** *remote_atm_address* **vpi** *vpi* **vci** *vci* [**follow-ifstate**]	Configures the soft PVC to the destination CES-IWF ATM addresses and VPI/VCI of the circuit. **Note** Use the destination's VPI/VCI, which you retrieved in Step 1. The **follow-ifstate** keyword configures the source (active) port circuit status to follow the status of the physical interface. The default circuit setting ignores the status of the physical interface.
Step 10	Switch(config-if)# **no shutdown**	Reenables the interface.

Example

The following example shows how to configure the source (active) side of a soft PVC, as shown in Figure 18-2:

```
CESwitch# show ces address

CES-IWF ATM Address(es):
47.0091.8100.0000.0060.5c71.1f01.4000.0c80.1034.10  CBR-PVC-B

CESwitch# configure terminal
CESwitch(config)# interface cbr 3/0/0
CESwitch(config-if)# shutdown
CESwitch(config-if)# ces aal1 service unstructured
CESwitch(config-if)# ces aal1 clock synchronous
CESwitch(config-if)# ces dsx1 clock source network-derived
CESwitch(config-if)# ces circuit 0 circuit-name CBR-PVC-A
CESwitch(config-if)# ces pvc 0 dest-address 47.0091.8100.0000.0060.5c71.1f01.4000.0c80.1034.10 vpi 0
vci 1040
CESwitch(config-if)# no shutdown
```

Verifying a Soft PVC for Unstructured CES

To verify the soft PVC configuration, use the following privileged EXEC commands:

Command	Purpose
show ces circuit	Shows the soft PVC configuration information.
show ces circuit interface cbr *card/subcard/port circuit-id*	Shows the detailed soft PVC interface configuration information.

Examples

The following example shows how to display the soft PVC configured in the previous section (shown in Figure 18-2), using the **show ces circuit** command:

```
CESwitch# show ces circuit
Interface  Circuit  Circuit-Type     X-interface   X-vpi   X-vci Status
 CBR3/0/0    0       Active SoftVC   ATM-P3/0/3      0        16 UP
 CBR3/0/1    0       Passive SoftVC  ATM-P3/0/3      0      1040 UP
```

The following example shows how to display the detailed circuit information for CBR 3/0/1, the destination (passive) side of the soft PVC (shown in Figure 18-2), using the **show ces circuit interface cbr** command:

```
CESwitch# show ces circuit interface cbr 3/0/1 0
Circuit: Name CBR-PVC-B, Circuit-state ADMIN_UP / oper-state UP
Interface CBR3/0/1, Circuit_id 0, Port-Type T1, Port-State UP
Port Clocking network-derived, aal1 Clocking Method CESIWF_AAL1_CLOCK_SYNC
Channel in use on this port: 1-24
Channels used by this circuit: 1-24
Cell-Rate: 4107, Bit-Rate 1544000
cas OFF, cell_header 0xC100 (vci = 3088)
Configured CDV 2000 usecs, Measured CDV 2378 usecs
De-jitter: UnderFlow 137, OverFlow 0
ErrTolerance 8, idleCircuitdetect OFF, onHookIdleCode 0x0
state: VcActive, maxQueueDepth      823, startDequeueDepth      435
Partial Fill:       47, Structured Data Transfer 0
Passive SoftVC
Src: atm addr 47.0091.8100.0000.00e0.4fac.b401.4000.0c81.803c.10 vpi 0, vci 1040
Dst: atm addr 47.0091.8100.0000.00e0.4fac.b401.4000.0c81.8030.00
```

The following example shows how to display the detailed circuit information for CBR 3/0/0, the source (active) side of the soft PVC (shown in Figure 18-2), using the **show ces circuit interface cbr** command:

```
CESwitch# show ces circuit interface cbr 3/0/0 0
Circuit: Name CBR-PVC-A, Circuit-state ADMIN_UP / oper-state UP
Interface CBR3/0/0, Circuit_id 0, Port-Type T1, Port-State UP
Port Clocking network-derived, aal1 Clocking Method CESIWF_AAL1_CLOCK_SYNC
Channel in use on this port: 1-24
Channels used by this circuit: 1-24
Cell-Rate: 4107, Bit-Rate 1544000
cas OFF, cell_header 0x100 (vci = 16)
Configured CDV 2000 usecs, Measured CDV 326 usecs
De-jitter: UnderFlow 1, OverFlow 0
ErrTolerance 8, idleCircuitdetect OFF, onHookIdleCode 0x0
state: VcAlarm, maxQueueDepth      823, startDequeueDepth      435
Partial Fill:       47, Structured Data Transfer 0
Active SoftVC
Src: atm addr 47.0091.8100.0000.00e0.4fac.b401.4000.0c81.8030.10 vpi 0, vci 16
Dst: atm addr 47.0091.8100.0000.00e0.4fac.b401.4000.0c81.803c.10
```

Configuring T1/E1 Structured (*n* x 64) Circuit Emulation Services

This section provides an overview of structured (*n* x 64 Kbps) circuit emulation services and describes how to configure CES modules for structured circuit emulation services.

Overview of Structured Circuit Emulation Services

An important distinction between structured and unstructured circuit emulation services is that structured circuit emulation services allow you to allocate T1/E1 bandwidth. Structured circuit emulation services only use the T1/E1 bandwidth actually required to support the active structured circuit(s) you configure.

For example, configuring a CES module for structured services allows you to define multiple hard PVCs or soft PVCs for any CES T1 or E1 port. In both module types, any bits not available for structured circuit emulation services are used for framing and out-of-band control.

n x 64 refers to a circuit bandwidth (data transmission speed) provided by the aggregation of *n* x 64-Kbps channels, where *n* is an integer greater than or equal to 1. The 64-Kbps data rate, or the DS0 channel, is the basic building block of the T carrier systems (T1, T2, and T3).

The T1/E1 structured (*n* x 64) circuit emulation services enable a CES module to function in the same way as a classic Digital Access and Crossconnect System (DACS) switch.

The Simple Gateway Control Protocol (SGCP) provides similar functionality by controlling structured CES circuits for voice over ATM. For additional information see the "Configuring SGCP" section on page 18-48.

For a detailed description of structured circuit emulation services, refer to the *Guide to ATM Technology*.

Configuring Network Clocking for Structured CES

Circuit emulation services require that the network clock be configured properly. For structured services, synchronous clocking is required. For instructions on configuring network clocking, see the "Configuring Network Clocking" section on page 3-10. For a discussion of clocking issues and network examples, refer to the network clock synchronization and network clocking for CES topics in the *Guide to ATM Technology*.

Configuring Synchronous Clocking With an OC-12c Interface Module

When synchronous clocking is being used and propagated via an OC-12c interface module, be sure to use the following configurations:

- For the Catalyst 8540 MSR, use the optional clocking module.
- For the Catalyst 8510 MSR and LightStream 1010 ATM switch routers, use feature card per flow queueing (FC-PFQ).

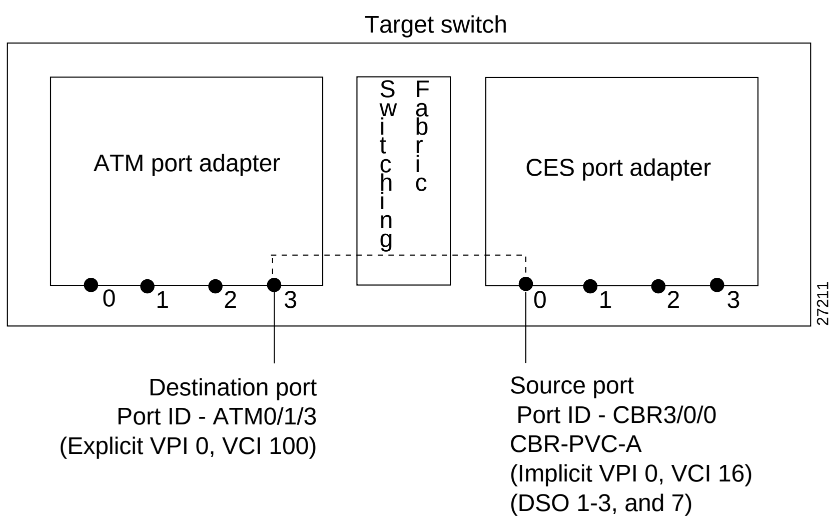
Configuring a Hard PVC for Structured CES

This section describes how to configure a hard permanent virtual channel (PVC) for structured circuit emulation services.

Figure 18-3 shows that the hard PVC for structured CES connection is configured with the following parameters:

- Four time slots (DS0 channels 1 to 3, and 7) are configured for a circuit named CBR-PVC-A.
- ATM port 0/1/3 in the ATM switch router is designated as the destination port of the hard PVC.
- The CES AAL1 service is structured and the clock source is network-derived.
- The framing is esf and the line code is b8zs.

Figure 18-3 Hard PVC Configured for Structured CES



To configure the CES port for structured CES, follow these steps, beginning in privileged EXEC mode:

	Command	Purpose
Step 1	Switch# **show ces status**	Displays information about current CBR interfaces. Use this command to choose the source port.
Step 2	Switch# **show atm status**	Displays information about current ATM interfaces. Use this command to choose the destination port.
Step 3	Switch# **configure terminal** Switch(config)#	At the privileged EXEC prompt, enters global configuration mode.
Step 4	Switch(config)# **interface cbr** *card*/*subcard*/*port* Switch(config-if)#	Selects the physical interface to be configured.
Step 5	Switch(config-if)# **shut**	

	Command	Purpose
Step 6	Switch(config-if)# **ces aal1 service** {**structured** | **unstructured**}	Configures the CES interface AAL1 service type.
Step 7	Switch(config-if)# **ces dsx1 clock source** {**loop-timed** | **network-derived**}	Configures the CES interface clock source.
Step 8	Switch(config-if)# **ces dsx1 framing** {**sf** | **esf**}	Configures the CES T1 framing type. The default is **esf**.
	Switch(config-if)# **ces dsx1 framing** {**e1_crc_mfCAS_lt** | **e1_crc_mf_lt** | **e1_lt** | **e1_mfCAS_lt**}	Configures the CES E1 framing type. For CES E1, the default is **e1_lt**.

Example

The following example shows how to configure the hard PVC for structured T1 CES, as shown in Figure 18-3:

```
CESwitch# show ces status

   Interface      IF      Admin        Port  Channels in
     Name       Status   Status        Type      use
------------- -------- --------- ----------- -----------
     CBR3/0/0       UP        UP          T1
     CBR3/0/1       UP        UP          T1
     CBR3/0/2       UP        UP          T1
     CBR3/0/3       UP        UP          T1

CESwitch# show atm status

NUMBER OF INSTALLED CONNECTIONS: (P2P=Point to Point, P2MP=Point to MultiPoint,
MP2P=Multipoint to Point)

Type     PVCs SoftPVCs     SVCs     TVCs     PVPs SoftPVPs     SVPs      Total
P2P        27        2       13        0        0        0        0         42
P2MP        0        0        2        0        0        0        0          2
MP2P        0        0        0        0        0        0        0          0
                                      TOTAL INSTALLED CONNECTIONS =         44

PER-INTERFACE STATUS SUMMARY AT 18:12:45 UTC Thu Jul 22 1999:
   Interface      IF        Admin  Auto-Cfg    ILMI Addr     SSCOP    Hello
     Name       Status      Status   Status    Reg State     State    State
------------- -------- ------------ -------- ------------ --------- --------
ATM0/0/1          DOWN         down  waiting          n/a      Idle      n/a
ATM0/0/5          DOWN     shutdown  waiting          n/a      Idle      n/a
ATM0/0/6          DOWN     shutdown  waiting          n/a      Idle      n/a
ATM0/0/7          DOWN     shutdown  waiting          n/a      Idle      n/a
ATM0/1/0          DOWN     shutdown  waiting          n/a      Idle      n/a
ATM0/1/1          DOWN     shutdown  waiting          n/a      Idle      n/a
ATM0/1/2          DOWN     shutdown  waiting          n/a      Idle      n/a
ATM0/1/3            UP           up     done  UpAndNormal    Active      n/a
ATM0/1/7          DOWN         down  waiting          n/a      Idle      n/a
ATM1/0/0          DOWN         down  waiting          n/a      Idle      n/a
ATM1/0/1          DOWN         down  waiting          n/a      Idle      n/a
ATM1/0/2          DOWN         down  waiting          n/a      Idle      n/a
ATM1/0/3            UP           up     done  UpAndNormal    Active      n/a
ATM1/1/0            UP           up     done  UpAndNormal    Active      n/a
ATM1/1/1          DOWN         down  waiting          n/a      Idle      n/a
ATM1/1/2          DOWN         down  waiting          n/a      Idle      n/a
ATM1/1/3          DOWN         down  waiting          n/a      Idle      n/a
```

```
ATM2/0/0           UP          up      n/a  UpAndNormal      Idle      n/a
ATM-P3/0/3         UP          up  waiting          n/a      Idle      n/a
ATM3/1/0         DOWN        down  waiting          n/a      Idle      n/a
ATM3/1/1           UP          up     done  UpAndNormal    Active 2way_in
ATM3/1/2         DOWN        down  waiting          n/a      Idle      n/a
ATM3/1/3         DOWN        down  waiting          n/a      Idle      n/a
ATM-P4/0/0         UP          up  waiting          n/a      Idle      n/a

CESwitch# configure terminal
CESwitch(config)# interface cbr 3/0/0
CESwitch(config-if)# shutdown
CESwitch(config-if)# ces aal1 service structured
CESwitch(config-if)# ces dsx1 clock source network-derived
CESwitch(config-if)# ces dsx1 framing esf
CESwitch(config-if)# ces dsx1 linecode b8zs
CESwitch(config-if)# ces circuit 1 timeslots 1-3,7
CESwitch(config-if)# ces circuit 1 circuit-name CBR-PVC-A
CESwitch(config-if)# ces pvc 1 interface atm 0/1/3 vpi 0 vci 100
CESwitch(config-if)# no shutdown
```

Note If you do not specify the circuit name and logical name parameters in the command line, the system automatically assigns a unique default name in the form CBR*x/y/z:#* for the circuit being configured. For example, the default name for this particular circuit is CBR3/0/0:1. For structured CES, the circuit number sequence always begins at 1 for each port in a CES module.

The virtual path identifier/virtual channel identifier (VPI/VCI) values shown in the example (vpi 0 vci 100) are for demonstration purposes only. The service provider you select gives you a virtual path for your data, but you must decide which VCI number to assign to the circuit.

Verifying a Hard PVC for Structured CES

To verify the hard PVC configured with structured services, use the following privileged EXEC commands:

Command	Purpose
show ces circuit	Shows the configuration information for the hard PVC.
show ces circuit interface cbr *card*/*subcard*/*port circuit-id*	Shows the detailed interface configuration information for the hard PVC.

Examples

The following example shows the details of the hard PVC, shown in Figure 18-3, using the **show ces circuit** command:

```
CESwitch# show ces circuit
Interface  Circuit  Circuit-Type     X-interface   X-vpi   X-vci Status
 CBR3/0/0    1       HardPVC           ATM0/1/3      0       100 UP
```

The output from this command verifies the source (CBR 3/0/0) and destination (ATM 0/1/3) port IDs of the hard PVC and indicates that the circuit is up.

The following example shows the interface details for port CBR 3/0/0 (shown in Figure 18-3), using the **show ces circuit interface cbr** command:

```
CESwitch# show ces circuit interface cbr 3/0/0 1
Circuit: Name CBR-PVC-A, Circuit-state ADMIN_UP / oper-state UP
Interface CBR3/0/0, Circuit_id 1, Port-Type T1, Port-State UP
Port Clocking network-derived, aal1 Clocking Method CESIWF_AAL1_CLOCK_SYNC
Channel in use on this port: 1-3, 7
Channels used by this circuit: 1-3, 7
Cell-Rate: 4107, Bit-Rate 1544000
cas OFF, cell_header 0x100 (vci = 16)
Configured CDV 2000 usecs, Measured CDV 326 usecs
De-jitter: UnderFlow 1, OverFlow 0
ErrTolerance 8, idleCircuitdetect OFF, onHookIdleCode 0x0
state: VcAlarm, maxQueueDepth      823, startDequeueDepth      435
Partial Fill:       47, Structured Data Transfer 1
HardPVC
Src: CBR3/0/0 vpi 0, vci 16
Dst: ATM0/1/3 vpi 0, vci 100
```

The output from this command verifies the following configuration information:

- The circuit named CBR-PVC-A is in an UP state.
- The interface CBR 3/0/0 has a circuit id of 1 (because structured CES services always begin at 1 for each port in a CES module).
- The channels being used by this circuit are 1-3 and 7.
- The source port for the hard PVC is CBR 3/0/0. The destination port is ATM 0/1/3.

Configuring a Hard PVC for Structured CES with a Shaped VP Tunnel

A shaped VP tunnel is a VP tunnel that, by default, carries only VCs of the constant bit rate (CBR) service category with a peak cell rate (PCR). However, it is possible to configure a shaped virtual path (VP) tunnel to carry VCs of other service categories. The overall output of the shaped VP tunnel is rate-limited, by hardware, to the PCR of the tunnel.

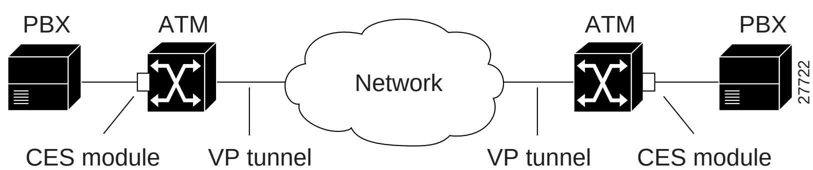
This section describes how to configure a hard PVC for structured CES with a shaped VP tunnel, which is a two-phase process, as follows:

- Phase 1—Configuring a Shaped VP Tunnel
- Phase 2—Configuring a Hard PVC

For more information about configuring shaped VP tunnels, see the "Configuring VP Tunnels" section on page 6-31.

Figure 18-4 shows an example of a how a structured CES circuit can be configured with a shaped VP tunnel.

Figure 18-4 Structured CES Circuit Configured with a Shaped VP Tunnel



Phase 1—Configuring a Shaped VP Tunnel

To configure a shaped VP tunnel, follow these steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch# **configure terminal** Switch(config)#	At the privileged EXEC prompt, enters global configuration mode.
Step 2	Switch(config)# **atm connection-traffic-table-row** [**index** *row-index*] **cbr pcr** *rate*	Configures the connection traffic table row for the desired PVP CBR cell rate.
Step 3	Switch(config)# **interface atm** *card*/*subcard*/*port* Switch(config-if)#	Selects the physical interface to be configured.
Step 4	Switch(config-if)# **shutdown**	Disables the interface.
Step 5	Switch(config-if)# **atm pvp** *vpi* [**hierarchical** | **shaped**] [**rx-cttr** *index*] [**tx-cttr** *index*]	Configures a shaped VP tunnel, as follows: • Specifies whether the tunnel is hierarchical or shaped. **Note** To configure a shaped VP tunnel to carry PVCs of other (non-CBR) service categories, the VP tunnel must be configured as a hierarchical tunnel. • Specifies the connection traffic table row in the received direction. The default is 1. • Specifies the connection traffic table row in the transmitted direction. The default is 1.
Step 6	Switch(config-if)# **no shutdown**	Reenables the interface.
Step 7	Switch(config-if)# **interface atm** *card*/*subcard*/*port***.***subinterface#* Switch(config-subif)#	Configures a subinterface. **Note** You cannot create a subinterface on the route processor interface ATM 0.
Step 8	Switch(config-subif)# **exit** Switch(config)#	Exits subinterface mode.

Note Even though the shaped VP tunnel is defined as CBR, it can carry PVCs of another service category by substituting the new service category after the tunnel interface has been initially configured. For information about configuring VP tunnels with other (non-CBR) service categories, see the"Configuring VP Tunnels" section on page 6-31.

Example

The following example shows how to configure a shaped VP tunnel.

```
CESwitch# configure terminal
CESwitch(config)# atm connection-traffic-table-row index 10 cbr pcr 4000
CESwitch(config)# interface atm 0/0/0
CESwitch(config-if)# shutdown
CESwitch(config-if)# atm pvp 1 shaped rx-cttr 10 tx-cttr 10
CESwitch(config-if)# no shutdown
CESwitch(config-if)# interface atm 0/0/0.1
CESwitch(config-subif)# exit
CESwitch(config)#
```

Note A shaped VP tunnel is defined as a CBR VP with a PCR. A maximum of 64 shaped VP tunnels can be defined on each of the following interface groups: (0/0/*x*, 1/0/*x*), (0/1/*x*, 1/1/*x*), (2/0/*x*, 3/0/*x*), (2/1/*x*, 3/1/*x*), (9/0/*x*, 10/0/*x*), (9/1/*x*, 10/1/*x*), (11/0/*x*, 12/0/*x*) and (11/1/*x*, 12/1/*x*). For further limitations on shaped VP tunnels, see the "Configuring a Shaped VP Tunnel" section on page 6-34.

Phase 2—Configuring a Hard PVC

To configure a hard PVC, follow these steps:

	Command	Purpose
Step 1	Switch# **show ces status**	Displays information about the current CBR interfaces. Use this command to choose the source CBR port.
Step 2	Switch# **show atm status**	Displays information about the current ATM interfaces. Use this command to choose the destination ATM port. **Note** The interface must be up.
Step 3	Switch# **configure terminal** Switch(config)#	At the privileged EXEC prompt, enters global configuration mode.
Step 4	Switch(config)# **interface cbr** *card/subcard/port* Switch(config-if)#	Selects the physical interface to be configured.
Step 5	Switch(config-if)# **shutdown**	Disables the interface.
Step 6	Switch(config-if)# **ces aal1 service** {**structured** | **unstructured**}	Configures the CES interface AAL1 service type.

	Command	Purpose
Step 7	Switch(config-if)# **ces circuit** *circuit-id* [**timeslots** *number*]	Configures the following CES connection attributes for the circuit: • Circuit id number. – For CES T1 structured service, use 1 through 24. – For CES E1 structured service, use 1 through 31. **Note** The 0 circuit identifier is reserved for unstructured service. • Time slots for the circuit for structured service only. – For CES T1, the range is 1 through 24. • For CES E1, the range is 1 through 31.
Step 8	Switch(config-if)# **ces pvc** *circuit-id* **interface atm** *card/subcard/port* **vpi** *vpi* **vci** *vci*	Configures the destination port for the circuit and configures a hard PVC, as follows: • Specifies the circuit identification. (Use the circuit id from the previous step.) • Specifies the card/subcard/port number of the ATM interface. • Specifies the VPI of the destination PVC. • Specifies the VCI of the destination PVC.
Step 9	Switch(config-if)# **no shutdown**	Reenables the interface.

Example

The following example shows how to configure hard PVCs for the shaped VP tunnel.

```
CESwitch# show ces status
Interface      IF      Admin        Port  Channels in
     Name       Status   Status        Type      use
------------- -------- --------- ----------- -----------
     CBR3/1/0       UP        UP          T1
     CBR3/1/1       UP        UP          T1
     CBR3/1/2       UP        UP          T1
     CBR3/1/3       UP        UP          T1
CESwitch# show atm status

NUMBER OF INSTALLED CONNECTIONS: (P2P=Point to Point, P2MP=Point to MultiPoint,
MP2P=Multipoint to Point)

Type      PVCs SoftPVCs      SVCs      TVCs      PVPs SoftPVPs      SVPs       Total
P2P         27        2        13         0         0        0         0          42
P2MP         0        0         2         0         0        0         0           2
MP2P         0        0         0         0         0        0         0           0
                                      TOTAL INSTALLED CONNECTIONS =          44
```

```
PER-INTERFACE STATUS SUMMARY AT 18:12:45 UTC Thu Jul 22 1999:
   Interface      IF         Admin  Auto-Cfg    ILMI Addr     SSCOP    Hello
     Name       Status      Status    Status    Reg State     State    State
------------- -------- ------------ -------- ------------ --------- --------
ATM0/0/1          DOWN        down  waiting          n/a      Idle      n/a
ATM0/0/5          DOWN    shutdown  waiting          n/a      Idle      n/a
ATM0/0/6          DOWN    shutdown  waiting          n/a      Idle      n/a
ATM0/0/7          DOWN    shutdown  waiting          n/a      Idle      n/a
ATM0/0/ima1         UP          up     done  UpAndNormal    Active 2way_in
ATM0/1/0          DOWN    shutdown  waiting          n/a      Idle      n/a
ATM0/1/1          DOWN    shutdown  waiting          n/a      Idle      n/a
ATM0/1/2          DOWN    shutdown  waiting          n/a      Idle      n/a
ATM0/1/3            UP          up     done  UpAndNormal    Active      n/a
ATM0/1/7          DOWN        down  waiting          n/a      Idle      n/a
ATM0/1/ima2         UP          up     done  UpAndNormal    Active 2way_in
ATM1/0/0          DOWN        down  waiting          n/a      Idle      n/a
ATM1/0/1          DOWN        down  waiting          n/a      Idle      n/a
ATM1/0/2          DOWN        down  waiting          n/a      Idle      n/a
ATM1/0/3            UP          up     done  UpAndNormal    Active      n/a
ATM1/1/0            UP          up     done  UpAndNormal    Active      n/a
ATM1/1/1          DOWN        down  waiting          n/a      Idle      n/a
ATM1/1/2          DOWN        down  waiting          n/a      Idle      n/a
ATM1/1/3          DOWN        down  waiting          n/a      Idle      n/a
ATM2/0/0            UP          up      n/a  UpAndNormal      Idle      n/a
ATM-P3/0/3          UP          up  waiting          n/a      Idle      n/a
ATM3/1/0          DOWN        down  waiting          n/a      Idle      n/a
ATM3/1/1            UP          up     done  UpAndNormal    Active 2way_in
ATM3/1/1.99         UP          up     done  UpAndNormal    Active 2way_in
ATM3/1/2          DOWN        down  waiting          n/a      Idle      n/a
ATM3/1/3          DOWN        down  waiting          n/a      Idle      n/a
ATM-P4/0/0          UP          up  waiting          n/a      Idle      n/a

CESwitch# configure terminal
CESwitch(config)# interface cbr 3/1/0
CESwitch(config-if)# shutdown
CESwitch(config-if)# ces aal1 service structured
CESwitch(config-if)# ces circuit 1 timeslots 1
CESwitch(config-if)# ces pvc 1 interface atm 0/0/0.1 vpi 1 vci 101

CESwitch(config-if)# ces circuit 2 timeslots 2
CESwitch(config-if)# ces pvc 2 interface atm 0/0/0.1 vpi 1 vci 102

CESwitch(config-if)# ces circuit 3 timeslots 3
CESwitch(config-if)# ces pvc 3 interface atm 0/0/0.1 vpi 1 vci 103
CESwitch(config-if)# no shutdown
```

Verifying a Hard PVC for Structured CES with a Shaped VP Tunnel

To verify the hard PVC configuration, use the following privileged EXEC commands:

Command	Purpose
show ces circuit	Shows configuration information for the hard PVC.
show ces circuit interface cbr *card*/*subcard*/*port circuit-id*	Shows detailed interface configuration information for the hard PVC.
show atm vp interface atm *card*/*subcard*/*port vpi*	Show detailed interface configuration information for the shaped VP tunnel.

Examples

The following example shows how to display the basic information about the hard PVC shown in Figure 18-4, using the **show ces circuit** command:

```
CESwitch# show ces circuit

Interface  Circuit  Circuit-Type     X-interface   X-vpi   X-vci Status
 CBR3/1/0    1       HardPVC         ATM0/0/0.1      1       101  DOWN
 CBR3/1/0    2       HardPVC         ATM0/0/0.1      1       102  DOWN
 CBR3/1/0    3       HardPVC         ATM0/0/0.1      1       103  DOWN
 CBR3/1/3    0       Active SoftVC      UNKNOWN      0         0  DOWN
```

The following example shows how to display detailed information about the hard PVC shown in Figure 18-4, using the **show ces circuit interface** command:

```
CESwitch# show ces circuit interface cbr 3/1/0 1
Circuit: Name CBR3/1/0:1, Circuit-state ADMIN_UP / oper-state UP Interface CBR3
Port Clocking loop-timed, aal1 Clocking Method CESIWF_AAL1_CLOCK_SYNC
Channel in use on this port: 1-3
Channels used by this circuit: 1
Cell-Rate: 172, Bit-Rate 64000
cas OFF, cell_header 0x100 (vci = 16)
Configured CDV 2000 usecs, Measured CDV unavailable
De-jitter: UnderFlow unavailable, OverFlow unavaliable
ErrTolerance 8, idleCircuitdetect OFF, onHookIdleCode 0x0
state: VcLoc, maxQueueDepth       81, startDequeueDepth       64
Partial Fill:       47, Structured Data Transfer 1
HardPVC
src: CBR3/1/0 vpi 0, vci 16
Dst: ATM0/0/0 vpi 1, vci 101
```

The following example shows how to display detailed information about the shaped VP tunnel shown in Figure 18-4, using the **show atm vp** command:

```
NewLs1010# show atm vp interface atm 0/0/0 1

Interface: ATM0/0/0, Type: oc3suni
VPI = 1
Status: SHAPED TUNNEL
Time-since-last-status-change: 13:59:23
Connection-type: PVP
Cast-type: point-to-point
Usage-Parameter-Control (UPC): pass
Wrr weight: 2
Number of OAM-configured connections: 0
OAM-configuration: disabled
OAM-states:  Not-applicable
Threshold Group: 1, Cells queued: 0
Rx cells: 0, Tx cells: 0
Tx Clp0:0,  Tx Clp1: 0
Rx Clp0:0,  Rx Clp1: 0
Rx Upc Violations:0, Rx cell drops:0
Rx Clp0 q full drops:0, Rx Clp1 qthresh drops:0
→ Rx connection-traffic-table-index: 10
→ Rx service-category: CBR (Constant Bit Rate)
→ Rx pcr-clp01: 4000
Rx scr-clp01: none
Rx mcr-clp01: none
Rx      cdvt: 1024 (from default for interface)
Rx       mbs: none
→ Tx connection-traffic-table-index: 10
→ Tx service-category: CBR (Constant Bit Rate)
→ Tx pcr-clp01: 4000
Tx scr-clp01: none
Tx mcr-clp01: none
Tx      cdvt: none
Tx       mbs: none
```

Configuring a Soft PVC for Structured CES

In a soft PVC, as well as a hard PVC, you configure both ends of the CES circuit. However, a soft PVC typically involves CES modules at opposite edges of an ATM network, so a soft PVC can be set up between any two CES modules anywhere in your network.

The destination address of a soft PVC can point to either of the following:

- Any ATM switch router external ATM port in the network
- A port in any other CES module in the network

For example, to set up a soft PVC involving a local node and a destination node at the opposite edge of the network, you need to determine the CES-IWF ATM address of the port in the destination node to complete a soft PVC setup.

To obtain the destination address for an already configured port in a CES module, log into the remote ATM switch router containing that module. Then use the **show ces address** command to display all the CES-IWF ATM addresses currently configured for that node.

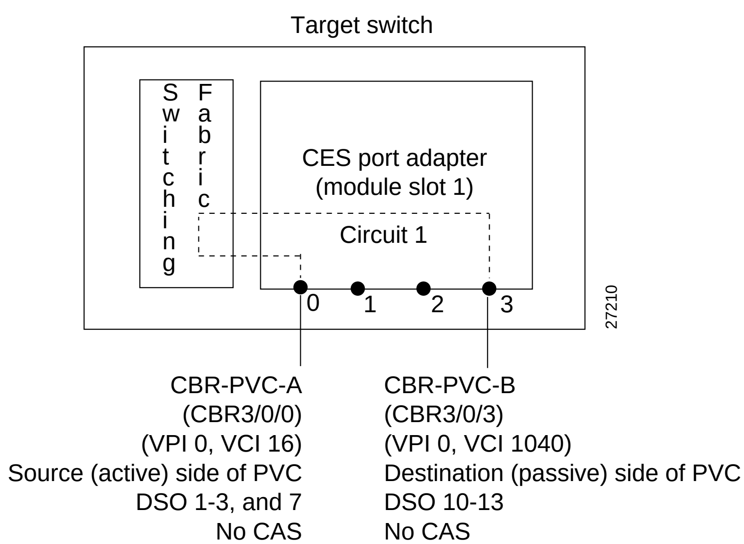
> **Note** Typically you will configure a soft PVC between CES modules anywhere in your network. For simplicity, this example and the accompanying procedure describe how to create a soft PVC between modules in the same ATM switch router chassis.

This section describes how to configure a soft PVC for structured service based on the following assumptions:

- The source (active) side of the soft PVC is named CBR-PVC-A.
- The destination (passive) side of the soft PVC is named CBR-PVC-B.
- Four time slots (DS0 channels) are configured for the soft PVC, as follows:
  - For circuit CBR-PVC-A: DS0 channels 1 to 3 and 7 are used on port CBR 3/0/0.
  - For circuit CBR-PVC-B: DS0 channels 10 to 13 are used on port CBR 3/0/3.
- Channel associated signalling (CAS) is not enabled. For information about configuring a soft PVC with CAS, see the "Configuring a Soft PVC for Structured CES" section on page 18-29.
- CES AAL1 service is structured and the clock source is network-derived.
- CES framing is esf and the line code is b8zs.
- The status of the circuit will follow the status of the physical interface.

Figure 18-5 shows an example of a soft PVC configured for structured CES.

Figure 18-5 Soft PVC Configured for Structured CES



Configuring a soft PVC for structured CES is a two-phase process:

- Phase 1—Configuring the Destination (Passive) Side of a Soft PVC
- Phase 2—Configuring the Source (Active) Side of a Soft PVC

Phase 1—Configuring the Destination (Passive) Side of a Soft PVC

To configure a destination (passive) side of a soft PVC for structured CES, follow these steps, beginning in privileged EXEC mode:

	Command	Purpose
Step 1	Switch# **show ces status**	Displays information about the current CBR interfaces. Use this command to choose the destination port.
Step 2	Switch# **configure terminal** Switch(config)#	At the privileged EXEC prompt, enters global configuration mode.
Step 3	Switch(config)# **interface cbr** *card*/*subcard*/*port* Switch(config-if)#	Selects the physical interface to be configured.
Step 4	Switch(config-if)# **shutdown**	Disables the interface.
Step 5	Switch(config-if)# **ces aal1 service** {**structured** | **unstructured**}	Configures the CES interface AAL1 service type.
Step 6	Switch(config-if)# **ces dsx1 clock source** {**loop-timed** | **network-derived**}	Configures the clock source.
Step 7	Switch(config-if)# **ces dsx1 framing** {**sf | esf**}	Configures the CES T1 framing type. The default is **esf**.
	Switch(config-if)# **ces dsx1 framing** {**e1_crc_mfCAS_lt** | **e1_crc_mf_lt** | **e1_lt** | **e1_mfCAS_lt**}	Configures the CES E1 framing type. For CES E1, the default is **e1_lt**.
Step 8	Switch(config-if)# **ces dsx1 linecode** {**ami** | **b8zs**}	Configures the CES T1 line code type. The default is **b8zs**.
	Switch(config-if)# **ces dsx1 linecode** {**ami** | **hdb3**}	Configures the CES E1 line code type. The default is **hdb3**.
Step 9	Switch(config-if)# **ces circuit** *circuit-id* **timeslots** *number*	Configures the following CES connection attributes for the circuit: • Circuit id number. – For CES T1 structured service, use 1 through 24. – For CES E1 structured service, use 1 through 31. • Time slots for the circuit for structured service only. – For CES T1, the range is 1 through 24. – For CES E1, the range is 1 through 31.
Step 10	Switch(config-if)# **ces circuit** *circuit-id* **circuit-name** *name*	Configures the CES interface circuit name.

	Command	Purpose
Step 11	Switch(config-if)# **ces pvc** *circuit-id* **passive follow-ifstate**	Configures the destination (passive) port circuit status to follow the status of the physical interface. The default circuit setting ignores the status of the physical interface.
Step 12	Switch(config-if)# **no shutdown**	Reenables the interface.

Example

The following example shows how to configure the destination (passive) side of a soft PVC for structured T1 CES, as shown in Figure 18-5:

```
CESwitch# show ces status

   Interface      IF      Admin        Port  Channels in
     Name       Status   Status        Type      use
------------- -------- --------- ----------- -----------
     CBR3/0/0       UP        UP          T1
     CBR3/0/1       UP        UP          T1
     CBR3/0/2       UP        UP          T1
     CBR3/0/3       UP        UP          T1

CESwitch# configure terminal
CESwitch(config)# interface cbr 3/0/3
CESwitch(config-if)# shutdown
CESwitch(config-if)# ces aal1 service structured
CESwitch(config-if)# ces dsx1 clock source network-derived
CESwitch(config-if)# ces dsx1 framing esf
CESwitch(config-if)# ces dsx1 linecode b8zs
CESwitch(config-if)# ces circuit 1 timeslots 10-13
CESwitch(config-if)# ces circuit 1 circuit-name CBR-PVC-B
CESwitch(config-if)# no shutdown
CESwitch(config-if)# ces pvc 1 passive follow-ifstate
```

Phase 2—Configuring the Source (Active) Side of a Soft PVC

To configure the source (active) side of a soft PVC for structured CES, follow these steps, beginning in privileged EXEC mode:

	Command	Purpose
Step 1	Switch# **show ces address**	Shows the CES address for the destination end of the circuit. Use this command to retrieve the VPI/VCI of the destination port.
Step 2	Switch# **configure terminal** Switch(config)#	At the privileged EXEC prompt, enters global configuration mode.
Step 3	Switch(config)# **interface cbr** *card/subcard/port* Switch(config-if)#	Selects the physical interface to be configured.
Step 4	Switch(config-if)# **shutdown**	Disables the interface.

	Command	Purpose
Step 5	Switch(config-if)# **ces circuit** *circuit-id* **timeslots** *number*	Configures the following CES connection attributes for the circuit: • Circuit id number. – For CES T1 structured service, use 1 through 24. – For CES E1 structured service, use 1 through 31. **Note** The 0 circuit identifier is reserved for unstructured service. • Time slots for the circuit for structured service only. – For CES T1, the range is 1 through 24. – For CES E1, the range is 1 through 31.
Step 6	Switch(config-if)# **ces circuit** *circuit-id* **circuit-name** *name*	Configures the CES interface circuit name.
Step 7	Switch(config-if)# **ces pvc** *circuit-id* **dest-address** *remote_atm_address* **vpi** *vpi* **vci** *vci* [**follow-ifstate**]	Configures the soft PVC to the destination CES-IWF ATM addresses and VPI/VCI of the circuit. Use the VPI/VCI of the destination port that was retrieved in Step 1. The **follow-ifstate** keyword configures the source (active) port circuit status to follow the status of the physical interface. The default circuit setting ignores the status of the physical interface.
Step 8	Switch(config-if)# **no shutdown**	Reenables the interface.

Example

The following example shows how to configure the source (active) side of a soft PVC for structured CES, as shown in Figure 18-5:

```
CESwitch# show ces address

CES-IWF ATM Address(es):
47.0091.8100.0000.00e0.4fac.b401.4000.0c81.803c.10  CBR3/0/3:1  vpi 0 vci 3088

CESwitch# configure terminal
CESwitch(config)# interface cbr 3/0/0
CESwitch(config-if)# shutdown
CESwitch(config-if)# ces circuit 1 timeslots 1-3, 7
CESwitch(config-if)# ces circuit 1 circuit-name CBR-PVC-A
CESwitch(config-if)# ces pvc 1 dest-address 47.0091.8100.0000.0060.5c71.1f01.4000.0c80.1034.10 vpi 0 vci 16
follow-ifstate
CESwitch(config-if)# no shutdown
```

If you do not specify the circuit name and logical name parameters in the command line, the system automatically assigns a unique default name in the form CBR*x/y/z:#* for the circuit being configured. For example, the default name for this particular circuit is CBR3/0/0:1. For structured circuit emulation services, the circuit number sequence always begins at 1 for each port in a CES module.

Verifying a Soft PVC for Structured CES

To verify the soft PVC configured with structured CES, use the following EXEC commands:

Command	Purpose
show ces circuit	Shows the configuration information for the soft PVC.
show ces circuit interface cbr *card*/*subcard*/*port circuit-id*	Shows the detailed interface configuration information for the soft PVC.

Examples

The following example shows the details of the CES circuit (shown in Figure 18-5), using the **show ces circuit** command:

```
CESwitch# show ces circuit

Interface  Circuit  Circuit-Type     X-interface   X-vpi   X-vci Status
 CBR3/0/0    1       Active SoftVC   ATM-P3/0/3      0      3088  UP
 CBR3/0/3    1       Passive SoftVC  ATM-P3/0/3      0        16  UP
```

The following example shows the interface details for the source port (CBR 3/0/0) (shown in Figure 18-5), using the **show ces circuit interface cbr** command:

```
CESwitch# show ces circuit interface cbr 3/0/0 1
Circuit: Name CBR-PVC-A, Circuit-state ADMIN_UP / oper-state UP
Interface CBR3/0/0, Circuit_id 1, Port-Type T1, Port-State UP
Port Clocking network-derived, aal1 Clocking Method CESIWF_AAL1_CLOCK_SYNC
Channel in use on this port: 1-3,7
Channels used by this circuit: 1-3,7
Cell-Rate: 698, Bit-Rate 256000
cas OFF, cell_header 0x100 (vci = 16)
Configured CDV 2000 usecs, Measured CDV unavailable
De-jitter: UnderFlow unavailable, OverFlow unavailable
ErrTolerance 8, idleCircuitdetect OFF, onHookIdleCode 0x0
state: VcActive, maxQueueDepth       45, startDequeueDepth       28
Partial Fill:       47, Structured Data Transfer 98
Active SoftVC
Src: atm addr 47.0091.8100.0000.00e0.4fac.b401.4000.0c81.8030.10 vpi 0, vci 16
Dst: atm addr 47.0091.8100.0000.00e0.4fac.b401.4000.0c81.803c.10
```

The following example shows the interface details for the destination port (CBR 3/0/3) (shown in Figure 18-5), using the **show ces circuit interface cbr** command:

```
CESwitch# show ces circuit interface cbr 3/0/3 1
Circuit: Name CBR-PVC-B, Circuit-state ADMIN_UP / oper-state UP
Interface CBR3/0/3, Circuit_id 1, Port-Type T1, Port-State UP
Port Clocking network-derived, aal1 Clocking Method CESIWF_AAL1_CLOCK_SYNC
Channel in use on this port: 10-13
Channels used by this circuit: 10-13
Cell-Rate: 698, Bit-Rate 256000
cas OFF, cell_header 0xC100 (vci = 3088)
Configured CDV 2000 usecs, Measured CDV unavailable
De-jitter: UnderFlow unavailable, OverFlow unavailable
ErrTolerance 8, idleCircuitdetect OFF, onHookIdleCode 0x0
state: VcActive, maxQueueDepth       45, startDequeueDepth       28
Partial Fill:       47, Structured Data Transfer 98
Passive SoftVC
Src: atm addr 47.0091.8100.0000.00e0.4fac.b401.4000.0c81.803c.10 vpi 0, vci 3088
Dst: atm addr 47.0091.8100.0000.00e0.4fac.b401.4000.0c81.8030.00
```

Configuring a Soft PVC for Structured CES with CAS Enabled

Since the CES T1/E1 port adapter emulates CBR services over ATM networks, it must be able to support channel-associated signalling (CAS) information that is introduced into structured CES circuits by PBXs and TDMs. An optional CAS feature for the CES T1/E1 port adapter meets this requirement.

CAS information carried in a CBR bit stream can be configured with a CES module, as follows:

- The optional CAS feature is not enabled (the default state). For information about configuring a soft PVC for structured CES without CAS enabled, see the Configuring a Soft PVC for Structured CES, page 18-29.
- The optional CAS feature is enabled, but without the optional, Cisco-proprietary on-hook detection feature enabled. This option is described in the following procedure.
- Both the optional CAS and on-hook detection features are enabled. For information about configuring a soft permanent virtual channel (soft PVC) for structured CES with both CAS and on-hook detection enabled, see the "Configuring a Soft PVC for Structured CES with CAS and On-Hook Detection Enabled" section on page 18-39.

Note For a detailed description of CAS operation and the on-hook detection feature, refer to the circuit emulation services topic in the *Guide to ATM Technology*.

This section describes how to configure a soft PVC for structured CES with channel-associated signalling (CAS) enabled.

Note Typically you will configure a soft PVC between CES modules anywhere in your network. For simplicity, this example and the accompanying procedure describe how to create a soft PVC between modules in the same ATM switch router chassis.

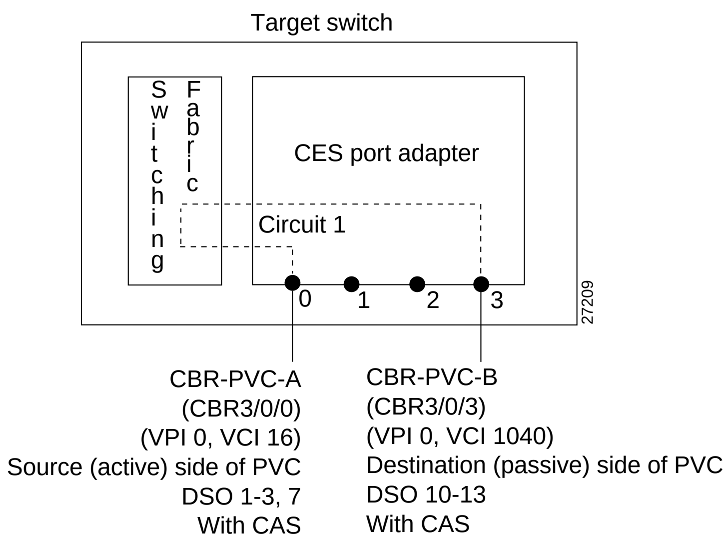
The following procedure is based on the following assumptions:

- The source (active) side of the soft PVC (CBR-PVC-A) remains as previously configured.
- The destination (passive) side of the soft PVC (CBR-PVC-B) remains as previously configured.

- Four time slots (DS0 channels) remain as previously configured for the soft PVC:
  - For circuit CBR-PVC-A: DS0 channels 1 to 3 and 7 are used on port CBR3/0/0.
  - For circuit CBR-PVC-B: DS0 channels 10 to 13 are used on port CBR3/0/3.
- CAS is enabled for the circuit.
- The signalling mode for the T1 CBR ports is set to "robbedbit."

Figure 18-6 shows a soft PVC configured for structured CES with CAS enabled.

Figure 18-6 Soft PVC Configured for Structured CES with CAS Enabled



To configure a soft PVC for structured CES with CAS enabled, follow these steps, beginning in privileged EXEC mode:

	Command	Purpose
Step 1	Switch# **show ces status**	Displays information about the current CBR interfaces. Use this command to choose the ports to be configured with CAS enabled.
Step 2	Switch# **configure terminal** Switch(config)#	At the privileged EXEC mode prompt, enters global configuration mode.
Step 3	Switch(config)# **interface cbr** *card*/*subcard*/*port* Switch(config-if)#	Selects the source interface to be configured.
Step 4	Switch(config-if)# **no shutdown**	Reenables the interface.
Step 5	Switch(config-if)# **ces dsx1 signalmode robbedbit**	Configures the signal mode to robbedbit (CES T1 only).
Step 6	Switch(config-if)# **ces circuit** *circuit-id* **cas**	Enables channel-associated signalling.
Step 7	Switch(config-if)# **exit** Switch(config)#	Returns to global configuration mode.

	Command	Purpose
Step 8	Switch(config)# **interface cbr** *card*/*subcard*/*port* Switch(config-if)#	Selects the destination interface to be configured.
Step 9	Switch(config-if)# **shutdown**	Disables the interface.
Step 10	Switch(config-if)# **ces dsx1 signalmode robbedbit**	Configures the signal mode to robbedbit (CES T1 only).
Step 11	Switch(config-if)# **ces circuit** *circuit-id* **cas**	Enables channel-associated signalling.
Step 12	Switch(config-if)# **no shutdown**	Reenables the interface.

Example

The following example shows how to enable channel-associated signalling (CAS) on a soft PVC (see Figure 18-6):

```
CESwitch# show ces status

   Interface      IF      Admin        Port  Channels in
     Name       Status   Status        Type      use
------------- -------- --------- ----------- -----------
     CBR3/0/0       UP        UP          T1   1-3,7
     CBR3/0/1     DOWN        UP          T1
     CBR3/0/2     DOWN        UP          T1
     CBR3/0/3       UP        UP          T1   10-13

CESwitch# configure terminal
CESwitch(config)# interface cbr 3/0/0
CESwitch(config-if)# shutdown
CESwitch(config-if)# ces dsx1 signalmode robbedbit
CESwitch(config-if)# ces circuit 1 cas
CESwitch(config-if)# no shutdown
CESwitch(config-if)# exit
CESwitch(config)# interface cbr 3/0/3
CESwitch(config-if)# shutdown
CESwitch(config-if)# ces dsx1 signalmode robbedbit
CESwitch(config-if)# ces circuit 1 cas
CESwitch(config-if)# no shutdown
```

Verifying a Soft PVC for Structured CES with CAS Enabled

To verify the soft PVC with structured CES and CAS enabled, use the following EXEC commands:

Command	Purpose
show ces circuit	Shows the configuration information for the soft PVC.
show ces circuit interface cbr *card*/*subcard*/*port circuit-id*	Shows the detailed interface configuration information for the soft PVC.

Examples

The following example displays the details of the CES circuit (shown in Figure 18-6), using the **show ces circuit** command at the privileged EXEC mode prompt:

```
CESwitch# show ces circuit

Interface  Circuit  Circuit-Type     X-interface   X-vpi   X-vci Status
 CBR3/0/0    0       Active SoftVC   ATM-P3/0/3      0        16 UP
 CBR3/0/1    0       Passive SoftVC  ATM-P3/0/3      0      1040 UP
```

The following example displays the CAS status for the source port CBR 3/0/0 (shown in Figure 18-6):

```
CESwitch# show ces circuit interface cbr 3/0/0 1
Circuit: Name CBR-PVC-A, Circuit-state ADMIN_UP / oper-state UP
Interface CBR3/0/0, Circuit_id 1, Port-Type T1, Port-State UP
Port Clocking network-derived, aal1 Clocking Method CESIWF_AAL1_CLOCK_SYNC
Channel in use on this port: 1-3,7
Channels used by this circuit: 1-3,7
Cell-Rate: 698, Bit-Rate 256000
➔ cas ON, cell_header 0x100 (vci = 16)
Configured CDV 2000 usecs, Measured CDV unavailable
De-jitter: UnderFlow unavailable, OverFlow unavailable
ErrTolerance 8, idleCircuitdetect OFF, onHookIdleCode 0x0
state: VcActive, maxQueueDepth       45, startDequeueDepth       28
Partial Fill:       47, Structured Data Transfer 98
Active SoftVC
Src: atm addr 47.0091.8100.0000.00e0.4fac.b401.4000.0c81.8030.10 vpi 0, vci 16
Dst: atm addr 47.0091.8100.0000.00e0.4fac.b401.4000.0c81.803c.10
```

The following example displays the CAS status for the destination port CBR 3/0/3 (shown in Figure 18-6):

```
CESwitch# show ces circuit interface cbr 3/0/3 1
Circuit: Name CBR-PVC-B, Circuit-state ADMIN_UP / oper-state UP
Interface CBR3/0/3, Circuit_id 1, Port-Type T1, Port-State UP
Port Clocking network-derived, aal1 Clocking Method CESIWF_AAL1_CLOCK_SYNC
Channel in use on this port: 10-13
Channels used by this circuit: 10-13
Cell-Rate: 698, Bit-Rate 256000
➔ cas ON, cell_header 0xC100 (vci = 3088)
Configured CDV 2000 usecs, Measured CDV unavailable
De-jitter: UnderFlow unavailable, OverFlow unavailable
ErrTolerance 8, idleCircuitdetect OFF, onHookIdleCode 0x0
state: VcActive, maxQueueDepth       45, startDequeueDepth       28
Partial Fill:       47, Structured Data Transfer 98
Passive SoftVC
Src: atm addr 47.0091.8100.0000.00e0.4fac.b401.4000.0c81.803c.10 vpi 0, vci 3088
Dst: atm addr 47.0091.8100.0000.00e0.4fac.b401.4000.0c81.8030.00
```

Configuring a Soft PVC for Structured CES with CAS and On-Hook Detection Enabled

This section outlines the additional steps that you must take to activate the on-hook detection (bandwidth-release) feature in a 1 x 64 structured CES circuit.

To configure a soft PVC for structured CES with CAS and on-hook detection enabled, follow these steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **interface cbr** *card*/*subcard*/*port* Switch(config-if)#	Selects the physical interface to be configured.
Step 2	Switch(config-if)# **shutdown**	Disables the interface.
Step 3	Switch(config-if)# **ces circuit** *circuit-id* [**cas**] [**on-hook-detect** *pattern*]	Configures channel-associated signalling and on-hook detection on the CES circuit.
Step 4	Switch(config-if)# **no shutdown**	Reenables the interface.

Example

The following example shows how to configure on-hook detection on the soft PVC with structured CES and CAS enabled in the "Configuring a Soft PVC for Structured CES with CAS Enabled" section on page 18-35 (shown in Figure 18-6):

```
CESwitch(config)# interface cbr 3/0/0
CESwitch(config-if)# shutdown
CESwitch(config-if)# ces circuit 1 cas on-hook-detect 2
CESwitch(config-if)# no shutdown
```

Note The four ABCD bits in the CAS mechanism are device-specific, depending on the manufacturer of the voice/video telephony device that generates the CBR traffic. The ABCD bits of the CAS mechanism are user-configurable.

Verifying a Soft PVC for Structured CES with CAS and On-Hook Detection Enabled

To show the on-hook detection configuration of a soft PVC configured with structured CES and CAS enabled, use the following EXEC command:

Command	Purpose
show ces circuit interface cbr *card*/*subcard*/*port circuit-id*	Shows the detailed interface configuration information for the soft PVC.

Example

The following example shows the soft PVC with CAS and on-hook detection enabled as hexadecimal number 2 (shown in Figure 18-6):

```
CESwitch# show ces circuit interface cbr 3/0/3 1
Circuit: Name CBR-PVC-B, Circuit-state ADMIN_UP / oper-state UP
Interface CBR3/0/3, Circuit_id 1, Port-Type T1, Port-State UP
Port Clocking network-derived, aal1 Clocking Method CESIWF_AAL1_CLOCK_SYNC
Channel in use on this port: 10-13
Channels used by this circuit: 10-13
Cell-Rate: 698, Bit-Rate 256000
➔ cas ON, cell_header 0xC100 (vci = 3088)
Configured CDV 2000 usecs, Measured CDV unavailable
De-jitter: UnderFlow unavailable, OverFlow unavailable
➔ ErrTolerance 8, idleCircuitdetect OFF, onHookIdleCode 0x2
state: VcActive, maxQueueDepth       45, startDequeueDepth       28
Partial Fill:       47, Structured Data Transfer 98
Passive SoftVC
Src: atm addr 47.0091.8100.0000.00e0.4fac.b401.4000.0c81.803c.10 vpi 0, vci 3088
Dst: atm addr 47.0091.8100.0000.00e0.4fac.b401.4000.0c81.8030.00
```

Creating Multiple Structured Soft PVCs on the Same CES Port

This section describes how to create more than one structured soft permanent virtual channel (soft PVC) on the same CES T1/E1 port. Figure 18-7 shows how you can configure multiple CES circuits on a single T1/E1 port.

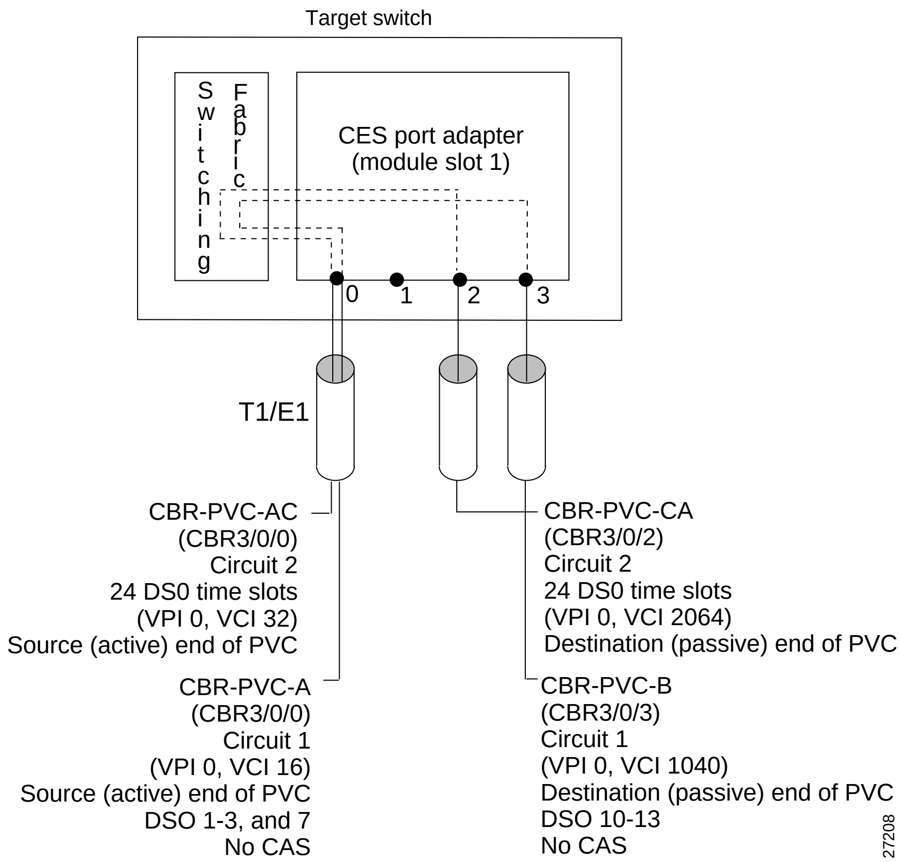
Note Typically you will configure a soft PVC between CES modules anywhere in your network. For simplicity, this example and the accompanying procedure describe how to create a soft PVC between modules in the same ATM switch router chassis.

Assume that certain configuration information has already been established for a soft PVC (see Figure 18-6) and that you are to create an additional soft PVC involving the same CES module.

The following assumptions apply to creating multiple soft PVCs on the same T1/E1 port (see Figure 18-7):

- The source (active) side of a soft PVC named CBR-PVC-A is already created on port CBR 3/0/0.
- The destination (passive) side of a soft PVC named CBR-PVC-B is already created on port CBR 3/0/3.
- A new source (active) side of a soft PVC named CBR-PVC-AC will be created on port CBR 3/0/0 of the CES module, thereby creating a multiple CES circuit on this particular port.
- A new destination (passive) side of a soft PVC named CBR-PVC-CA will be created on port CBR 3/0/2 of the CES module.
- The CES AAL1 service is structured and the clock source is network-derived.
- The CES framing is esf and the line code is b8zs.

Figure 18-7 Configuring Multiple Structured Soft PVCs on the Same CES T1/E1 Port



Configuring multiple soft PVCs for structured CES is a two-phase process:

- Phase 1—Configuring the Destination (Passive) Side of Multiple Soft PVCs
- Phase 2—Configuring the Source (Active) Side of Multiple Soft PVCs

Phase 1—Configuring the Destination (Passive) Side of Multiple Soft PVCs

To configure multiple soft PVCs on the destination (passive) side of the same port, follow these steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **interface cbr** *card*/*subcard*/*port* Switch(config-if)#	Selects the physical interface to be configured.
Step 2	Switch(config-if)# **shutdown**	Disables the interface.
Step 3	Switch(config-if)# **ces aal1 service** {**structured** | **unstructured**}	Configures the CES interface AAL1 service type.
Step 4	Switch(config-if)# **ces dsx1 clock source** {**loop-timed** | **network-derived**}	Configures the clock source.

	Command	Purpose
Step 5	Switch(config-if)# **ces dsx1 framing** {**sf** | **esf**}	Configures the CES T1 framing type. The default is **esf**.
	Switch(config-if)# **ces dsx1 framing** {**e1_crc_mfCAS_lt** | **e1_crc_mf_lt** | **e1_lt** | **e1_mfCAS_lt**}	Configures the CES E1 framing type. The default is **e1_lt**.
Step 6	Switch(config-if)# **ces dsx1 linecode** {**ami** | **b8zs**}	Configures the CES T1 line code type. The default is **b8zs**.
	Switch(config-if)# **ces dsx1 linecode** {**ami** | **hdb3**}	Configures the CES E1 line code type. The default is **hdb3**.
Step 7	Switch(config-if)# **ces circuit** *circuit-id* [**circuit-name** *name*] [**timeslots** *number*]	Configures the following CES connection attributes for the circuit: • Circuit id number. – For CES T1 structured service, use 1 through 24. – For CES E1 structured service, use 1 through 31. **Note** The 0 circuit identifier is reserved for unstructured service. • Configures the CES interface circuit name. • Configures the time slots for the circuit for structured service only. – For CES T1, the range is 1 through 24. – For CES E1, the range is 1 through 31.
Step 8	Switch(config-if)# **ces pvc** *circuit-id* **passive follow-ifstate**	Configures the destination (passive) port circuit status to follow the status of the physical interface. The default circuit setting ignores the status of the physical interface.
Step 9	Switch(config-if)# **no shutdown**	Reenables the interface.

Example

The following example shows how to configure multiple soft PVCs on the destination (passive) side of the same port (shown in Figure 18-7):

```
CESwitch(config)# interface cbr 3/0/2
CESwitch(config-if)# shutdown
CESwitch(config-if)# ces aal1 service structured
CESwitch(config-if)# ces dsx1 clock source network-derived
CESwitch(config-if)# ces dsx1 framing esf
CESwitch(config-if)# ces dsx1 linecode b8zs
CESwitch(config-if)# ces circuit 2 timeslots 24 circuit-name CBR-PVC-CA
CESwitch(config-if)# no shutdown
```

Note If you do not specify the circuit name and logical name parameters in the command line, the system automatically assigns a unique default name in the form CBR*x/y/z:#* for the circuit being configured. For example, the default name for this particular circuit is CBR3/0/2:1. For structured circuit emulation services, the circuit number sequence always begins at 1 for each port in a CES module.

Phase 2—Configuring the Source (Active) Side of Multiple Soft PVCs

To configure multiple soft PVCs on the source (active) side of the same port, follow these steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **interface cbr** *card*/*subcard*/*port* Switch(config-if)#	Selects the source interface to be configured.
Step 2	Switch(config-if)# **shutdown**	Disables the interface.
Step 3	Switch(config-if)# **ces circuit** *circuit-id* [**circuit-name** *name*] [**timeslots** *number*]	Configures the following CES connection attributes for the circuit: • Circuit id number. – For CES T1 structured service, use 1 through 24. – For CES E1 structured service, use 1 through 31. • Configures the CES interface circuit name. • Configures the time slots for the circuit for structured service only. – For CES T1, the range is 1 through 24. – For CES E1, the range is 1 through 31.
Step 4	Switch(config-if)# **no shutdown**	Reenables the interface.
Step 5	Switch(config-if)# **end** Switch#	Exits interface configuration mode.
Step 6	Switch# **show ces address**	Shows the CES address for the destination end of the circuit. Use this command to retrieve the VPI/VCI of the destination port.
Step 7	Switch# **configure terminal** Switch(config)#	At the privileged EXEC prompt, enters configuration mode.
Step 8	Switch(config)# **interface cbr** *card*/*subcard*/*port* Switch(config-if)#	Selects the destination interface to be configured.
Step 9	Switch(config-if)# **shutdown**	Disables the interface.

	Command	Purpose
Step 10	Switch(config-if)# **ces pvc** *circuit-id* **dest-address** *remote_atm_address* **vpi** *vpi* **vci** *vci* [**follow-ifstate**]	Configures the soft PVC to the destination CES-IWF ATM addresses and VPI/VCI of the circuit. Use the VPI/VCI of the destination port that was retrieved in Step 4.
Step 11	Switch(config-if)# **no shutdown**	Reenables the interface.

Example

The following example shows how to configure multiple soft PVCs on the source (active) side of the same port (shown in Figure 18-7):

```
CESwitch(config)# interface cbr 3/0/0
CESwitch(config-if)# shutdown
CESwitch(config-if)# ces circuit 2 timeslots 24
CESwitch(config-if)# ces circuit 2 circuit-name CBR-PVC-AC
CESwitch(config-if)# no shutdown
CESwitch(config-if)# end
CESwitch# show ces address

CES-IWF ATM Address(es):
47.0091.8100.0000.0060.5c71.1f01.4000.0c80.1030.10  CBR-PVC-A
47.0091.8100.0000.0060.5c71.1f01.4000.0c80.1030.20  CBR-PVC-AC
47.0091.8100.0000.0060.5c71.1f01.4000.0c80.1034.10  CBR-PVC-B
47.0091.8100.0000.0060.5c71.1f01.4000.0c80.1038.10  CBR-PVC-CA

CESwitch# configure terminal
CESwitch(config)# interface cbr 3/0/2
CESwitch(config-if)# shutdown
CESwitch(config-if)# ces pvc 2 dest-address
 47.0091.8100.0000.0060.5c71.1f01.4000.0c80.1038.10 vpi 0 vci 2064
CESwitch(config-if)# no shutdown
```

If you do not specify the circuit name and logical name parameters in the command line, the system automatically assigns a unique default name in the form CBR*x/y/z:#* for the circuit being configured. For example, the default name for this particular circuit is CBR3/0/2:1. For structured circuit emulation services, the circuit number sequence always begins at 1 for each port in a CES module.

Verifying the Creation of Multiple Structured Soft PVCs on the Same CES Port

To verify multiple structured soft PVCs with CAS enabled, use the following EXEC commands:

Command	Purpose
show ces circuit	Shows the configuration information for the soft PVC.
show ces address	Shows the CES address for the destination end of the circuit.
show ces circuit interface cbr *card/subcard/port circuit-id*	Shows the detailed interface configuration information for the soft PVC.

Examples

The following example displays the circuit details for the soft PVCs that you created in the previous procedure (shown in Figure 18-7) using the **show ces circuit** command in privileged EXEC mode:

```
CESwitch# show ces circuit
Interface  Circuit  Circuit-Type     X-interface   X-vpi   X-vci Status
 CBR3/0/0    1       Active SoftVC   ATM-P3/0/3      0      3088  UP
 CBR3/0/0    2       Active SoftVC   ATM-P3/0/3      0      2080  UP
 CBR3/0/2    2       Passive SoftVC  ATM-P3/0/3      0        32  UP
 CBR3/0/3    1       Passive SoftVC  ATM-P3/0/3      0        16  UP
```

The following example displays the CES-IWF addresses of the soft PVCs that you configured (shown in Figure 18-7), using the **show ces address** command in privileged EXEC mode:

```
CESwitch# show ces address

CES-IWF ATM Address(es):
47.0091.8100.0000.00e0.4fac.b401.4000.0c81.8030.10  CBR3/0/0:1  vpi 0 vci 16
47.0091.8100.0000.00e0.4fac.b401.4000.0c81.8030.20  CBR3/0/0:2  vpi 0 vci 32
47.0091.8100.0000.00e0.4fac.b401.4000.0c81.8038.20  CBR3/0/2:2  vpi 0 vci 2080
47.0091.8100.0000.00e0.4fac.b401.4000.0c81.803c.10  CBR3/0/3:1  vpi 0 vci 3088
```

The following example displays the interface details for the new circuit 2 soft PVC that you set up on port CBR 3/0/0 (shown in Figure 18-7), using the **show ces circuit interface cbr** command:

```
CESwitch# show ces circuit interface cbr 3/0/0 2
Circuit: Name CBR-PVC-AC, Circuit-state ADMIN_UP / oper-state UP
Interface CBR3/0/0, Circuit_id 2, Port-Type T1, Port-State UP
Port Clocking network-derived, aal1 Clocking Method CESIWF_AAL1_CLOCK_SYNC
Channel in use on this port: 24
Channels used by this circuit: 24
Cell-Rate: 172, Bit-Rate 64000
cas OFF, cell_header 0x200 (vci = 32)
Configured CDV 2000 usecs, Measured CDV unavailable
De-jitter: UnderFlow unavailable, OverFlow unavailable
ErrTolerance 8, idleCircuitdetect OFF, onHookIdleCode 0x0
state: VcActive, maxQueueDepth       81, startDequeueDepth       64
Partial Fill:       47, Structured Data Transfer 1
Active SoftVC
Src: atm addr 47.0091.8100.0000.00e0.4fac.b401.4000.0c81.8030.20 vpi 0, vci 32
Dst: atm addr 47.0091.8100.0000.00e0.4fac.b401.4000.0c81.8038.20
```

The following example displays the interface details for the new circuit 1 soft PVC that you configured on port CBR3/0/2 (shown in Figure 18-7), using the **show ces circuit interface cbr** command:

```
CESwitch# show ces circuit interface cbr 3/0/2 2
Circuit: Name CBR-PVC-CA, Circuit-state ADMIN_UP / oper-state UP
Interface CBR3/0/2, Circuit_id 2, Port-Type T1, Port-State UP
Port Clocking network-derived, aal1 Clocking Method CESIWF_AAL1_CLOCK_SYNC
Channel in use on this port: 24
Channels used by this circuit: 24
Cell-Rate: 172, Bit-Rate 64000
cas OFF, cell_header 0x8200 (vci = 2080)
Configured CDV 2000 usecs, Measured CDV unavailable
De-jitter: UnderFlow unavailable, OverFlow unavailable
ErrTolerance 8, idleCircuitdetect OFF, onHookIdleCode 0x0
state: VcActive, maxQueueDepth       81, startDequeueDepth       64
Partial Fill:       47, Structured Data Transfer 1
Passive SoftVC
Src: atm addr 47.0091.8100.0000.00e0.4fac.b401.4000.0c81.8038.20 vpi 0, vci 2080
Dst: atm addr 47.0091.8100.0000.00e0.4fac.b401.4000.0c81.8030.00
```

Reconfiguring a Previously Established Circuit

Once you have configured a circuit, you cannot change the circuit's configuration while the circuit is up. You must first bring the interface down. Then you can change the circuit configuration. After entering these configuration changes, you must bring the interface back up. To change an enabled circuit's configuration, follow these steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **interface cbr** *card*/*subcard*/*port* Switch(config-if)#	Selects the physical interface to be configured.
Step 2	Switch(config-if)# **shutdown**	Disables the CES interface.
Step 3	For example, to specify the clock source as network-derived and to change the AAL1 clocking mode from adaptive to synchronous, enter: Switch(config-if)# **ces dsx1 clock source network-derived** Switch(config-if)# **ces aal1 clock synchronous**	Configures the clock source as network-derived and reconfigures the AAL1 clock mode to synchronous.
Step 4	Switch(config-if)# **no shutdown**	Enables the CES interface.
Step 5	Switch(config-if)# **end** Switch#	Exits interface configuration mode and returns to privileged EXEC mode.
Step 6	Switch# **show ces circuit interface cbr** *card*/*subcard*/*port circuit-id*	Shows detailed interface configuration information for the circuit. Use this command to verify your configuration changes.

Note The **no ces circuit** *circuit-id* **shutdown** command *deletes* the circuit. If you use this command, you must reenter all of the configuration information for the circuit. Do not use this command unless you intend to delete the circuit.

Examples

The following example disables interface cbr 3/0/0, specifies the clock source as network-derived, changes the AAL1 clocking method to synchronous, and reenables the interface.

```
CESwitch(config)# interface cbr 3/0/0
CESwitch(config-if)# shutdown
CESwitch(config-if)# ces dsx1 clock source network-derived
CESwitch(config-if)# ces aal1 clock synchronous
CESwitch(config-if)# no shutdown
```

The following example displays the changed configuration information for the circuit, using the **show ces circuit interface cbr** command:

```
CESwitch# show ces circuit interface cbr 3/0/0 0
Circuit: Name CBR-PVC-A, Circuit-state ADMIN_UP /
Interface CBR3/0/0, Circuit_id 0, Port-Type T1, Port-State UP
Port Clocking network-derived, aal1 Clocking Method CESIWF_AAL1_CLOCK_SYNC
Channel in use on this port: 1-24
Channels used by this circuit: 1-24
Cell-Rate: 4107, Bit-Rate 1544000
cas OFF, cell_header 0x100 (vci = 16)
cdv 2000 usecs, Measured cdv 350 usecs
ErrTolerance 8, idleCircuitdetect OFF, onHookIdleCode 0x0
state: VcAlarm, maxQueueDepth      879, startDequeueDepth      491
Partial Fill:       47, Structured Data Transfer 0
HardPVC
src: CBR3/0/0 vpi 0, vci 16
Dst: ATM0/1/3 vpi 0, vci 100
```

The output from this command verifies the following configuration information:

- The circuit named CBR-PVC-A is UP.
- The clock source is network-derived.
- The AAL1 clocking method is synchronous.

Deleting a Previously Established Circuit

This section describes how to delete a previously established circuit.

To delete a previously established circuit, follow these steps, beginning in privileged EXEC mode:

	Command	Purpose
Step 1	Switch# **show ces circuit**	Shows the configuration information for the circuit.
Step 2	Switch# **configure terminal** Switch(config)#	Enters global configuration mode from the terminal.
Step 3	Switch(config)# **interface cbr** *card*/*subcard*/*port* Switch(config-if)#	Selects the physical interface where the circuit is to be deleted.
Step 4	Switch(config-if)# **no ces circuit** *circuit-id*	Deletes the CES circuit.
Step 5	Switch(config-if)# **exit** Switch(config)#	Exits interface configuration mode and returns to global configuration mode.
Step 6	Switch(config)# **interface cbr** *card*/*subcard*/*port* Switch(config-if)#	Selects the other physical interface where the circuit is to be deleted.
Step 7	Switch(config-if)# **no ces circuit** *circuit-id*	Deletes the other end of CES circuit.

Example

The following example shows how to delete a previously established circuit:

```
CESwitch# show ces circuit

Interface  Circuit  Circuit-Type X-interface X-vpi X-vci Status
CBR3/0/0     0        HardPVC      ATM0/0      0     100   UP
CBR3/0/3     0        HardPVC      ATM0/0      0     101   UP

CESwitch# configure terminal
CESwitch(config)# interface cbr 3/0/0
CESwitch(config-if)# no ces circuit 0
CESwitch(config-if)# exit
CESwitch(config)# interface cbr 3/0/3
CESwitch(config-if)# no ces circuit 0
```

Verifying Deletion of a Previously Established Circuit

To verify the deletion of a previously configured circuit, use the following privileged EXEC commands:

Command	Purpose
show ces circuit	Shows the configuration information for the circuit.
show ces address	Shows the configuration information for any CES addresses.

Examples

The following example displays the configuration of any CES circuits:

```
CESwitch# show ces circuit
```

The absence of output verifies that all CES circuits are deleted.

The following example displays the configuration of any CES addresses:

```
CESwitch# show ces address

CES-IWF ATM Address(es):
```

The absence of output verifies that all CES circuits are deleted.

Configuring SGCP

The Simple Gateway Control Protocol (SGCP) controls voice-over-IP gateways by an external call control element (called a call-agent). This has been adapted to allow SGCP to control ATM switch router circuit emulation services (CES) circuits (called endpoints in SGCP). The resulting system (call-agents and gateways) allows for the call-agent to engage in common channel signalling (CCS) over a 64-Kbps CES circuit, governing the interconnection of bearer channels on the CES interface. In this system the ATM switch router acts as a voice-over-ATM gateway.

For overview information about configuring the SCGP feature, refer to the *Guide to ATM Technology*.

Operation

The network operator can globally enable or disable SGCP operation for the switch. By default, SGCP is disabled. When SGCP is enabled, the ATM switch router begins listening on the well-known User Datagram Protocol (UDP) port for SGCP packets. The endpoint ID in an SGCP packet identifies the CES circuit. The CES circuit endpoint can be used by SGCP if the following conditions exist:

- The parent CES interface is enabled, and the LineState field indicates NoAlarm (determined via the **show ces interface** command).
- The CES circuit is allocated a single time slot.
- The CES circuit is enabled (not shut).
- The CES circuit is not configured as an active soft PVC.
- The CES circuit is not configured as part of a hard PVC.

The following sections describe SGCP configuration tasks:

- Configuring SGCP on the Entire Switch, page 18-49
- Displaying SGCP, page 18-49
- Configuring CES Circuits for SGCP, page 18-50
- Displaying SGCP Endpoints, page 18-51
- Displaying SGCP Connections, page 18-52
- Configuring SGCP Request Handling, page 18-53
- Configuring Call-Agent Address, page 18-53
- Shutting Down SGCP, page 18-54

Configuring SGCP on the Entire Switch

To enable SGCP operations for the entire switch, use the following global configuration command:

Command	Purpose
sgcp	Enables or disables SGCP operations for the entire switch.

Example

The following example shows how to enable SGCP for the entire switch:

```
Switch(config)# sgcp
```

Displaying SGCP

To display SGCP configuration, operational state, and a summary of connection activity, use the following privileged EXEC command:

Command	Purpose
show sgcp	Displays the global SGCP configuration.

Example

The following example displays the SGCP configuration:

```
Switch# show sgcp

SGCP Admin State ACTIVE, Oper State ACTIVE
SGCP call-agent:none  , SGCP graceful-shutdown enabled? FALSE
SGCP request timeout 2000, SGCP request retries 6
74 CES endpoint connections created
74 CES endpoints in active connections
```

Configuring CES Circuits for SGCP

Any single time slot (64 Kbps) allocated to a circuit on a CES T1/E1 interface can be configured for SGCP with these restrictions:

- CES is not the active source end of a soft PVC.
- CES is not part of a hard PVC.

Note Configuration on the call-agent can restrict the range of circuits designated for signalling on a CES circuit interface.

When you configure a CES circuit for SGCP, signalling should be given the proper time slot. For T1 CES circuits, a time slot can be given a number from 1 to 24; for E1 CES, a number from 1 to 31.

Although no keyword identifies a CES circuit as allocatable by SGCP, there is normally a simple configuration rule to ensure that signalling allocates the proper time slot:

circuit *x* is allocated time slot *x*, 1<=*x*<=24 (or 31 for E1).

Note The endpoint specifier used by SGCP refers to the CES circuit ID (not the time slot). If a time slot is not allocated to a circuit, that time slot cannot be used by SGCP (or CES, either).

To configure SGCP operation on a CES circuit interface, follow these steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **interface cbr** *card*/*subcard*/*port* Switch(config-if)#	Selects the physical interface to be configured.
Step 2	Switch(config-if)# **ces aal1 service structured**	Configures the AAL1 service type.
Step 3	Switch(config-if)# **ces circuit** *circuit-id* **timeslot** *number*	Allocates a time slot number to the circuit identifier.

Example

The following example shows how to configure the CES port for structured CES with all time slots available for SGCP. CES circuit 16 is configured for common channel signalling and specified as a soft permanent virtual channel (soft PVC) to a circuit on the CES port adapter connected to the call-agent.

```
Switch(config)# interface cbr 1/1/2
Switch(config-if)# ces aal1 service structured
Switch(config-if)# ces circuit 1 timeslot 1
Switch(config-if)# ces circuit 2 timeslot 2
Switch(config-if)# ces circuit 3 timeslot 3
Switch(config-if)# ces circuit 4 timeslot 4
Switch(config-if)# ces circuit 5 timeslot 5
Switch(config-if)# ces circuit 6 timeslot 6
Switch(config-if)# ces circuit 7 timeslot 7
Switch(config-if)# ces circuit 8 timeslot 8
Switch(config-if)# ces circuit 9 timeslot 9
Switch(config-if)# ces circuit 10 timeslot 10
Switch(config-if)# ces circuit 11 timeslot 11
Switch(config-if)# ces circuit 12 timeslot 12
Switch(config-if)# ces circuit 13 timeslot 13
Switch(config-if)# ces circuit 14 timeslot 14
Switch(config-if)# ces circuit 15 timeslot 15
Switch(config-if)# ces circuit 16 timeslot 16
Switch(config-if)# ces pvc 16 dest-address
47.0091.8100.0000.0060.3e64.fd01.4000.0c80.1038.10 vpi 0 vci 2064
Switch(config-if)# ces circuit 17 timeslot 17
Switch(config-if)# ces circuit 18 timeslot 18
Switch(config-if)# ces circuit 19 timeslot 19
Switch(config-if)# ces circuit 20 timeslot 20
Switch(config-if)# ces circuit 21 timeslot 21
Switch(config-if)# ces circuit 22 timeslot 22
Switch(config-if)# ces circuit 23 timeslot 23
Switch(config-if)# ces circuit 24 timeslot 24
Switch(config-if)# end
```

Displaying SGCP Endpoints

SGCP endpoints are all the CES circuits that might be eligible for SGCP connections. To display SGCP endpoints, use the following EXEC command:

Command	Purpose
show sgcp endpoint [**interface cbr** *card/subcard/port* [*circuit-id*]]	Displays the SGCP endpoints.

Note SGCP cannot allocate a CES circuit to a connection if it is already part of a hard or soft PVC.

Example

The following example displays the possible SGCP endpoints on CES interface CBR 1/1/0:

```
Switch> show sgcp endpoint interface cbr 1/1/0

Endpt         Timeslots Conn State            Call ID
CBR1.1.0/1         1    no connection
CBR1.1.0/2         1    no connection
CBR1.1.0/3         1    no connection
CBR1.1.0/4         1    no connection
CBR1.1.0/5         1    no connection
CBR1.1.0/6         1    no connection
CBR1.1.0/7         1    no connection
CBR1.1.0/8         1    no connection
CBR1.1.0/9         1    no connection
CBR1.1.0/10        1    no connection
CBR1.1.0/11        1    active
CBR1.1.0/12        1    no connection
CBR1.1.0/14        1    active             1234abc
CBR1.1.0/15        1    active             2234abc
CBR1.1.0/16        1    active             3234abc
CBR1.1.0/17        1    active             4234abc
CBR1.1.0/18        1    active             5234abc
CBR1.1.0/19        1    active             6234abc
CBR1.1.0/20        1    active             7234abc
CBR1.1.0/21        1    active             8234abc
CBR1.1.0/22        1    active             9234abc
CBR1.1.0/23        1    active             a234abc
CBR1.1.0/24        1    active             b234abc
```

Displaying SGCP Connections

To display SGCP connections (either globally or per single interface), use the following EXEC command:

Command	Purpose
show sgcp connection [**interface cbr** *card*/*subcard*/*port*]	Displays the SGCP connections.

Example

The following example displays all SGCP connections created on the ATM switch router:

```
Switch> show sgcp connection

Conn Endpt          Soft VC State        Call Id
CBR0.0.0/1         Dest- active VC       d234ab
CBR0.0.0/2         Dest- active VC       12345bc
CBR0.0.0/3         Dest- active VC       1284ab
CBR0.0.0/4         Dest- active VC       9234abc
```

Configuring SGCP Request Handling

When the ATM switch router initiates an SGCP request (for example, to disconnect the circuit), default request timer and request retry values are in operation. To change the default value of SGCP requests, use the global configuration commands, as shown in the following table:

Command	Purpose
sgcp request timeout *msecs*	Configures the SGCP request timeout value.
sgcp request retries *number*	Configures the SGCP request retry value.

Examples

The following example shows how to change the request timeout to 2000 milliseconds:

```
Switch(config)# sgcp request timeout 2000
```

The following example shows how to change the request retry value to 5:

```
Switch(config)# sgcp request retries 5
```

Configuring Call-Agent Address

By default the SGCP call agents perform the following tasks:

- The ATM switch router sends a response to an SGCP request in a UDP packet with the destination address the same as the source address of the request UDP packet.
- To send a DeleteConnection request for a connection that exists, the ATM switch router specifies the destination address of the UDP packet as the source UDP address in the CreateConnection request.

To alter this behavior, and send responses and requests to a specific IP address and UDP port, use the following global configuration command:

Command	Purpose
sgcp call-agent *ip-address udp-port*	Configures the call-agent IP address and UDP port.

Note If the IP address is specified without the UDP port number, the well-known SGCP port 2427 is used.

Example

The following example shows how to set the call-agent with IP address 133.20.5.122 and UDP port 12000:

```
Switch(config)# sgcp call-agent 133.20.5.122 12000
```

Shutting Down SGCP

When SGCP is disabled with the **no sgcp** command, active SGCP connections are terminated; however DeleteConnection requests are not sent to the call-agent for these active connections.
To notify call-agent and perform a graceful SGCP shutdown, use the following global configuration command:

Command	Purpose
sgcp graceful-shutdown	Shuts down SGCP and notifies call-agent.

Example

The following example shows how to perform a graceful shutdown:

```
Switch(config)# sgcp graceful-shutdown
```

CHAPTER 19

Configuring Frame Relay to ATM Interworking Port Adapter Interfaces

This chapter describes Frame Relay to ATM interworking and the required steps to configure the channelized Frame Relay port adapters in the Catalyst 8510 MSR and LightStream 1010 ATM switch routers. These port adapters facilitate interworking between a Frame Relay network, an ATM network, and network users. Existing Frame Relay users can also migrate to higher bandwidth ATM using channelized Frame Relay port adapters. Additionally, these port adapters extend the ATM network across a wide area over a frame-based serial line or intervening Frame Relay WAN.

Note This chapter provides advanced configuration instructions for the Catalyst 8540 MSR, Catalyst 8510 MSR, and LightStream 1010 ATM switch routers. For an overview of Frame Relay to ATM interworking, refer to the *Guide to ATM Technology*. For complete descriptions of the commands mentioned in this chapter, refer to the *ATM Switch Router Command Reference* publication. For hardware installation and cabling instructions, refer to the *ATM and Layer 3 Port Adapter and Interface Module Installation Guide*.

For a more information on how to configure your Frame Relay specific network equipment, refer to the Cisco IOS 11.3 publications on the Documentation CD-ROM.

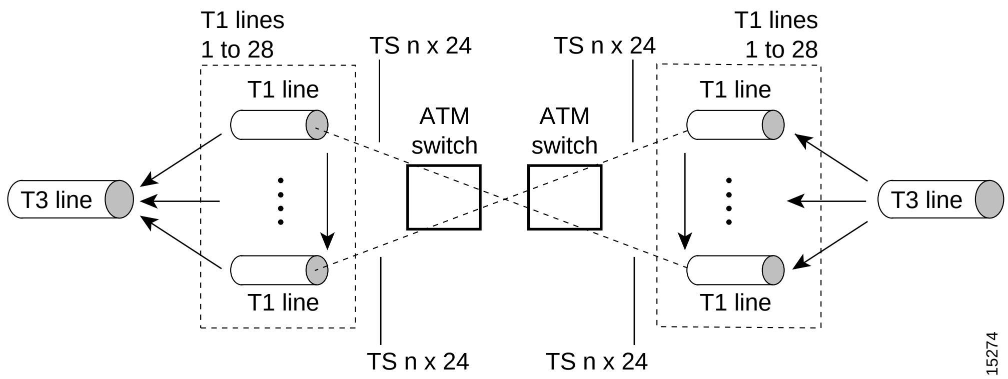
This chapter includes the following sections:

Configuring the Channelized DS3 Frame Relay Port Adapter

The channelized DS3 (CDS3) Frame Relay port adapter provides one physical port (45 Mbps). Each DS3 interface consists of 28 T1 lines multiplexed through a single T3 trunk. Each T1 line operates at 1.544 Mbps, which equates to 24 time slots (DS0 channels). A DS0 time slot provides 56 or 64 kbps of usable bandwidth. You can combine one or more DS0 time slots into a channel group to form a serial interface. A channel group provides *n* x 56 or 64 kbps of usable bandwidth, where *n* is the number of time slots, from 1 to 24. You can configure a maximum of 127 serial interfaces, or channel groups, per port adapter.

Figure 19-1 illustrates how a T3 trunk demultiplexes into 28 T1 lines that provide single or multiple time slots mapped across the ATM network. These time slots are then multiplexed to form an outgoing T3 bit stream.

Figure 19-1 T3/T1 Time Slot Mapping



Configuration Guidelines

In order to configure the CDS3 Frame Relay port adapter physical interface you need the following information:

- Digital transmission link information, for example, T3 and T1 clock source and framing type
- Channel information and time slot mapping
- Protocols and encapsulations you plan to use on the new interfaces

Default CDS3 Frame Relay Port Adapter Interface Configuration

The following defaults are assigned to all CDS3 Frame Relay port adapter interfaces:

- Framing — M23
- Clock source — loop-timed
- Cable length — 224

The following defaults are assigned to all T1 lines on the CDS3 Frame Relay port adapter:

- Framing — esf
- Speed — 64 kbps
- Clock source — internal
- Line coding — b8zs
- T1 yellow alarm — detection and generation

Configuring the CDS3 Frame Relay Port Adapter Interface

To manually change any of your default configuration values, perform the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **controller t3** *card*/*subcard*/*port* Switch(config-controller)#	Specifies the controller interface port and enters controller configuration mode.
Step 2	Switch(config-controller)# **clock source** {**free-running** | **loop-timed** | **network-derived** | **reference**}	Configures the type of clocking.
Step 3	Switch(config-controller)# **framing** {**c-bit** | **m23**}	Configures the CDS3 Frame Relay port adapter framing type.
Step 4	Switch(config-controller)# **cablelength** *cablelength*	Configures the CDS3 Frame Relay port adapter cable length.
Step 5	Switch(config-controller)# **mdl** {**transmit** {**path** | **idle-signal** | **test-signal**} | **string** {**eic** | **lic** | **fic** | **unit** | **pfi** | **port** | **generator** *string*}[1]	Configures the maintenance data link (MDL) message.

1. MDL messages are only supported when framing on the CDS3 Frame Relay port adapter is set for c-bit parity.

Example

The following example shows how to change the cable length configuration to 300 using the **cablelength** command.

```
Switch(config)# controller t3 3/0/0
Switch(config-controller)# cablelength 300
```

When using the cable length option, note that user-specified T3 cable lengths are structured into ranges as follows: 0 to 224 and 225 to 450. If you enter a cable length value that falls into one of these ranges, the range for that value is used.

For example, if you enter 150 feet, the 0 to 224 range is used. If you later change the cable length to 200 feet, there is no change because 200 is within the 0 to 224 range. However, if you change the cable length to 250, the 225 to 450 range is used. The actual number you enter is stored in the configuration file.

Configuring the T1 Lines on the CDS3 Frame Relay Port Adapter

To configure the T1 lines, perform the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **controller t3** *card*/*subcard*/*port* Switch(config-controller)#	Specifies the controller interface port and enters controller configuration mode.
Step 2	Switch(config-controller)# **t1** *line-number* **framing** {**esf** | **sf**}	Configures the T1 framing type.
Step 3	Switch(config-controller)# **t1** *line-number* **yellow** {**detection** | **generation**}	Configures yellow alarms for the T1 line.

Configuring the Channel Group on the CDS3 Frame Relay Port Adapter

A channel group, also referred to as a serial interface, is configured on a T1 line by associating time slots to it. The channel group can have from 1 to 24 time slots (DS0s). The transmission rate or bandwidth of the channel group is calculated by multiplying the number of time slots times 56 kbps or 64 kbps.

Note A time slot can be part of only one channel group. Additionally, all time slots within a channel group must be on the same T1 line.

To configure the channel group on a T1 line, perform the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **controller t3** *card*/*subcard*/*port*	Specifies the controller interface port and enters controller configuration mode.
Step 2	Switch(config-controller)# **channel-group** *number* **t1** *line-number* **timeslots** *list* [**speed** {**56** | **64**}]	Creates the channel group with the specified time slots and speed.

Note You can group either contiguous or noncontiguous time slots on a T1 line.

Example

The following example shows how to configure a channel group (with identifier 5), assigning time slots 1 through 5 on T1 line 1 using the **channel-group** command.

```
Switch(config)# controller t3 0/1/0
Switch(config-controller)# channel-group 5 t1 1 timeslots 1-5
Switch(config-controller)#
```

Note The example above creates the serial interface 0/1/0:5.

Displaying the CDS3 Frame Relay Port Adapter Controller Information

To display the controller configuration, use one of the following EXEC commands:

Command	Purpose
show controllers t3 *card*/*subcard*/*port*[**:***t1-line*] [**brief** | **tabular**]	Displays T3 and T1 configuration.

Example

The following example displays the configuration, status, and statistics of T1 line number 1 on controller 0/1/0:

```
Switch# show controllers t3 0/1/0:1 tabular
➔ T3 0/1/0:1 is up.
 PAM state is Up
 1CT3 H/W Version: 1.7
 1CT3 F/W Version: 2.7
➔ T3 0/1/0 T1 1
  Transmitter is sending LOF Indication (RAI).
  Receiver has loss of frame.
  Framing is ESF, Line Code is B8ZS, Clock Source is line.
  INTERVAL      LCV   PCV   CSS  SELS   LES    DM    ES   BES   SES   UAS    SS
  12:43-12:51     0     0     0     0     0     0     0     0     0   434     0
  12:28-12:43     0     0     0     0     0     0     0     0     0   900     0
  12:13-12:28     0     0     0     0     0     0     0     0     0   900     0
  11:58-12:13     0     0     0     0     0     0     0     0     0   900     0
  11:43-11:58     0     0     0     0     0     0     0     0     0   900     0
  11:28-11:43     0     0     0     0     0     0     0     0     0   900     0
  11:13-11:28     0     0     0     0     0     0     0     0     0   900     0
  10:58-11:13     0     0     0     0     0     0     0     0     0   900     0
  Total           0     0     0     0     0     0     0     0     0  6300     0
```

Deleting a Channel Group on the CDS3

This section describes two ways to delete a channel group on the CDS3 after it has been configured.

If you want to delete individual channel groups without shutting down the controller, use method one.

If you want to delete several channels groups on a controller, use method two. However, if you use method two, you must first shut down the controller, which shuts down all channel groups on the controller.

Method One

Perform the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **interface serial** *card*/*subcard*/*port***:***cgn*	Selects the Frame Relay serial port and channel group number to be deleted.
Step 2	Switch(config-if)# **shutdown**	Shuts down the serial interface.

	Command	Purpose
Step 3	Switch(config-if)# **exit** Switch(config)#	Exits serial interface configuration mode.
Step 4	Switch(config)# **controller t3** *card*/*subcard*/*port* Switch(config-controller)#	Selects the controller interface port and enters controller configuration mode.
Step 5	Switch(config-controller)# **no channel-group** *cgn*	Deletes the selected channel group number.

Method Two

Perform the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **controller t3** *card*/*subcard*/*port* Switch(config-controller)#	Selects the controller interface port and enters controller configuration mode.
Step 2	Switch(config-controller)# **shutdown**	Shuts down the controller interface.
Step 3	Switch(config-controller)# **no channel-group** *cgn*	Deletes the selected channel group number.
Step 4	Switch(config-controller)# **no shutdown**	Reenables the controller interface.

Examples

The following example shuts down the serial interface and deletes channel group 1:

```
Switch(config)# interface serial 4/0/0:1
Switch(config-if)# shutdown
Switch(config-if)# exit
Switch(config)# controller t3 4/0/0
Switch(config-controller)# no channel-group 1
Switch(config-controller)# end
Switch#
```

The following example shuts down the T3 controller, deletes channel group 1, and then reenables the T3 controller:

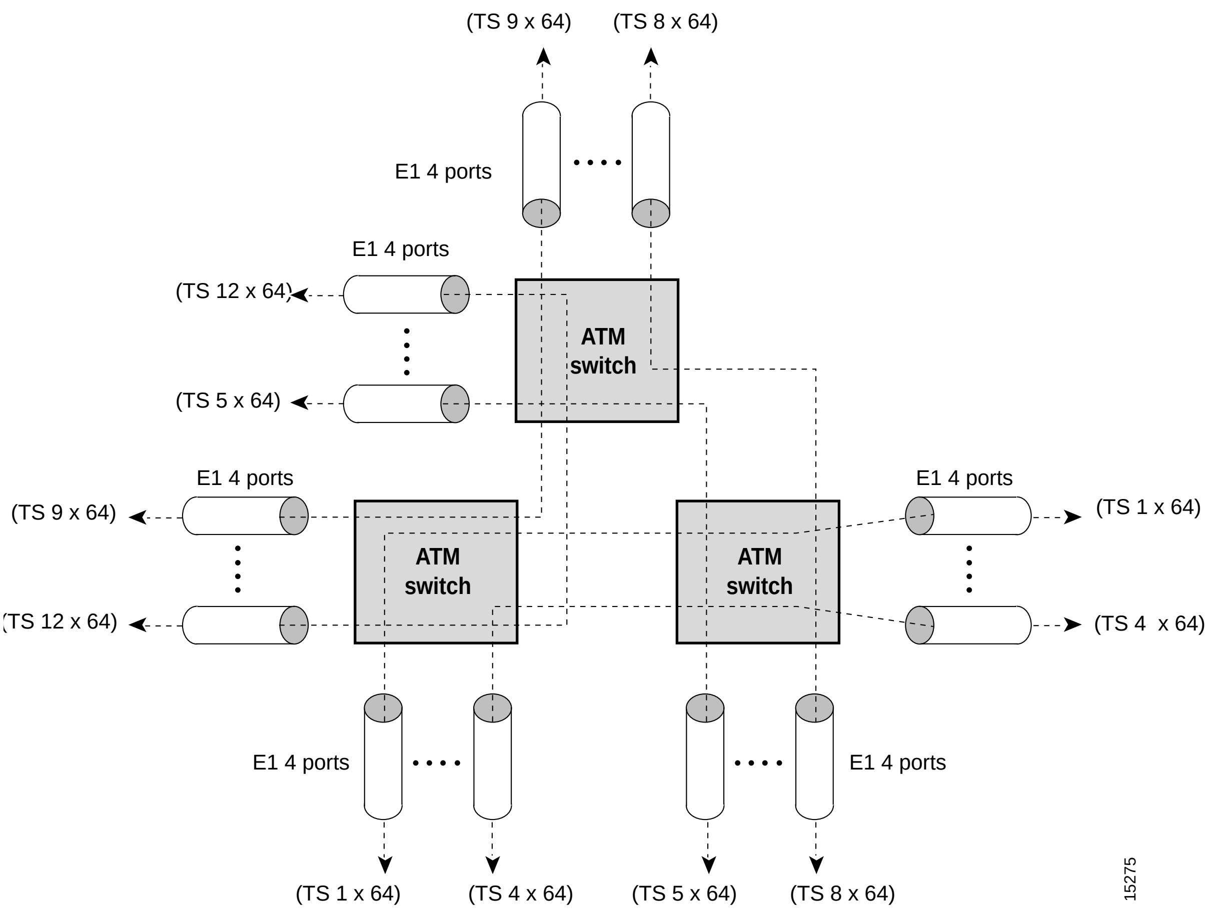
```
Switch(config)# controller t3 4/0/0
Switch(config-controller)# shutdown
Switch(config-controller)# no channel-group 1
Switch(config-controller)# no shutdown
Switch(config-controller)# end
Switch#
```

Configuring the Channelized E1 Frame Relay Port Adapter

The channelized E1 (CE1) Frame Relay port adapter provides four physical ports. Each port supports up to 31 E1 serial interfaces, also referred to as channel groups, totalling 124 serial interfaces per port adapter. The E1 line operates at 2.048 Mbps, which is equivalent to 31 time slots (DS0 channels). The E1 time slot provides usable bandwidth of *n* x 64 kbps, where *n* is the time slot from 1 to 31.

Figure 19-2 illustrates how an E1 trunk (with four ports) provides single or multiple time slots mapped across the ATM network. Each time slot represents a single *n* x 64 circuit that transmits data at a rate of 64 kbps. Multiple *n* x 64 circuits can be connected to a single port, using separate time slots.

Figure 19-2 E1 Time Slot Mapping



Default CE1 Frame Relay Port Adapter Interface Configuration

The following defaults are assigned to all CE1 Frame Relay port adapter interfaces:

- Framing—crc4
- Clock source—loop-timed
- Line coding—HDB3

Configuring the CE1 Frame Relay Port Adapter Interface

If your CE1 Frame Relay port adapter needs to be configured, you must have the following information:

- Digital transmission link information, for example, E1 clock source and framing type
- Channel information and time slot mapping
- Protocols and encapsulations you plan to use on the new interfaces

To manually change any of your default configuration values, perform the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **controller e1** *card*/*subcard*/*port* Switch(config-controller)#	Specifies the controller interface port and enters controller configuration mode.
Step 2	Switch(config-controller)# **clock source** {**free-running** | **loop-timed** | **reference** | **network-derived**}	Configures the type of clocking.
Step 3	Switch(config-controller)# **framing** {**crc4** | **no-crc4**}	Configures the E1 framing type.

Example

The following example shows how to change the clock source to free-running using the **clock source** command.

```
Switch(config)# controller e1 1/0/0
Switch(config-controller)# clock source free-running
```

Configuring the Channel Group on the CE1 Frame Relay Port Adapter

A channel group, also referred to as a serial interface, is configured on an E1 line by associating time slots to it. The channel group can have from 1 to 31 time slots (DS0s). The transmission rate or bandwidth of the channel group is calculated by multiplying the number of time slots times 64 kbps.

To configure the channel group, perform the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **controller e1** *card*/*subcard*/*port* Switch(config-controller)#	Specifies the controller interface port and enters controller configuration mode.
Step 2	Switch(config-controller)# **channel-group** *number* {**timeslots** *range* | **unframed**}	Configures the identifier and range of E1 time slot number(s) that comprise the channel group. The keyword **unframed** configures a CE1Frame Relay interface as clear channel (unframed).

Example

The following example shows how to configure time slots 1 through 5 and 20 through 23 on E1 channel group 5 using the **channel-group** command.

```
Switch(config)# controller e1 0/1/0
Switch(config-controller)# channel-group 5 timeslots 1-5, 20-23
```

Displaying the CE1 Frame Relay Port Adapter Controller Information

To display your controller configuration, use the following EXEC command:

Command	Purpose
show controllers e1 *card*/*subcard*/*port* [**brief** | **tabular**]	Displays E1 controller configuration.

Example

The configuration for controller E1 is displayed in the following example:

```
Switch# show controllers e1 0/0/0 tabular
E1 0/0/0 is up.
E1 0/0/0 is up.
 PAM state is Up
 4CE1 H/W Version: 3.1
  4CE1 F/W Version: 2.0
  No alarms detected.
  Framing is crc4, Line Code is HDB3, Clock Source is line.
  INTERVAL       LCV  PCV  CS  SELS LES  DM   ES  BES  SES  UAS  SS
  18:38-18:51     0    0   0    0    0    0    2    0   10  704   0
```

Configuring Frame Relay to ATM Interworking Functions

You must follow the required steps to enable Frame Relay to ATM interworking on your ATM switch router. In addition, you can customize Frame Relay to ATM for your particular network needs and monitor Frame Relay-to-ATM connections. The following sections outline these tasks:

- Enabling Frame Relay Encapsulation on an Interface, page 19-10
- Configuring Frame Relay Serial Interface Type, page 19-11

For information on how to customize your Frame Relay-to-ATM connections, see the “Configuring LMI” section on page 19-12 and the “Configuring Frame Relay to ATM Resource Management” section on page 19-16.

Enabling Frame Relay Encapsulation on an Interface

To set Frame Relay encapsulation on the serial interface, perform the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **interface serial** *card/subcard/port***:***cgn* Switch(config-if)#	Selects the interface to be configured.
Step 2	Switch(config-if)# **encapsulation frame-relay ietf**	Configures Frame Relay encapsulation.

Frame Relay supports encapsulation of all supported protocols in conformance with RFC 1490, allowing interoperability between multiple vendors.

Note You must shut down the interface prior to Frame Relay encapsulation.

Example

```
Switch(config)# interface serial 0/1/0:5
Switch(config-if)# shutdown
Switch(config-if)# encapsulation frame-relay ietf
Switch(config-if)# no shutdown
```

Displaying Frame Relay Encapsulation

To display Frame Relay encapsulation, use the following user EXEC command:

Command	Purpose
show interfaces serial *card/subcard/port***:***cgn*	Displays Frame Relay encapsulation.

Example:

The following example displays the Frame Relay encapsulation configuration on serial interface 0/1/0:5:

```
Switch# show interfaces serial 0/1/0:5
Serial0/1/0:5 is up, line protocol is up
  Hardware is FRPAM-SERIAL
  MTU 4096 bytes, BW 320 Kbit, DLY 0 usec, rely 0/255, load 1/255
Encapsulation FRAME-RELAY IETF, loopback not set, keepalive not set
  Last input never, output never, output hang never
  Last clearing of "show interface" counters never
  Input queue: 0/75/0 (size/max/drops); Total output drops:
<information deleted>
```

Configuring Frame Relay Serial Interface Type

To configure an interface as a data communications equipment (DCE) or Network-Network Interface (NNI) type, perform the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **interface serial** *card*/*subcard*/*port***:***cgn* Switch(config-if)#	Selects the interface to be configured.
Step 2	Switch(config-if)# **frame-relay intf-type** {**dce** | **nni**}	Selects a Frame Relay interface type.

Example

The following example shows how to configure Frame Relay interface type NNI for serial interface 0/1/0:5:

```
Switch(config)# interface serial 0/1/0:5
Switch(config-if)# frame-relay intf-type nni
```

Displaying Frame Relay Interface Configuration

To display the Frame Relay interface configuration, use the following EXEC command:

Command	Purpose
more system:running-config	Displays the Frame Relay interface configuration.

Example

The Frame Relay configuration is displayed in the following example:

```
Switch# more system:running-config
Building configuration...

Current configuration:
!
version 11.3
no service pad
no service password-encryption
!
hostname Switch
!
<information deleted>
!
interface Serial0/1/0:5
 no ip address
 no ip directed-broadcast
 encapsulation frame-relay IETF
 no arp frame-relay
➔ frame-relay intf-type nni
<information deleted>
```

Configuring LMI

Three industry-accepted standards are supported for addressing the Local Management Interface (LMI), including the Cisco specification. By default, the Cisco ILMI option is active on your Frame Relay interface.

Configuring the LMI Type

To manually set an LMI type on your Frame Relay port adapter, perform the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **interface serial** *card*/*subcard*/*port*:*cgn* Switch(config-if)#	Selects the interface to be configured.
Step 2	Switch(config-if)# **frame-relay lmi-type** [**cisco** | **ansi** | **q933a**]	Selects Frame Relay LMI type.
Step 3	Switch(config-if)# **end** Switch#	Exits interface configuration mode.
Step 4	Switch# **copy system:running-config nvram:startup-config**	Writes the LMI type to NVRAM.

Example

The following example changes the LMI type to ansi on serial interface 1/1/0:1:

```
Switch(config)# interface serial 1/1/0:1
Switch(config-if)# frame-relay lmi-type ansi
Switch(config-if)# end
Switch# copy system:running-config nvram:startup-config
```

Displaying LMI Type

To display the LMI type configuration, perform the following task in user EXEC mode:

Command	Purpose
show frame-relay lmi interface serial *card*/*subcard*/*port*:*cgn*	Displays LMI type configuration.

Example

The following example displays the LMI type configuration of a Frame Relay port adapter:

```
Switch> show frame-relay lmi interface serial 1/1/0:1

LMI Statistics for interface Serial1/1/0:1 (Frame Relay NNI) LMI TYPE = ANSI
  Invalid Unnumbered info 0             Invalid Prot Disc 0
  Invalid dummy Call Ref 0              Invalid Msg Type 0
  Invalid Status Message 0              Invalid Lock Shift 0
  Invalid Information ID 0              Invalid Report IE Len 0
  Invalid Report Request 0              Invalid Keep IE Len 0
  Num Status Enq. Rcvd 5103             Num Status msgs Sent 5103
  Num Update Status Rcvd 0              Num St Enq. Timeouts 10
  Num Status Enq. Sent 5118             Num Status msgs Rcvd 5103
  Num Update Status Sent 0              Num Status Timeouts 14
```

Configuring the LMI Keepalive Interval

A keepalive interval must be set to configure the LMI. By default, this interval is 10 seconds and, per the LMI protocol, must be set as a positive integer that is less than the lmi-t392dce interval set on the interface of the neighboring switch.

To set the keepalive interval, perform the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **interface serial** *card/subcard/port***:***cgn* Switch(config-if)#	Selects the interface to be configured.
Step 2	Switch(config-if)# **keepalive** *number*	Selects the keepalive interval.

Example

The following example configures the LMI keepalive interval to 30 seconds:

```
Switch(config)# interface serial 1/1/0:1
Switch(config-if)# keepalive 30
```

Displaying LMI Keepalive Interval

To display the LMI keepalive interval, perform the following task in user EXEC mode:

Command	Purpose
show frame-relay lmi interface serial *card/subcard/port***:***cgn*	Displays LMI keepalive interval.

Example

The following example displays the LMI keepalive interval of a Frame Relay port adapter:

```
Switch> show interfaces serial 1/1/0:1
Serial1/1/0:1 is up, line protocol is up
  Hardware is FRPAM-SERIAL
  MTU 4096 bytes, BW 640 Kbit, DLY 0 usec, rely 255/255, load 1/255
➔ Encapsulation FRAME-RELAY IETF, loopback not set, keepalive set (30 sec)
  LMI enq sent  5163, LMI stat recvd 5144, LMI upd recvd 0, DTE LMI up
  LMI enq recvd 5154, LMI stat sent  5154, LMI upd sent  0, DCE LMI up
  LMI DLCI 1023  LMI type is CISCO  frame relay NNI
  Last input 00:00:04, output 00:00:20, output hang never

<Information Deleted>
```

Configuring the LMI Polling and Timer Intervals (Optional)

You can set various optional counters, intervals, and thresholds to fine-tune the operation of your LMI on your Frame Relay devices. Set these attributes by performing one or more of the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **interface serial** *card/subcard/port***:***cgn* Switch(config-if)#	Selects the interface to be configured.
Step 2	Switch(config-if)# **frame-relay lmi-n391dte** *keep-exchanges*	Configures an NNI full status polling interval.
Step 3	Switch(config-if)# **frame-relay lmi-n392dce** *threshold*	Configures the DCE and the NNI error threshold.
Step 4	Switch(config-if)# **frame-relay lmi-n392dte** *threshold*	Configures the NNI error threshold.
Step 5	Switch(config-if)# **frame-relay lmi-n393dce** *events*	Configures the DCE and NNI monitored events count.
Step 6	Switch(config-if)# **frame-relay lmi-n393dte** *events*	Configures the monitored event count on an NNI interface.
Step 7	Switch(config-if)# **frame-relay lmi-t392dce** *seconds*	Configures the polling verification timer on a DCE or NNI interface.

Example

The following example shows how to change the default polling verification timer on a Frame Relay interface to 20 seconds using the **frame-relay lmi-t392dce** command.

```
Switch(config)# interface serial 0/1/0:5
Switch(config-if)# frame-relay lmi-t392dce 20
```

Displaying Frame Relay Serial Interface

To display information about a serial interface, perform the following task in user EXEC mode:

Command	Purpose
show interfaces serial *card*/*subcard*/*port***:***cgn*	Displays Frame Relay serial interface configuration.

Example

The following example displays serial interface configuration information for an interface with Cisco LMI enabled:

```
Switch> show interfaces serial 0/1/0:5
Serial 0/1/0:5 is up, line protocol is up
Hardware is FRPAM-SERIAL
MTU 4096 bytes, BW 1536 Kbit, DLY 0 usec, rely 229/255, load 14/255
Encapsulation FRAME-RELAY IETF, loopback not set, keepalive set (10 sec)
LMI enq sent 0, LMI stat recvd 0, LMI upd recvd 0
➔ LMI DLCI 1023 LMI type is CISCO frame relay DCE
<information deleted>
```

Displaying LMI Statistics

To display statistics about the LMI, perform the following task in user EXEC mode:

Command	Purpose
show frame-relay lmi interface serial *card*/*subcard*/*port***:***cgn*	Displays LMI statistics.

Example

The following example displays the LMI statistics of a Frame Relay port adapter with an NNI interface:

```
Switch> show frame-relay lmi interface serial 0/1/0:5
LMI Statistics for interface serial 0/1/0:5 (Frame Relay NNI) LMI Type = Cisco
Invalid Unnumberred info 0Invalid Prot Disc 0
Invalid dummy Call Ref 0Invalid msg Type 0
Invalid Status Message 0Invalid Lock Shift 0
Invalid Information ID 0Invalid Report IE Len 0
Invalid Report Request 0Invalid Keep IE Len 0
Num Status Enq. Rcvd 11Num Status msgs Sent 11
Num Update Status Rcvd 0Num St Enq Timeouts 0
Num Status Enq. Sent 10Num Status msgs Rcvd 10
Num Update Status Sent 0Num Status Timeouts 0
```

Configuring Frame Relay to ATM Resource Management

This section describes the following resource management tasks specifically for your Frame Relay to ATM interworking network needs:

- Configuring Frame Relay-to-ATM Connection Traffic Table Rows, page 19-16
- Creating a Frame Relay-to-ATM CTT Row, page 19-17
- Configuring the Interface Resource Management Tasks, page 19-18

For information about how to configure your ATM Connection Traffic Table rows, see the “Configuring the Connection Traffic Table” section on page 8-10.

Configuring Frame Relay-to-ATM Connection Traffic Table Rows

A row in the Frame Relay-to-ATM Connection Traffic Table (CTT) must be created for each unique combination of Frame Relay traffic parameters. All Frame Relay to ATM interworking virtual connections then provide traffic parameters for each row in the table per flow (receive and transmit). Multiple virtual connections can refer to the same traffic table row.

The Frame Relay traffic parameters (specified in the command used to create the row) are converted into equivalent ATM traffic parameters. Both parameters are stored internally and used for interworking virtual connections.

The formula used for Frame Relay to ATM traffic conversions are specified in the B-ICI specification, V2.0. Use a frame size (*n*) of 250 bytes and a header size of 2 bytes. See Table 19-1.

Table 19-1 Frame Relay to ATM Traffic Conversion

Peak Cell Rate (0+1) (Cells Per Second) =	Peak Information Rate[1] /8 * (6/260)
Sustainable Cell Rate (0) (Cells Per Second) =	Committed Information Rate[1] /8 * (6/250)
Maximum Burst Size (0) (Cells) =	(Committed Burst Size[2] /8 * (1/(1-Committed InformationRate/PeakInformationRate))+1)*(6/250)

1. In bits per second
2. In bits

PVC Connection Traffic Rows

Permanent virtual channel (PVC) connection traffic rows, or stable rows, are used to specify traffic parameters for PVCs.

Note PVC connection traffic rows cannot be deleted while in use by a connection.

SVC Connection Traffic Rows

SVC connection traffic rows, or transient rows, are used by the signalling software to obtain traffic parameters for soft SVCs.

Note SVC connection traffic rows cannot be deleted from the CLI or SNMP. They are automatically deleted when the connection is removed.

To make the CTT management software more efficient, the CTT row-index space is split into space allocated by the CLI/SNMP and signalling. See Table 19-2.

Table 19-2 CTT Row-Index Allocation

Allocated By	Row-Index Range
CLI/SNMP	1 through 1,073,741,823
Signalling	1,073,741,824 through 2,147,483,647

Predefined Rows

Table 19-3 describes the predefined row:

Table 19-3 Default Frame Relay to ATM Connection Traffic Table Row

CTT Row-Index	CIR (bits/s)	Bc (bits)	Be (bits)	PIR (bits/s)	Service Category	ATM Row-Index
100	64,000	32,768	32,768	64,000	VBR-NRT	100

Creating a Frame Relay-to-ATM CTT Row

To create a Frame Relay-to-ATM CTT row, perform the following task in global configuration mode:

Command	Purpose
frame-relay connection-traffic-table-row [**index** *row-index*] *cir-value bc-value pir-value be-value* {**abr** | **vbr-nrt** | **ubr**} [*atm-row-index*]	Configures a Frame Relay-to-ATM CTT row.

If you do not specify an index row number, the system software determines if one is free. The index row number is then displayed in the allocated index field if the command is successful.

If the ATM row index is not specified, system software tries to use the same row index used by Frame Relay. If not possible, a free ATM row index is used.

Example

The following example shows how to configure a Frame Relay-to-ATM CTT row with non-real-time variable bit rate (VBR-NRT) service category, committed information rate of 64000 bits per second, a peak information rate of 1536000 bits per second, and a committed burst size of 8192 bits per second:

```
Switch(config)# frame-relay connection-traffic-table-row 64000 8192 1536000 vbr-nrt
Allocated index = 64000
Switch(config)#
```

Displaying the Frame Relay-to-ATM Connection Traffic Table

To display the Frame Relay-to-ATM CTT configuration, use the following EXEC command:

Command	Purpose
show frame-relay connection-traffic-table-row [**from-row** *row* | **row** *row*]	Displays the Frame Relay-to-ATM CTT configuration.

Example

The following example shows how to display the Frame Relay-to-ATM CTT configuration table:

```
Switch# show frame-relay connection-traffic-table-row
Row      cir      bc        be       pir      FR-ATM    Service Category  ATM row
100      64000   32768     32768     64000                   vbr-nrt        100
```

Configuring the Interface Resource Management Tasks

The following resource management tasks configure queue thresholds, committed burst size, and service overflow on Frame Relay interfaces. To change any of these interface parameters, perform the following steps, in interface configuration mode:

	Command	Purpose
Step 1	Switch(config-if)# **frame-relay input-queue** {**abr** | **ubr** | **vbr-nrt**} {**discard-threshold** | **marking-threshold**} *threshold*	Configures discard and marking thresholds for the inbound direction.
Step 2	Switch(config-if)# **frame-relay output-queue** {**abr** | **ubr** | **vbr-nrt**} {**discard-threshold** | **marking-threshold**} *threshold*	Configures discard and marking thresholds for the outbound direction.
Step 3	Switch(config-if)# **frame-relay bc-default** *bc-value*	Configures the committed burst size (in bits) used for ABR/UBR soft VCs on the destination interface.

	Command	Purpose
Step 4	Switch(config-if)# **frame-relay accept-overflow**	Configures to accept or discard overflow traffic (exceeding CIR) for VBR circuits. **Note** Unavailable on CE1 Frame Relay interfaces.
Step 5	Switch(config-if)# **frame-relay overbooking** *percent*	Configures the percentage of CIR overbooking.

Note Steps 1, 2, 4, and 5 affect existing and future connections on the Frame Relay interface, but Step 3 affects only future connections.

Displaying Frame Relay Interface Resources

To display your Frame Relay interface resource configuration, use the following EXEC command:

Command	Purpose
show frame-relay interface resource serial *card*/*subcard*/*port***:***cgn*	Displays resource allocation on a Frame Relay interface.

Example

The resource information for Frame Relay serial interface 0/1/0:5 is displayed in the following example:

```
Switch# show frame-relay interface resource serial 0/1/0:5
Encapsulation: FRAME-RELAY
Input queues (PAM to switch fabric):
              Discard threshold: 87% vbr-nrt, 87% abr, 87% ubr
              Marking threshold: 75% vbr-nrt, 75% abr, 75% ubr
        Output queues (PAM to line):
              Discard threshold: 87% vbr-nrt, 87% abr, 87% ubr
              Marking threshold: 75% vbr-nrt, 75% abr, 75% ubr
            Overflow servicing for VBR: enabled
Resource Management state:
          Available bit rates (in bps):
             320000 vbr-nrt RX, 320000 vbr-nrt TX
             320000 abr RX,     320000 abr TX
             320000 ubr RX,     320000 ubr TX
          Allocated bit rates (in bps):
             0 vbr-nrt RX, 0 vbr-nrt TX
             0 abr RX,     0 abr TX
             0 ubr RX,     0 ubr TX
```

Configuring Frame Relay-to-ATM Virtual Connections

This section describes how to configure virtual connections (VCs) for Frame Relay to ATM interworking and Frame Relay-to-Frame Relay switching.

The tasks to configure virtual connections are described in the following sections:

- Configuration Guidelines, page 19-20
- Characteristics and Types of Virtual Connections, page 19-20
- Configuring Frame Relay to ATM Network Interworking PVCs, page 19-21
- Configuring Frame Relay to ATM Service Interworking PVCs, page 19-23
- Configuring Terminating Frame Relay to ATM Service Interworking PVCs, page 19-25
- Configuring Frame Relay Transit PVCs, page 19-27

Configuration Guidelines

Perform the following tasks in a prescribed order before configuring a Frame Relay to ATM interworking permanent virtual channel (PVC), soft PVC, or a Frame Relay-to-Frame Relay PVC:

Step 1 Configure the controller on the Frame Relay port adapter.

Step 2 Configure the T1 channel or E1 interface and channel group on the Frame Relay port adapter.

Step 3 Configure Frame Relay encapsulation and Frame Relay LMI on the serial port corresponding to the channel group configured in Step 2.

Step 4 Configure Frame Relay resource management tasks including Frame Relay connection traffic table rows.

Step 5 Configure Frame Relay to ATM interworking VC tasks.

Characteristics and Types of Virtual Connections

The characteristics of the Frame Relay to ATM interworking VC, established when the VC is created, include the following:

- Frame Relay to ATM interworking parameters
- Committed information rate (CIR), committed burst size (Bc), excess burst size (Be), peak information rate (PIR) (that is, access rate [AR]) for Frame Relay
- Peak and average transmission rates for ATM
- Service category
- Cell sequencing integrity
- ATM adaption Layer 5 (AAL5) for terminating interworking PVC

These switching features can be turned off with the interface configuration commands.

Note For information about ATM VCCs, see Chapter 6, “Configuring Virtual Connections.”

Note You can configure a maximum of 2000 virtual connections on a CDS3 or CE1 Frame Relay port adapter.

Table 19-4 lists the types of supported virtual connections.

Table 19-4 Supported Frame Relay to ATM Virtual Connection Types

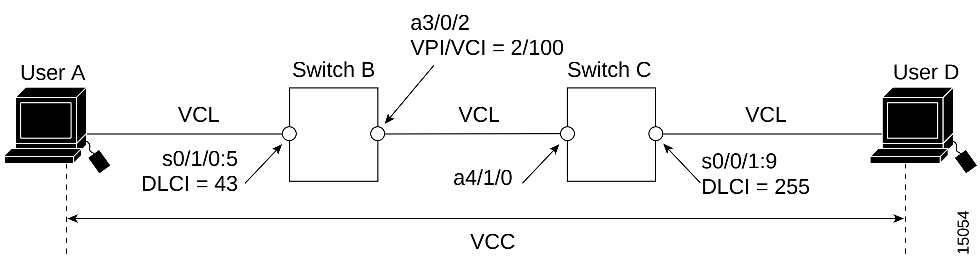
Connection	Point-to-Point	Point-to-Multipoint	Transit	Terminate
Permanent virtual channel	3	–	3	3
Soft permanent virtual channel	3	–	3	–

Configuring Frame Relay to ATM Network Interworking PVCs

This section describes configuring Frame Relay to ATM network interworking PVCs. This type of connection establishes a bidirectional facility that transfers Frame Relay traffic between two Frame Relay users through an ATM network.

Figure 19-3 shows an example of a Frame Relay to ATM network interworking PVC between Frame Relay User A and ATM User D through an ATM network.

Figure 19-3 Network Interworking PVC Example



To configure a Frame Relay to ATM network interworking PVC, perform the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **interface serial** *card/subcard/port***:***cgn*[1] Switch(config-if)#	Selects the interface to be configured.
Step 2	Switch(config-if)# **frame-relay pvc** *dlci*[2] [**upc** {**pass** | **drop**}] [**rx-cttr** *index*] [**tx-cttr** *index*] **network** [**clp-bit** {**0** | **1** | **map-de**}] [**de-bit** {**map-de** | **map-clp-or-de**}] [**interface atm** *card/subcard/port vpi vci* [**upc** *upc*] [**pd** {**off** | **on**}] [**rx-cttr** *index*] [**tx-cttr** *index*]]	Configures a Frame Relay to ATM network interworking PVC.

1. The serial interface is created with the **channel-group** command and configured using the **encapsulation frame-relay ietf** command. *cgn* is the channel group number of a channel group configured using the **channel-group** command.
2. The *dlci* value appears in the **Conn-Id** and **X-Conn-Id** columns of the **show vc** command.

Note The row index for **rx-cttr** and **tx-cttr** must be configured before using this optional parameter. See the "Configuring the Connection Traffic Table" section on page 8-10.

Note When configuring PVC connections, configure the lowest virtual path identifier (VPI) and virtual channel identifier (VCI) numbers first.

Examples

The following example shows how to configure the internal cross-connect Frame Relay to ATM network interworking PVC on Switch B between serial interface 0/1/0:5, DLCI = 43 and ATM interface 3/0/2, VPI = 2, VCI = 100 (see Figure 19-3):

```
Switch-B(config)# interface serial 0/1/0:5
Switch-B(config-if)# frame-relay pvc 43 network interface atm 3/0/2 2 100
```

The following example shows how to configure the internal cross-connect PVC on Switch C between serial interface 0/0/1:9, DLCI = 255 and ATM interface 4/1/0, VPI = 2, VCI = 100:

```
Switch-C(config)# interface serial 0/0/1:9
Switch-C(config-if)# frame-relay pvc 255 network interface atm 4/1/0 2 100
```

Note The Frame Relay to ATM network interworking PVC must be configured from the serial interface and cross-connected to the ATM interface.

Displaying Frame Relay to ATM Network Interworking PVCs

To display the network interworking configuration, use the following EXEC command:

Command	Purpose
show vc [**interface** {**atm** *card*/*subcard*/*port* [*vpi vci*] | **serial** *card*/*subcard*/*port*:*cgn* [*dlci*]}]	Shows the PVC interface configuration.

Example

The following example displays the Switch B PVC configuration for serial interface 0/1/0:5:

```
Switch-B# show vc interface serial 0/1/0:5
Interface      Conn-Id    Type   X-Interface     X-Conn-Id   Encap  Status
Serial0/1/0:5     43      PVC     ATM3/0/2          2/100             UP
```

The following example displays the configuration of the Switch B PVC on serial interface 0/1/0:5, DLCI = 43:

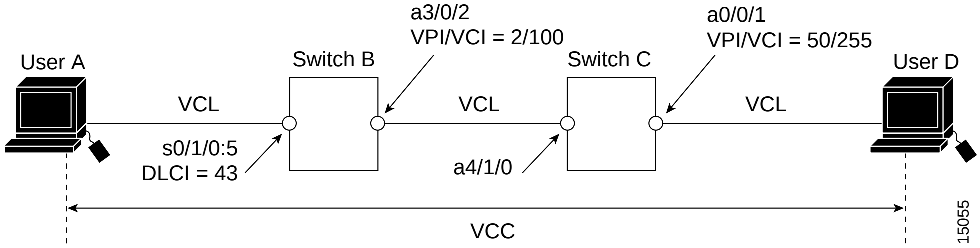
```
Switch-B# show vc interface serial 0/1/0:5 43
Interface: Serial0/1/0:5, Type: FRPAM-SERIAL
DLCI = 43     Status : ACTIVE
Connection-type: PVC
Cast-type: point-to-point
Usage-Parameter-Control (UPC): tag-drop
pvc-create-time : 00:00:10      Time-since-last-status-change : 00:00:03
Interworking Function Type : network
de-bit Mapping : map-clp-or-de       clp-bit Mapping : map-de
ATM-P Interface: ATM-P0/1/0, Type: ATM-PSEUDO
ATM-P VPI = 82  ATM-P VCI = 11
ATM-P Connection Status: UP
Cross-connect-interface: ATM0/0/0, Type: oc3suni
Cross-connect-VPI = 2
Cross-connect-VCI = 100
Cross-connect-UPC: pass
Cross-connect OAM-configuration: disabled
Cross-connect OAM-state:  Not-applicable
tx Frames : 0   Rx Frames : 0
tx Bytes : 0    Rx Bytes : 0
tx Frames Discarded : 0         Rx Frames Discarded : 0
tx Bytes Discarded  : 0         Rx Bytes Discarded  : 0
Rx connection-traffic-table-index: 100
Rx service-category: VBR-NRT (Non-Realtime Variable Bit Rate)
Rx pir: 64000
Rx cir: 64000
Rx Bc : 32768
Rx Be : 32768
Tx connection-traffic-table-index: 100
Tx service-category: VBR-NRT (Non-Realtime Variable Bit Rate)
Tx pir: 64000
Tx cir: 64000
Tx Bc : 32768
Tx Be : 32768
```

Configuring Frame Relay to ATM Service Interworking PVCs

This section describes configuring Frame Relay to ATM service interworking permanent virtual channels (PVCs). A Frame Relay to ATM service interworking PVC is established as a bidirectional facility to transfer Frame Relay to ATM traffic between a Frame Relay user and an ATM user. The upper user protocol encapsulation (FRF.3, RFC 1483, RFC 1490, RFC 1577) mapping can be enabled with the translation option of the **frame-relay pvc** command.

Figure 19-4 shows an example of a Frame Relay to ATM service interworking PVC between Frame Relay User A and ATM User D through an ATM network.

Figure 19-4 Service Interworking PVC Example



Note VPI and VCI values can change when traffic is relayed through the ATM network.

To configure a Frame Relay to ATM service interworking PVC, perform the following steps beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **interface serial** *card*/*subcard*/*port*:*cgn* Switch(config-if)#	Selects the interface to be configured.
Step 2	Switch(config-if)# **frame-relay pvc** *dlci* [**upc** {**pass** | **drop**}] [**rx-cttr** *index*] [**tx-cttr** *index*] **service** {**transparent** | **translation**} [**clp-bit** {**0** | **1** | **map-de**}] [**de-bit** {**0** | **1** | **map-clp**}] [**efci-bit** {**0** | **map-fecn**}] [**interface atm** *card*/*subcard*/*port* *vpi* [*vci* | **any-vci**[1]] [**upc** {**pass** | **drop**}] [**pd** {**off** | **on**}] [**rx-cttr** *index*] [**tx-cttr** *index*] [**encap** *aal-encap*] [**inarp** *minutes*]]	Configures a Frame Relay to ATM service interworking PVC.

1. The **any-vci** option is only available on interface atm0. See note below.

Note Since release 12.0(1a)W5(5b) of the ATM switch software, addressing the interface on the route processor has changed. The ATM interface is now called atm0, and the Ethernet interface is now called ethernet0. Old formats (atm 2/0/0 and ethernet 2/0/0) are still supported.

Note The row index for **rx-cttr** and **tx-cttr** must be configured before using this optional parameter. See the "Configuring the Connection Traffic Table" section on page 8-10.

Examples

The following example shows how to configure the internal cross-connect PVC on Switch B between serial interface 0/1/0:5, DLCI = 43, and ATM interface 3/0/2, VPI = 2, VCI = 100 (with the translation option):

```
Switch-B(config)# interface serial 0/1/0:5
Switch-B(config-if)# frame-relay pvc 43 service translation interface atm 3/0/2 2 100
```

The following example shows how to configure the internal cross-connect PVC on Switch C between ATM interface 4/1/0, VPI = 2, VCI = 100 and ATM interface 0/0/1, VPI 50, VCI = 255:

```
Switch-C(config)# interface atm 4/1/0
Switch-C(config-if)# atm pvc 2 100 interface atm 0/0/1 50 255
```

Each subsequent VC cross connection and link must be configured until the VC is terminated to create the entire PVC.

Note The Frame Relay to ATM service interworking PVC must be configured from the serial interface and then cross-connected to the ATM interface.

Displaying Frame Relay to ATM Service Interworking PVCs

To display the service interworking PVC configuration, use the following EXEC commands:

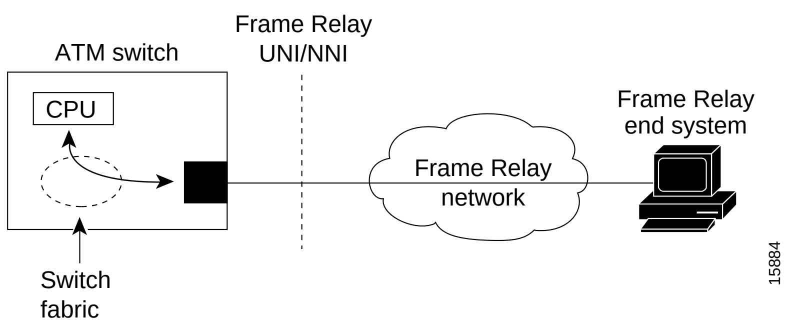
Command	Purpose
show interfaces [**serial** *card*/*subcard*/*port*:*cgn*]	Shows the serial interface configuration.
show vc [**interface** {**atm** *card*/*subcard*/*port* [*vpi vci*] | **serial** *card*/*subcard*/*port*:*cgn* [*dlci*]}]	Shows the PVC interface configuration.

Configuring Terminating Frame Relay to ATM Service Interworking PVCs

This section describes configuring terminating Frame Relay to ATM service interworking permanent virtual channels (PVCs). This type of terminating connection provides the connection from IP over Frame Relay to the ATM switch router used for IP over ATM and network management.

Figure 19-5 shows an example of transmit and terminating connections.

Figure 19-5 Frame Relay to ATM Transmit and Terminating Connections



Terminating connections are configured using the **frame-relay pvc** command; however, all switch terminating connections use atm0 to connect to the ATM switch route processor.

To configure terminating Frame Relay to ATM service interworking PVC connections, perform the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **interface serial** *card*/*subcard*/*port*:*cgn* Switch(config-if)#	Selects the interface to be configured.
Step 2	Switch(config-if)# **frame-relay pvc** *dlci* [**upc** {**pass** | **drop**}] [**rx-cttr** *index*] [**tx-cttr** *index*] **service** {**transparent** | **translation**} [**clp-bit** {**0** | **1** | **map-de**}] [**de-bit** {**0** | **1** | **map-clp**}] [**efci-bit** {**0** | **map-fecn**}] [**interface atm** *card*/*subcard*/*port* *vpi vci* | **any-vci**[1]] [**upc** {**pass** | **drop**}] [**pd** {**off** | **on**}] [**rx-cttr** *index*] [**tx-cttr** *index*] [**encap** *aal-encap*] [**inarp** *minutes*]]	Configures a Frame Relay to ATM service interworking PVC.

1. The **any-vci** option is only available on interface atm0.

Example

The following example shows how to configure the internal cross-connect PVC on Switch B between serial interface 0/1/0:5, DLCI = 50, and the terminating connection on ATM interface 0, VPI = 0, and an unspecified VCI:

```
Switch-B(config)# interface serial 0/1/0:5
Switch-B(config-if)# frame-relay pvc 50 service translation interface atm 0 0 any-vci encap aal5snap
```

Note The Frame Relay to ATM service interworking PVC must be configured from the serial interface and then cross connected to the ATM interface.

Displaying Terminating Frame Relay to ATM Service Interworking PVCs

To display the service interworking PVC configuration, use the following EXEC commands:

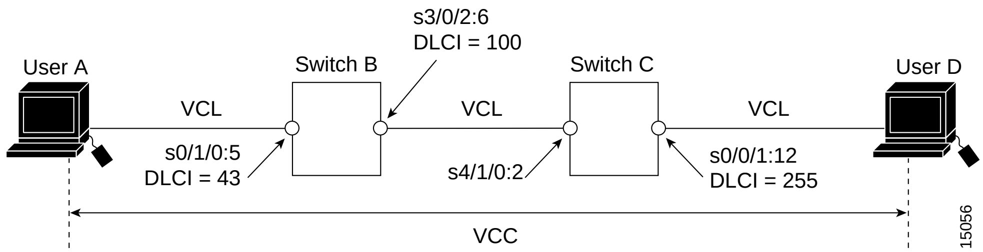
Command	Purpose
show interfaces [**serial** *card*/*subcard*/*port*:*cgn*]	Shows the serial interface configuration.
show vc [**interface** {**atm** *card*/*subcard*/*port* [*vpi vci*] | **serial** *card*/*subcard*/*port*:*cgn* [*dlci*]}]	Shows the PVC interface configuration.

Note See the "Displaying Frame Relay to ATM Network Interworking PVCs" section on page 19-22 for examples of the **show vc** command.

Configuring Frame Relay Transit PVCs

This section describes configuring internal cross-connect Frame Relay-to-Frame Relay transit permanent virtual channels (PVCs). This type of PVC is used to establish a bidirectional facility to transfer Frame Relay traffic between two Frame Relay users. Figure 19-6 shows a Frame Relay transit PVC between Frame Relay users A and D.

Figure 19-6 Transit PVC Example



To configure a Frame Relay transit PVC, perform the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **interface serial** *card*/*subcard*/*port***:***cgn* Switch(config-if)#	Selects the interface to be configured.
Step 2	Switch(config-if)# **frame-relay pvc** *dlci* [**upc** {**pass** | **drop**}] [**rx-cttr** *index*] [**tx-cttr** *index*] **interface serial** *card*/*subcard*/*port***:***cgn* **dlci** *dlci* [**upc** {**pass** | **drop**}] [**rx-cttr** *index*] [**tx-cttr** *index*]	Configures a Frame Relay-to-Frame Relay transit PVC.

Examples

The following example shows how to configure the internal cross-connect Frame Relay PVC on Switch B between serial interface 0/1/0:5, DLCI = 43, and serial interface 3/0/2:6, DLCI = 100:

```
Switch-B(config)# interface serial 0/1/0:5
Switch-B(config-if)# frame-relay pvc 43 interface serial 3/0/2:6 100
```

The following example shows how to configure the internal cross-connect Frame Relay on Switch C between serial interface 4/1/0:2, DLCI = 100,0 and serial interface 0/0/1:12, DLCI = 255:

```
Switch-C(config)# interface serial 4/1/0:2
Switch-C(config-if)# frame-relay pvc 100 interface serial 0/0/1:12 255
```

Each subsequent VC cross-connection and link must be configured until the VC is terminated to create the entire VCC.

To display Frame Relay transit PVCs, use the **show interfaces** and **show vc** commands.

Configuring Frame Relay Soft PVC Connections

This section describes configuring Frame Relay to ATM interworking soft permanent virtual channels (soft PVC) connections.

You can configure the following soft PVC connections:

- Frame Relay-to-Frame Relay soft PVC connection, configured as network interworking
- Frame Relay to ATM soft PVC connection, configured as network interworking
- Frame Relay to ATM soft PVC connection, configured as service interworking

Configuration Guidelines

These guidelines are appropriate for both network and service interworking soft PVC connections.

Note Frame Relay interworking soft PVCs can only be configured from a Frame Relay interface.

Perform the following steps, and see Figure 19-7:

Step 1 Determine which two switches you want to define as participants in the soft PVC.

Step 2 Determine the source (active) side of the soft PVC.

Step 3 Determine an available data-link connection identifier (DLCI) for value *dlci_a* on the source end of the soft PVC.

Step 4 Determine the destination (passive) side of the soft PVC.

Step 5 Determine the ATM address of the destination side of the soft PVC. Use the **show atm addresses** command on the destination switch.

Step 6 If the destination side of the soft PVC is a Frame Relay interface, choose an available DLCI value. Use the **show vc interface serial** command.

If the destination side of the soft PVC is an ATM interface, choose an available VPI/VCI value.

Step 7 Choose the interworking function type, and the relevant interworking parameters (for example, de-bit/clp-bit mapping options).

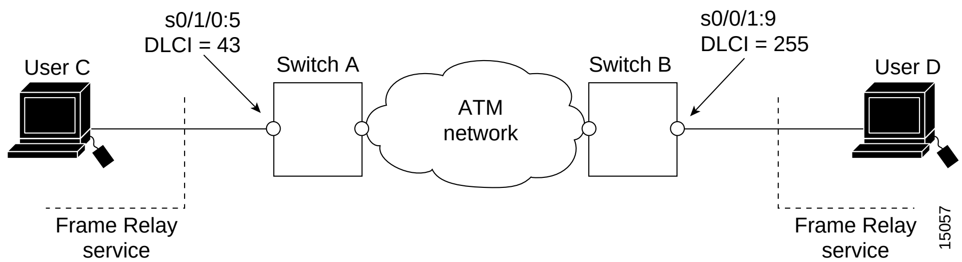
Note If the soft PVC terminates on a Frame Relay interface, the soft PVC can only be configured as a network interworking connection. If the soft PVC terminates on an ATM interface, the soft PVC can be configured either as a network interworking connection or a service interworking connection.

Step 8 Configure the Frame Relay interworking soft PVC on the source side. See the following sections for configuration steps and examples.

Configuring Frame Relay-to-Frame Relay Network Interworking Soft PVCs

This section describes how to configure a Frame Relay-to-Frame Relay network interworking soft PVC terminating on two Frame Relay interfaces. Figure 19-7 shows a Frame Relay-to-Frame Relay network interworking soft PVC between Switch A and Switch B.

Figure 19-7 Frame Relay-to-Frame Relay Network Interworking Soft PVC Example



To configure a Frame Relay-to-Frame Relay network interworking soft PVC, perform the following steps, beginning in EXEC mode:

	Command	Purpose
Step 1	Switch# **show interfaces**	Determines source and destination interfaces.
Step 2	Switch# **show vc interface serial** *card*/*subcard*/*port*:*cgn* [*dlci*]	Determines the DLCI available for Step 3.
Step 3	Switch# **show vc interface serial** *card*/*subcard*/*port*:*cgn* [*dlci*]	Determines the DLCI available for Step 7.
Step 4	Switch# **show atm addresses**	Determines soft PVC destination address.
Step 5	Switch# **configure terminal** Switch(config)#	From the source (active) side at the privileged EXEC prompt, enter configuration mode from the terminal.
Step 6	Switch(config)# **interface serial** *card*/*subcard*/*port*:*cgn* Switch(config-if)#	Selects the source Frame Relay port and channel group number.
Step 7	Switch(config-if)# **frame-relay soft-vc** *dlci-a* **dest-address** *address* **dlci** *dlci_b* [**upc** {**pass** | **drop**}] [**rx-cttr** *index*] [**tx-cttr** *index*] [**retry-interval** [**first** *first-retry-interval*] [**maximum** *max-retry-interval*]] [**network** [**clp-bit** {**0** | **1** | **map-de**}] **de-bit** {**map-de** | **map-clp-or-de**}]]	Configures a network interworking soft PVC terminating on a Frame Relay serial interface.

The previous configuration steps are illustrated in the following section.

Note The row index for **rx-cttr** and **tx-cttr** must be configured before using this optional parameter. See the “Configuring the Connection Traffic Table” section on page 8-10.

Frame Relay-to-Frame Relay Interworking Soft PVC Configuration Example

This section provides an example of a Frame Relay-to-Frame Relay network interworking soft PVC configured between Switch A and Switch B, as shown in Figure 19-7. The source (active) side is serial interface 0/1/0:5 on Switch A.

Step 1 Use the **show vc interface serial** command to determine that data-link connection identifier (DLCI) 43 is available on serial interface 0/1/0:5 on Switch A:

```
Switch-A# show vc interface serial 0/1/0:5
Interface      Conn-Id Type    X-Interface    X-Conn-Id  Encap  Status
Serial0/1/0:5     54  SoftVC   Serial3/0/0:3    54       SoftVC  UP
Serial0/1/0:5     55  SoftVC   Serial3/0/0:2    55       SoftVC  UP
Serial0/1/0:5     56  SoftVC   ATM0/1/3        0/45      SVC     UP
Serial0/1/0:5     66  SoftVC   ATM1/1/0        0/100     SoftVC  UP
```

Step 2 The destination (passive) side is a Frame Relay serial interface 0/0/1:9 on Switch B.

Step 3 The ATM address for the destination serial interface 0/0/1:9 on Switch B is 47.0091.8100.0000.00e0.1e79.8803.4000.0c81.8010.00.

```
Switch-B# show atm addresses
Switch Address(es):
47.00918100000000E01E798803.00E01E808601.00 active

Soft VC Address(es) :
47.0091.8100.0000.00e0.1e79.8803.4000.0c80.0000.00 ATM1/0/0
47.0091.8100.0000.00e0.1e79.8803.4000.0c80.0010.00 ATM1/0/1
47.0091.8100.0000.00e0.1e79.8803.4000.0c80.0020.00 ATM1/0/2
47.0091.8100.0000.00e0.1e79.8803.4000.0c80.0030.00 ATM1/0/3
<information deleted>

Soft VC Address(es) for Frame Relay Interfaces :
47.0091.8100.0000.00e0.1e79.8803.4000.0c81.8010.00 Serial0/0/1:9
47.0091.8100.0000.00e0.1e79.8803.4000.0c81.8020.00 Serial0/0/1:10

ILMI Switch Prefix(es):
47.0091.8100.0000.00e0.1e79.8803
<information deleted>
```

Step 4 DLCI 255 is available on serial interface 0/0/1:9 Switch B.

```
Switch-B# show vc interface serial 0/0/1:9
Interface Conn-Id Type X-Interface X-Conn-Id Encap Status
Serial0/0/1:9 44 SoftVC Serial3/0/0:3 54 SoftVC UP
Serial0/0/1:9 45 SoftVC Serial3/0/0:2 55 SoftVC UP
Serial0/0/1:9 76 SoftVC ATM0/1/3 0/45 SVC UP
Serial0/0/1:9 86 SoftVC ATM1/1/0 0/100 SoftVC UP
```

Step 5 Configure the network interworking soft PVC from Switch A beginning in global configuration mode.

```
Switch-A(config)# interface serial 0/1/0:5
Switch-A(config-if)# frame-relay soft-vc 43 dest-address
47.0091.8100.0000.00e0.1e79.8803.4000.0c81.8010.00 dlci 255
```

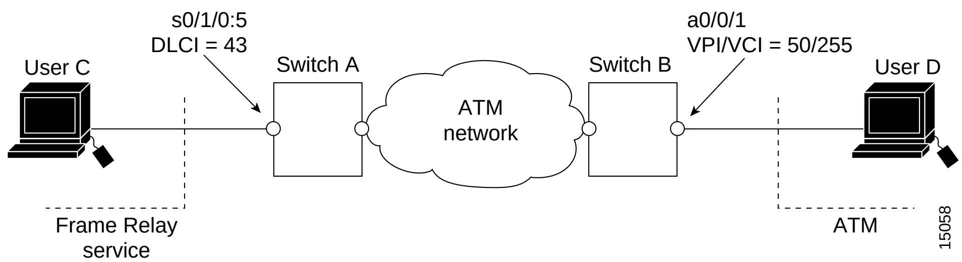
Note If the soft PVC originates and terminates on a Frame Relay interface, the default interworking type is network interworking. You do not need to specify the interworking type explicitly.

After you complete the soft VC configuration, proceed to the "Display Frame Relay Interworking Soft PVCs" section on page 19-35 and verify the connection.

Configuring Frame Relay to ATM Network Interworking Soft PVCs

This section describes how to configure a Frame Relay to ATM network interworking soft permanent virtual channel (soft PVC). Figure 19-8 shows a Frame Relay to ATM network interworking soft PVC between Switch A and Switch B.

Figure 19-8 Frame Relay to ATM Network Interworking Soft PVC Example



To configure a Frame Relay to ATM network interworking soft PVC, perform the following steps, beginning in EXEC mode:

	Command	Purpose
Step 1	Switch# **show interfaces**	Determines source and destination interfaces.
Step 2	Switch# **show vc interface serial** *card*/*subcard*/*port***:***cgn* [*dlci*]	Determines the DLCI available for Step 3.
Step 3	Switch# **show vc interface serial** *card*/*subcard*/*port***:***cgn* [*dlci*]	Determines the DLCI available for Step 7.
Step 4	Switch# **show atm addresses**	Determines soft PVC destination address.
Step 5	Switch# **configure terminal** Switch(config)#	From the source (active) side, at the privileged EXEC prompt, enter configuration mode from the terminal.
Step 6	Switch(config)# **interface serial** *card*/*subcard*/*port***:***cgn* Switch(config-if)#	Selects the source Frame Relay port and channel group number.
Step 7	Switch(config-if)# **frame-relay soft-vc** *dlci-a* **dest-address** *address* **dlci** *dlci_b* [**upc** {**pass** | **drop**}] [**rx-cttr** *index*] [**tx-cttr** *index*] [**retry-interval** [**first** *first-retry-interval*] [**maximum** *max-retry-interval*]] [**network** [**clp-bit** {**0** | **1** | **map-de**}] **de-bit** {**map-de** | **map-clp-or-de**}]] [**explicit-path** *precedence* {**name** *path-name* | **identifier** *path-id*} [**upto** *partial-entry-index*]] [**only-explicit**]	Configures a network interworking soft PVC terminating on an ATM interface.

The previous configuration steps are illustrated in the following section.

Note The row index for **rx-cttr** and **tx-cttr** must be configured before using this optional parameter. See the "Configuring the Connection Traffic Table" section on page 8-10.

Frame Relay to ATM Network Interworking Soft PVC Configuration Example

This section provides an example of a network interworking soft PVC configured between switch A and Switch B and shown in Figure 19-9. The source (active) side is serial interface 0/1/0:5 on Switch A.

Step 1 Use the **show vc interface serial** command to determine that DLCI 43 is available on serial interface 0/1/0:5 Switch A.

```
Switch-A# show vc interface serial 0/1/0:5
Interface      Conn-Id Type    X-Interface    X-Conn-Id  Encap  Status
Serial0/1/0:5     54  SoftVC   Serial3/0/0:3    54       SoftVC  UP
Serial0/1/0:5     55  SoftVC   Serial3/0/0:2    55       SoftVC  UP
Serial0/1/0:5     56  SoftVC   ATM0/1/3        0/45      SVC     UP
Serial0/1/0:5     66  SoftVC   ATM1/1/0        0/100     SoftVC  UP
```

Step 2 On Switch B, use the **show atm addresses** command to determine the destination ATM address for ATM interface 0/0/1, which is 47.0091.8100.0000.00e0.1e19.9904.4000.0c80.0010.00.

```
Switch-B# show atm addresses
Switch Address(es):
47.00918100000000E01E199904.00E01E808601.00 active
Soft VC Address(es) :
47.0091.8100.0000.00e0.1e19.9904.4000.0c80.0000.00 ATM0/0/0
47.0091.8100.0000.00e0.1e19.9904.4000.0c80.0010.00 ATM0/0/1
47.0091.8100.0000.00e0.1e19.9904.4000.0c80.0020.00 ATM0/0/2
47.0091.8100.0000.00e0.1e19.9904.4000.0c80.0030.00 ATM0/0/3
<information deleted>
```

Step 3 On Switch B, use the **show vc interface atm** command to determine that VPI/VCI 50/255 is available for use on ATM interface 0/0/1.

```
Switch-B# show vc interface atm 0/0/1
Interface    Conn-Id    Type   X-Interface     X-Conn-Id   Encap  Status
ATM0/0/1      0/5       PVC     ATM2/0/0          0/58     QSAAL    UP
ATM0/0/1      0/16      PVC     ATM2/0/0          0/44     ILMI     UP
ATM0/0/1      0/18      PVC     ATM2/0/0          0/71     PNNI     UP
```

Step 4 Configure the network interworking soft PVC from Switch A beginning in global configuration mode.

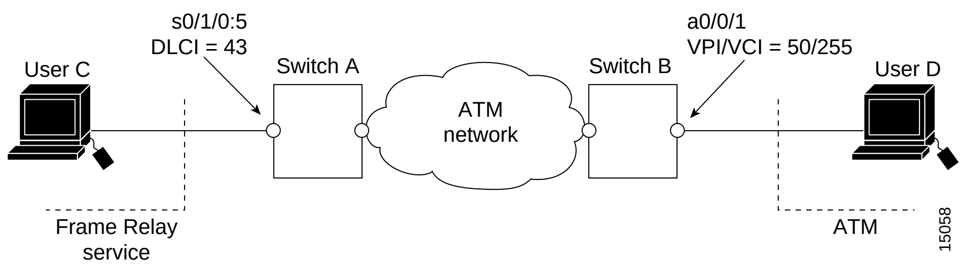
```
Switch-A(config)# interface serial0/1/0:5
Switch-A(config-if)# frame-relay soft-vc 43 dest-address
47.0091.8100.0000.00e0.1e19.9904.4000.0c80.0010.00 vc 50 255 network
```

After you complete the soft VC configuration, go to the "Display Frame Relay Interworking Soft PVCs" section on page 19-35 and verify the connection.

Configuring Frame Relay to ATM Service Interworking Soft PVCs

This section describes configuring a Frame Relay to ATM service interworking soft PVC terminating on an ATM interface. Figure 19-9 shows a Frame Relay to ATM service interworking soft PVC between Switch A and Switch B.

Figure 19-9 Frame Relay to ATM Service Interworking Soft PVC Example



To configure a Frame Relay to ATM service interworking soft PVC, perform the following steps, beginning in EXEC mode:

	Command	Purpose
Step 1	Switch# **show interfaces**	Determines source and destination interfaces.
Step 2	Switch# **show vc interface serial** *card*/*subcard*/*port***:***cgn* [*dlci*]	Determines the DLCI available for Step 3.
Step 3	Switch# **show vc interface serial** *card*/*subcard*/*port***:***cgn* [*dlci*]	Determines the DLCI available for Step 7.
Step 4	Switch# **show atm addresses**	Determines the soft PVC destination address.
Step 5	Switch# **configure terminal** Switch(config)#	From the source (active) side, at the privileged EXEC prompt, enter configuration mode from the terminal.
Step 6	Switch(config)# **interface serial** *card*/*subcard*/*port***:***cgn* Switch(config-if)#	Selects the Frame Relay serial port and channel group number.
Step 7	Switch(config-if)# f**rame-relay soft-vc** *dlci_a* **dest-address** *address* **vc** *vpi vci* [**upc** {**pass** | **drop**}] [**rx-cttr** *index*] [**tx-cttr** *index*] [**retry-interval** [**first** *first-retry-interval*] [**maximum** *max-retry-interval*]] [**service** [**translation** | **transparent**]] [**clp-bit** {**0** | **1** | **map-de**}] [**de-bit** {**0** | **1** | **map-clp**}] [**efci-bit** {**0** | **map-fecn**}] [**explicit-path precedence** {**name** *path-name* | **identifier** *path-id*} [**upto** *partial-entry-index*]] [**only-explicit**]	Configures a service interworking soft PVC.

Note The row index for **rx-cttr** and **tx-cttr** must be configured before using this optional parameter. See the "Configuring the Connection Traffic Table" section on page 8-10.

Note If the interworking soft PVC terminates on an ATM interface, the default interworking type is service interworking in translation mode.

Frame Relay to ATM Service Interworking Soft PVC Configuration Example

Use the following steps to configure the service interworking soft PVC between Switch A and switch B as shown in Figure 19-9.

Note In the following process the source (active) side is serial interface 0/1/0:5 on Switch A and the destination (passive) side is ATM interface 0/0/1 on Switch B.

Step 1 On Switch A, use the **show vc interface serial** command to determine that DLCI 43 is available for use on serial interface 0/1/0:5 Switch A:

```
Switch-A# show vc interface serial 0/1/0:5
Interface      Conn-Id Type    X-Interface    X-Conn-Id  Encap  Status
Serial0/1/0:5     54  SoftVC   Serial3/0/0:3    54       SoftVC  UP
Serial0/1/0:5     55  SoftVC   Serial3/0/0:2    55       SoftVC  UP
Serial0/1/0:5     56  SoftVC   ATM0/1/3        0/45      SVC     UP
Serial0/1/0:5     66  SoftVC   ATM1/1/0        0/100     SoftVC  UP
```

Step 2 On Switch B, use the **show atm addresses** command to determine the destination ATM address for ATM interface 0/0/1, which is 47.0091.8100.0000.00e0.1e19.9904.4000.0c80.0010.00.

```
Switch-B# show atm addresses
Switch Address(es):
47.00918100000000E01E199904.00E01E808601.00 active
Soft VC Address(es) :
47.0091.8100.0000.00e0.1e19.9904.4000.0c80.0000.00 ATM0/0/0
➔ 47.0091.8100.0000.00e0.1e19.9904.4000.0c80.0010.00 ATM0/0/1
47.0091.8100.0000.00e0.1e19.9904.4000.0c80.0020.00 ATM0/0/2
47.0091.8100.0000.00e0.1e19.9904.4000.0c80.0030.00 ATM0/0/3
<information deleted>
```

Step 3 On Switch B, use the **show vc interface atm** command to determine that VPI/VCI 50/255 is available for use on ATM interface 0/0/1:

```
Switch-B# show vc interface atm 0/0/1
Interface    Conn-Id    Type   X-Interface     X-Conn-Id   Encap  Status
ATM0/0/1      0/5       PVC     ATM2/0/0          0/58     QSAAL    UP
ATM0/0/1      0/16      PVC     ATM2/0/0          0/44     ILMI     UP
ATM0/0/1      0/18      PVC     ATM2/0/0          0/71     PNNI     UP
```

Step 4 The following example configures a service interworking soft PVC in transparent mode on Switch A using the information obtained in the previous steps:

```
Switch-A(config)# interface serial 0/1/0:5
Switch-A(config-if)# frame-relay soft-vc 43 dest-address
47.0091.8100.0000.00e0.1e19.9904.4000.0c80.0010.00 vc 50 255 service transparent
```

After you complete the soft VC configuration, go to the "Display Frame Relay Interworking Soft PVCs" section on page 19-35 and verify the connection.

Display Frame Relay Interworking Soft PVCs

To display your Frame Relay interworking soft PVCs configuration, use the following EXEC command:

Command	Purpose
show vc [**interface** {**atm** *card*/*subcard*/*port* [*vpi vci*] | **serial** *card*/*subcard*/*port*:*cgn* [*dlci*]}]	Shows the PVC interface configuration.

Examples

The following example displays serial interface 1/1/0:2 soft PVC status:

```
Switch# show vc interface serial 1/1/0:2
Interface      Conn-Id    Type   X-Interface     X-Conn-Id   Encap  Status
Serial1/1/0:2     34      SoftVC  ATM0/0/0        100/255            UP
```

The following example displays ATM interface 0/0/0 soft PVC status:

```
Switch# show vc interface atm 0/0/0
Interface      Conn-Id    Type   X-Interface     X-Conn-Id   Encap  Status
ATM0/0/0          0/5       PVC     ATM2/0/0          0/43     QSAAL    UP
ATM0/0/0          0/16      PVC     ATM2/0/0          0/35     ILMI     UP
ATM0/0/0          0/200     PVC     ATM0/0/1          0/200             DOWN
ATM0/0/0        100/255     SoftVC  Serial1/1/0:2       34                UP
```

Configuring the Soft PVC Route Optimization Feature

This section describes the soft permanent virtual channel (soft PVC) route optimization feature for Frame Relay interfaces. Most soft PVCs have a much longer lifetime than switched virtual channels (SVCs). The route chosen during the soft connection setup remains the same even though the network topology might change.

Soft connections, with the route optimization percentage threshold set, provide the following features:

- When a better route is available, soft permanent virtual paths (soft PVPs) or soft PVCs are dynamically rerouted.
- Route optimization can be triggered manually.

Note Soft PVC route optimization should not be configured with constant bit rate (CBR) connections.

Configuring a Frame Relay Interface with Route Optimization

Soft PVC route optimization must be enabled and configured to determine the point at which a better route is found and the old route is reconfigured.

To enable and configure a Frame Relay interface with route optimization, perform the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **atm route-optimization percentage-threshold** *value*	Configures the ATM route optimization threshold.
Step 2	Switch(config)# **interface serial** *card*/*subcard*/*port*:*cgn* Switch(config-if)#	Selects the interface to configure. Enter the interface number of the source end of the soft PVC. Route optimization works for the source end of a soft PVC only and is ignored if configured on the destination interface.
Step 3	Switch(config-if)# **atm route-optimization soft-connection** [**interval** *minutes*] [**time-of-day** {**anytime** | *start-time end-time*}]	Configures the interface for route optimization.

Example

The following example shows how to configure an interface with a route optimization interval configured as every 30 minutes between the hours of 6:00 P.M. and 5:00 A.M.:

```
Switch(config)# atm route-optimization percentage-threshold 45
Switch(config)# interface serial 1/0/0:1
Switch(config-if)# atm route-optimization soft-connection interval 30 time-of-day 18:00 5:00
```

Displaying a Frame Relay Interface Route Optimization Configuration

To display the Frame Relay interface route optimization configuration, use the following privileged EXEC commands:

Command	Purpose
show running-config	Shows the serial interface configuration route optimization configuration.
show interfaces [**serial** *card*/*subcard*/*port*:*cgn*]	Shows the serial interface configuration.

Example

The following example shows the route optimization configuration of serial interface 1/0/0:1:

```
Switch# show running-config
Building configuration...

<information deleted>

!
interface Serial1/0/0:1
 description Engineering connections
 no ip address
 no ip directed-broadcast
 encapsulation frame-relay IETF
 no arp frame-relay
 no snmp trap link-status
 frame-relay intf-type nni

atm route-optimization soft-connection interval 30 time-of-day 18:0 5:0
!

Switch# show interfaces serial 3/0/0:1
Serial3/0/0:1 is up, line protocol is up
  Hardware is FRPAM-SERIAL
  MTU 4096 bytes, BW 1536 Kbit, DLY 0 usec, rely 128/255, load 1/255
  Encapsulation FRAME-RELAY IETF, loopback not set, keepalive not set
  Last input 00:00:08, output never, output hang never
  Last clearing of "show interface" counters never
  Input queue: 0/75/0 (size/max/drops); Total output drops: 0
  Queueing strategy: weighted fair
  Output queue: 0/1000/64/0 (size/max total/threshold/drops)
     Conversations  0/0/256 (active/max active/max total)
     Reserved Conversations 0/0 (allocated/max allocated)
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
     12963 packets input, 12963 bytes, 0 no buffer
     Received 0 broadcasts, 0 runts, 0 giants, 0 throttles
     12963 input errors, 7638 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
     0 packets output, 0 bytes, 0 underruns
     0 output errors, 0 collisions, 0 interface resets
     0 output buffer failures, 0 output buffers swapped out
     2 carrier transitions
  Timeslots(s) Used: 1-24 on T1 1
  Frames Received with:
   DE set: 0, FECN set :0, BECN set: 0
  Frames Tagged :
   DE: 0, FECN: 0 BECN: 0
  Frames Discarded Due to Alignment Error: 0
  Frames Discarded Due to Illegal Length: 0
  Frames Received with unknown DLCI: 0
  Frames with illegal Header : 0
  Transmit Frames with FECN set :0,  BECN Set :0
  Transmit Frames Tagged FECN : 0 BECN : 0
  Transmit Frames Discarded due to No buffers : 0
  Default Upc Action : tag-drop
  Default Bc (in Bits) : 32768
➔ Soft vc route optimization is enabled
    Soft vc route optimization interval = 50 minutes
    Soft vc route optimization time-of-day range = (20:10 - 23:40)
```

Respecifying Existing Frame Relay to ATM Interworking Soft PVCs

For existing Frame Relay to ATM interworking soft permanent virtual channels (soft PVCs), a connection is disabled to prevent an explicit path from being used for routing while it is reconfigured. The **redo_explicit** keyword is used to allow respecifying of the explicit path configuration without bringing down connections. Existing connections remain unaffected unless a reroute takes place. If rerouting occurs, the new explicit path configuration takes affect.

To enable or disable soft PVC and respecify explicit-path configuration, use the following interface command:

Command	Purpose
frame-relay soft-vc *dlci_a* [**enable** | **disable**] [**redo-explicit** [**explicit-path** *precedence* {**name** *path-name* | **identifier** *path-id*} [**upto** *partial-entry-index*]] [**only-explicit**]]	Respecifies the explicit path on a Frame Relay to ATM interworking soft PVC.

CHAPTER 20

Configuring IMA Port Adapter Interfaces

This chapter describes inverse multiplexing over ATM (IMA) and the steps required to configure the IMA port adapters in the Catalyst 8540 MSR, Catalyst 8510 MSR, and LightStream 1010 ATM switch routers. These port adapters group multiple low-speed links into one larger virtual trunk or IMA group.

Note This chapter provides advanced configuration instructions for the Catalyst 8540 MSR, Catalyst 8510 MSR and LightStream 1010 ATM switch routers. For complete descriptions of the commands mentioned in this chapter, refer to the *ATM Switch Router Command Reference* publication. For hardware installation and cabling instructions, refer to the *ATM and Layer 3 Port Adapter and Interface Module Installation Guide*.

For more information on how to configure your IMA-specific network equipment, refer to the Cisco IOS publications on the Documentation CD-ROM.

This chapter includes the following sections:

Note IMA is only possible on switches with FC-PFQ installed.

Overview of IMA

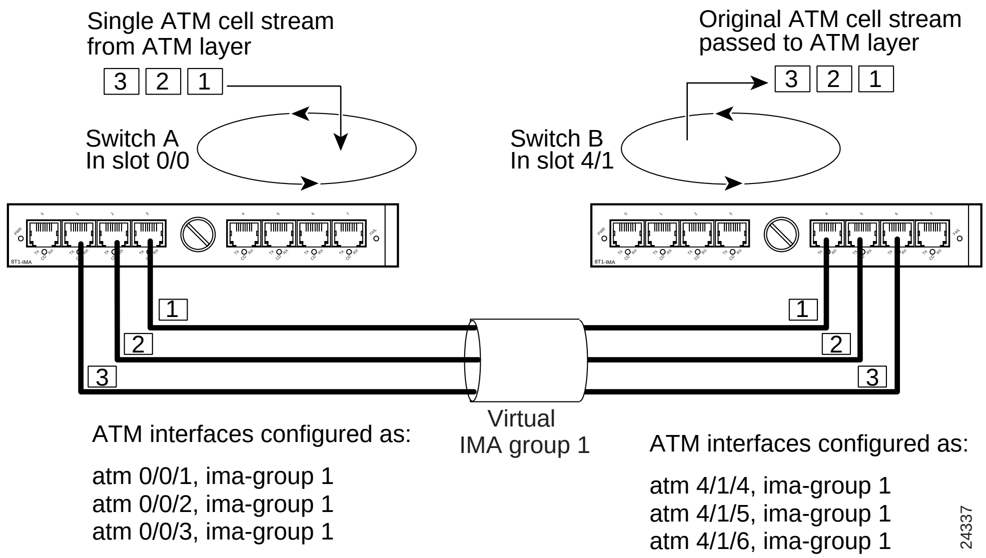
IMA allows you to aggregate multiple low-speed links into one larger virtual trunk or IMA group. An inverse multiplexer appears to your ATM switch router as one logical pipe. This IMA group provides modular bandwidth for user access to ATM networks for connections between ATM network elements at rates between the traditional order multiplex levels, such as between T1 or E1 and T3 or E3.

IMA involves inverse multiplexing and demultiplexing of ATM cells in a cyclical fashion among links grouped to form a higher bandwidth logical group with a rate approximately the sum of the link rates. This group of links is called an IMA group.

Inverse multiplexing in the transmit direction controls the distribution of cells onto the group of physical links available to the IMA group interface. It also handles differential delays and deals with links that are added or dropped, or fail and are later restored. In the receive direction, the IMA interface performs differential delay compensation and recombines the cells into the original ATM cell stream while allowing minimal cell delay variation (CDV). The IMA process of splitting and recombining the ATM cell stream is as transparent to the layer above as a traditional single-link physical layer interface.

Figure 20-1 illustrates the configuration of the T1 IMA port adapters (with eight ports each) on two switches which create a virtual IMA group connection.

Figure 20-1 IMA Grouping Example



IMA groups terminate at each end of the IMA virtual link. The transmit IMA receives the ATM cell stream from the ATM layer and distributes it on a cell-by-cell basis across the multiple T1 or E1 links within the IMA group. At the far-end, the receiving IMA recombines the cells from each link, also on a cell-by-cell basis, recreating the original ATM cell stream. The aggregate cell stream is then passed to the ATM layer.

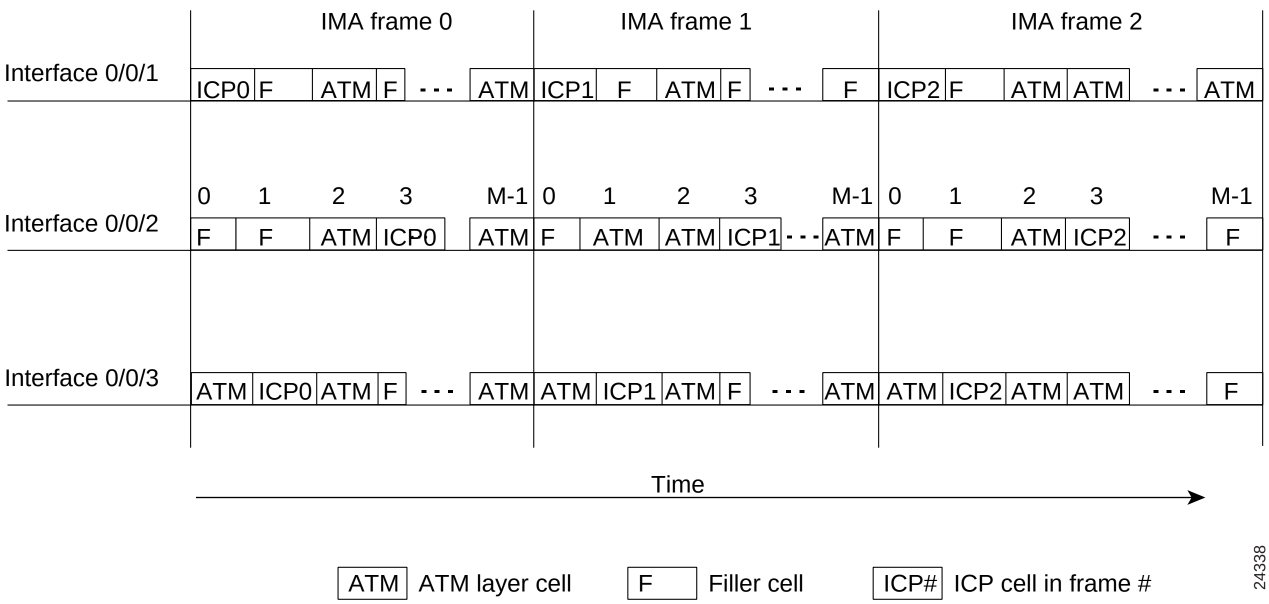
The IMA frame is the unit of control in the IMA protocol. An IMA frame is a series of consecutive cells. Periodically, the transmit IMA sends special cells that permit reconstruction of the ATM cell stream at the receiving IMA. These cells, defined as IMA Control Protocol (ICP) cells, provide the definition of an IMA frame. The transmitter must align the transmission of IMA frames on all links (shown in Figure 20-2) to allow the receiver to adjust for differential link delays among the constituent physical links. Based on this required behavior, the receiver can detect the differential delays by measuring the arrival times of the IMA frames on each link.

The transmitting end sends cells continuously. If no ATM layer cells are sent between ICP cells within an IMA frame, the transmit IMA sends filler cells to maintain a continuous stream of cells at the physical layer. Filler cells, which provide cell rate decoupling at the IMA sublayer, are discarded by the receiving IMA.

A new OAM cell is defined for use by the IMA protocol. This cell has codes that define it as either an ICP cell or a filler cell.

Within the IMA frame, the ICP cell appears at the ICP cell offset position, which can vary among the links. Figure 20-2 shows an example of the transmission of IMA frames over three links. On interface 0/0/1, the ICP cells have their cell offset set to 0 and are the first cells in each IMA frame. On interface 0/0/2, the ICP cells have the ICP cell offset set to 3 and are the fourth cells in each IMA frame. On interface 0/0/3, the ICP cells have their ICP cell offset set to 1 and are the second cells in each IMA frame.

Figure 20-2 IMA Frames



Note These ICP cells are distributed more evenly over the IMA frame but are shown closer for illustration purposes. Within an IMA frame, the ICP cells on all links have the same IMA frame sequence number.

Configuring the T1/E1 IMA Port Adapter

The T1/E1 IMA port adapter provides eight physical ports. Each port adapter supports up to four IMA groups and independent ATM interfaces. The following are possible combinations:

- Four IMA groups
- Three IMA groups and one independent ATM interface
- Two IMA groups and two independent ATM interfaces
- One IMA group and three independent ATM interfaces
- No IMA group and four independent ATM interfaces

The T1 line operates at 1.544 Mbps, which is equivalent to 24 time slots (DS0 channels). The T1 time slot provides usable bandwidth of *n* x 64 kbps, where *n* is the time slot from 1 to 24. The E1 line operates at 2.048 Mbps.

T1/E1 IMA port adapters support interface overbooking. For configuration information, see the "Configuring Interface Overbooking" section on page 8-37.

Note By default, T1/E1 IMA interfaces are shut down when the port adapter is installed.

Default T1/E1 IMA Interface Configuration

The following defaults are assigned to all T1/E1 IMA port adapter interfaces:

- Clock source = system clock
- Transmit clock source = network derived
- Loopback = no loopback
- BERT = disabled

The following port adapter types have specific defaults assigned.

T1 port adapter:

- Framing = extended super frame (ESF)
- Line build-out (LBO) = short 133
- Linecode = b8zs
- Facilities Data Link (FDL) = no FDL
- Yellow = enabled

E1 port adapter:

- Framing = pcm30adm
- Line build-out (LBO) = short gain12 22db
- Linecode = hdb3
- National bits = 1 1 1 1 1 1

The following defaults are assigned to all IMA groups:

- Minimum number of active links = 1
- Clock mode = common
- Differential delay = 25 milliseconds
- Frame length = 128 cells
- Test link = first link in the group
- Test pattern = value of test link

Configuring the T1/E1 IMA Interface

To manually change any of your default configuration values, perform the following steps, beginning in global configuration mode:

Note IMA is only possible on switches with FC-PFQ installed.

	Command	Purpose
Step 1	Switch(config)# **interface atm** *card*/*subcard*/*port* Switch(config-if)#	Specifies the ATM interface and enters interface configuration mode.
Step 2	Switch(config-if)# **bert pattern** {**2^15** | **2^20** | **2^23** | **0s** | **1s** | **2^11** | **2^20-QRSS** | **alt-0-1**} **interval** *minutes*	Configures the bit error rate test pattern.
Step 3	Switch(config-if)# **clock source** {**free-running** | **loop-timed** | **network-derived**}	Configures the type of clocking.
Step 4	Switch(config-if)# **framing** {**esfadm** | **sfadm**}	Modifies the T1 IMA framing type.
	Switch(config-if)# **framing** {**cleare1** | **crc4adm** | **pcm30adm**}	Modifies the E1 IMA framing type.
Step 5	Switch(config-if)# **lbo** {**long** {**gain26** | **gain36**} {**-15db** | **-22.5db** | **-7.5db** | **0db**}} | {**short** {**133ft** | **266ft** | **399ft** | **533ft** | **655ft**}}	Modifies the T1IMA line build-out.
	Switch(config-if)# **lbo** {**long gain43** {**120db** | **75db**} | **short gain12 22db**}	Modifies the E1 IMA line build-out.
Step 6	Switch(config-if)# **loopback** {**cell** | **diagnostic** | **line** | **local** | **payload** | **pif** | **remote** {**line** {**inband** | **fdl** {**ansi** | **bellcore**}} | **payload** [**fdl ansi**]}}	Configures the T1 line loopback.
	Switch(config-if)# **loopback** {**cell** | **diagnostic** | **line** | **payload** | **pif**}	Configures the E1 line loopback.
Step 7	Switch(config-if)# **linecode** {**ami** | **b8zs**}	Modifies the T1 line code format.
	Switch(config-if)# **linecode** {**ami** | **hdb3**}	Modifies the E1 line code format.
Step 8	Switch(config-if)# **fdl** {**ansi** | **att**}	Configures T1 FDL format.
Step 9	Switch(config-if)# **yellow** {**detection** | **generation**}	Enables T1 yellow alarm detection.
Step 10	Switch(config-if)# **national reserve** *bit-pattern*	Modifies the E1 national bits.

Example

The following example shows how to change the clock source to free running:

```
Switch(config)# interface atm 0/0/3
Switch(config-if)# clock source free-running
```

Displaying the T1/E1 IMA Interface Configuration

To display the physical T1/E1 IMA interface configuration, use the following EXEC command:

Command	Purpose
show controllers atm *card*/*subcard*/*port*	Displays the physical interface configuration and status.

Example

The following example shows a T1 IMA ATM interface 0/0/3 configuration, including the change to the clock source configuration from the previous section:

```
Switch# show controller atm 0/0/3
ATM0/0/3 is up
        PAM State is UP
        Firmware Version: 1.6
        FPGA Version : 1.2
        Boot version : 1.2
Port type: T1    Port rate: 1.5 Mbps    Port medium: UTP
Port status:Good Signal    Loopback:None    Flags:8000
fdl is DISABLED
Yellow alarm enabled in both tx and rx
linecode is B8ZS
TX Led: Traffic Pattern    RX Led: Traffic Pattern   CD Led: Green
TX clock source:  free-running
T1 Framing Mode:  ESF ADM format
LBO (Cablelength) is short 133
Counters:
    Key: txcell  - # cells transmitted
       rxcell    - # cells received
       hcs       - # uncorrectable HEC errors
       chece     - # rx Correctable HEC errors
       uicell    - # unassigned/idle cells dropped
       oocd      - # rx out of cell deliniation
       rx_fovr   - # rx FIFO over run
       tx_fovr   - # tx FIFO over run
       coca      - # tx Change of cell allignment
       pcv       - # path code violations
       lcv       - # line code violations
       es        - #
--More--
```

Configuring IMA Group Functions

To configure IMA group functions on an ATM switch router, perform the tasks in the following sections:

- Creating an IMA Group Interface, page 20-7
- Adding an Interface to an Existing IMA Group, page 20-8
- Deleting an Interface from an IMA Group, page 20-10
- Deleting an IMA Group, page 20-11

Creating an IMA Group Interface

To create an IMA group interface, first link a physical interface to the IMA group. After configuring the physical interface as part of an IMA group, you can then create the IMA group interface. An IMA group interface is identified by its card, subcard, and IMA group number. For example, IMA group 1 configured on the physical interface card 0 and subcard 0 is identified as 0/0/ima1. IMA group numbers range from 0 to 3.

Note You must create the IMA group at both ends of the connection.

To create an IMA group interface at both ends of the connection, perform the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **interface atm** *card*/*subcard*/*port* Switch(config-if)#	Specifies the ATM port and enters interface configuration mode.
Step 2	Switch(config-if)# **shutdown**	Shuts down the interface prior to configuring the IMA group.
Step 3	Switch(config-if)# **ima-group** *number*	Assigns the interface to an IMA group number.
Step 4	Switch(config-if)# **no shutdown**	Reenables the interface.
Step 5	Switch(config-if)# **exit** Switch(config)#	Returns to global configuration mode.
Step 6	Switch(config)# **interface atm** *card*/*subcard*/**ima***group* Switch(config-if)#	Specifies the IMA group 0 to 3 and enters interface configuration mode.
Step 7	Switch(config-if)# **no shutdown**	Creates the IMA group.
Step 8	—	Repeat this procedure on the other end of the connection.

Note The IMA group numbers on each end of the interface can differ. For example, you can configure the interfaces in IMA group 1 on Switch A and in IMA group 2 on Switch B.

Example

The following example shows how to create the IMA group interface 0/0/ima1 shown in Figure 20-1 starting with Switch A, ATM interface 0/0/1:

```
SwitchA(config)# interface atm 0/0/1
SwitchA(config-if)# shutdown
SwitchA(config-if)# ima-group 1
SwitchA(config-if)# no shutdown
SwitchA(config-if)# exit
SwitchA(config)# interface atm 0/0/ima1
SwitchA(config-if)# no shutdown
```

The following example shows how to create the IMA group interface 4/1/ima1 shown in Figure 20-1 on Switch B, ATM interface 4/1/4:

```
SwitchB(config)# interface atm 4/1/4
SwitchB(config-if)# shutdown
SwitchB(config-if)# ima-group 1
SwitchB(config-if)# no shutdown
SwitchB(config-if)# exit
SwitchB(config)# interface atm 4/1/ima1
SwitchB(config-if)# no shutdown
```

Adding an Interface to an Existing IMA Group

An interface can be added to an existing IMA group link by assigning the IMA group number.

Note You must configure the IMA group at both ends of the physical connection.

To configure the interfaces at both ends of the connection as members of an existing IMA group, perform the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **interface atm** *card/subcard/port* Switch(config-if)#	Specifies the ATM port and enters interface configuration mode.
Step 2	Switch(config-if)# **shutdown**	Prior to configuring the IMA group, shuts down the interface.
Step 3	Switch(config-if)# **ima-group** *number*	Assigns the interface to an IMA group number.
Step 4	Switch(config-if)# **no shutdown**	Reenables the interface.
Step 5	—	Repeat this procedure on the other end of the connection.

Note You can use the **ima-group** command to move an interface from one IMA group to another.

Examples

The following example shows how to configure ATM interface 0/0/2 on Switch A as part of the IMA group 1 shown in Figure 20-1:

```
SwitchA(config)# interface atm 0/0/2
SwitchA(config-if)# shutdown
SwitchA(config-if)# ima-group 1
SwitchA(config-if)# no shutdown
```

The following example shows how to configure ATM interface 4/1/5 on Switch B as part of the IMA group 1 shown in Figure 20-1:

```
SwitchB(config)# interface atm 4/1/5
SwitchB(config-if)# shutdown
SwitchB(config-if)# ima-group 1
SwitchB(config-if)# no shutdown
```

The following example shows how to move ATM interface 4/1/5 on Switch B to the IMA group 3:

```
SwitchB(config)# interface atm 4/1/5
SwitchA(config-if)# shutdown
SwitchB(config-if)# ima-group 3
SwitchB(config-if)# no shutdown
```

Displaying the IMA Group Configuration

To display the IMA group configuration, use the following EXEC commands:

Command	Purpose
show ima interface [**atm** *card*/*subcard*/**ima***group* [**detailed**]]	Displays IMA group interface configuration and status.
show interfaces atm *card*/*subcard*/**ima***group*	Displays IMA interface configuration and status.

Example

The following example shows the IMA group interface configuration for IMA group 0/0/ima1 interface:

```
SwitchA# show ima interface atm 0/0/ima1
ATM0/0/ima1 is up
        Group Index    = 2
        State: NearEnd = operational, FarEnd = operational
        FailureStatus  = noFailure
IMA Group Current Configuration:
        MinNumTxLinks = 1    MinNumRxLinks = 1
        DiffDelayMax  = 25   FrameLength   = 128
        NeTxClkMode   = common(ctc)  CTC_Reference_Link = ATM0/0/3
        TestLink      = 3       Testpattern      = Not Specified
        TestProcStatus     = disabled   GTSM change timestamp = 990426154350
IMA Link Information:
Link          Physical Status        NearEnd Rx Status       Test Status
-----         ---------------        -----------------      ---------------
ATM0/0/2      up                     active                 disabled
ATM0/0/3      up                     active                 disabled
```

The following example shows the interface configuration for T1 IMA group 0/0/ima1:

```
SwitchA# show interfaces atm 0/0/ima1
ATM0/0/ima1 is up, line protocol is up
  Hardware is imapam_t1_ima
  MTU 4470 bytes, sub MTU 4470, BW 1500 Kbit, DLY 0 usec, rely 255/255, load 1/255
  Encapsulation ATM, loopback not set, keepalive not supported
  Last input 00:00:00, output 00:00:00, output hang never
  Last clearing of "show interface" counters never
  Input queue: 0/75/0 (size/max/drops); Total output drops: 0
  Queueing strategy: weighted fair
  Output queue: 0/1000/64/0 (size/max total/threshold/drops)
     Conversations  0/0/256 (active/max active/max total)
     Reserved Conversations 0/0 (allocated/max allocated)
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
     223 packets input, 11819 bytes, 0 no buffer
     Received 0 broadcasts, 0 runts, 0 giants, 0 throttles
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
     215 packets output, 11395 bytes, 0 underruns
     0 output errors, 0 collisions, 1 interface resets
     0 output buffer failures, 0 output buffers swapped out
```

The following example shows the ATM layer interface configuration of the T1 IMA group 0/0/ima1:

```
SwitchA# show atm interface atm 0/0/ima1

Interface:      ATM0/0/ima1     Port-type:      imapam_t1_ima
IF Status:      UP              Admin Status:   up
Auto-config:    enabled         AutoCfgState:   completed
IF-Side:        Network         IF-type:        NNI
Uni-type:       not applicable  Uni-version:    not applicable
Max-VPI-bits:   8               Max-VCI-bits:   14
Max-VP:         255             Max-VC:         16383
ConfMaxSvpcVpi: 255             CurrMaxSvpcVpi: 255
ConfMaxSvccVpi: 255             CurrMaxSvccVpi: 255
ConfMinSvccVci: 35              CurrMinSvccVci: 35
Svc Upc Intent: pass            Signalling:     Enabled
ATM Address for Soft VC: 47.0091.8100.0000.0040.0b0a.2a81.4000.0c80.0090.00
Configured virtual links:
  PVCLs SoftVCLs   SVCLs   TVCLs   PVPLs SoftVPLs   SVPLs Total-Cfgd Inst-Conns
      3        0       0       0       0        0       0          3          3
Logical ports(VP-tunnels):     0
Input cells:    105             Output cells:   109
5 minute input rate:             0 bits/sec,       0 cells/sec
5 minute output rate:            0 bits/sec,       0 cells/sec
Input AAL5 pkts: 58, Output AAL5 pkts: 60, AAL5 crc errors: 0
```

Deleting an Interface from an IMA Group

To delete an interface from an IMA group, perform the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **interface atm** *card*/*subcard*/*port* Switch(config-if)#	Specifies the ATM port and enters interface configuration mode.
Step 2	Switch(config-if)# **no ima-group**	Deleted the interface from an IMA group number.

Example

The following example shows how to delete an interface from an IMA group:

```
Switch(config)# interface atm 0/0/1
Switch(config-if)# no ima-group
```

Confirming the Interface Deletion

To confirm the interface deletion from the IMA group, use the following EXEC command:

Command	Purpose
show ima interface atm *card*/*subcard*/*port*	Displays IMA group interface configuration and status.

Example:

The following example shows how to verify that the interface is deleted from the IMA group:

```
SwitchA# show ima interface atm 0/0/1
ATM0/0/1 is not a part of IMA group
```

Deleting an IMA Group

To delete an IMA group, use the following global configuration command:

Command	Purpose
no interface atm *card*/*subcard*/**ima***group*	Deletes the IMA group from the T1/E1 IMA interface.

Note When you delete an IMA group, the interfaces remain configured as members of the IMA group. When you recreate the IMA group, the member interfaces reinitialize automatically.

Example

The following example shows how to delete ATM interface 0/0/ima1 and administratively shut down the member interfaces:

```
Switch(config)# no interface atm 0/0/ima1
```

Confirming the IMA Group Deletion

To confirm the IMA group deletion, perform the following steps in user EXEC mode:

Command	Purpose
show ima interface [**atm** *card*/*subcard*/**ima***group* [**detailed**]]	Displays IMA group interface configuration and status.

Example

The following example shows how to verify that the interface is deleted from the IMA group:

```
Switch# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)# interface atm 0/0/2
Switch(config-if)# shut
Switch(config-if)# ima-group 0
Switch(config-if)# no shut
Switch(config-if)# exit
Switch(config)# interface atm 0/0/ima0
Switch(config-if)# no shut
Switch(config-if)# end
Switch# show ima interface atm 0/0/ima0
ATM0/0/ima0 is up
        Group Index    = 5
        State: NearEnd = operational, FarEnd = operational
        FailureStatus  = noFailure
IMA Group Current Configuration:
        MinNumTxLinks = 1    MinNumRxLinks = 1
        DiffDelayMax  = 25   FrameLength   = 128
        NeTxClkMode   = common(ctc)  CTC_Reference_Link = ATM0/0/2
        TestLink      = 2       Testpattern      = Not Specified
        TestProcStatus     = disabled   GTSM change timestamp = 000210165420
IMA Link Information:
Link           Physical Status        NearEnd Rx Status       Test Status
-----          ---------------        -----------------      ---------------
ATM0/0/2       up                     active                 disabled
Switch# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)# interface atm 0/0/ima0
Switch(config-if)# end
Switch(config)# no interface atm 0/0/ima0
Switch(config)# exit
Switch# show ima interface atm 0/0/ima0
                           ^
% Invalid input detected at '^' marker.

Switch#
```

Configuring IMA Group Parameters

This section describes how to configure inverse multiplexing over ATM (IMA) group parameters after configuring an IMA group at the interface level. These tasks include configuring active minimum links, interface clock mode, link differential delay, frame length, and test pattern.

Configuring IMA Group Minimum Active Links

You can configure an IMA group to require a minimum number of active links. This number is the minimum number of links required for the IMA group to become operational and provides a guaranteed minimum bandwidth. For example, if the **active-minimum-links** command number is configured as 3, the minimum number of active links necessary for the IMA group to be active is three and the minimum bandwidth available is approximately 3 x T1 speed.

To configure the minimum active links on the IMA group, perform the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **interface atm** *card*/*subcard*/**ima***group* Switch(config-if)#	Specifies the IMA group to configure and enters interface configuration mode.
Step 2	Switch(config-if)# **ima active-links-minimum** *number*	Specifies the minimum number of active links for an IMA group.

Note Only when the minimum number of links are active in the IMA group does the group come up. The IMA group remains down if the IMA group has fewer active links than the minimum number of active links configured.

Example

The following example shows how to configure the minimum number of active links that must be up for the IMA group to function as 3:

```
SwitchA(config)# interface atm 0/0/ima1
SwitchA(config-if)# ima active-links-minimum 3
```

Displaying the IMA Group Minimum Active Links Configuration

To display the IMA group minimum active links configuration, use the following EXEC command:

Command	Purpose
show ima interface [**atm** *card*/*subcard*/**ima***group* [**detailed**]]	Displays IMA group interface configuration and status.

Example

The following example shows the IMA group interface minimum active links configuration:

```
SwitchA# show ima interface
ATM0/0/ima1 is up
        Group Index    = 5
        State: NearEnd = operational, FarEnd = operational
        FailureStatus  = noFailure
IMA Group Current Configuration:
➔       MinNumTxLinks = 3   MinNumRxLinks = 3
        DiffDelayMax  = 25   FrameLength   = 128
        NeTxClkMode   = common(ctc)  CTC_Reference_Link = ATM0/0/2
        TestLink      = 2       Testpattern      = Not Specified
        TestProcStatus     = disabled   GTSM change timestamp = 990427165502
IMA Link Information:
Link           Physical Status        NearEnd Rx Status       Test Status
-----          ---------------        -----------------      ---------------
ATM0/0/2       up                     active                 disabled
ATM0/0/3       up                     active                 disabled
ATM0/0/4       up                     active                 disabled
ATM0/0/5       up                     active                 disabled
```

Configuring IMA Group Interface Clock Mode

The links configured as part of a IMA group interface can derive their clocking from one single clock source using common transmit clocking (CTC) mode, or the link clocking can be derived individually from different clock sources using independent transmit clocking (ITC) mode. For example, if three interfaces are configured as members of an IMA group interface, one can be configured to use the reference clock, and the remaining links can derive their clocking from the local oscillator.

To configure the clocking mode on the IMA group, perform the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **interface atm** *card*/*subcard*/**ima***group* Switch(config-if)#	Specifies the IMA group to configure and enters interface configuration mode.
Step 2	Switch(config-if)# **ima clock-mode** {**common** | **independent**}	Specifies the transmit clock mode for the IMA group.

Example

The following example shows how to configure the IMA group clocking mode as independent:

```
SwitchA(config)# interface atm 0/0/ima1
SwitchA(config-if)# ima clock-mode independent
```

Displaying the IMA Group Interface Clock Mode Configuration

To display the IMA group transmit clock mode configuration, use the following EXEC command:

Command	Purpose
show ima interface [**atm** *card*/*subcard*/**ima***group* [**detailed**]]	Displays IMA group interface configuration and status.

Example

The following example shows the IMA group clock mode configuration:

```
SwitchA# show ima interface
ATM0/0/ima1 is up
        Group Index    = 4
        State: NearEnd = operational, FarEnd = operational
        FailureStatus  = noFailure
IMA Group Current Configuration:
        MinNumTxLinks = 1    MinNumRxLinks = 1
        DiffDelayMax  = 25   FrameLength   = 128
➔        NeTxClkMode   = independent(itc)
        TestLink      = 3       Testpattern      = Not Specified
        TestProcStatus     = disabled   GTSM change timestamp = 990427121150
IMA Link Information:
Link          Physical Status        NearEnd Rx Status       Test Status
-----         ---------------        -----------------      ---------------
ATM0/0/2      up                     active                 disabled
ATM0/0/3      up                     active                 disabled
```

Configuring IMA Group Link Differential Delay

The transmitter on the T1/E1 IMA port adapter must align the transmission of IMA frames on all links as shown in Figure 20-2. Alignment allows the receiver to adjust for differential delays among the members of the IMA group. Based on this required behavior, the receiver can detect the differential delays by measuring the arrival times of the IMA frames on each link.

The transmitting end of the IMA group connection sends cells continuously. If there are no ATM layer cells to send between ICP cells within an IMA frame, the transmit IMA sends filler cells to maintain a continuous stream of cells at the physical layer.

The receiving end of the IMA group connection must allocate sufficient buffer space to compensate for the differential delay between the member links. The maximum differential delay value configured for the IMA group determines the size of these buffers.

To configure the maximum differential delay allowed in the IMA group, perform the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **interface atm** *card*/*subcard*/**ima***group* Switch(config-if)#	Specifies the IMA group and enters interface configuration mode.
Step 2	Switch(config-if)# **ima differential-delay-maximum** *msecs*	Specifies the maximum link differential delay tolerated for the IMA group in milliseconds. For T1, the range is 25 to 250 milliseconds, and for E1, the range is 25 to 190 milliseconds.

Example

The following example shows how to configure the maximum allowable differential delay to 100 milliseconds between all interfaces assigned to the IMA group.

```
SwitchA(config)# interface atm 0/0/ima1
SwitchA(config-if)# ima differential-delay-maximum 100
```

Displaying the IMA Group Link Differential Delay Configuration

To display the IMA group maximum differential delay configuration, use the following EXEC command:

Command	Purpose
show ima interface [**atm** *card*/*subcard*/**ima***group* [**detailed**]]	Displays IMA group interface configuration and status.

Example

The following example shows the IMA group maximum differential delay configuration:

```
SwitchA# show ima interface
ATM0/0/ima1 is up
        Group Index    = 4
        State: NearEnd = operational, FarEnd = operational
        FailureStatus  = noFailure
IMA Group Current Configuration:
        MinNumTxLinks = 1    MinNumRxLinks = 1
➔        DiffDelayMax  = 100  FrameLength   = 128
        NeTxClkMode   = common(ctc)  CTC_Reference_Link = ATM0/0/3
        TestLink      = 3       Testpattern      = Not Specified
        TestProcStatus     = disabled   GTSM change timestamp = 990427135611
IMA Link Information:
Link           Physical Status        NearEnd Rx Status       Test Status
-----          ---------------        -----------------      ---------------
ATM0/0/2       up                     active                 disabled
ATM0/0/3       up                     active                 disabled
```

Configuring IMA Group Frame Length

The IMA protocol uses the frame length parameter to determine the number of cells that make up an IMA frame.The IMA group frame length determines the amount of framing overhead and the amount of data lost in case of frame corruption or loss. A small frame length causes more overhead but loses less data if a problem occurs. The recommended frame length is 128.

To configure the frame length on the IMA group, perform the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **interface atm** *card*/*subcard*/**ima***group* Switch(config-if)#	Specifies the IMA group to configure and enters interface configuration mode.
Step 2	Switch(config-if)# **ima frame-length** {**128** | **256** | **32** | **64**}	Specifies the frame length of the IMA group transmit frames, in number of cells.

Example

The following example shows how to configure the frame length transmitted as 256 cells for IMA group 0/0/ima1:

```
SwitchA(config)# interface atm 0/0/ima1
SwitchA(config-if)# ima frame-length 256
```

Displaying the IMA Group Frame Length Configuration

To display the IMA group frame length configuration, use the following EXEC command:

Command	Purpose
show ima interface [**atm** *card*/*subcard*/**ima***group* [**detailed**]]	Displays IMA group interface configuration and status.

Example

The following example shows the IMA group frame length configuration:

```
SwitchA# show ima interface
ATM0/0/ima1 is up
        Group Index    = 4
        State: NearEnd = operational, FarEnd = operational
        FailureStatus = noFailure
IMA Group Current Configuration:
        MinNumTxLinks = 1    MinNumRxLinks = 1
→        DiffDelayMax  = 25   FrameLength   = 256
        NeTxClkMode   = common(ctc)  CTC_Reference_Link = ATM0/0/3
        TestLink      = 3       Testpattern      = Not Specified
        TestProcStatus     = disabled   GTSM change timestamp = 990427143739
IMA Link Information:
Link           Physical Status        NearEnd Rx Status       Test Status
-----          ---------------        -----------------      ---------------
ATM0/0/2       up                     active                 disabled
ATM0/0/3       up                     active                 disabled
```

Configuring IMA Group Test Pattern

An IMA group can have a test pattern defined to provide extra support to verify the connectivity of links within an IMA group. It uses a test pattern sent over one link to verify connectivity to the rest of the group. The test pattern should be looped over all the other links in the group at the far end of the connection. The test procedure is performed using the ICP cells exchanged between both ends of the IMA virtual links.

To configure the test pattern to be transmitted on the IMA group, perform the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **interface atm** *card*/*subcard*/**ima***group* Switch(config-if)#	Specifies the IMA group and enters interface configuration mode.
Step 2	Switch(config-if)# **ima test** [**link** *link-value*] [**pattern** *pattern-value*]	Specifies the specific link and pattern or test pattern only for the IMA group.
Step 3	Switch(config-if)# **no ima test**	Stops the test on the IMA group.

Examples

The following example shows how to configure the test pattern 8 to transmit over link 3 of IMA group 0/0/ima1:

```
SwitchA(config)# interface atm 0/0/ima1
SwitchA(config-if)# ima test link 3 pattern 8
```

The following example shows how to stop the test on IMA group 0/0/ima1:

```
SwitchA(config)# interface atm 0/0/ima1
SwitchA(config-if)# no ima test
```

Displaying the IMA Group Test Pattern Configuration

To display the IMA group test pattern configuration, use the following EXEC command:

Command	Purpose
show ima interface [**atm** *card*/*subcard*/**ima***group* [**detailed**]]	Displays IMA group interface configuration and status.

Example

The following example shows the IMA group test pattern configuration:

```
SwitchA# show ima interface
ATM0/0/ima1 is up
        Group Index    = 4
        State: NearEnd = operational, FarEnd = operational
        FailureStatus  = noFailure
IMA Group Current Configuration:
        MinNumTxLinks = 1    MinNumRxLinks = 1
        DiffDelayMax  = 25   FrameLength   = 128
        NeTxClkMode   = common(ctc)  CTC_Reference_Link = ATM0/0/3
➔        TestLink      = 3    TestPattern   = 8
        TestProcStatus     = operating  GTSM change timestamp = 990427143950
IMA Link Information:
Link          Physical Status       NearEnd Rx Status      Test Status
-----         ---------------       -----------------      ---------------
ATM0/0/2      up                    active                 operating
ATM0/0/3      up                    active                 operating
```

CHAPTER 21

Configuring ATM Router Module Interfaces

This chapter describes steps required to configure the ATM router module on the Catalyst 8540 MSR, Catalyst 8510 MSR, and LightStream 1010 ATM switch routers, and the enhanced ATM router module for the Catalyst 8540 MSR. The ATM router module allows you to integrate Layer 3 switching with ATM switching on the same ATM switch router.

Note This chapter provides advanced configuration instructions for the Catalyst 8540 MSR, Catalyst 8510 MSR, and LightStream 1010 ATM switch routers. For complete descriptions of the commands mentioned in this chapter, refer to the *ATM Switch Router Command Reference* publication. For hardware installation and cabling instructions, refer to the *ATM and Layer 3 Module Installation Guide*.

Note The LightStream 1010 system software image does not include support for the ATM router module or Layer 3 features. You can download the Catalyst 8510 MSR image to a LightStream 1010 ATM switch router with a multiservice ATM switch processor installed.

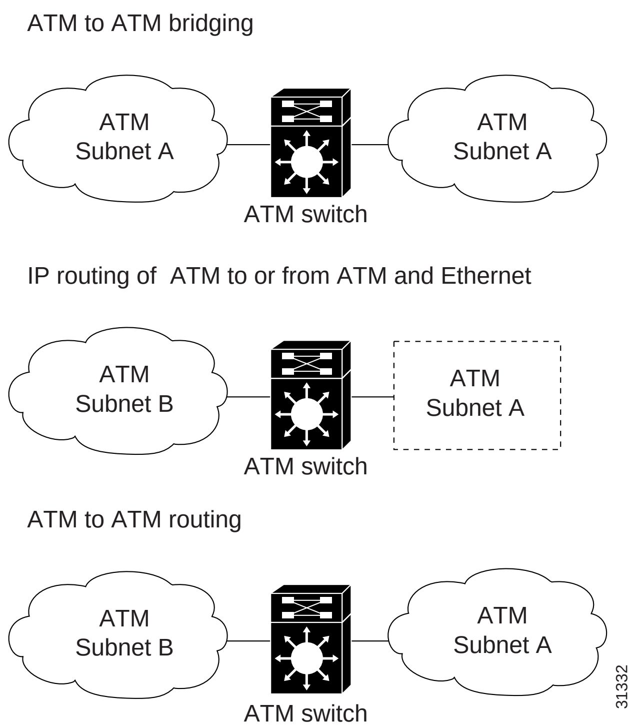
This chapter includes the following sections:

Overview of the ATM Router Module

The ATM router module allows you to integrate Layer 3 routing and ATM switching within a single chassis. When you install the ATM router module, you no longer need to choose either Layer 3 or ATM technology, as is frequently the case with enterprise, campus, and MAN applications.

The ATM router module can perform one or more of the functions described in Figure 21-1.

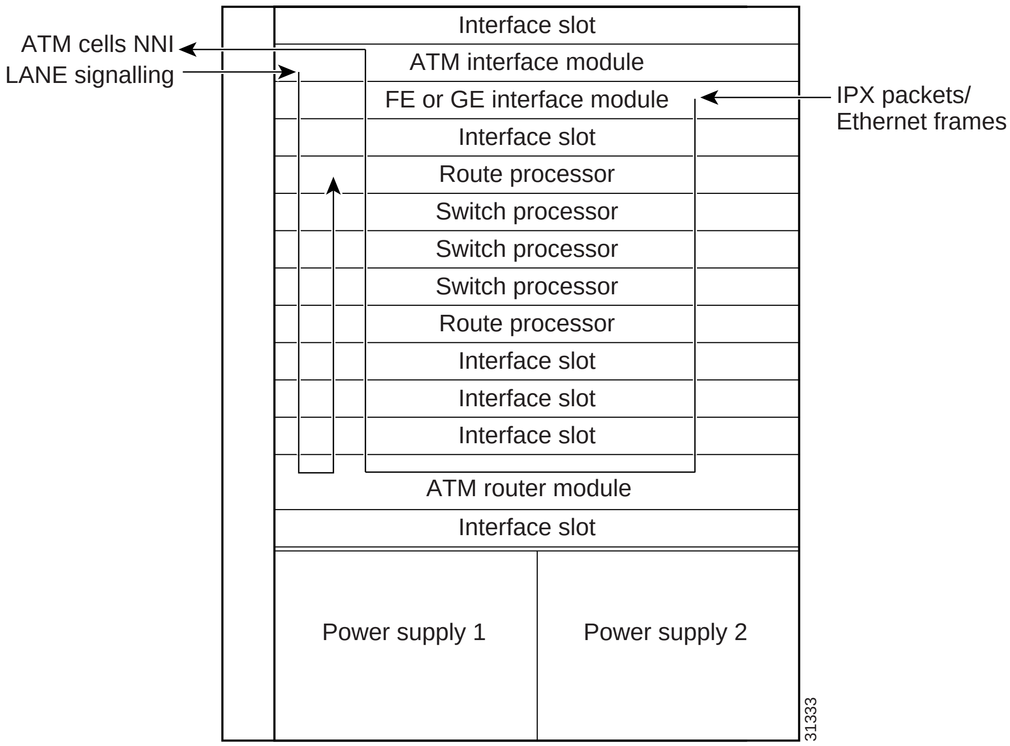
Figure 21-1 ATM Router Module Routing and Bridging Functions



The ATM router module receives Address Resolution Protocol (ARP) messages and route broadcasts from connected ATM peers and sends the appropriate control information to the route processor. On the ATM side, the ATM router module connects to the switching fabric as would any other interface module.

On the Catalyst 8540 MSR, the ATM router module supports LANE clients (LECs), but not LANE servers (LES, LECS, and BUS). It separates the control and data path so that all LANE control messages are handled by the route processor, and data messages are switched on the ATM router module port, as shown in Figure 21-2. The LEC is configured on the ATM router module interface, but control message traffic is sent to the route processor by the ATM router module. The ATM router module sends all ATM data traffic to the appropriate VCs.

Figure 21-2 ATM Router Module Traffic Flow (Catalyst 8540 MSR)



Catalyst 8540 MSR Enhanced ATM Router Module Features

The Catalyst 8540 MSR enhanced ATM router module offers the following benefits:

- Interoperates with all of the Layer 3 switching interface modules available for the Catalyst 8540 CSR chassis. For more information on the Catalyst 8540 CSR Layer 3 interface modules, refer to the *ATM and Layer 3 Module Installation Guide*.
- Provides an integrated high performance link between ATM and Layer 3 cards. The ATM router module provides an aggregate switching capacity of 2 Gbps between ATM and Layer 3 ports (2 x 1-Gbps interfaces per module). Data transfers to the switch core at the rate of 1 Gbps.
- Simplifies management.
- Hot-swappable.
- Occupies only one slot in the chassis.
- Supports multiprotocol encapsulation over ATM (RFC 1483) switched virtual connections (SVCs) and permanent virtual circuits (PVCs) with either ATM adaptation layer 5 (AAL5) Subnetwork Access Protocol (SNAP) or AAL5 MUX encapsulation.
- Supports classical ATM over IP (RFC 1577) SVCs and PVCs.
- Standard and extended access control list (ACL) support for IP, and standard ACL support for IPX.

  For information configuring on IP ACLs, see the "Filtering IP Packets at the IP Interfaces" section on page 11-9, and refer to the "Configuring IP Services" chapter in the *Cisco IOS IP and IP Routing Configuration Guide.* For information configuring on IPX ACLs, refer to the "Configuring Novell IPX" chapter in the *Cisco IOS AppleTalk and Novell IPX Configuration Guide*.

- IP fragmentation support.
- IP 6-path load balancing support.
- Supports OAM-based PVC management.
- Supports Bridge Group Virtual Interface (BVI).
- Supports integrated routing and bridging (IRB).

Note The Catalyst 8540 MSR enhanced ATM router module does not support LANE clients.

The ATM router module has no external interfaces. All traffic is sent and received through internal interfaces to the switching fabric. The Catalyst 8540 MSR enhanced ATM router module has two internal ports.

Catalyst 8540 MSR ATM Router Module Features

The Catalyst 8540 MSR ATM router module offers the following benefits:

- Interoperates with all of the Layer 3 switching interface modules available for the Catalyst 8540 CSR chassis. For more information on the Catalyst 8540 CSR Layer 3 interface modules, refer to the *ATM and Layer 3 Module Installation Guide*.
- Provides an integrated high performance link between ATM and Layer 3 cards. The ATM router module provides an aggregate switching capacity of 2 Gbps between ATM and Layer 3 ports (2 x 1-Gbps interfaces per module). Data transfers to the switch core at the rate of 1 Gbps.
- Simplifies management.
- Hot-swappable.
- Occupies only one slot in the chassis.
- Supports LANE clients (LECs).
- Supports RFC 1483 SVCs and PVCs with AAL5 SNAP encapsulation.
- Supports RFC 1577 SVCs and PVCs.
- Supports OAM-based PVC management.
- Supports BVI.
- Supports IRB.

The ATM router module has no external interfaces. All traffic is sent and received through internal interfaces to the switching fabric. The Catalyst 8540 MSR enhanced ATM router module has two internal ports.

Catalyst 8510 MSR and LightStream 1010 ATM Router Module Features

The Catalyst 8510 MSR and LightStream 1010 ATM router module offers the following benefits:

- Interoperates with all of the Layer 3 switching interface modules available for the Catalyst 8510 CSR chassis. For more information on the Catalyst 8510 CSR Layer 3 interface modules, refer to the *ATM and Layer 3 Module Installation Guide*.
- Provides an integrated high performance link between ATM and Layer 3 cards. The ATM router module provides a switching capacity of 1 Gbps between ATM and Layer 3 ports. Data transfers to the switch core at the rate of 1 Gbps.
- Simplifies management.
- Hot-swappable.
- Occupies only one slot in the chassis.
- Supports RFC 1483 SVCs and PVCs with AAL5 SNAP encapsulation.
- Supports RFC 1577 SVCs and PVCs.
- Supports OAM-based PVC management.
- Supports BVI.
- Supports IRB.

The ATM router module has no external interfaces. All traffic is sent and received through internal interfaces to the switching fabric. The Catalyst 8510 MSR and LightStream 1010 ATM router module has one internal port.

Hardware and Software Restrictions of the ATM Router Module

Hardware Restrictions

The following hardware restrictions apply to the Catalyst 8540 MSR, Catalyst 8510 MSR, and LightStream 1010 ATM router modules, and the Catalyst 8540 MSR enhanced ATM router modules:

- You can install the ATM router module in any slot except a route processor slot, and, in the case of the Catalyst 8540 MSR, a switch processor slot.
- The ATM router module is only supported on LightStream 1010 ATM switches with multiservice ATM switch rout processor with FC-PFQ and the Catalyst 8510 MSR system software image.
- You can install up to two ATM router modules per chassis.
- When you hot swap an ATM router module, wait one minute after removing the module before inserting a new module.

Note The ATM router module is only supported on ATM switches which have multiservice ATM switch processor installed.

Catalyst 8540 MSR Enhanced ATM Router Module Software Restrictions

The following software restrictions apply to the Catalyst 8540 MSR enhanced ATM router module:

- LANE is not supported.
- Use tag switching functionality with caution. Do not distribute routes learned through tag switching to Fast Ethernet (FE) or Gigabit Ethernet (GE), or vice versa. Otherwise, you might have unreachable route destinations.
- The ATM router module does not initialize if it replaces an ATM port adapter or interface module when hierarchical VP tunnels are globally enabled. Reboot the switch to initialize the ATM router module.
- IP multicast is only supported over 1483 LLC/SNAP encapsulated PVCs.
- ATM Director does not support any PVC commands.
- Even though each ATM router module interface supports a maximum of 2048 VCs, only 1400 to 1500 external VCs can be configured. Internal VCs use up the rest.
- Do not install an ATM router module in a slot pair where hierarchical VP tunnels are configured. Slot pairs 0 and 1, 2 and 3, 9 and 10, and 11 and 12 use the same switching modules for scheduling. For example, do not install an ATM router module in slot 10 when hierarchical VP tunnels are configured on slot 9. For more information on hierarchical VP tunneling restrictions, see the "Configuring a Hierarchical VP Tunnel for Multiple Service Categories" section on page 6-35.

The Catalyst 8540 MSR enhanced ATM router modules do not support the following features:

- Point-to-point subinterfaces. Only point-to-multipoint subinterfaces are supported
- Tag-edged router functionality
- Fast Simple Server Redundancy Protocol (FSSRP)
- Bridging for multiplexing device encapsulation
- Protocol Independent Multicast (PIM) IP multipoint signalling
- PIM nonbroadcast multiaccess (NBMA)
- PIM over ATM multipoint signalling
- Translation from IP quality of service (QoS) to ATM QoS
- Resource Reservation Protocol (RSVP) to ATM SVC
- PVC management using ILMI
- IP multicast over RFC 1483 SVCs
- Access lists for ATM to ATM routing
- Half-bridge devices
- Layer 2 ACLs

Catalyst 8540 MSR ATM Router Module Software Restrictions

The following software restrictions apply to the Catalyst 8540 MSR ATM router module:

- Use tag switching functionality with caution. Do not distribute routes learned through tag switching to FE or GE, or vice versa. Otherwise, you might have unreachable route destinations.
- The ATM router module does not initialize if it replaces an ATM port adapter or interface module when hierarchical VP tunnels are globally enabled. Reboot the switch to initialize the ATM router module.
- ATM Director does not support any PVC commands.
- Only LANE clients or RFC 1483, not both, can be configured on an ATM router module interface.
- RFC 1483 on the ATM router module supports only AAL5 SNAP encapsulation.
- Even though each ATM router module interface supports a maximum of 2048 VCs, only 1400 to 1500 external VCs can be configured. Internal VCs use up the rest.
- IP multicast is only supported over 1483 LLC/SNAP encapsulated PVCs.
- You can have a maximum of 64 LECs per chassis.
- Do not install an ATM router module in a slot pair where hierarchical VP tunnels are configured. Slot pairs 0 and 1, 2 and 3, 9 and 10, and 11 and 12 use the same switching modules for scheduling. For example, do not install an ATM router module in slot 10 when hierarchical VP tunnels are configured on slot 9. For more information on hierarchical VP tunneling restrictions, see the "Configuring a Hierarchical VP Tunnel for Multiple Service Categories" section on page 6-35.
- Token Ring LANE is not supported.

The Catalyst 8540 MSR ATM router modules do not support the following features:

- Point-to-point subinterfaces. Only point-to-multipoint subinterfaces are supported.
- Tag-edged router functionality
- Fast Simple Server Redundancy Protocol (SSRP)
- Bridging for multiplexing device encapsulation
- PIM IP multipoint signalling
- PIM NBMA
- PIM over ATM multipoint signalling
- Translation from IP QoS to ATM QoS
- RSVP to ATM SVC
- PVC management using ILMI
- Access lists for ATM to ATM routing
- Half-bridge devices
- RFC 1483 MUX encapsulation
- IP multicast over RFC 1483 SVCs
- ACLs for IP, and standard ACLs for IPX
- IP fragmentation.
- IP 6-path load balancing.

Catalyst 8540 MSR and LightStream 1010 ATM Router Module Software Restrictions

The following software restrictions apply to the Catalyst 8540 MSR enhanced ATM router module:

- Use tag switching functionality with caution. Do not distribute routes learned through tag switching to FE or GE, or vice versa. Otherwise, you might have unreachable route destinations.
- The ATM router module does not initialize if it replaces an ATM port adapter or interface module when hierarchical VP tunnels are globally enabled. Reboot the switch to initialize the ATM router module.
- ATM Director does not support any PVC commands.
- RFC 1483 on the ATM router module supports only AAL5 SNAP encapsulation.
- Even though each ATM router module interface supports a maximum of 2048 VCs, only 1400 to 1500 external VCs can be configured. Internal VCs use up the rest.
- Do not install an ATM router module in a slot pair where hierarchical VP tunnels are configured. Slot pair 0 and 1 and slot pair 3 and 4 use the same switching modules for scheduling. For example, do not install an ATM router module in slot 1 when hierarchical VP tunnels are configured on slot 0. For more information on hierarchical VP tunneling restrictions, see the “Configuring a Hierarchical VP Tunnel for Multiple Service Categories” section on page 6-35.
- RFC 1577 SVCs
- LANE clients are not supported.
- Only UBR PVCs are supported.
- IP multicast is only supported over 1483 LLC/SNAP encapsulated PVCs.

The Catalyst 8510 MSR and LightStream 1010 ATM router modules do not support the following features:

- Point-to-point subinterfaces. Only point-to-multipoint subinterfaces are supported.
- Tag-edged router functionality
- SSRP
- Bridging for multiplexing device encapsulation
- Protocol Independent Multicast (PIM) IP multipoint signalling
- PIM nonbroadcast multiaccess (NBMA)
- PIM over ATM multipoint signalling
- Translation from IP quality of service (QoS) to ATM QoS
- Resource Reservation Protocol (RSVP) to ATM SVC
- PVC management using ILMI
- Access lists for ATM to ATM routing
- Half-bridge devices
- RFC 1483 MUX encapsulation
- IP multicast over RFC 1483 SVCs
- ACLs for IP, and standard ACLs for IPX

- IP fragmentation.
- IP 6-path load balancing.

Note The ATM router module is only supported on ATM switches which have a multiservice ATM switch processor installed.

Note The LightStream 1010 system software image does not include support for the ATM router module or Layer 3 features. You can download this image to a LightStream 1010 ATM switch router with a multiservice ATM switch processor installed.

Configuring ATM Router Module Interfaces

The you can configure the following features directly on the ATM router module interfaces:

- Maximum virtual channel identifier (VCI) bits
- Maximum Transmission Units (MTUs) (enhanced Catalyst 8540 MSR)
- LANE clients (Catalyst 8540 MSR)
- RFC 1483
- Classical IP over ATM (RFC 1577)
- Bridging
- IP multicast

Note This document describes how to configure ATM software features combined with Layer 3 features only. For more detailed information on how to configure the Layer 3 modules that interoperate with the ATM router module in the Catalyst 8540 MSR chassis, refer to the *Layer 3 Switching Software Feature and Configuration Guide*, which is available on the Documentation CD-ROM that came with your ATM switch router, online at Cisco.com, or when ordered separately as a hard copy document.

Note ATM router modules have internal interfaces, but no external ports. Use the **interface atm** *card/subcard/port* command to specify these interfaces.

Note Virtual path identifier (VPI) 2 is reserved for ATM router module interfaces, which allows up to 2048 external VCs on each ATM router module interface. Using VPI 0 would have allowed less than 1024 external VCs on an ATM router module interface because the ATM router module external VCs would have been forced to share the VC space within VPI 0 with the internal PVCs.

Even though each ATM router module interface supports a maximum of 2048 VCs, only 1400 to 1500 external VCs can be configured. Internal VCs use up the rest.

Default ATM Router Module Interface Configuration Without Autoconfiguration

If ILMI is disabled or if the connecting end node does not support ILMI, the following defaults are assigned to all ATM router module interfaces:

- ATM interface type = UNI
- UNI version = 3.0
- Maximum VCI bits = 11
- MTU size = 1500 bytes
- ATM interface side = network
- ATM UNI type = private

Note Only Catalyst 8540 MSR enhanced ATM router module interfaces support IP unicast and IP multicast fragmentation. For IP unicast fragmentation, the packet must ingress on an enhanced ATM router module interface and egress on any interface. For IP multicast fragmentation, IP multicast data packets greater than 1500 bytes are fragmented to 1500 bytes on the ingress enhanced ATM router module interface before being switched to other members in the multicast group. All the members in the multicast group must have an MTU equal to or greater than 1500 bytes.

Manual ATM Router Module Interface Configuration

To manually change the default configuration values, perform the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **interface atm** *card*/*subcard*/*port* Switch(config-if)#	Specifies an ATM interface and enters interface configuration mode.
Step 2	Switch(config-if)# **atm maxvci-bits** *max-vci-bits*	Modifies the maximum number of active VCI bits.
Step 3	Switch(config-if)# **mtu** *bytes*	Modifies the MTU size. The default MTU size is 1500 bytes. **Note** Only Catalyst 8540 MSR enhanced ATM router modules support variable MTU sizes.

Example

The following example shows how to change the default number of active VCI bits:

```
Switch(config)# interface atm 0/0/0
Switch(config-if)# atm maxvci-bits 10
```

Configuring LECs on ATM Router Module Interfaces (Catalyst 8540 MSR)

The procedures for configuring LANE clients (LECs) on the ATM router module are the same as for the configuration of LECs on the route processor, with one exception: To specify an ATM router module interface, rather than the route processor interface, use the **interface atm** *card*/*subcard*/*port* command. On the route processor, you would use the **interface atm 0** command.

Note To route traffic between an emulated LAN and a Fast Ethernet (FE) or Gigabit Ethernet (GE) interface, you must configure the LEC on an ATM router module interface rather than a route processor interface.

Note An ATM router module interface can be configured for either LECs or RFC 1483 PVCs, not both. For both features to operate on the same ATM router module, configure LECs on one interface and RFC 1483 PVCs on the other.

Note LANE clients are not supported on the Catalyst 8540 MSR enhanced ATM router module.

To configure a LEC on an ATM router module interface, use the following commands, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **interface atm** *card*/*subcard*/*port*.*subinterface#* **multipoint** Switch(config-subif)#	Creates the ATM router module point-to-multipoint subinterface and enters subinterface mode. **Note** The ATM router module only supports point-to-multipoint subinterfaces.
Step 2	Switch(config-subif)# **ip address** *ip-address mask*	Provides a protocol address and subnet mask for the client on this subinterface.
Step 3	Switch(config-subif)# **lane client ethernet** *elan-name*	Enables a LANE client for an emulated LAN.

Example

The following example shows how to configure two LECs on an ATM router module interface:

```
Switch# configure terminal
Switch(config)# interface atm 1/0/0.4 multipoint
Switch(config-subif)# ip address 40.0.0.1 255.0.0.0
Switch(config-subif)# lane client ethernet VLAN4
Switch(config-subif)# exit
Switch(config)# interface atm 1/0/0.5 multipoint
Switch(config-subif)# ip address 50.0.0.1 255.0.0.0
Switch(config-subif)# lane client ethernet VLAN5
Switch(config-subif)# exit
Switch(config)# router ospf 1
Switch(config-router)# network 40.0.0.0 0.255.255.255 area 0
Switch(config-router)# network 50.0.0.0 0.255.255.255 area 0
```

For more information on configuring LECs on ATM router module interfaces, see the "Configuring a LAN Emulation Client on the ATM Switch Router" section on page 13-14. For a detailed description of LANE and its components, refer to *Cisco IOS Switching Services Configuration Guide: Virtual LANs.*

LEC Configuration Examples

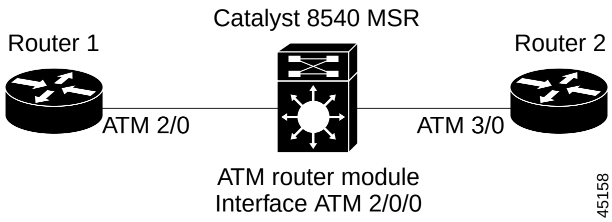
The examples in this section show how to configure LANE clients (LECs) on networks with two routers and one Catalyst 8540 MSR. For detailed information on configuring the LANE server (LES), LANE configuration server (LECS), and broadcast-and-unknown server (BUS), see Chapter 13, "Configuring LAN Emulation."

Caution For performance reasons, avoid configuring the LANE server components on ATM switch routers. Instead, configure the LANE server components on a router such as a Cisco 7500 series router or a Catalyst 5500 router with a LANE module installed.

LANE Routing Over ATM

The following example shows how to configure LANE routing over ATM using the ATM router module. Figure 21-3 shows an example of a network for LANE routing over ATM.

Figure 21-3 Example Network for LANE Routing over ATM



Router 1 ATM Interface

```
Router1# configure terminal
Router1(config)# interface atm 2/0
Router1(config-if)# ip address 1.0.0.1 255.0.0.0
Router1(config-if)# atm pvc 1 0 5 qsaal
Router1(config-if)# atm pvc 2 0 16 ilmi
Router1(config-if)# lane client ethernet happy
Router1(config-if)# end
Router1#
```

ATM Switch Router ATM Router Module Interface

```
Switch# configure terminal
Switch(config)# interface atm 2/0/0
Switch(config-if)# ip address 1.0.0.2 255.0.0.0
Switch(config-if)# lane client ethernet BACKBONE
Switch(config-if)# end
Switch#
```

Router 2 ATM Interface

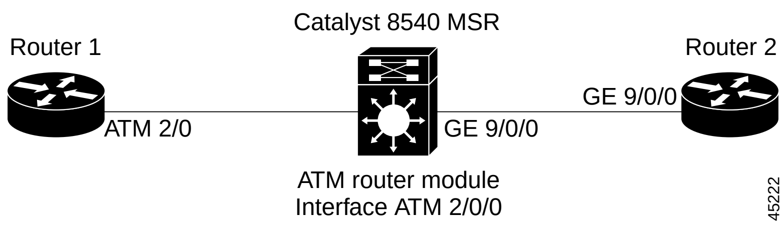
```
Router2# configure terminal
Router2(config)# interface atm 3/0
Router2(config-if)# ip address 1.0.0.3 255.0.0.0
Router2(config-if)# no ip mroute-cache
Router2(config-if)# atm pvc 1 0 5 qsaal
Router2(config-if)# atm pvc 2 0 16 ilmi
Router2(config-if)# no atm ilmi-keepalive
Router2(config-if)# lane client ethernet BACKBONE
Router2(config-if)# end
Router2#
```

For detailed information on configuring LANE clients (LECs), see Chapter 13, "Configuring LAN Emulation."

LANE Routing from ATM to Ethernet

The following example shows how to configure LANE routing from ATM to Ethernet using the ATM router module. Figure 21-4 shows an example of a LANE network for LANE routing from ATM to Ethernet.

Figure 21-4 Example Network for LANE Routing from ATM to Ethernet



Router 1 ATM Interface

```
Router1# configure terminal
Router1(config)# interface atm 2/0
Router1(config-if)# ip address 1.0.0.1 255.0.0.0
Router1(config-if)# atm pvc 1 0 5 qsaal
Router1(config-if)# atm pvc 2 0 16 ilmi
Router1(config-if)# lane client ethernet happy
Router1(config-if)# end
Router1#
```

ATM Switch Router ATM Router Module Interface

```
Switch# configure terminal
Switch(config)# interface atm 2/0/0
Switch(config-if)# ip address 1.0.0.2 255.0.0.0
Switch(config-if)# lane client ethernet BACKBONE
Switch(config-if)# end
Switch#
```

ATM Switch Router Ethernet Interface

```
Switch# configure terminal
Switch(config)# interface gigabitethernet 9/0/0
Switch(config-if)# ip address 129.1.0.1 255.255.255.0
Switch(config-if)# no ip directed-broadcast
Switch(config-if)# end
Switch#
```

Router 2 Ethernet Interface

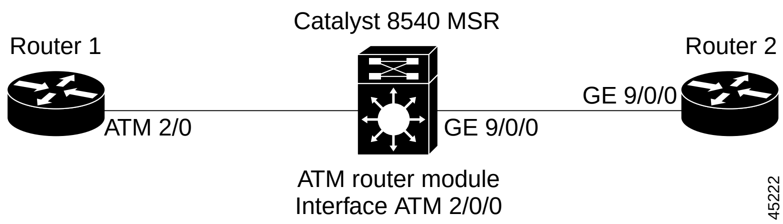
```
Router2# configure terminal
Router2(config)# interface gigabitethernet 9/0/0
Router2(config-if)# ip address 129.1.0.2 255.255.255.0
Router2(config-if)# no ip directed-broadcast
Router2(config-if)# end
Router2#
```

Configure the desired network routing protocol, such as RIP, OSPF, or EIGRP, on Ethernet interfaces. For more information on configuring networking protocols and routing, refer to the *Layer 3 Software Configuration Guide*.

LANE Bridging Between ATM and Ethernet

The following example show how to configure LANE bridging between ATM and Ethernet using the ATM router module. Figure 21-5 shows an example of a network for LANE bridging between ATM and Ethernet.

Figure 21-5 Example Network for LANE Bridging Between ATM and Ethernet



Router 1 ATM Interface

```
Router1# configure terminal
Router1(config)# interface atm 2/0
Router1(config-if)# atm pvc 1 0 5 qsaal
Router1(config-if)# atm pvc 2 0 16 ilmi
Router1(config-if)# lane client ethernet happy
Router1(config-if)# bridge-group 1
Router1(config-if)# end
Router1#
```

Router 1 Bridge Interface

```
Router1# configure terminal
Router1(config)# interface BVI1
Router1(config-if)# ip address 130.2.3.1 255.255.255.0
Router1(config-if)# exit
Router1(config)# bridge 1 protocol ieee
Router1(config)# bridge 1 route ip
Router1(config)# bridge irb
Router1(config)# end
Router1#
```

ATM Switch Router ATM Router Module Interface

```
Switch# configure terminal
Switch(config)# interface atm 2/0/0
Switch(config-if)# lane client ethernet BACKBONE
Switch(config-if)# bridge-group 1
Switch(config-if)# exit
Switch(config)# bridge 1 protocol ieee
Switch(config)# end
Switch#
```

ATM Switch Router Ethernet Interface

```
Switch# configure terminal
Switch(config)# interface gigabitethernet9/0/0
Switch(config-if)# bridge-group 1
Switch(config-if)# end
Switch#
```

Router 2 Ethernet Interface

```
Router2# configure terminal
Router2(config)# interface ethernet 9/0/0
Router2(config-if)# bridge-group 1
Router2(config-if)# end
Router2#
```

Router 2 Bridge Interface

```
Router2# configure terminal
Router2(config)# interface BVI1
Router2(config-if)# ip address 130.2.3.4 255.255.255.0
Router2(config-if)# exit
Router2(config)# bridge 1 protocol ieee
Router2(config)# bridge 1 route ip
Router2(config)# bridge irb
Router2(config)# end
Router2#
```

For more information on configuring bridging, refer to the *Layer 3 Software Configuration Guide*.

Confirming the LEC Configuration

To confirm the LEC configuration on the ATM switch router, use the following EXEC commands:

Command	Purpose
show lane [**interface atm** *card*/*subcard*/*port*[**.***subinterface#*] | **name** *elan-name*] [**brief**]	Displays the global and per-virtual channel connection LANE information for all the LANE components and emulated LANs configured on an interface or any of its subinterfaces.
show lane client [**interface atm** *card*/*subcard*/*port*[**.***subinterface#*] | **name** *elan-name*] [**brief**]	Displays the global and per-VCC LANE information for all LANE clients configured on any subinterface or emulated LAN.
show lane config [**interface atm** *card*/*subcard*/*port*[**.***subinterface#*]]	Displays the global and per-VCC LANE information for the configuration server configured on any interface.

Configuring Multiprotocol Encapsulation over ATM

This section describes how to configure multiprotocol encapsulation over ATM, as defined in RFC 1483, on the ATM router module.

The primary use of multiprotocol encapsulation over ATM, also know as RFC 1483, is carrying multiple Layer 3 and bridged frames over ATM. RFC 1483 traffic is routed through an ATM router module interface using static map lists. Static map lists provide an alternative to using the ATM Address Resolution Protocol (ARP) and ATM Inverse ARP (InARP) mechanisms. For more information on static map lists, see the “Mapping a Protocol Address to a PVC Using Static Map Lists” section on page 12-7.

For a detailed description of multiprotocol encapsulation over ATM, refer to the *Guide to ATM Technology*.

Note Traffic shaping and policing are not supported on the ATM router module interfaces; for traffic shaping and policing on ATM connections, use VP tunnels. For more information on VP tunnels, see the "Configuring VP Tunnels" section on page 6-31.

To configure multiprotocol encapsulation over ATM on the ATM router module interface, use the following commands, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **interface atm** *card*/*subcard*/*port*.*subinterface#* **multipoint** Switch(config-subif)#	Creates the ATM router module point-to-multipoint subinterface and enters subinterface mode. **Note** The ATM router module only supports point-to-multipoint subinterfaces.
Step 2	Switch(config-subif)# **ip address** *ip-address mask*	Enters the IP address and subnet mask associated with this interface.
Step 3	Switch(config-subif)# **map-group** *name*	Enters the map group name associated with this PVC.
Step 4	Switch(config-subif)# **atm pvc 2** *vci-a* [**upc** *upc*] [**pd** *pd*] [**rx-cttr** *index*] [**tx-cttr** *index*] **interface atm** *card*/*subcard*/*port*[**.***vpt#*] *vpi-b vci-b* [**upc** *upc*] **encap** {**aal5mux**[1] | **aal5snap**}	Configures the PVC. **Note** The VPI number on the ATM router module interface must be 2.
Step 5	Switch(config-subif)# **exit** Switch(config)#	Returns to global configuration mode.
Step 6	Switch(config)# **map-list** *name* Switch(config-map-list)#	Creates a map list by naming it, and enters map-list configuration mode.
Step 7	Switch(config-map-list)# **ip** *ip-address* {**atm-nsap** *address* | **atm-vc** *vci*} [**broadcast**]	Associates a protocol and address with a specific virtual circuit.

1. Only the Catalyst 8540 MSR enhanced ATM router module supports AAL5 MUX encapsulation.

Example

The following example shows how to configure RFC 1483 on an ATM router module interface, beginning in global configuration mode:

```
Switch(config)# interface atm 1/0/0.1011 multipoint
Switch(config-subif)# ip address 10.1.1.1 255.255.255.0
Switch(config-subif)# map-group net1011
Switch(config-subif)# atm pvc 2 1011 interface atm 3/0/0 0 1011 encap aal5snap
Switch(config-subif)# exit
Switch(config)# map-list net1011
Switch(config-map-list)# ip 10.1.1.2 atm-vc 1011
Switch(config-map-list)# end
Switch#
```

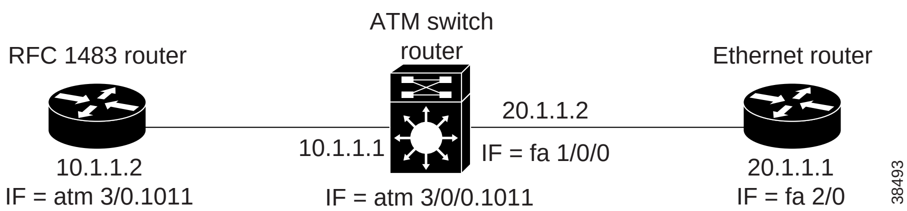
Multiprotocol Encapsulation over ATM Configuration Example

The following example shows how to configure for multiprotocol encapsulation over ATM with two routers and a ATM switch router.

The ATM switch router has an ATM router module in slot 0, a Fast Ethernet interface module in slot 1, and an ATM interface module in slot 3. One router has an ATM interface processor in slot 3. The other router has a Fast Ethernet interface module in slot 2.

Figure 21-6 shows an example of an RFC 1483 network.

Figure 21-6 Example Network for RFC 1483



Router with ATM Interface

```
RouterA# configure terminal
RouterA(config)# interface atm 3/0.1011 multipoint
RouterA(config-subif)# ip address 10.1.1.2 255.255.255.0
RouterA(config-subif)# atm pvc 1011 0 1011 aal5snap
RouterA(config-subif)# map group net1011
RouterA(config-subif)# ipx network 1011
RouterA(config-subif)# exit
RouterA(config)# map-list net1011
RouterA(config-map-list)# ip 10.1.1.1 atm-vc 1011
RouterA(config-map-list)# ipx 1011.1111.1111.1111 atm-vc 1011
RouterA(config-map-list)# exit
RouterA(config)#
```

ATM Switch Router

```
Switch# configure terminal
Switch(config)# interface atm 0/0/0.1011 multipoint
Switch(config-subif)# ip address 10.1.1.1 255.255.255.0
Switch(config-subif)# ipx network 1011
Switch(config-subif)# map-group net1011
Switch(config-subif)# atm pvc 2 1011 interface atm 3/0/0 0 1011
Switch(config-subif)# map-list net1011
Switch(config-map-list)# ip 10.1.1.2 atm-vc 1011
Switch(config-map-list)# ipx 1011.2222.2222.2222 atm-vc 1011
Switch(config-map-list)# exit
Switch(config)# interface fastethernet 1/0/0
Switch(config-if)# ip address 20.1.1.2 255.255.255.0
Switch(config-if)# ipx network 2011
Switch(config-if)# end
Switch#
```

Note The VCI in the **atm pvc** command must match the **atm-vc** VCI in the map list.

Ethernet Router

```
RouterB# configure terminal
RouterB(config)# ipx routing
RouterB(config)# interface fastethernet 2/0
RouterB(config-if)# ip address 20.1.1.1 255.255.255.0
RouterB(config-if)# ipx network 2011
RouterB(config-if)# end
RouterB#
```

Configuring Classical IP over ATM in a PVC Environment

This section describes how to configure classical IP over ATM, as described in RFC 1577, in a PVC environment on the ATM router module. The ATM Inverse ARP (InARP) mechanism is applicable to networks that use permanent virtual connections (PVCs), where connections are established but the network addresses of the remote ends are not known. For more information on configuring ATM ARP and ATM InARP, see the "Configuring Classical IP over ATM" section on page 12-1.

For a description of classical IP over ATM and RFC 1577, refer to the *Guide to ATM Technology*.

In a PVC environment, configure the ATM InARP mechanism on the ATM router module by performing the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **interface atm** *card/subcard/port* Switch(config-if)#	Specifies the ATM router module interface to configure.
Step 2	Switch(config-if)# **ip address** *ip-address mask*	Specifies the IP address of the interface.
Step 3	Switch(config-if)# **atm pvc 2** *vci* **interface atm** *card/subcard/port vpi vci* **encap** {**aal5mux**[1] | **aal5snap**} [**inarp** *minutes*]	Creates a PVC and enables ATM InARP. **Note** The VPI number on the ATM router module interface must be 2.

1. Only the Catalyst 8540 MSR enhanced ATM router module supports AAL5 MUX encapsulation.

Repeat these tasks for each PVC you want to create.

The **inarp** *minutes* interval specifies how often inverse ARP datagrams are sent on this virtual circuit. The default value is 15 minutes.

Example

The following example shows how to configure an IP-over-ATM interface on interface ATM 3/0/0, using a PVC with AAL5SNAP encapsulation, InARP set to ten minutes, VPI = 2, and VCI = 100:

```
Switch(config)# interface atm 3/0/0
Switch(config-if)# ip address 11.11.11.11 255.255.255.0
Switch(config-if)# atm pvc 2 100 interface atm 0/0/0 50 100 encap aal5snap inarp 10
```

Configuring Classical IP over ATM in an SVC Environment

This section describes how to configure classical IP over ATM in an SVC environment on your ATM router module. It requires configuring only the device's own ATM address and that of a single ATM Address Resolution Protocol (ARP) server into each client device.

For a detailed description of the role and operation of the ATM ARP server, refer to the *Guide to ATM Technology*.

The ATM switch router can be configured as an ATM ARP client, thereby being able to work with any ATM ARP server conforming to RFC 1577. Alternatively, one of the ATM switch routers in a logical IP subnet (LIS) can be configured to act as the ATM ARP server itself. In that case, it automatically acts as a client as well. The following sections describe configuring the ATM switch router in an SVC environment as either an ATM ARP client or an ATM ARP server.

Configuring as an ATM ARP Client

In an SVC environment, configure the ATM ARP mechanism on the interface by performing the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **interface atm** *card*/*subcard*/*port* Switch(config-if)#	Selects the ATM router module interface.
Step 2	Switch(config-if)# **atm nsap-address** *nsap-address* or Switch(config-if)# **atm esi-address** *esi***.***selector*	Specifies the network service access point (NSAP) ATM address of the interface. or Specifies the end-system-identifier (ESI) address of the interface.
Step 3	Switch(config-if)# **ip address** *ip-address mask*	Specifies the IP address of the interface.
Step 4	Switch(config-if)# **atm arp-server nsap** *nsap-address*	Specifies the ATM address of the ATM ARP server.
Step 5	Switch(config-if)# **exit** Switch(config)#	Exits interface configuration mode.
Step 6	Switch(config)# **atm route** *addr-prefix*[1] **atm** *card*/*subcard*/*port* **internal**	Configures a static route through the ATM router module interface. See the note that follows this table.

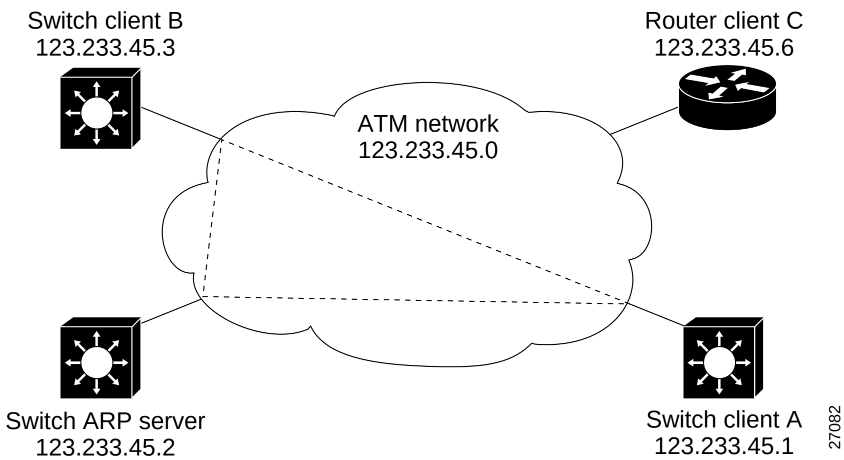
1. The address prefix is the first 19 bytes of the NSAP address.

Note The end system identifier (ESI) address form is preferred, in that it automatically handles the advertising of the address. Use the network service access point (NSAP) form of the command when you need to define a full 20-byte unique address with a prefix unrelated to the network prefix on that interface. You only need to specify a static route when configuring an ARP client using an NSAP address.

NSAP Address Example

Figure 21-7 shows three ATM switch routers and a router connected using classical IP over ATM.

Figure 21-7 Classical IP over ATM Connection Setup



The following example shows how to configure the ATM router module interface ATM 1/0/0 of Client A in Figure 21-7, using the NSAP address:

```
Client A(config)# interface atm 1/0/0
Client A(config-if)# atm nsap-address 47.0091.8100.0000.1111.1111.1111.1111.1111.1111.00
Client A(config-if)# ip address 123.233.45.1 255.255.255.0
Client A(config-if)# atm arp-server nsap 47.0091.8100.0000.1111.1111.1111.2222.2222.2222.00
Client A(config-if)# exit
Client A(config)# atm route 47.0091.8100.0000.1111.1111.1111.1111.1111.1111 atm 1/0/0 internal
```

ESI Example

The following example shows how to configure the ATM router module interface ATM 1/0/0 of Client A in Figure 21-7, using the ESI:

```
Client A(config)# interface atm 1/0/0
Client A(config-if)# atm esi-address 0041.0b0a.1081.40
Client A(config-if)# ip address 123.233.45.1 255.255.255.0
Client A(config-if)# atm arp-server nsap 47.0091.8100.0000.1111.1111.1111.2222.2222.2222.00
Client A(config-if)# exit
```

Configuring as an ATM ARP Server

Cisco's implementation of the ATM ARP server supports a single, nonredundant server per LIS, and one ATM ARP server per subinterface. Thus, a single ATM switch router can support multiple ARP servers by using multiple interfaces.

To configure the ATM ARP server, perform the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **interface atm** *card*/*subcard*/*port*[**.***subinterface#*] Switch(config-if)#	Selects the Catalyst 8540 MSR enhanced ATM router module interface.
Step 2	Switch(config-if)# **atm nsap-address** *nsap-address* or Switch(config-if)# **atm esi-address** *esi***.***selector*	Specifies the NSAP ATM address of the interface. or Specifies the end-system-identifier address of the interface.
Step 3	Switch(config-if)# **ip address** *ip-address mask*	Specifies the IP address of the interface.
Step 4	Switch(config-if)# **atm arp-server time-out** *minutes*[1]	Configures the ATM ARP server optional idle timer.
Step 5	Switch(config-if)# **atm route** *addr-prefix*[2] **atm** *card*/*subcard*/*port* **internal**	Configures a static route through the optional ATM router module interface.

1. This form of the **atm arp-server** command indicates that this interface performs the ATM ARP server functions. When you configure the ATM ARP client (described earlier), the **atm arp-server** command is used—with a different keyword and argument—to identify a different ATM ARP server to the client.
2. Address prefix is the first 19 bytes of the NSAP address.

Note The ESI address form is preferred in that it automatically handles the advertising of the address. Use the NSAP form of the command when you need to define a full 20-byte unique address with a prefix unrelated to the network prefix on that interface. You only need to specify a static route when configuring an ARP server using an NSAP address.

The idle timer interval is the number of minutes a destination entry listed in the ATM ARP server's ARP table can be idle before the server takes any action to timeout the entry.

Example

The following example configures the route processor interface ATM 0 as an ARP server (shown in Figure 21-7):

```
ARP_Server(config)# interface atm 1/0/0
ARP_Server(config-if)# atm esi-address 0041.0b0a.1081.00
ARP_Server(config-if)# atm arp-server self
ARP_Server(config-if)# ip address 123.233.45.2 255.255.255.0
```

Displaying the IP-over-ATM Interface Configuration

To show the IP-over-ATM interface configuration, use the following EXEC commands:

Command	Purpose
show atm arp-server	Shows the ATM interface ARP configuration.
show atm map	Shows the ATM map list configuration.

Examples

In the following example, the **show atm arp-server** command displays the configuration of the interface ATM 1/0/0:

```
Switch# show atm arp-server

Note that a '*' next to an IP address indicates an active call

      IP Address        TTL     ATM Address
ATM1/0/0:
    * 10.0.0.5          19:21   47009181005670000000000112200410b0a108140
```

The following example displays the map-list configuration of the static map and IP-over-ATM interfaces:

```
Switch# show atm map
Map list ATM1/0/0_ATM_ARP : DYNAMIC
arp maps to NSAP 36.0091810000000003D5607900.0003D5607900.00
        , connection up, VPI=0 VCI=73, ATM2/0/0
ip 5.1.1.98 maps to s 36.0091810000000003D5607900.0003D5607900.00
        , broadcast, connection up, VPI=0 VCI=77, ATM2/0/0

Map list ip : PERMANENT
ip 5.1.1.99 maps to VPI=0 VCI=200
```

Configuring Bridging

All PVCs configured on ATM router module interfaces are used for bridging.

To configure bridging on an ATM router module interface, use the following commands, beginning in global configuration mode:

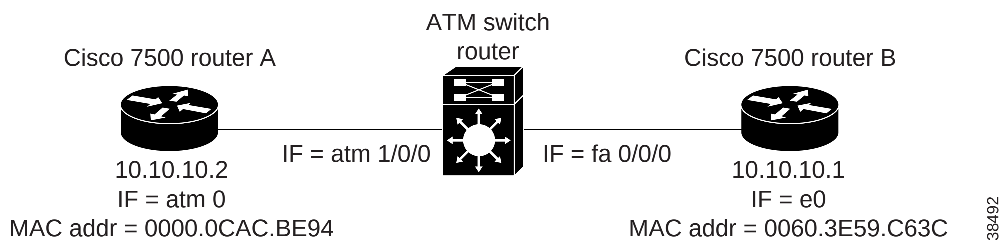
	Command	Purpose
Step 1	Switch(config)# **interface atm** *card*/*subcard*/*port* Switch(config-if)#	Specifies the interface on the ATM router module to configure.
Step 2	Switch(config-if)# **atm pvc 2** *vci* **interface atm** *card*/*subcard*/*port vpi*	Configures a PVC. **Note** The VPI number on the ATM router module interface must be 2.
Step 3	Switch(config-if)# **bridge-group** *number*	Assigns the interface to a bridge group.
Step 4	Switch(config-if)# **end** Switch(config)#	Returns to global configuration mode.
Step 5	Switch(config)# **interface fastethernet** *card*/*subcard*/*port* Switch(config-if)#	Specifies the Fast Ethernet interface to configure.
Step 6	Switch(config-if)# **no cdp enable**	Disables Cisco Discovery Protocol on the interface.
Step 7	Switch(config-if)# **bridge-group** *number*	Assigns the interface to a bridge group.
Step 8	Switch(config-if)# **end** Switch(config)#	Returns to global configuration mode.
Step 9	Switch(config)# **bridge** *number* **protocol ieee**	Specifies the IEEE 802.1D Spanning-Tree Protocol for the bridge group.

Example

The following example shows how to configure bridging on a Catalyst 8540 MSR with a Fast Ethernet interface module in slot 0, an ATM interface module in slot 1, and an ATM router module in slot 3.

Figure 21-8 shows an example bridging network.

Figure 21-8 Example Network for Bridging



```
Switch(config)# interface atm 3/0/0
Switch(config-if)# atm pvc 2 200 interface atm 1/0/0 0 200
Switch(config-if)# bridge-group 5
Switch(config-if)# end
Switch(config)# interface fastethernet 0/0/0
Switch(config-if)# no cdp enable
Switch(config-if)# bridge-group 5
Switch(config-if)# end
Switch(config)# bridge 5 protocol ieee
```

Configuring Packet Flooding on a PVC

Typically, a specific static map list configuration is not required for bridging to occur. In case of packet flooding, the bridging mechanism individually sends the packet to be flooded on all PVCs configured on the interface. To restrict the broadcast of the packets to only a subset of the configured PVCs you must define a separate static map list. Use the **broadcast** keyword in the **static-map** command to restrict packet broadcasting.

	Command	Purpose
Step 1	Switch(config)# **interface atm** *card*/*subcard*/*port* Switch(config-if)#	Specifies the interface to configure on the ATM router module.
Step 2	Switch(config-if)# **no ip address**	Disables IP processing.
Step 3	Switch(config-if)# **no ip directed-broadcast**	Disables the translation of directed broadcasts to physical broadcasts.
Step 4	Switch(config-if)# **map-group** *number*	Enters the map group name associated with this PVC.
Step 5	Switch(config-if)# **atm pvc 2** *vci-A* **interface atm** *card*/*subcard*/*port vpi-B*	Configures a PVC. **Note** The VPI number on the ATM router module interface must be 2.
Step 6	Switch(config-if)# **bridge-group** *number*	Assigns the interface to a bridge group.
Step 7	Switch(config-if)# **end** Switch(config)#	Returns to global configuration mode.
Step 8	Switch(config)# **map-list** *name* Switch(config-map-list)#	Creates a map list by naming it, and enters map-list configuration mode.
Step 9	Switch(config-map-list)# **bridge atm-vc** *number* **broadcast**	Enables packet flooding on a PVC.

Example

In the following example only PVC 2, 200 is used for packet flooding:

```
Switch(config)# interface atm 3/0/0
Switch(config-if)# no ip address
Switch(config-if)# no ip directed-broadcast
Switch(config-if)# map-group bg_1
Switch(config-if)# atm pvc 2 200 interface atm 1/0/1 0 200
Switch(config-if)# atm pvc 2 201 interface atm 1/0/1 0 300
Switch(config-if)# bridge-group 5
Switch(config-if)# end
Switch(config)# map-list bg_1
Switch(config-map-list)# bridge atm-vc 200 broadcast
```

Note For more information about bridging, refer to the *Layer 3 Software Configuration Guide*.

Displaying the Bridging Configuration

To display the bridging configuration on the ATM router module interface, use the following privileged EXEC command:

Command	Purpose
show bridge verbose	Displays the entries in the bridge forwarding database.

Example

```
Switch# show bridge verbose

Total of 300 station blocks, 297 free
Codes: P - permanent, S - self
BG Hash      Address      Action  Interface         VC    Age   RX count   TX count
 5 28/0   0000.0ce4.341c forward  Fa0/0/0              -
 5 2A/0   0000.0cac.be94 forward  ATM3/0/0             200
 5 FA/0   0060.3e59.c63c forward  Fa0/0/0              -
```

Configuring IP Multicast

To configure IP multicast over an RFC 1483 permanent virtual connection (PVC) on an ATM router module, use the following commands, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **ip multicast-routing**	Enables IP multicast routing.
Step 2	Switch(config)# **interface atm** *card*/*subcard*/*port***.***subinterface#* **multipoint** Switch(config-subif)#	Creates the ATM router module point-to-multipoint subinterface, and enters subinterface mode. **Note** The ATM router module only supports point-to-multipoint subinterfaces.
Step 3	Switch(config-subif)# **map-group** *name*	Enters the map group name associated with this PVC.
Step 4	Switch(config-subif)# **atm pvc 2** *vci-a* [**upc** *upc*] [**pd** *pd*] **interface atm** *card*/*subcard*/*port*[**.***vpt#*] *vpi-b vci-b* [**upc** *upc*] **encap aal5snap**	Configures the PVC. **Note** The VPI number on the ATM router module interface must be 2.
Step 5	Switch(config-subif)# **ip pim dense-mode**	Enables Protocol Independent Multicast dense mode on the subinterface.
Step 6	Switch(config-subif)# **exit** Switch(config)#	Returns to global configuration mode.
Step 7	Switch(config)# **map-list** *name* Switch(config-map-list)#	Creates a map list by naming it, and enters map-list configuration mode.
Step 8	Switch(config-map-list)# **ip** *ip-address* {**atm-nsap** *address* | **atm-vc** *vci*} **broadcast**	Associates a protocol and address with a specific virtual circuit.
Step 9	Switch(config-map-list)# **end** Switch#	Returns to privileged EXEC mode.

Example

```
Switch(config)# ip multicast-routing
Switch(config)# interface atm 1/0/0.1011 multipoint
Switch(config-subif)# ip address 10.1.1.1 255.255.255.0
Switch(config-subif)# map-group net1011
Switch(config-subif)# atm pvc 2 1011 interface atm 3/0/0 0 1011 encap aal5snap
Switch(config-subif)# ip pim dense-mode
Switch(config-subif)# exit
Switch(config)# map-list net1011
Switch(config-map-list)# ip 10.1.1.2 atm-vc 1011 broadcast
```

Note For more information on IP multicast, refer to the *Layer 3 Software Configuration Guide*.

CHAPTER 22

Managing Configuration Files, System Images, and Functional Images

This chapter describes some fundamental tasks you perform to maintain the configuration files, system images, and hardware functional images used by your ATM switch router.

Note This chapter provides advanced configuration instructions for the Catalyst 8540 MSR, Catalyst 8510 MSR, and LightStream 1010 ATM switch routers. For complete descriptions of the commands mentioned in this chapter, refer to the *ATM Switch Router Command Reference* publication.

This chapter includes the following sections:

Check the information in the first sections of the chapter to determine if it applies to your installation. Also, familiarize yourself with the Cisco IOS File System section, as this describes new features in this release. If you are an experienced IOS user, you can skip the third section.

Configuring a Static IP Route

If you are managing the ATM switch router through an Ethernet interface or ATM subinterface on the multiservice route processor, and your management station or Trivial File Transfer Protocol (TFTP) server is on a different subnet than the ATM switch router, you must first configure a static IP route.

Caution Failure to configure a static IP route prior to installing the new image will result in a loss of remote administrative access to the ATM switch router. If this happens, you can regain access from a direct console connection, although this requires physical access to the console port.

To configure a static IP route, perform the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	Switch(config)# **ip route** *prefix*[1] *mask*[2] **ethernet 0** | **atm 0**[**.***subinterface#*]	Configures a static IP route on the Ethernet interface or ATM subinterface of the route processor.
Step 2	Switch(config)# **end** Switch#	Returns to privileged EXEC mode.
Step 3	Switch# **copy system:running-config nvram:startup-config**	Saves the configuration to NVRAM.

1. The IP route prefix of the remote network where the management station or TFTP server resides.
2. The subnet mask of the remote network where the management station or TFTP server resides.

The following example shows how to configure an IP address on the main Ethernet port, then save the configuration.

```
Switch(config)# interface ethernet 0
Switch(config-if)# ip address 172.20.52.11 255.255.255.224
Switch(config-if)# end
Switch# copy system:running-config nvram:startup-config
```

Understanding the Cisco IOS File System

This release of the ATM switch router system software uses the Cisco IFS (IOS File System). With IFS, you now access files on a storage device by specifying a filename and the file system containing the file. The following old command, for example, accesses the running-config and startup-config files:

```
Switch# copy running-config startup-config
```

With IFS, you additionally specify the system containing the files using the syntax *filesystem***:***filename*. For example:

```
Switch# copy system:running-config nvram:startup-config
```

The syntax *filesystem***:***filename* is called the file URL. In addition, remote file systems (such as TFTP, FTP, and rcp) allow you to specify additional options in the file URL, such as username, password, remote host, and so on. This way, you can enter all the required information at once without having to respond to prompts.

With IFS, some show commands have been replaced with more commands. For example:

```
Switch# show running-config
```

has been replaced with the following command:

```
Switch# more system:running-config
```

For complete information on using file URLs and the new IFS commands and syntax, refer to the *Configuration Fundamentals Configuration Guide* and the *Configuration Fundamentals Command Reference* publications.

File Systems and Memory Devices

File systems on the ATM switch router include read-only memory (RAM, or system), Flash memory (such as bootflash and the Flash PC cards in slot0 and slot1), and remote file systems (such as TFTP or rcp servers).

You can use the **show file systems** privileged EXEC command to display the valid file systems on your ATM switch router. The following example shows the file systems on a Catalyst 8540 MSR:

```
Switch# show file systems
File Systems:

     Size(b)     Free(b)      Type  Flags  Prefixes
*   20578304     8984376     flash     rw   slot0: flash:
     7995392      118192     flash     rw   slot1:
     7602176      636256     flash     rw   bootflash:
           -           -   unknown     rw   rcsf:
           -           -    opaque     rw   null:
           -           -    opaque     rw   system:
           -           -   network     rw   tftp:
      520184      517855     nvram     rw   nvram:
           -           -   network     rw   rcp:
           -           -   network     rw   ftp:
     5242880           0    opaque     ro   atm-acct-ready:
     5242880     5242880    opaque     ro   atm-acct-active:
    20578304     5264212     flash     rw   sec-slot0:
           -           -     flash     rw   sec-slot1:
     7602176      641048     flash     rw   sec-bootflash:
      520184      517855     nvram     rw   sec-nvram:
           -           -     nvram     rw   sec-rcsf:
```

File System Tasks

Refer to the *Configuration Fundamentals Configuration Guide* for details on the following frequently performed tasks:

- Format flash memory on a new Flash PC card or on any Flash memory device that has locked blocks or failed sectors
- Manage files on file systems, including setting the default file system, listing files on a file system, deleting and recovering files, and so on.

Maintaining System Images and Configuration Files

The following sections list common tasks you perform to maintain system images and configuration files on your ATM switch router:

For detailed instructions on performing these tasks, refer to the *Configuration Fundamentals Configuration Guide*.

Modifying, Downloading, and Maintaining Configuration Files

The following are frequently performed tasks to maintain configuration files:

- Copy configuration files from the ATM switch router to a network server—You can copy files to a TFTP server or rcp server for backup purposes or to store alternative configurations.
- Copy configuration files from a network server to the ATM switch router—You can copy configuration files from a TFTP server or an rcp server to the running configuration or startup configuration of the ATM switch router to restore a configuration, to use a configuration from another device, or to ensure that you have the same configuration on several devices.
- Maintain configuration files larger than NVRAM—You can maintain configuration files larger than NVRAM by compressing them, storing them on Flash memory devices, or storing them on TFTP or rcp servers for downloading at system startup.
- Copy configuration files between different locations—You can copy configuration files from Flash memory to the startup or running configuration, copy configuration files between Flash memory devices, or copy a configuration file from a server to Flash memory.
- Reexecute the configuration commands in startup configuration or clear the configuration information.

Modifying, Downloading, and Maintaining System Images

The following are frequently performed tasks to maintain system image files:

- Copy images from Flash memory to a network server—You can store system images for backup or other purposes by copying them from a Flash memory device to a TFTP or rcp server.
- Copy images from a network server to Flash memory—You perform this procedure when upgrading your system image or functional image.
- Copy images between local Flash memory devices.

Rebooting and Specifying Startup Information

The following commonly performed tasks are used to reboot the ATM switch router and specify startup information:

- Modify the configuration register boot field—You use the configuration register boot field to specify whether the ATM switch router loads a system image, and where it obtains the system image, or whether the system image loads from ROM.
- Specify the system startup image—You can enter multiple **boot** commands in the startup configuration file or in the BOOT environment variable to provide main and alternative methods for loading a system image onto the ATM switch router.
- Specify the startup configuration file—You can configure the CONFIG_FILE environment variable to load the startup configuration file from NVRAM (the default), from a Flash memory device, or from a network server.
- Enter ROM monitor mode or manually load a system image from ROM monitor if a valid system image is not found or if the configuration file is corrupted.

Additional File Transfer Features

The following file configuration file transfer options are also available:

- Configure the ATM switch router as a TFTP server to provide other devices on the network with system images and configuration files.
- Configure the ATM switch router to use the remote copy protocol (rcp) and remote shell (rsh) protocol—With rsh you can execute commands remotely; with rcp, you can copy files to and from a file system residing on a remote host or network server.

Maintaining Functional Images (Catalyst 8540 MSR)

You can load functional images used by certain hardware controllers in the ATM switch router. This section describes the function and maintenance of functional image.

Understanding Functional Images (Catalyst 8540 MSR)

Functional images provide the low-level operating functionality for various hardware controllers. On hardware controllers with insystem programmable devices, such as field programmable gate arrays (FPGAs) and Erasable Programmable Logic Devices (EPLDs), the hardware functional images can be reprogrammed independently of loading the system image and without removing the devices from the controller.

On the ATM switch router, you can reprogram the functional images on the route processors, rommon, switch processors, switch processor feature cards, carrier modules, full-width modules, and network clock modules.

All new hardware is shipped with functional images preloaded. Loading a different functional image is required only when upgrading or downgrading functional image versions.

Loading Functional Images (Catalyst 8540 MSR)

You load a functional image in two steps:

Step 1 Copy the image to a Flash memory device (bootflash, slot0, or slot1). For instructions on copying files to a Flash memory device, refer to the *Configuration Fundamentals Configuration Guide*.

Step 2 Load the image from the Flash memory device to the hardware controller.

Note The command for loading functional images on the ATM switch router differs from that described in the Cisco IOS documentation.

To download a functional image from a Flash memory device to a hardware controller, use the following command in privileged EXEC mode:

Command	Purpose
reprogram *device***:***filename* {*slot* [*subcard*] | **rommon**}	Loads the functional image with the specified filename to a device.

The **reprogram** command checks the compatibility of the image for the selected card type before downloading the functional image. If you have specified a slot number without a subcard, the functional image is downloaded to the full-width module that occupies that slot.

Note After loading a new functional image on the primary route processor or on one of the switch processors, you must power-cycle the switch for the hardware to reconfigure itself with the new image.

Caution Do not interrupt the download procedure. Wait until it has finished before attempting any commands on the switch.

Example

The following example demonstrates loading the functional image fi_c8540_rp.B.3_91 from the Flash PC card in slot 0 to the controller for the route processor in slot 4.

```
Switch# reprogram slot0:fi_c8540_rp.B.3_91 4
```

Displaying the Functional Image Information (Catalyst 8540 MSR)

To display the functional image version in a hardware controller, use the following command in privileged EXEC mode:

Command	Purpose
show functional-image-info {**slot** *slot* | **subslot** *slot*/*subslot*}	Displays the functional image information.

The following example shows the functional image information in the controller for the route processor module in slot 4:

```
Switch# show functional-image-info slot 4

Details for cpu Image on slot: 4

Functional Version of the FPGA Image: 4.8
 #Jtag-Distribution-Format-B
 #HardwareRequired: 100(3.0-19,4.0-19,5.0-19)
 #FunctionalVersion: 4.8
 #Sections: 1
 #Section1Format: MOTOROLA_EXORMAX

  Copyright (c) 1996-00 by cisco Systems, Inc.
  All rights reserved.
```

```
  generated by:       holliday
  on:                 Mon Mar  6 13:59:17 PST 2000
  using:              /vob/cougar/bin/jtag_script Version 1.13
  config file:        cpu.jcf

  Chain description:
  Part type Bits Config file
  10k50     10   ../cidrFpga2/max/cidr_fpga.ttf
  xcs4062   3    ../cubiFpga2/xil/cubi.bit
  xcs4062   3    ../cubiFpga2/xil/cubi.bit
  generic   2
  XC4005    3    /vob/cougar/custom/common/jtcfg/xil/jtcfg_r.bit
  Number devices             = 5
  Number of instruction bits = 21

  FPGA config file information:
  Bitgen date/time  Sum   File
  100/03/02 19:14:49 7068  ../cidrFpga2/max/cidr_fpga.ttf
  1999/04/15 18:46:32 36965 ../cubiFpga2/xil/cubi.bit
  1999/04/15 18:46:32 36965 ../cubiFpga2/xil/cubi.bit
  98/06/11 16:56:44 49904 /vob/cougar/custom/common/jtcfg/xil/jtcfg_r.bit
#End-Of-Header
```

Maintaining Functional Images (Catalyst 8510 MSR and LightStream 1010)

You can load functional images used by certain hardware controllers in the ATM switch router. This section describes the function and maintenance of functional images.

Understanding Functional Images (Catalyst 8510 MSR and LightStream 1010)

Functional images provide the low-level operating functionality for various hardware controllers. On hardware controllers with insystem programmable devices, such as Field Programmable Gate Arrays (FPGAs) and Erasable Programmable Logic Devices (EPLDs), the hardware functional images can be reprogrammed independently of loading the system image and without removing the devices from the controller.

Note You can currently reprogram the functional image on the channelized DS3 and channelized E1 Frame Relay port adapters.

All new hardware is shipped with functional images preloaded. Loading a different functional image is required only when upgrading or downgrading functional image versions.

Loading Functional Images (Catalyst 8510 MSR and LightStream 1010)

You load a functional image in two steps:

Step 1 Copy the image to a Flash memory device (bootflash, slot0, or slot1). For instructions on copying files to a Flash memory device, refer to the *Configuration Fundamentals Configuration Guide*.

Step 2 Load the image from the Flash memory device to the hardware controller.

Note The command for loading functional images on the ATM switch router differs from that described in the Cisco IOS documentation.

To download a functional image from a Flash memory device to a hardware controller, use the following command in privileged EXEC mode:

Command	Purpose
reprogram *device***:***filename* {*slot* [*subcard*] | **rommon**}	Loads the functional image with the specified filename to a device.

The **reprogram** command checks the compatibility of the image for the selected card type before downloading the functional image.

Caution Do not interrupt the download procedure. Wait until it has finished before attempting any commands on the switch.

Example

The following example demonstrates loading the functional image abr_tmp.exo from the Flash PC card in slot 0 to the controller in slot 0, subcard 1:

```
Switch# reprogram slot0:abr_tmp.exo 0 1
```

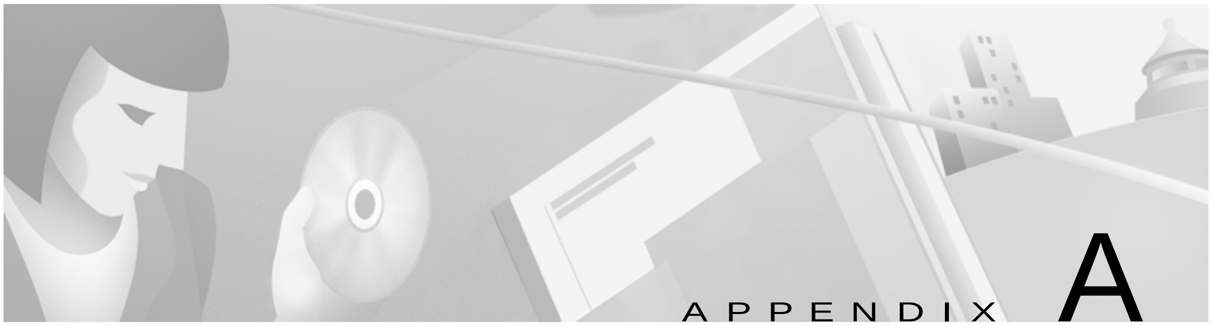
Displaying the Functional Image Information (Catalyst 8510 MSR and LightStream 1010)

To display the functional image version in a hardware controller, use the following command in privileged EXEC mode:

Command	Purpose
show functional-image-info {**slot** *slot* | **subslot** *slot*/*subcard*}	Displays the functional image information.

The following example shows the functional image information for the module in slot 4, subcard 0:

```
Switch# show functional-image-info subslot 4/0
###HardwareRequired   : B8(3.2)
##FunctionalVersion  : 2.3
##Sections           : 1
##Section1Format     : BINARY, length = 303016
# PUMA-4CE1 Firmware image
# Firmware Image       : fi-c8510-4e1fr.2_3
#
# EPLD config file  : C85MS-4E1-FRRJ48.jcf
# Chain description:
# Part type   Bits  Config file
# EPM7256S    10    /cougar/custom/puma/pld/testbench/PROG_FILES/4CE1/PLD/DB/7256.pof
# EPM7064S    10    /cougar/custom/puma/pld/testbench/PROG_FILES/4CE1/PLD/DB/7064.pof
# EPM7064S    10    /cougar/custom/puma/pld/testbench/PROG_FILES/4CE1/PLD/MB/7064.pof
# Number devices    = 3
# Number of instruction bits = 30
#
# FPGA config file information:
###End-of-header
```



APPENDIX A

PNNI Migration Examples

This appendix provides examples of how to migrate a flat network topology to a Private Network-Network Interface (PNNI) hierarchical network topology, and includes the following sections:

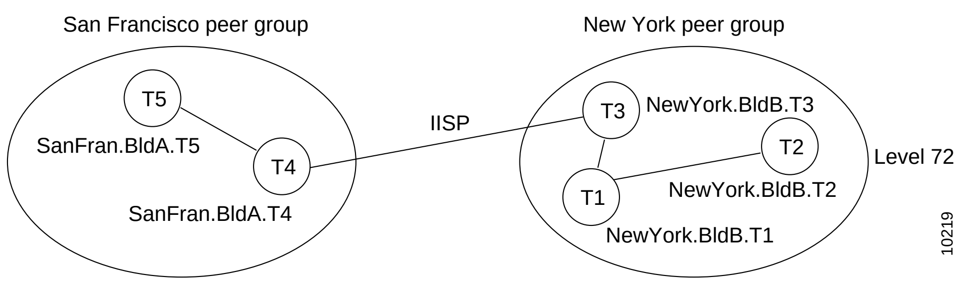
- Adding a Higher Level of PNNI Hierarchy, page A-1
- Adding a New Lowest Level of PNNI Hierarchy, page A-7

Note Detailed PNNI configuration instructions are described in the chapter Chapter 10, “Configuring ATM Routing and PNNI.” For a functional description of hierarchical PNNI, refer to the *Guide to ATM Technology*. For a complete description of the commands mentioned in this chapter, refer to the *ATM Switch Router Command Reference* publication.

Adding a Higher Level of PNNI Hierarchy

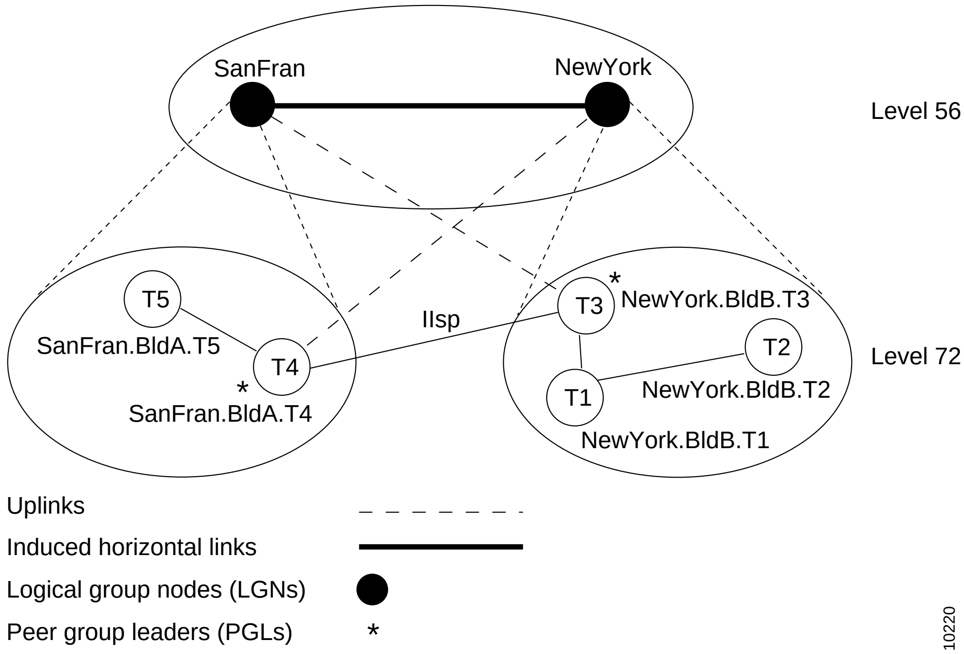
Figure A-1 shows an example network with two PNNI peer groups connected by an Interim Inter-Switch Signalling Protocol (IISP) interface.

Figure A-1 Two PNNI Peer Groups Connected by an IISP Interface



You can convert the network to a single hierarchical PNNI routing domain by configuring a second level of hierarchy in each peer group and converting the IISP interface to a PNNI interface, as shown in Figure A-2.

Figure A-2 Two-Level PNNI Hierarchical Network



The initial configuration for each ATM switch router is shown in the sections that follow. The commands used to migrate the network to a two-level PNNI hierarchical network (shown in Figure A-2) are also provided.

Switch T1 Initial Configuration

The initial configuration for switch NewYork BldB.T1 follows:

```
hostname NewYork.BldB.T1
atm address 47.0091.4455.6677.1144.1011.1233.0060.3e7b.3a01.00
atm router pnni
 node 1 level 72 lowest
  redistribute atm-static
```

Switch T2 Initial Configuration

The initial configuration for switch NewYork BldB.T2 follows:

```
hostname NewYork.BldB.T2
atm address 47.0091.4455.6677.1144.1011.1244.0060.3e5b.bc01.00
atm router pnni
 node 1 level 72 lowest
  redistribute atm-static
```

To display the reachability information, use the **show atm route** command.

```
NewYork.BldB.T2# show atm route

Codes: P - installing Protocol (S - Static, P - PNNI, R - Routing control),
       T - Type (I - Internal prefix, E - Exterior prefix, SE -
                 Summary Exterior prefix, SI - Summary Internal prefix,
                 ZE - Suppress Summary Exterior, ZI - Suppress Summary Internal)

P  T Node/Port        St Lev Prefix
~ ~~ ~~~~~~~~~~~~~~~~ ~~ ~~~ ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
P  I 9   0            UP 0   47.0091.4455.6677.1144.1011.1233/104
P SI 1   0            UP 0   47.0091.4455.6677.1144.1011.1244/104
R  I 1   ATM2/0/0     UP 0   47.0091.4455.6677.1144.1011.1244.0060.3e5b.bc01/152
R  I 1   ATM2/0/0     UP 0   47.0091.4455.6677.1144.1011.1244.0060.3e5b.bc02/152
R  I 1   ATM2/0/0     UP 0   47.0091.4455.6677.1144.1011.1244.4000.0c/128
P  I 11  0            UP 0   47.0091.4455.6677.1144.1011.1255/104
P  E 11  0            UP 0   47.0091.4455.6677.22/64
S  E 1   ATM0/0/1     DN 0   47.0091.8200.0001.1/60
```

Switch T3 Initial Configuration

The initial configuration for switch NewYork BldB.T3 follows:

```
hostname NewYork.BldB.T3
atm address 47.0091.4455.6677.1144.1011.1255.0060.3e5b.c401.00
atm router pnni
 node 1 level 72 lowest
  redistribute atm-static

interface ATM0/0/2
 no ip address
atm route 47.0091.4455.6677.22... ATM0/0/2
```

To display the reachability information, use the **show atm route command**. To display the interface type, use the **show atm interface** command:

```
NewYork.BldB.T3# show atm interface atm 0/0/2

Interface:      ATM0/0/2        Port-type:      oc3suni
IF Status:      UP              Admin Status:   up
Auto-config:    enabled         AutoCfgState:   completed
IF-Side:        Network         IF-type:        IISP
Uni-type:       not applicable  Uni-version:    V4.0

<information deleted>
```

Note In the example, the interface type of interface atm 0/0/2 on NewYork.BldB.T3 is determined using Integrated Local Management Interface (ILMI) autoconfiguration. Because the other side of the link on SanFran.BldA.T4 is configured as IISP, the interface type is determined to be IISP. When using ILMI autoconfiguration on one side of the link and manually configuring the other side as IISP, be careful to specify the configured side as either the user or network side, depending on whether it has the larger value of atmfMySystemIdentifier.

Switch T4 Initial Configuration

The initial configuration for switch SanFran.BldA.T4 follows:

```
hostname SanFran.BldA.T4
atm address 47.0091.4455.6677.2233.1011.1266.0060.3e7b.2001.00
atm router pnni
 node 1 level 72 lowest
  redistribute atm-static

interface ATM0/0/3
 no ip address
 no atm auto-configuration
 atm iisp side user version 4.0
atm route 47.0091.4455.6677.11... ATM0/0/3
```

To display the reachability information, use the **show atm route command**. To display the interface type, side, and version, use the **show atm interface** command:

```
SanFran.BldA.T4# show atm interface atm 0/0/3

Interface:      ATM0/0/3        Port-type:      oc3suni
IF Status:      UP              Admin Status:   up
Auto-config:    disabled        AutoCfgState:   not applicable
IF-Side:        User            IF-type:        IISP
Uni-type:       not applicable  Uni-version:    V4.0
```

Switch T5 Initial Configuration

The initial configuration for switch SanFran.BldA.T5 follows:

```
hostname SanFran.BldA.T5
atm address 47.0091.4455.6677.2233.1011.1244.0060.3e7b.2401.00
atm router pnni
 node 1 level 72 lowest
  redistribute atm-static
```

Configuring Second Level of PNNI Hierarchy on Switches T3 and T4

The following example shows how to configure and display the second level of PNNI hierarchy on switches NewYork.BldB.T3 and SanFran.BldA.T4 (see Figure A-2):

Note In this example, the configuration of the second level of PNNI hierarchy on switch NewYork.BldB.T3 or switch SanFran.BldA.T4 has no effect on new or existing connections.

```
NewYork.BldB.T3# configure terminal
NewYork.BldB.T3(config)# atm router pnni
NewYork.BldB.T3(config-atm-router)# node 2 level 56
NewYork.BldB.T3(config-pnni-node)# name NewYork
NewYork.BldB.T3(config-pnni-node)# exit
NewYork.BldB.T3(config-atm-router)# node 1
NewYork.BldB.T3(config-pnni-node)# parent 2
NewYork.BldB.T3(config-pnni-node)# election leadership-priority 45
NewYork.BldB.T3(config-pnni-node)# end
NewYork.BldB.T3#
```

```
SanFran.BldA.T4# configure terminal
SanFran.BldA.T4(config)# atm router pnni
SanFran.BldA.T4(config-atm-router)# node 2 level 56
SanFran.BldA.T4(config-pnni-node)# name SanFran
SanFran.BldA.T4(config-pnni-node)# exit
SanFran.BldA.T4(config-atm-router)# node 1
SanFran.BldA.T4(config-pnni-node)# parent 2
SanFran.BldA.T4(config-pnni-node)# election leadership-priority 45
SanFran.BldA.T4(config-pnni-node)# end
SanFran.BldA.T4#
```

Use the following commands to confirm the creation of the PNNI hierarchy:

```
SanFran.BldA.T4# show atm pnni local-node

PNNI node 1 is enabled and running
  Node name: SanFran.BldA.T4
  System address          47.00914455667722331011126 6.00603E7B2001.01
  Node ID          72:160:47.009144556677223310111266.00603E7B2001.00
  Peer group ID        72:47.0091.4455.6677.2233.0000.0000
  Level 72, Priority 45 95, No. of interfaces 3, No. of neighbors 1
  Parent Node Index: 2

<information deleted>

PNNI node 2 is enabled and running
  Node name: SanFran
  System address          47.009144556677223310111266.00603E7B2001.02
  Node ID           56:72:47.009144556677223300000000.00603E7B2001.00
  Peer group ID        56:47.0091.4455.6677.0000.0000.0000
  Level 56, Priority 0 0, No. of interfaces 0, No. of neighbors 0
  Parent Node Index: NONE

<information deleted>

SanFran.BldA.T4# show atm pnni hierarchy
  Locally configured parent nodes:
  Node          Parent
  Index  Level  Index   Local-node Status     Node Name
  ~~~~~  ~~~~~  ~~~~~~  ~~~~~~~~~~~~~~~~~~~~  ~~~~~~~~~~~~~~~~~~~~~~
  1      72     2       Enabled/ Running      SanFran.BldA.T4
  2      56     N/A     Enabled/ Running      SanFran

SanFran.BldA.T4# show atm pnni hierarchy network
  Summary of active parent LGNs in the routing domain:
  Node  Level  Parent  Node Name
  ~~~~  ~~~~~  ~~~~~~  ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
  1     72     2       SanFran.BldA.T4
  2     56     0       SanFran

SanFran.BldA.T4# show atm pnni hierarchy network detail
  Detailed hierarchy network display:
  Number Of Network LGN Ancestors: 1

  Lowest Level (72) information:
  Node No.....: 1    Node Name: SanFran.BldA.T4
  Node’s ID...:   72:160:47.009144556677223310111266.00603E7B2001.00
  Node’s Addr.:          47.009144556677223310111266.00603E7B2001.01
  Node’s PG ID:       72:47.0091.4455.6677.2233.0000.0000
  PGL No......: 1    PGL Name: SanFran.BldA.T4
  PGL ID......:   72:160:47.009144556677223310111266.00603E7B2001.00
```

```
Level 56 ancestor information:
Parent LGN..: 2    LGN Name: SanFran
LGN’s ID....:    56:72:47.00914455667722330000000.00603E7B2001.00
LGN’s Addr..:          47.00914455667722331011126 6.00603E7B2001.02
LGN’s PG ID.:       56:47.0091.4455.6677.0000.0000.0000
LGN PGL No..:      Unelected or unknown
LGN’s PGL ID:      0:0:00.000000000000000000000000.000000000000.00
```

Configuring the Link Between Switch T3 and Switch T4 for PNNI

The following example shows how to configure the link between switch NewYorkBldB.T3 and SanFran.BldA.T4 for PNNI.

Note In this example, only one side of the IISP interface is configured to change the link from IISP to PNNI because the other side of the link is using ILMI autoconfiguration for the interface type. You can use either the **atm auto-configuration** or **atm nni** command to change the link from IISP to PNNI.

```
SanFran.BldA.T4# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
SanFran.BldA.T4(config)# interface atm 0/0/3
SanFran.BldA.T4(config-if)# atm auto-configuration
SanFran.BldA.T4(config-if)# end
SanFran.BldA.T4#
%ATM-5-ATMSOFTSTART: Restarting ATM signalling and ILMI on ATM0/0/3.
```

Note When you change the link from IISP to PNNI, all existing connections across the interface are cleared. The ability to route new connections across the link is restored within a few seconds, when the PNNI uplinks and induced horizontal link come up.

Verifying Connectivity to All ATM Addresses and Deleting an Old Static Route on Switches T4 and T3

The following example shows how to verify connectivity to all ATM addresses before deleting an old static route on switch T4:

```
SanFran.BldA.T4# show atm route

Codes: P - installing Protocol (S - Static, P - PNNI, R - Routing control),
       T - Type (I - Internal prefix, E - Exterior prefix, SE -
                 Summary Exterior prefix, SI - Summary Internal prefix,
                 ZE - Suppress Summary Exterior, ZI - Suppress Summary Internal)

P  T Node/Port        St Lev Prefix
~ ~~ ~~~~~~~~~~~~~~~~ ~~ ~~~ ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
S  E 1   ATM0/0/3     DN 0   47.0091.4455.6677.11/64
P  I 12  0            UP 0   47.0091.4455.6677.1144/72
P SI 2   0            UP 0   47.0091.4455.6677.2233/72
P  I 9   0            UP 0   47.0091.4455.6677.2233.1011.1244/104
P SI 1   0            UP 0   47.0091.4455.6677.2233.1011.1266/104
R  I 1   ATM2/0/0     UP 0   47.0091.4455.6677.2233.1011.1266.0060.3e7b.2001/152
R  I 1   ATM2/0/0     UP 0   47.0091.4455.6677.2233.1011.1266.0060.3e7b.2002/152
R  I 1   ATM2/0/0     UP 0   47.0091.4455.6677.2233.1011.1266.4000.0c/128
```

The following example shows how to delete the old static route from switch T4:

```
SanFran.BldA.T4# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
SanFran.BldA.T4(config)# no atm route 47.0091.4455.6677.11 atm0/0/3
SanFran.BldA.T4(config)# end
SanFran.BldA.T4#
```

The following example verifies that the old static route on switch T4 has been deleted:

```
SanFran.BldA.T4# show atm route

Codes: P - installing Protocol (S - Static, P - PNNI, R - Routing control),
       T - Type (I - Internal prefix, E - Exterior prefix, SE -
                 Summary Exterior prefix, SI - Summary Internal prefix,
                 ZE - Suppress Summary Exterior, ZI - Suppress Summary Internal)

P  T Node/Port        St Lev Prefix
~ ~~ ~~~~~~~~~~~~~~~~ ~~ ~~~ ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
P  I 12  0            UP 0   47.0091.4455.6677.1144/72
P SI 2   0            UP 0   47.0091.4455.6677.2233/72
P  I 9   0            UP 0   47.0091.4455.6677.2233.1011.1244/104
P SI 1   0            UP 0   47.0091.4455.6677.2233.1011.1266/104
R  I 1   ATM2/0/0     UP 0   47.0091.4455.6677.2233.1011.1266.0060.3e7b.2001/152
R  I 1   ATM2/0/0     UP 0   47.0091.4455.6677.2233.1011.1266.0060.3e7b.2002/152
R  I 1   ATM2/0/0     UP 0   47.0091.4455.6677.2233.1011.1266.4000.0c/128
```

The following example shows how to delete the old static route from switch T3:

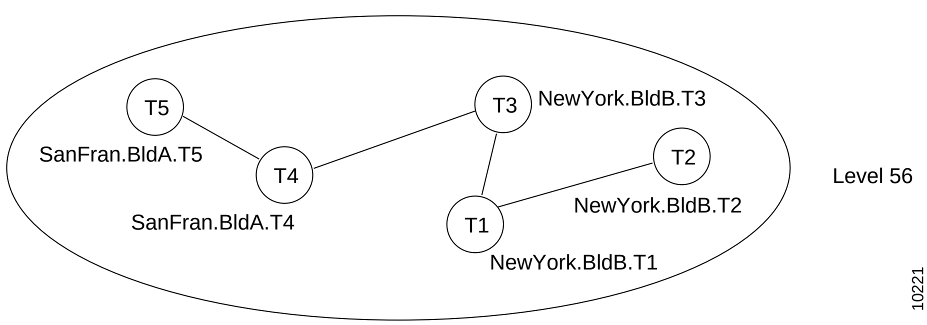
```
NewYork.BldB.T3# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
NewYork.BldB.T3(config)# no atm route 47.0091.4455.6677.22 atm 0/0/2
NewYork.BldB.T3(config)# end
NewYork.BldB.T3#
```

To verify the deletion of the old static route on switch T3, use the **show atm route** command.

Adding a New Lowest Level of PNNI Hierarchy

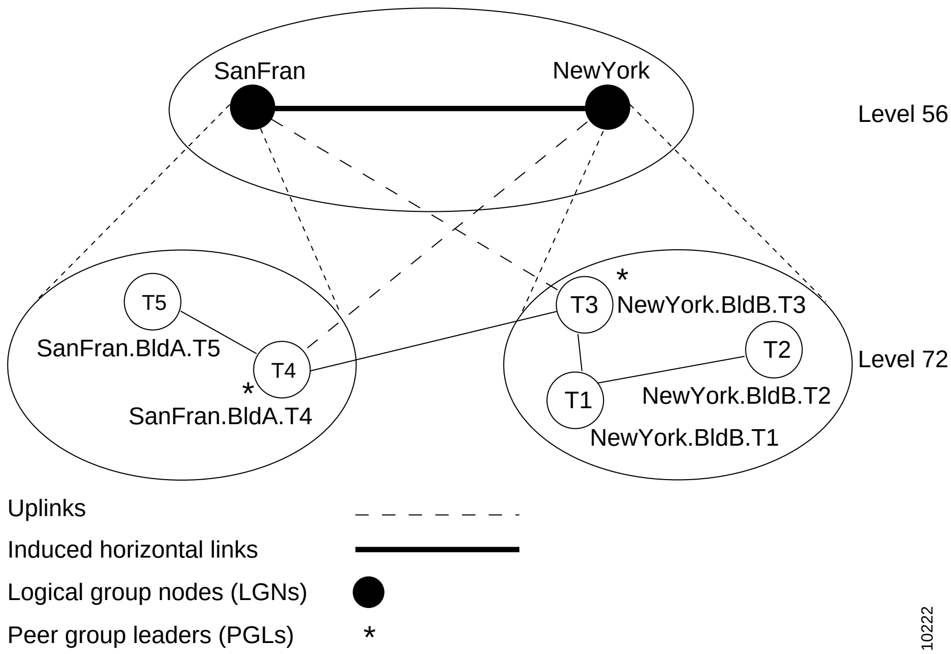
Figure A-3 shows an example network configured with only one level of PNNI hierarchy at level 56.

Figure A-3 One-Level PNNI Hierarchical Network



You can convert the network into a two-level hierarchical PNNI network by bringing each lowest level node down to level 72 and splitting the network into two peer groups. At the same time, you can add a second level of hierarchy at level 56. The resulting network topology is shown in Figure A-4.

Figure A-4 Two-Level PNNI Hierarchical Network



Note This example assumes that all addresses have already been assigned according to a hierarchical ATM address plan. All the ATM switch routers share the same 56-bit prefix. The ATM switch routers in Building A in San Francisco share the same 72-bit prefix. The ATM switch routers in Building B in New York share a different 72-bit prefix. As a result, no renumbering is necessary to migrate the network from a single level of PNNI hierarchy to two levels of PNNI hierarchy.

Note If no renumbering is necessary and all ATM switch routers are peer group leader/logical group node (PGL/LGN)-capable (Cisco IOS Release 11.3T, WA4, or later releases), existing connections are not affected by the migration process. The existing connections remain active while you modify the PNNI configuration.

You can implement the migration process one ATM switch router at a time. As each ATM switch router is moved down to level 72, the ability to establish new connections across that ATM switch router is lost temporarily and then automatically restored. You can pause for long periods of time during the migration process without any harmful effects.

The initial configuration for each ATM switch router is shown in the sections that follow. The commands used to migrate the network to the two-level PNNI hierarchical network (shown in Figure A-4) are also provided.

Switch T1 Initial Configuration

The initial configuration for switch NewYork BldB.T1 follows:

```
hostname NewYork.BldB.T1
atm address 47.0091.4455.6677.1144.1011.1233.0060.3e7b.3a01.00
atm router pnni
 node 1 level 56 lowest
  redistribute atm-static
```

The following example shows the output from the **show atm route** command for the switch:

```
NewYork.BldB.T1# show atm route

Codes: P - installing Protocol (S - Static, P - PNNI, R - Routing control),
       T - Type (I - Internal prefix, E - Exterior prefix, SE -
                 Summary Exterior prefix, SI - Summary Internal prefix,
                 ZE - Suppress Summary Exterior, ZI - Suppress Summary Internal)

P  T Node/Port        St Lev Prefix
~ ~~ ~~~~~~~~~~~~~~~~ ~~ ~~~ ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
P SI 1   0            UP 0   47.0091.4455.6677.1144.1011.1233/104
R  I 1   ATM2/0/0     UP 0   47.0091.4455.6677.1144.1011.1233.0060.3e7b.3a01/152
R  I 1   ATM2/0/0     UP 0   47.0091.4455.6677.1144.1011.1233.0060.3e7b.3a02/152
R  I 1   ATM2/0/0     UP 0   47.0091.4455.6677.1144.1011.1233.0060.3e7b.3a03/152
R  I 1   ATM2/0/0     UP 0   47.0091.4455.6677.1144.1011.1233.0060.3e7b.3a04/152
R  I 1   ATM2/0/0     UP 0   47.0091.4455.6677.1144.1011.1233.0060.3e7b.3a05/152
R  I 1   ATM2/0/0     UP 0   47.0091.4455.6677.1144.1011.1233.4000.0c/128
P  I 9   0            UP 0   47.0091.4455.6677.1144.1011.1244/104
P  I 10  0            UP 0   47.0091.4455.6677.1144.1011.1255/104
P  I 12  0            UP 0   47.0091.4455.6677.2233.1011.1244/104
P  I 11  0            UP 0   47.0091.4455.6677.2233.1011.1266/104
```

Switch T2 Initial Configuration

The initial configuration for switch NewYork BldB.T2 follows:

```
hostname NewYork.BldB.T2
atm address 47.0091.4455.6677.1144.1011.1244.0060.3e5b.bc01.00
atm router pnni
 node 1 level 56 lowest
  redistribute atm-static
```

Switch T3 Initial Configuration

The initial configuration for switch NewYork BldB.T3 follows:

```
hostname NewYork.BldB.T3
atm address 47.0091.4455.6677.1144.1011.1255.0060.3e5b.c401.00
atm router pnni
 node 1 level 56 lowest
  redistribute atm-static
```

Switch T4 Initial Configuration

The initial configuration for switch SanFran.BldA.T4 follows:

```
hostname SanFran.BldA.T4
atm address 47.0091.4455.6677.2233.1011.1266.0060.3e7b.2001.00
atm router pnni
 node 1 level 56 lowest
  redistribute atm-static
```

Switch T5 Initial Configuration

The initial configuration for switch SanFran.BldA.T5 follows:

```
hostname SanFran.BldA.T5
atm address 47.0091.4455.6677.2233.1011.1244.0060.3e7b.2401.00
atm router pnni
 node 1 level 56 lowest
  redistribute atm-static
```

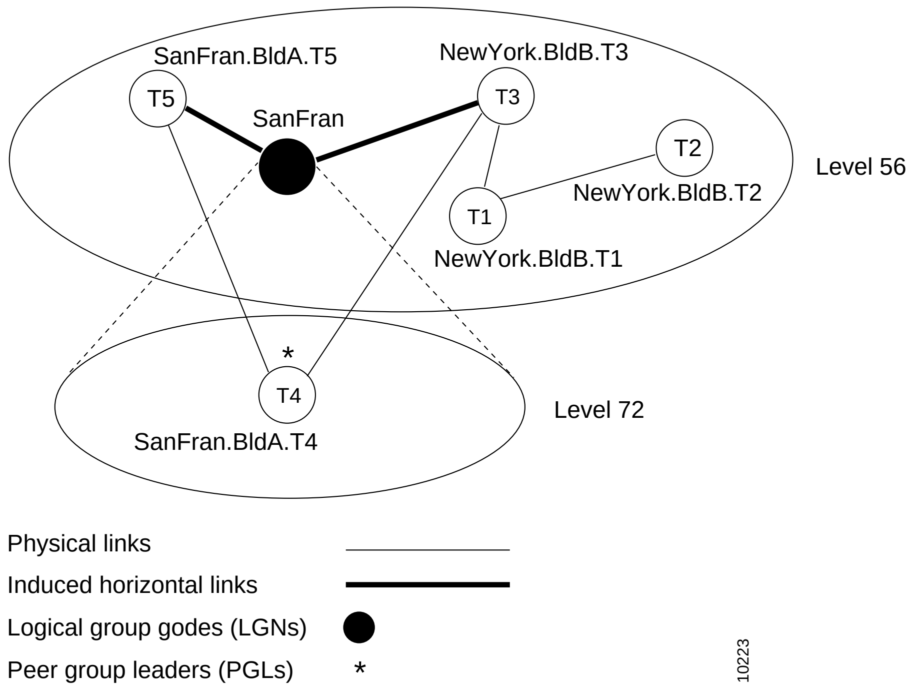
Moving Switch T4 Down into a New Peer Group

The first ATM switch router you move down into a new peer group at level 72 should be the ATM switch router you prefer as the peer group leader (PGL). Before moving down the first ATM switch router, configure the logical group node (LGN) for the second level of hierarchy on the ATM switch router.

Note We recommend that you enter the **no auto-summary** command to disable **auto-summary** on all new LGNs during the migration process. PNNI always routes to the node that advertises the longest matching reachable address prefix; therefore, auto-summary is not required. Furthermore, debugging is easier when **auto-summary** is disabled. If anything goes wrong during the migration process, you can use the **show atm route** command to debug the problem. After all the nodes have been moved into the child peer group represented by the LGN, restore **auto-summary** to reduce the number of reachable address prefixes advertised by the LGN.

Figure A-5 shows the network topology after moving ATM switch router SanFran.BldA.T4 down into a new peer group at level 72 and establishing an LGN representing that peer group at level 56.

Figure A-5 Moving a Switch Down in the PNNI Hierarchy



Although ATM switch router SanFran.BldA.T5 and NewYork.BldB.T3 are not running any PGLs or LGNs in this example, these ATM switch routers must be capable of establishing the PNNI hierarchy. This capability allows them to bring up the induced horizontal links to the LGN SanFran, maintaining PNNI connectivity across the network. For this reason, we recommend that you upgrade all ATM switch routers to Cisco IOS Release 11.3T, WA4 or later, before configuring PNNI hierarchy.

The following example shows how to move switch SanFran.BldA.T4 down into a new peer group:

```
SanFran.BldA.T4# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
SanFran.BldA.T4(config)# atm router pnni
SanFran.BldA.T4(config-atm-router)# node 2 level 56
SanFran.BldA.T4(config-pnni-node)# name SanFran
SanFran.BldA.T4(config-pnni-node)# no auto-summary
SanFran.BldA.T4(config-pnni-node)# exit
SanFran.BldA.T4(config-atm-router)# node 1
SanFran.BldA.T4(config-pnni-node)# election leadership-priority 45
SanFran.BldA.T4(config-pnni-node)# node 1 disable
SanFran.BldA.T4(config-pnni-node)# node 1 level 72
SanFran.BldA.T4(config-pnni-node)# parent 2
SanFran.BldA.T4(config-pnni-node)# node 1 enable
SanFran.BldA.T4(config-pnni-node)# end
SanFran.BldA.T4#
```

Note When you move down the first switch into a new peer group, the ATM switch router cannot establish new connections until it can elect itself PGL. By default, this election process takes approximately 90 seconds, or less if a second ATM switch router is brought into the peer group quickly. After the new configuration on this ATM switch router is stable, the PNNI network is fully functional and new connections can be accepted across all ATM switch routers.

Moving Switch SanFran.BldA.T5 Down into an Existing Peer Group

After you move the first ATM switch router down to form a new peer group, you can move the remaining ATM switch routers down into the peer group one by one. You should move the ATM switch routers down in an order that keeps the peer group contiguous.

The following example shows how to move switch SanFran.BldA.T5 down into a peer group at level 72:

```
SanFran.BldA.T5# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
SanFran.BldA.T5(config)# atm router pnni
SanFran.BldA.T5(config-atm-router)# node 1 disable
SanFran.BldA.T5(config-pnni-node)# node 1 level 72 enable
SanFran.BldA.T5(config-pnni-node)# end
SanFran.BldA.T5#
```

Note When you move an ATM switch router down into an existing peer group, the ability to establish new connections across that ATM switch router is lost temporarily (up to several seconds).

To verify the configuration, use the **show atm pnni local-node** and **show atm pnni hierarchy** commands. For examples of these commands, see the "Configuring Second Level of PNNI Hierarchy on Switches T3 and T4" section on page A-4.

You can configure one or more of the ATM switch routers that have been moved down into the peer group as a backup PGL. The following example shows how to configure SanFran.BldA.T5 as a backup PGL for the peer group SanFran (see Figure A-4):

```
SanFran.BldA.T5# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
SanFran.BldA.T5(config)# atm router pnni
SanFran.BldA.T5(config-atm-router)# node 2 level 56
SanFran.BldA.T5(config-pnni-node)# name SanFran
SanFran.BldA.T5(config-pnni-node)# no auto-summary
SanFran.BldA.T5(config-pnni-node)# exit
SanFran.BldA.T5(config-atm-router)# node 1
SanFran.BldA.T5(config-pnni-node)# election leadership-priority 10
SanFran.BldA.T5(config-pnni-node)# parent 2
SanFran.BldA.T5(config-pnni-node)# end
SanFran.BldA.T5#
SanFran.BldA.T5# show atm pnni local-node

PNNI node 1 is enabled and running
  Node name: SanFran.BldA.T5
  System address          47.00914455667722331011244.00603E7B2401.01
  Node ID           72:160:47.00914455667722331011244.00603E7B2401.00
  Peer group ID        72:47.0091.4455.6677.2233.0000.0000
  Level 72, Priority 10 10, No. of interfaces 2, No. of neighbors 1
  Parent Node Index: 2

<information deleted>

PNNI node 2 is enabled and not running
  Node name: SanFran
  System address          47.00914455667722331011244.00603E7B2401.02
  Node ID            56:72:47.00914455667722330000000.00603E7B2401.00
  Peer group ID        56:47.0091.4455.6677.0000.0000.0000
  Level 56, Priority 0 0, No. of interfaces 0, No. of neighbors 0
  Parent Node Index: NONE

<information deleted>
```

```
SanFran.BldA.T5# show atm pnni hierarchy
  Locally configured parent nodes:
  Node          Parent
  Index  Level  Index   Local-node Status     Node Name
  ~~~~~  ~~~~~  ~~~~~~  ~~~~~~~~~~~~~~~~~~~~  ~~~~~~~~~~~~~~~~~~~~~~
  1      72     2       Enabled/ Running      SanFran.BldA.T5
  2      56     N/A     Enabled/ Not Running  SanFran

SanFran.BldA.T5# show atm pnni hierarchy network
  Summary of active parent LGNs in the routing domain:
  Node  Level  Parent  Node Name
  ~~~~  ~~~~~  ~~~~~~  ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
  1     72     14      SanFran.BldA.T5
  14    56     0       SanFran
```

Restoring Auto-Summary on the LGN SanFran

After all the nodes destined for the new peer group migrate into the peer group, you can restore **auto-summary** to reduce the number of reachable address prefixes advertised by the LGN.

The following example shows how to enable **auto-summary** on the LGN SanFran:

```
SanFran.BldA.T5# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
SanFran.BldA.T5(config)# atm router pnni
SanFran.BldA.T5(config-atm-router)# node 2
SanFran.BldA.T5(config-pnni-node)# auto-summary
SanFran.BldA.T5(config-pnni-node)# end
SanFran.BldA.T5#
```

The following example shows how to verify the configuration:

```
SanFran.BldA.T5# show atm pnni summary

Codes: Node - Node index advertising this summary
       Type - Summary type (INT - internal, EXT - exterior)
       Sup  - Suppressed flag (Y - Yes, N - No)
       Auto - Auto Summary flag (Y - Yes, N - No)
       Adv  - Advertised flag (Y - Yes, N - No)

 Node Type Sup Auto Adv  Summary Prefix
 ~~~~ ~~~~ ~~~ ~~~~ ~~~  ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
  1   Int   N   Y    Y   47.0091.4455.6677.2233.1011.1244/104
  2   Int   N   Y    N   47.0091.4455.6677.2233/72
```

The switch that contains the active PGL is configured similarly:

```
SanFran.BldA.T4# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
SanFran.BldA.T4(config)# atm router pnni
SanFran.BldA.T4(config-atm-router)# node 2
SanFran.BldA.T4(config-pnni-node)# auto-summary
SanFran.BldA.T4(config-pnni-node)# end
SanFran.BldA.T4#
```

The following examples show how to verify the configuration:

```
SanFran.BldA.T4# show atm pnni summary

Codes: Node - Node index advertising this summary
       Type - Summary type (INT - internal, EXT - exterior)
       Sup  - Suppressed flag (Y - Yes, N - No)
       Auto - Auto Summary flag (Y - Yes, N - No)
       Adv  - Advertised flag (Y - Yes, N - No)

 Node Type Sup Auto Adv  Summary Prefix
 ~~~~ ~~~~ ~~~ ~~~~ ~~~  ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
  1   Int   N   Y    Y   47.0091.4455.6677.2233.1011.1266/104
  2   Int   N   Y    Y   47.0091.4455.6677.2233/72

SanFran.BldA.T4# show atm route

Codes: P - installing Protocol (S - Static, P - PNNI, R - Routing control),
       T - Type (I - Internal prefix, E - Exterior prefix, SE -
                 Summary Exterior prefix, SI - Summary Internal prefix,
                 ZE - Suppress Summary Exterior, ZI - Suppress Summary Internal)

P  T Node/Port        St Lev Prefix
~ ~~ ~~~~~~~~~~~~~~~~ ~~ ~~~ ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
P  I 12  0            UP 0   47.0091.4455.6677.1144.1011.1233/104
P  I 11  0            UP 0   47.0091.4455.6677.1144.1011.1244/104
P  I 9   0            UP 0   47.0091.4455.6677.1144.1011.1255/104
P SI 2   0            UP 0   47.0091.4455.6677.2233/72
P  I 13  0            UP 0   47.0091.4455.6677.2233.1011.1244/104
P SI 1   0            UP 0   47.0091.4455.6677.2233.1011.1266/104
R  I 1   ATM2/0/0     UP 0   47.0091.4455.6677.2233.1011.1266.0060.3e7b.2001/152
R  I 1   ATM2/0/0     UP 0   47.0091.4455.6677.2233.1011.1266.0060.3e7b.2002/152
R  I 1   ATM2/0/0     UP 0   47.0091.4455.6677.2233.1011.1266.4000.0c/128
```

Moving Switches T3, T1, and T2 Down into a New Peer Group

The following example shows how to move switch NewYork.BldB.T3 down into a new peer group:

```
NewYork.BldB.T3# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
NewYork.BldB.T3(config)# atm router pnni
NewYork.BldB.T3(config-atm-router)# node 2 level 56
NewYork.BldB.T3(config-pnni-node)# name NewYork
NewYork.BldB.T3(config-pnni-node)# no auto-summary
NewYork.BldB.T3(config-pnni-node)# exit
NewYork.BldB.T3(config-atm-router)# node 1
NewYork.BldB.T3(config-pnni-node)# election leadership-priority 45
NewYork.BldB.T3(config-pnni-node)# node 1 disable
NewYork.BldB.T3(config-pnni-node)# node 1 level 72
NewYork.BldB.T3(config-pnni-node)# parent 2
NewYork.BldB.T3(config-pnni-node)# node 1 enable
NewYork.BldB.T3(config-pnni-node)# end
NewYork.BldB.T3#
```

The following example shows how to move switch NewYork.BldB.T1 down into a new peer group:

```
NewYork.BldB.T1# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
NewYork.BldB.T1(config)# atm router pnni
NewYork.BldB.T1(config-atm-router)# node 1 disable
NewYork.BldB.T1(config-pnni-node)# node 1 level 72 enable
NewYork.BldB.T1(config-pnni-node)# end
NewYork.BldB.T1#
```

The following example shows how to move switch NewYork.BldB.T2 down into a new peer group:

```
NewYork.BldB.T2# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
NewYork.BldB.T2(config)# atm router pnni
NewYork.BldB.T2(config-atm-router)# node 1 disable
NewYork.BldB.T2(config-pnni-node)# node 1 level 72 enable
NewYork.BldB.T2(config-pnni-node)# end
NewYork.BldB.T2#
```

The following examples show how to verify the results of the configuration:

```
NewYork.BldB.T2# show atm pnni local-node

PNNI node 1 is enabled and running
  Node name: NewYork.BldB.T2
  System address          47.00914455667711441011244.00603E5BBC01.01
  Node ID          72:160:47.00914455667711441011244.00603E5BBC01.00
  Peer group ID        72:47.0091.4455.6677.1144.0000.0000
  Level 72, Priority 0 0, No. of interfaces 3, No. of neighbors 1
  Parent Node Index: NONE

<information deleted>

NewYork.BldB.T2# show atm route

Codes: P - installing Protocol (S - Static, P - PNNI, R - Routing control),
       T - Type (I - Internal prefix, E - Exterior prefix, SE -
                 Summary Exterior prefix, SI - Summary Internal prefix,
                 ZE - Suppress Summary Exterior, ZI - Suppress Summary Internal)

P  T Node/Port        St Lev Prefix
~ ~~ ~~~~~~~~~~~~~~~~ ~~ ~~~ ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
P  I 9   0            UP 0   47.0091.4455.6677.1144.1011.1233/104
P  I 13  0            UP 0   47.0091.4455.6677.1144.1011.1233/104
P SI 1   0            UP 0   47.0091.4455.6677.1144.1011.1244/104
P  I 13  0            UP 0   47.0091.4455.6677.1144.1011.1244/104
R  I 1   ATM2/0/0     UP 0   47.0091.4455.6677.1144.1011.1244.0060.3e5b.bc01/152
R  I 1   ATM2/0/0     UP 0   47.0091.4455.6677.1144.1011.1244.0060.3e5b.bc02/152
R  I 1   ATM2/0/0     UP 0   47.0091.4455.6677.1144.1011.1244.4000.0c/128
P  I 11  0            UP 0   47.0091.4455.6677.1144.1011.1255/104
P  I 13  0            UP 0   47.0091.4455.6677.1144.1011.1255/104
P  I 12  0            UP 0   47.0091.4455.6677.2233/72


NewYork.BldB.T2# show atm pnni hierarchy network
  Summary of active parent LGNs in the routing domain:
  Node  Level  Parent  Node Name
  ~~~~  ~~~~~  ~~~~~~  ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
  1     72     13      NewYork.BldB.T2
  13    56     0       NewYork

NewYork.BldB.T2# show atm pnni hierarchy network detail
  Detailed hierarchy network display:
  Number Of Network LGN Ancestors: 1
```

```
Lowest Level (72) information:
Node No.....: 1    Node Name: NewYork.BldB.T2
Node’s ID...:   72:160:47.00914455667711441011244.00603E5BBC01.00
Node’s Addr.:          47.00914455667711441011244.00603E5BBC01.01
Node’s PG ID:       72:47.0091.4455.6677.1144.0000.0000
PGL No......: 11   PGL Name: NewYork.BldB.T3
PGL ID......:   72:160:47.00914455667711441011255.00603E5BC401.00

Level 56 ancestor information:
Parent LGN..: 13   LGN Name: NewYork
LGN’s ID....:    56:72:47.00914455667711440000000.00603E5BC401.00
LGN’s Addr..:          47.00914455667711441011255.00603E5BC401.02
LGN’s PG ID.:       56:47.0091.4455.6677.0000.0000.0000
LGN PGL No..:      Unelected or unknown
LGN’s PGL ID:      0:0:00.000000000000000000000000.000000000000.00
```

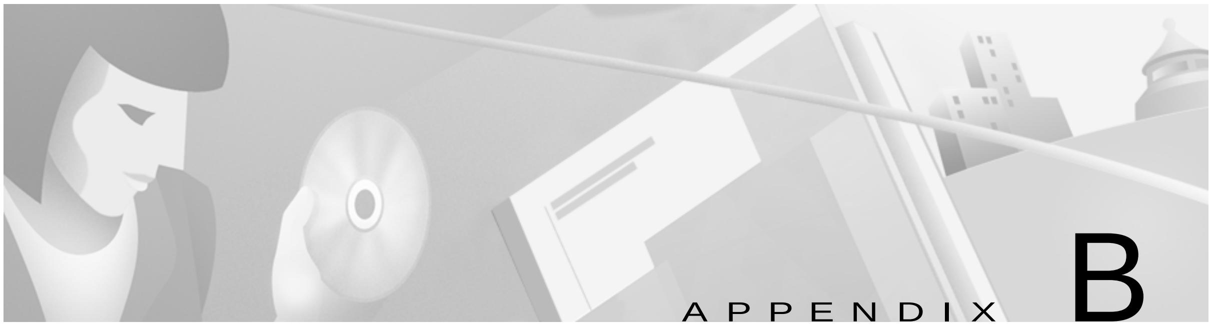
Restoring Autosummary on the LGN NewYork

The following example shows how to restore autosummary on the LGN NewYork:

```
NewYork.BldB.T3# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
NewYork.BldB.T3(config)# atm router pnni
NewYork.BldB.T3(config-atm-router)# node 2
NewYork.BldB.T3(config-pnni-node)# auto-summary
NewYork.BldB.T3(config-pnni-node)# end
NewYork.BldB.T3#
```

The following examples show how to verify the configuration:

```
NewYork.BldB.T3# show atm pnni summary

Codes: Node - Node index advertising this summary
       Type - Summary type (INT - internal, EXT - exterior)
       Sup  - Suppressed flag (Y - Yes, N - No)
       Auto - Auto Summary flag (Y - Yes, N - No)
       Adv  - Advertised flag (Y - Yes, N - No)

 Node Type Sup Auto Adv  Summary Prefix
 ~~~~ ~~~~ ~~~ ~~~~ ~~~  ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
  1   Int   N   Y    Y   47.0091.4455.6677.1144.1011.1255/104
  2   Int   N   Y    Y   47.0091.4455.6677.1144/72

NewYork.BldB.T3# show atm route

Codes: P - installing Protocol (S - Static, P - PNNI, R - Routing control),
       T - Type (I - Internal prefix, E - Exterior prefix, SE -
                 Summary Exterior prefix, SI - Summary Internal prefix,
                 ZE - Suppress Summary Exterior, ZI - Suppress Summary Internal)

P  T Node/Port        St Lev Prefix
~ ~~ ~~~~~~~~~~~~~~~~ ~~ ~~~ ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
P SI 2   0            UP 0   47.0091.4455.6677.1144/72
P  I 12  0            UP 0   47.0091.4455.6677.1144.1011.1233/104
P  I 9   0            UP 0   47.0091.4455.6677.1144.1011.1244/104
P SI 1   0            UP 0   47.0091.4455.6677.1144.1011.1255/104
R  I 1   ATM2/0/0     UP 0   47.0091.4455.6677.1144.1011.1255.0060.3e5b.c401/152
R  I 1   ATM2/0/0     UP 0   47.0091.4455.6677.1144.1011.1255.0060.3e5b.c402/152
R  I 1   ATM2/0/0     UP 0   47.0091.4455.6677.1144.1011.1255.4000.0c/128
P  I 10  0            UP 0   47.0091.4455.6677.2233/72
```



APPENDIX B

Acronyms

The acronyms in this appendix apply to the Catalyst 8540 MSR, Catalyst 8510 MSR, and LightStream 1010. Table B-1 lists the acronyms used in this publication, along with their expansions.

Table B-1 List of Acronyms

Acronym	Definition
AAA	authentication, authorization, and accounting
AAL	ATM adaptation layer
ABR	available bit rate
ACK	acknowledge
AESA	ATM end system address
AIS	alarm indication signal
AR	access rate
ARP	Address Resolution Protocol
ATM ARP	ATM Address Resolution Protocol
AW	administrative weight
Bc	committed burst size
Be	excess burst size
BER	bit error rate
BERT	bit error rate test
BITS	Building Integrated Timing Supply
BOOTP	Bootstrap Protocol
BUS	broadcast and unknown server
CAC	connection admission control
CAS	channel associated signalling
CBR	constant bit rate
CCO	Cisco Connection Online
CDP	Cisco Distribution Protocol
CDS3	channelized DS3
CDV	cell delay variation

Table B-1 List of Acronyms (continued)

Acronym	Definition
CDVT	cell delay variation tolerance
CE1	channelized E1
CES	circuit emulation services
CES-IWF	circuit emulation services interworking function
CHAP	Challenge Handshake Authentication Protocol
CIR	committed information rate
Cisco IFS	Cisco IOS File System
CLI	command-line interface
CLP	cell loss priority
CLR	cell loss ration
CoS	class of service
CRC	cyclic redundancy check
CSR	campus switch router
CTC	common transmit clocking
CTD	cell transfer delay
CTT	Connection Traffic Table
CTTR	Connection Traffic Table row
CUG	closed user group
DACS	digital access and crossconnect system
DCC	Data Country Code
DIP	dual in-line package
DLCI	data-link connection identifier
EFCI	Explicit Forward Congestion Indication
EHSA	Enhanced High System Availability
EIGRP	Enhanced Interior Gateway Routing Protocol
ELAN	emulated LAN
EPD	early packet discard
ESI	end system identifier
FC-PCQ	feature card per-class queuing
FC-PFQ	feature card per-flow queuing
FDL	facility data link
FE	Fast Ethernet
FPGA	Field Programmable Gate Array
FTP	File Transfer Protocol
GE	Gigabit Ethernet
ICD	International Code Designator

Table B-1 List of Acronyms (continued)

Acronym	Definition
ICMP	International Control Message Protocol
ICP	IMA Control Protocol
ID	identifier
IE	information element
IISP	Interim Interswitch Signaling Protocol
ILMI	Integrated Local Management Interface
IMA	inverse multiplexing over ATM
InARP	Inverse ARP
IPX	Internet Packet Exchange
LANE	LAN emulation
LBO	line build-out
LCD	loss of cell delineation
LEC	LAN emulation client
LECS	LAN emulation configuration server
LES	LAN emulation server
LGN	logical group node
LIS	logical IP subnet
LMI	Local Management Interface
LOS	loss of signal
MaxCR	maximum cell rate
MBS	maximum burst size
MCR	minimum cell rate
MDL	maintenance data link
MMF	multimode fiber
MSR	multiservice ATM switch router
NCDP	Network Clock Distribution Protocol
NE	network element
NMS	network management system
NNI	Network-Network Interface
NSAP	network service access point
NTP	Network Time Protocol
NVRAM	nonvolatile random-access memory
OAM	operation, administration, and management
OC	optical carrier
OSF	oversubscription factor
OSPF	Open Shortest Path First

Table B-1 List of Acronyms (continued)

Acronym	Definition
OVC	output virtual circuit
PAP	Password Authentication Protocol
PCR	peak cell rate
PD	packet discard
PG	peer group
PGL	peer group leader
PIF	physical interface
PIM	Protocol Independent Multicast
PIR	peak information rate
PNNI	Private Network-Network Interface
PPP	Point-to-Point Protocol
PRS	primary reference source
PTSE	PNNI topology state element
PVC	permanent virtual channel
PVCL	permanent virtual channel link
PVP	permanent virtual path
PVPL	permanent virtual path link
QoS	quality of service
QSAAL	Q.2931 protocol over signalling ATM adaptation layer
RADIUS	Remote Dial-In User Service
RAIG	Resource Availability Information Groups
RCAC	Resource Call Admission Control
rcp	remote copy protocol
RDI	remote defect indication
RISC	reduced instruction set computing
RM	resource management
RMON	Remote Monitoring
RR	relative rate
RS	rate scheduler
SCR	sustainable cell rate
SDH	Synchronous Digital Hierarchy
SGCP	Simple Gateway Control Protocol
SIN	ships in the night
SNAP	Subnetwork Access Protocol
SNMP	Simple Network Management Protocol
SONET	Synchronous Optical Network

Table B-1 List of Acronyms (continued)

Acronym	Definition
SRTS	synchronous residual time stamp
SSRP	Simple Server Redundancy Protocol
STM	Synchronous Transfer Module
STS	Synchronous Transfer Signal
SVC	switched virtual channel
SVCC	switched virtual channel connection
SVPC	switched virtual path connection
TACACS	Terminal Access Controller Access Control System
TBR	tag bit rate
TDM	time-division multiplexer
TDP	Tag Distribution Protocol
TVC	tag virtual channel
UBR	unspecified bit rate
UBR+	unspecified bit rate plus
UDP	User Datagram Protocol
UNI	User-Network Interface
UPC	usage parameter control
UTP	unshielded twisted-pair
VBR	variable bit rate
VBR-NRT	variable bit rate non-real time
VBR-RT	variable bit rate real time
VC	virtual channel
VCC	virtual channel connection
VCI	virtual channel identifier
VCL	virtual channel link
VP	virtual path
VPCI	virtual path connection identifier
VPI	virtual path identifier
VPN	virtual private network
WK	well-known
WRR	weighted round-robin

INDEX

Symbols

Numerics

A

B

C

D

E

G

H

K

L

M

N

O

P

Q

R

S

T

U

V

W

Y