

Configuring ATM

This chapter describes how to configure ATM on the Cisco ASR 1000 Series Aggregation Services Routers.

- Restrictions for Configuring ATM, on page 1
- ATM Interface, on page 1
- How to Configure ATM, on page 4
- ATM Configuration Examples, on page 30
- Monitoring and Maintaining the ATM Interface, on page 36
- Feature Information for Configuring ATM, on page 37
- Additional References, on page 40

Restrictions for Configuring ATM

Restrictions for PVC Bundle Management

ATM PVC Bundling Support on ASR1K

- A maximum of 8 PVCs can be configured in one PVC bundle.

ATM Interface

Asynchronous Transfer Mode (ATM) uses one Virtual Circuit (VC) to carry all traffic to the next hop address. Even with VC multiplexing, a single VC carries all traffic of the same protocol to the next hop address. Though Weighted Random Early Discard (Per-VC (D)WRED) and WFQ can classify and prioritize the packets, they all share one single Quality of Service (QoS) VC.

Enabling the ATM Interface

This section describes how to configure an ATM interface. For the AIP, all ATM port adapters, and the 1-port ATM-25 network module, the port number is always 0. For example, the *slot*/*port* address of an ATM interface on an AIP installed in slot 1 is 1/0.

To configure the ATM interface, use the following commands beginning in privileged EXEC mode:

SUMMARY STEPS

1. Device# **configure terminal**
2. Do one of the following:
   - Device(config)# **interface atm** *slot* **/0**
   - Device(config)# **interface atm** *slot* **/** *port-adapter* **/0**
3. Device(config-if)# **ip address** *ip-address mask*
4. Device(config-if)# **no shutdown**

DETAILED STEPS

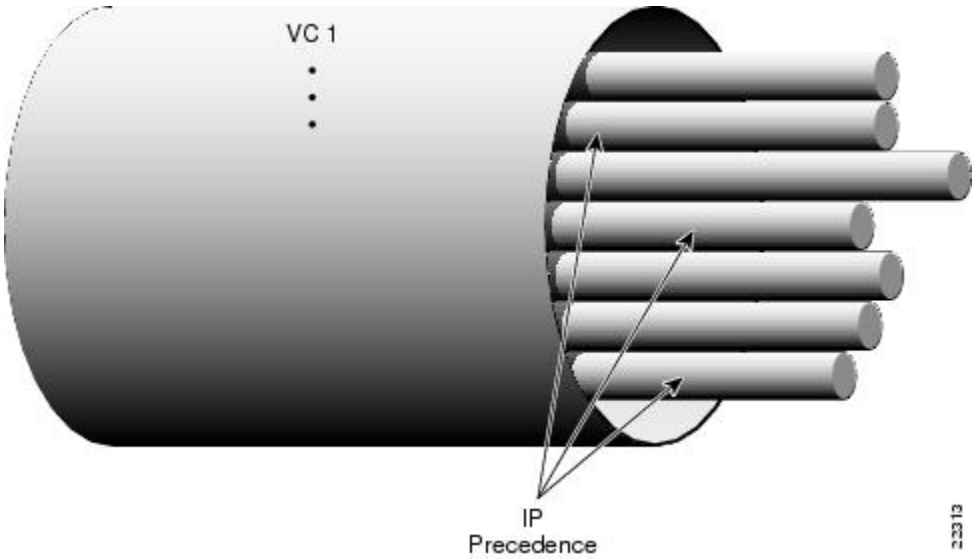
Procedure

	Command or Action	Purpose
Step 1	Device# **configure terminal**	Enters global configuration mode from the terminal.
Step 2	Do one of the following: • Device(config)# **interface atm** *slot* **/0** • Device(config)# **interface atm** *slot* **/** *port-adapter* **/0** **Example:** `Device(config)# interface atm number`	Specifies the ATM interface using the appropriate format of the **interface atm** command. To determine the correct form of the interface atm command, consult your ATM network module, port adapter, or device documentation.
Step 3	Device(config-if)# **ip address** *ip-address mask*	(Optional) If IP routing is enabled on the system, assigns a source IP address and subnet mask to the interface.
Step 4	Device(config-if)# **no shutdown**	Changes the shutdown state to up and enables the ATM interface, thereby beginning the segmentation and reassembly (SAR) operation on the interface. • The **no shutdown** command passes an **enable** command to the ATM interface, which then begins segmentation and reassembly (SAR) operations. It also causes the ATM interface to configure itself based on the previous configuration commands sent.

VC Bundle Support and Bundle Management

ATM VC bundle management allows you to configure multiple VCs that have different QoS characteristics between any pair of ATM-connected routers. As shown in the figure below, these VCs are grouped in a bundle and are referred to as bundle members.

Figure 1: ATM VC Bundle

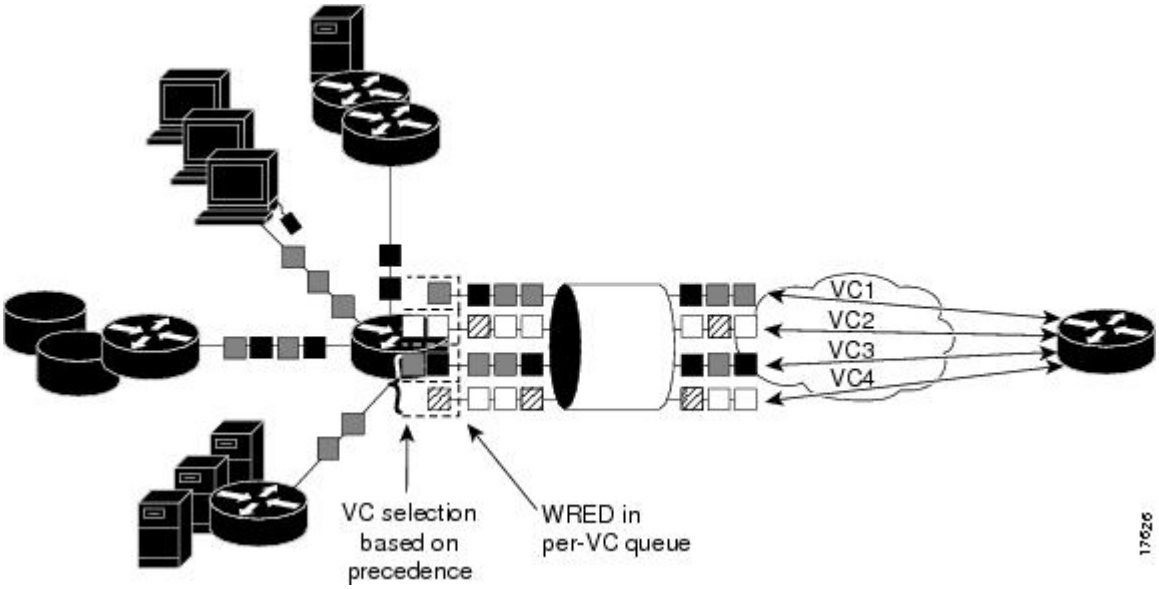


ATM VC bundle management allows you to define an ATM VC bundle and add VCs to it. Each VC of a bundle has its own ATM traffic class and ATM traffic parameters. You can apply attributes and characteristics to discrete VC bundle members or you can apply them collectively at the bundle level.

Using VC bundles, you can create differentiated service by flexibly distributing IP precedence levels over the different VC bundle members. You can map a single precedence level or a range of levels to each discrete VC in the bundle, thereby enabling individual VCs in the bundle to carry packets marked with different precedence levels. You can use WRED (or DWRED) to further differentiate service across traffic that has different IP precedences but that uses the same VC in a bundle.

To determine which VC in the bundle to use to forward a packet to its destination, the ATM VC bundle management software matches precedence levels between packets and VCs (see the figure below). IP traffic is sent to the next hop address for the bundle because all VCs in a bundle share the same destination, but the VC used to carry a packet depends on the value set for that packet in the IP Precedence bits of the type of service (ToS) byte of its header. The ATM VC bundle management software matches the IP precedence of the packet to the IP Precedence value or range of values assigned to a VC, sending the packet out on the appropriate VC. Moreover, the ATM VC bundle management feature allows you to configure how traffic will be redirected when the VC the packet was matched to goes down. The figure below illustrates how the ATM VC bundle management software determines which permanent virtual circuit (PVC) bundle member to use to carry a packet and how WRED (or DWRED) is used to differentiate traffic on the same VC.

Figure 2: ATM VC Bundle PVC Selection for Packet Transfer



The support of multiple parallel ATM VCs allows you to create stronger service differentiation at the IP layer. For instance, you might want to provide IP traffic belonging to real-time CoS (such as Voice over IP traffic) on an ATM VC with strict constraints (constant bit rate (CBR) or variable bit rate real-time (VBR-rt), for example), while transporting traffic other than real-time traffic over a more elastic ATM available bit rate (ABR) PVC. Using a configuration such as this would allow you to fully utilize your network capacity. You could also elect to transport best-effort IP traffic over an unspecified bit rate (UBR) PVC--UBR is effectively the ATM version of best-effort service.

How to Configure ATM

To configure ATM, complete the tasks in the following sections. The first task is required, and then you must configure at least one PVC or SVC. The virtual circuit options you configure must match in three places: on the router, on the ATM switch, and at the remote end of the PVC or SVC connection. The remaining tasks are optional.

Configuring a VC Bundle

Creating a VC Bundle

Command	Purpose
`Device(config-if)# bundle bundle-name`	Creates the specified bundle and enters bundle configuration mode.

Applying Parameters to Individual VCs

Configuring a VC Bundle Member Directly

Command	Purpose
`Device(config-if-atm-member)# ubr output-pcr [input-pcr]`	Configures the VC for unspecified bit rate (UBR) QoS and specifies the output peak cell rate (PCR) for it.
`Device(config-if-atm-member)# ubr+ output-pcr output-mcr [input-pcr] [input-mcr]`	Configures the VC for UBR QoS and specifies the output PCR and output minimum guaranteed cell rate for it.
`Device(config-if-atm-member)# vbr-nrt output-pcr output-scr output-mbs [input-pcr] [input-scr] [input-mbs]`	Configures the VC for variable bit rate nonreal-time (VBR-nrt) QoS and specifies the output PCR, output sustainable cell rate, and output maximum burst cell size for it.
`Device(config-if-atm-member)# precedence [other | range]`	Configures the precedence levels for the VC.
`Device(config-if-atm-member)# bump {implicit | explicit precedence-level | traffic}`	Configures the bumping rules for the VC.
`Device(config-if-atm-member)# protect {group | vc}`	Configures the VC to belong to the protected group of the bundle or to be an individually protected VC bundle member.

Configuring VC Class Parameters to Apply to a VC Bundle Member

Command	Purpose
`Device(config-vc-class)# bump {implicit | explicit precedence-level | traffic}`	Specifies the bumping rules for the VC member to which the class is applied. These rules determine to which VC in the bundle traffic is directed when the carrier VC bundle member goes down. **Note** You can also add the following commands to a VC class to be used to configure a VC bundle member: **ubr**, **ubr+**, and **vbr-nrt**. When a VC is a member of a VC bundle, the following commands cannot be used in vc-class mode to configure the VC: **encapsulation**, **protocol**, **inarp**, and **broadcast**. These commands are useful only at the bundle level, not the bundle member level. Configuration for an individual VC overrides the collective configuration applied to all VC bundle members through application of a VC class to the bundle.

Command	Purpose
`Device(config-vc-class)# precedence precedence min-threshold max-threshold mark-probability-denominator`	Defines precedence levels for the VC member to which the class is applied.
`Device(config-vc-class)# protect {group | vc}`	Configures the VC as a member of the protected group of the bundle or as an individually protected VC.

Applying a VC Class to a Discrete VC Bundle Member

Command	Purpose
`Device(config-if-atm-member)# class-vc vc-class-name`	Assigns a VC class to a VC bundle member.

Configuring a VC Not to Accept Bumped Traffic

Command	Purpose
`Device(config-if-atm-member)# no bump traffic`	Configures the VC not to accept any bumped traffic that would otherwise be redirected to it.

Monitoring and Maintaining VC Bundles and Their VC Members

Command	Purpose
`Device# show atm bundle bundle-name`	Displays the bundle attributes assigned to each bundle VC member and the current working status of the VC members.
`Device# show atm bundle bundle-name statistics [detail]`	Displays statistics or detailed statistics on the specified bundle.
`Device# show atm map`	Displays a list of all configured ATM static maps to remote hosts on an ATM network and on ATM bundle maps.
`Device# debug atm bundle errors`	Displays information on bundle errors.
`Device# debug atm bundle events`	Displays a record of bundle events.

Applying Bundle-Level Parameters

Configuring Bundle-Level Parameters

Command	Purpose
`Device(config-atm-bundle)# protocol protocol {protocol-address | inarp} [[no] broadcast]`	Configures a static map or enables Inverse Address Resolution Protocol (Inverse ARP) or Inverse ARP broadcasts for the bundle. **Note** Bundle-level parameters can be applied either by assigning VC classes or by directly applying them to the bundle. Parameters applied through a VC class assigned to the bundle are superseded by those applied at the bundle level. Bundle-level parameters are superseded by parameters applied to an individual VC.
`Device(config-atm-bundle)# encapsulation aal-encap`	Configures the ATM adaptation layer (AAL) and encapsulation type for the bundle.
`Device(config-atm-bundle)# inarp minutes`	Configures the Inverse ARP time period for all VC bundle members.
`Device(config-atm-bundle)# broadcast`	Enables broadcast forwarding for all VC bundle members.
`Device(config-atm-bundle)# oam retry up-count down-count retry frequency`	Configures the VC bundle parameters related to operation, administration, and maintenance (OAM) management.
`Device(config-atm-bundle)# oam-bundle [manage] [frequency]`	Enables end-to-end F5 OAM loopback cell generation and OAM management for all VCs in the bundle.

Configuring VC Class Parameters to Apply to a VC Bundle Member

Command	Purpose
`Device(config-vc-class)# bump {implicit | explicit precedence-level | traffic}`	Specifies the bumping rules for the VC member to which the class is applied. These rules determine to which VC in the bundle traffic is directed when the carrier VC bundle member goes down. **Note** You can also add the following commands to a VC class to be used to configure a VC bundle member: **ubr**, **ubr+**, and **vbr-nrt**. When a VC is a member of a VC bundle, the following commands cannot be used in vc-class mode to configure the VC: **encapsulation**, **protocol**, **inarp**, and **broadcast**. These commands are useful only at the bundle level, not the bundle member level. Configuration for an individual VC overrides the collective configuration applied to all VC bundle members through application of a VC class to the bundle.

Command	Purpose
`Device(config-vc-class)# precedence precedence min-threshold max-threshold mark-probability-denominator`	Defines precedence levels for the VC member to which the class is applied.
`Device(config-vc-class)# protect {group | vc}`	Configures the VC as a member of the protected group of the bundle or as an individually protected VC.

Attaching a Class to a Bundle

Command	Purpose
`(config-atm-bundle)# class-bundle vc-class-name`	Configures a bundle with the bundle-level commands contained in the specified VC class. **Note** Parameters set through bundle-level commands contained in the VC class are applied to the bundle and all of its VC members. Bundle-level parameters applied through commands configured directly on the bundle supersede those applied through a VC class. Some bundle-level parameters applied through a VC class or directly to the bundle can be superseded by commands that you directly apply to individual VCs in bundle-vc configuration mode.

Committing a VC to a Bundle

Command	Purpose
`Device(config-atm-bundle)# pvc-bundle pvc-name [vpi/] [vci]`	Adds the specified VC to the bundle and enters bundle-vc configuration mode in order to configure the specified VC bundle member.

Configuring PVCs

To use a permanent virtual circuit (PVC), you must configure the PVC into both the router and the ATM switch. PVCs remain active until the circuit is removed from either configuration.

When a PVC is configured, all the configuration options are passed on to the ATM interface. These PVCs are writable into the nonvolatile RAM (NVRAM) as part of the Route Processor (RP) configuration and are used when the RP image is reloaded.

Some ATM switches might have point-to-multipoint PVCs that do the equivalent of broadcasting. If a point-to-multipoint PVC exists, then that PVC can be used as the sole broadcast PVC for all multicast requests.

To configure a PVC, perform the tasks in the following sections.

Creating a Permanent Virtual Circuit

To use a permanent virtual circuit (PVC), configure the PVC in both the router and the ATM switch. PVCs remain active until the circuit is removed from either configuration. To create a PVC on the ATM interface

and enter interface ATM VC configuration mode, perform the following procedure beginning in global configuration mode:

SUMMARY STEPS

1. Device(config)# **interface atm** *slot*/*subslot*/*port*.*subinterface* [.*subinterface-number* {**multipoint** | **point-to-point**}]
2. (config-if)# **ipaddress***address mask*
3. Device(config-if)# **pvc** [*name*] *vpi* /*vci*
4. Device(config-if-atm-vc)# **protocol** *protocol* {*protocol-address* | **inarp**} [[**no**] **broadcast**]
5. Device(config-if-atm-vc)# **inarp** *minutes*
6. Device(config-if-atm-vc)# **encapsulation** {**aal5mux**|**aal5nlpidaal5snap**}
7. Device(config-if-atm-vc)# **end**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	Device(config)# **interface atm** *slot*/*subslot*/*port*.*subinterface* [.*subinterface-number* {**multipoint** | **point-to-point**}]	Enters subinterface configuration mode for the specified port on the ATM Interface Module (IM), where: • *slot*—Specifies the chassis slot number where the SIP is installed. • *subslot*—Specifies the secondary slot of the SIP where the IM is installed. • *port*—Specifies the number of the individual interface port on an IM. • *subinterface*—Specifies the number of the subinterface.
Step 2	(config-if)# **ipaddress***address mask*	Assigns the specified IP address and subnet mask to the interface or subinterface.
Step 3	Device(config-if)# **pvc** [*name*] *vpi* /*vci*	Configures a new ATM PVC by assigning its VPI/VCI numbers and enters ATM VC configuration mode. The valid values for *vpi* /*vci* are: • *name*—(Optional) An arbitrary string that identifies this PVC. • *vpi*—Specifies the VPI ID. The valid range is 0 to 255. • *vci*—Specifies the VCI ID. The valid range is 32 to 65535. Values 1 to 31 are reserved and should not be used, except for 5 for the QSAAL PVC and 16 for the ILMI PVC. ILMI is not supported. **Note**

	Command or Action	Purpose
		When using the **pvc** command, remember that the *vpi* /*vci* combination forms a unique identifier for the interface and all of its subinterfaces. If you specify a *vpi* /*vci* combination that has been used on another subinterface, the Cisco IOS software assumes that you want to modify that PVC's configuration and automatically switches to its parent subinterface.
Step 4	Device(config-if-atm-vc)# **protocol** *protocol* {*protocol-address* | **inarp**} [[**no**] **broadcast**] **Example:**	Configures the PVC for a particular protocol and maps it to a specific *protocol-address* . • *protocol*—Typically set to **ip**or **pppoe**, but other values are possible. **Note** PPP is not supported. • *protocol-address*—Destination address or virtual interface template for this PVC (if appropriate for the *protocol*). • **inarp** —Specifies that the PVC uses Inverse ARP to determine its address. • [**no**] **broadcast**—(Optional) Specifies that this mapping should (or should not) be used for broadcast packets.
Step 5	Device(config-if-atm-vc)# **inarp** *minutes*	(Optional) If using Inverse ARP, configures how often the PVC transmits Inverse ARP requests to confirm its address mapping. The valid range is 1 to 60 minutes, with a default of 15 minutes.
Step 6	Device(config-if-atm-vc)# **encapsulation** {**aal5mux**|**aal5nlpidaal5snap**}	(Optional) Configures the ATM adaptation layer (AAL) and encapsulation type, where: • **aal5mux** *protocol*—AAL and encapsulation type for multiplex (MUX)-type VCs. A protocol must be specified when you use this encapsulation type. Possible values for the *protocol* argument are as follows: • **aal5nlpid** —AAL and encapsulation type that allows ATM interfaces to interoperate with High-Speed Serial Interfaces (HSSIs) that are using an ATM data service unit (ADSU) and running ATM-Data Exchange Interface (DXI). Supported on ATM PVCs only. • **aal5snap** —AAL and encapsulation type that supports Inverse Address Resolution Protocol (ARP). Logical

	Command or Action	Purpose
		link control/Subnetwork Access Protocol (LLC/SNAP) precedes the protocol datagram. This is the default.
Step 7	Device(config-if-atm-vc)# **end**	Exits ATM VC configuration mode and returns to privileged EXEC mode.

Creating a PVC on a Point-to-Point Subinterface

Use point-to-point subinterfaces to provide each pair of routers with its own subnet. When you create a PVC on a point-to-point subinterface, the router assumes it is the only point-to-point PVC that is configured on the subinterface, and it forwards all IP packets with a destination IP address in the same subnet to this VC. To configure a point-to-point PVC, perform the following procedure beginning in global configuration mode:

SUMMARY STEPS

1. Device(config)# **interfaceatm***slot*/*subslot*/*port***.***subinterface***point-to-point**
2. Device(config-subif)# **ipaddress***address mask*
3. Device(config-subif)# **pvc** [*name*] *vpi* /*vci*
4. Device(config-if-atm-vc)# **protocol***protocol protocol-address* [[**no**] **broadcast**]
5. Device(config-if-atm-vc)# **encapsulation**{**aal5mux***protocol*|**aal5nlpid**|**aal5snap**}
6. Device(config-if)# **end**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	Device(config)# **interfaceatm***slot*/*subslot*/*port***.***subinterface***point-to-point**	Creates the specified point-to-point subinterface on the given port on the specified ATM SPA, and enters subinterface configuration mode, where: • *slot*—Specifies the chassis slot number in the Cisco ASR 1000 Series Router where the SIP is installed. • *subslot*—Specifies the secondary slot of the SIP where the SPA is installed. • *port*—Specifies the number of the individual interface port on a SPA. • *subinterface*—Specifies the number of the subinterface.
Step 2	Device(config-subif)# **ipaddress***address mask*	Assigns the specified IP address and subnet mask to this subinterface.
Step 3	Device(config-subif)# **pvc** [*name*] *vpi* /*vci*	Configures a new ATM PVC by assigning its VPI/VCI numbers and enters ATM VC configuration mode. The valid values for *vpi* /*vci* are:

	Command or Action	Purpose
		• *name*—(Optional) An arbitrary string that identifies this PVC. • *vpi*—Specifies the VPI ID. The valid range is 0 to 255. • *vci*—Specifies the VCI ID. The valid range is 32 to 65535. Values 1 to 31 are reserved and should not be used, except for 5 for the QSAAL PVC and 16 for the ILMI PVC. ILMI is unsupported on the Cisco ASR 1000 Series Routers. **Note** When using the **pvc** command, remember that the *vpi* /*vci* combination forms a unique identifier for the interface and all of its subinterfaces. If you specify a *vpi* /*vci* combination that has been used on another subinterface, the Cisco IOS XE software assumes that you want to modify that PVC's configuration and automatically switches to its parent subinterface.
Step 4	Device(config-if-atm-vc)# **protocol***protocol protocol-address* [[**no**] **broadcast**]	Configures the PVC for a particular protocol and maps it to a specific *protocol-address* . • *protocol*—Typically set to **ip** or **pppoe** for point-to-point subinterfaces, but other values are possible. **Note** PPP is not supported on the Cisco ASR 1000 Series Routers, and PPPoE is supported beginning in Cisco IOS XE Release 2.5. • *protocol-address*—Destination address or virtual template interface for this PVC (as appropriate for the specified *protocol*). • [**no**] **broadcast**—(Optional) Specifies that this mapping should (or should not) be used for broadcast packets. The **protocol** command also has an **inarp** option, but this option is not meaningful on point-to-point PVCs that use a manually configured address.
Step 5	Device(config-if-atm-vc)# **encapsulation**{**aal5mux***protocol*|**aal5nlpid**|**aal5snap**}	(Optional) Configures the ATM adaptation layer (AAL) and encapsulation type, where: • **aal5mux** *protocol*—AAL and encapsulation type for multiplex (MUX)-type VCs. A protocol must be specified when you use this encapsulation type.

	Command or Action	Purpose
		• **aal5nlpid** —(IP only) AAL and encapsulation type that allows ATM interfaces to interoperate with High-Speed Serial Interfaces (HSSIs) that are using an ATM data service unit (ADSU) and running ATM-Data Exchange Interface (DXI). Supported on ATM PVCs only. • **aal5snap** —AAL and encapsulation type that supports Inverse Address Resolution Protocol (ARP). Logical link control/Subnetwork Access Protocol (LLC/SNAP) precedes the protocol datagram. This is the default. **Note** Repeat Step 1 through Step 5 for each point-to-point subinterface to be configured on this ATM SPA.
Step 6	Device(config-if)# **end**	Exits interface configuration mode and returns to privileged EXEC mode.

What to do next

Tip To verify the configuration and current status of all PVCs on a particular interface, you can also use the **showatmvcinterfaceatm** command.

Verifying a Multipoint PVC Configuration

To verify the configuration of a particular PVC, use the **show atm pvc** command:

```
Device# show atm pvc 1/120

ATM3/1/0.120: VCD: 1, VPI: 1, VCI: 120
UBR, PeakRate: 149760
AAL5-LLC/SNAP, etype:0x0, Flags: 0xC20, VCmode: 0x0
OAM frequency: 0 second(s), OAM retry frequency: 1 second(s)
OAM up retry count: 3, OAM down retry count: 5
OAM Loopback status: OAM Disabled
OAM VC status: Not Managed
ILMI VC status: Not Managed
InARP frequency: 15 minutes(s)
Transmit priority 3
InPkts: 1394964, OutPkts: 1395069, InBytes: 1833119, OutBytes: 1838799
InPRoc: 1, OutPRoc: 1, Broadcasts: 0
InFast: 0, OutFast: 0, InAS: 94964, OutAS: 95062
InPktDrops: 0,  OutPktDrops: 0
CrcErrors: 0, SarTimeOuts: 0, OverSizedSDUs: 0, LengthViolation: 0, CPIErrors: 0
Out CLP=1 Pkts: 0
OAM cells received: 0
F5 InEndloop: 0, F5 InSegloop: 0, F5 InAIS: 0, F5 InRDI: 0
F4 InEndloop: 0, F4 InSegloop: 0, F4 InAIS: 0, F4 InRDI: 0
OAM cells sent: 0
F5 OutEndloop: 0, F5 OutSegloop: 0, F5 OutRDI: 0
```

```
F4 OutEndloop: 0, F4 OutSegloop: 0, F4 OutRDI: 0
OAM cell drops: 0
Status: UP
```

Note To verify the configuration and current status of all PVCs on a particular interface, you can also use the **show atm vc interface** command.

Mapping a Protocol Address to a PVC

The ATM interface supports a static mapping scheme that identifies the network address of remote hosts or routers. This section describes how to map a PVC to an address, which is a required task for configuring a PVC.

Note If you enable or disable broadcasting directly on a PVC using the **protocol** command, this configuration will take precedence over any direct configuration using the **broadcast** command.

See examples of PVC configurations in the section "ATM Configuration Examples, on page 30".

To map a protocol address to a PVC, use the following command in interface-ATM-VC configuration mode:

Command	Purpose
`Router(config-if-atm-vc)# protocol` *protocol protocol-address* `[[no] broadcast]`	Maps a protocol address to a PVC.

Configuring the AAL and Encapsulation Type

To configure the ATM adaptation layer (AAL) and encapsulation type, use the following command beginning in interface-ATM-VC configuration mode:

Command	Purpose
`Router(config-if-atm-vc)# encapsulation aal5` *encap*	Configures the ATM adaptation layer (AAL) and encapsulation type. • For a list of AAL types and encapsulations supported for the *aal-encap* argument, refer to the **encapsulation aal5** command in the "ATM Commands" chapter of the *Cisco IOS Wide-Area Networking Command Reference*. The global default is AAL5 with SNAP encapsulation.

Configuring PVC Traffic Parameters

The supported traffic parameters are part of the following service categories: Constant Bit Rate (CBR), Unspecified Bit Rate (UBR), Variable Bit Rate Non Real-Time (VBR-NRT), and real-time Variable Bit Rate (VBR). Only one of these categories can be specified per PVC connection so if a new one is entered, it will replace the existing one.

Note The commands in this section are not supported on the ATM port adapter (PA-A1 series). The 1-port ATM-25 network module only supports UBR.

The *-pcr* and *-mcr* arguments are the peak cell rate and minimum cell rate, respectively. The *-scr* and *-mbs* arguments are the sustainable cell rate and maximum burst size, respectively.

To configure PVC traffic parameters, use one of the following commands beginning in interface-ATM-VC configuration mode:

Command	Purpose
`Router(config-if-atm-vc)# cbr peak_cell_rate_KBPS`	Configures the Constant Bit Rate (CBR).
`Router(config-if-atm-vc)# ubr output-pcr`	Configures the Unspecified Bit Rate (UBR).
`Router(config-if-atm-vc)# vbr-nrt output-pcr output-scr output-mbs`	Configures the Variable Bit Rate-Non Real Time (VBR-NRT) QOS.
`Router(config-if-atm-vc)# vbr-rt peak-rate average-rate burst`	Configures the real-time Variable Bit Rate (VBR). (Cisco MC3810 and Multiport T1/E1 ATM Network Module only.)

Enabling Inverse ARP

Inverse ARP is enabled by default when you create a PVC using the **pvc** command. Once configured, a protocol mapping between an ATM PVC and a network address is learned dynamically as a result of the exchange of ATM Inverse ARP packets.

Inverse ARP is supported on PVCs running IP or IPX and no static map is configured. If a static map is configured, Inverse ARP will be disabled.

When PVC discovery is enabled on an active PVC and the router terminates that PVC, the PVC will generate an ATM Inverse ARP request. This allows the PVC to resolve its own network addresses without configuring a static map.

Address mappings learned through Inverse ARP are aged out. However, mappings are refreshed periodically. This period is configurable using the **inarp** command, which has a default of 15 minutes.

You can also enable Inverse ARP using the **protocol** command. This is necessary only if you disabled Inverse ARP using the **no protocol** command. For more information about this command, refer to the "ATM Commands" chapter in the *Cisco IOS Wide-Area Networking Command Reference*.

For an example of configuring Inverse ARP, see the section "Enabling Inverse ARP Example, on page 32".

To enable Inverse ARP on an ATM PVC, use the following commands beginning in global configuration mode:

SUMMARY STEPS

1. Do one of the following:

- Router(config)# **interface atm** *slot* **/0**[**.** *subinterface-number* {**multipoint** | **point-to-point**}]
-
-
- Router(config)# **interface atm** *slot* **/** *port-adapter* **/0**[**.** *subinterface-number*{**multipoint** | **point-to-point**}]

2. Router(config-if)# **pvc** [*name*] *vpi* **/** *vci*
3. Router(config-if-atm-vc)# **encapsulation aal5snap**
4. Router(config-if-atm-vc)# **inarp** *minutes*

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	Do one of the following: • Router(config)# **interface atm** *slot* **/0**[**.** *subinterface-number* {**multipoint** | **point-to-point**}] • • • Router(config)# **interface atm** *slot* **/** *port-adapter* **/0**[**.** *subinterface-number*{**multipoint** | **point-to-point**}] **Example:** **Example:** **Example:** `Router(config)# interface atm number[. subinterface-number {multipoint | point-to-point}]`	Specifies the ATM interface using the appropriate format of the **interface atm** command.[1]
Step 2	Router(config-if)# **pvc** [*name*] *vpi* **/** *vci*	Specifies an ATM PVC by name (optional) and VPI/VCI numbers.
Step 3	Router(config-if-atm-vc)# **encapsulation aal5snap**	Configures AAL5 LLC-SNAP encapsulation if it is not already configured.
Step 4	Router(config-if-atm-vc)# **inarp** *minutes*	(Optional) Adjusts the Inverse ARP time period.

Enabling ANCP on an ATM Interface

The **ancp enable** command should be configured only for the control VCs on which the Access Node Control Protocol (ANCP) message is sent from the DSLAM. Use the following procedure to enable ANCP on ATM interfaces. For an example of enabling ANCP, see the section Enabling ANCP on ATM Interfaces Example, on page 32.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ancp adjacency timer** *interval*
4. **interface atm** *slot* / *subslot* / *port* **.** *subinterface*
5. **ip address** *ip-address mask*
6. **pvc** *vpi* / *vci*
7. **ancp enable**
8. **exit**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**ancp adjacency timer** *interval* **Example:** `Router(config)# ancp adjacency timer 100`	Sets the ANCP adjacency timer interval, which specifies the amount of time to wait before sending an ANCP hello packet to the DSLAM. • Valid values are defined in units of 100 milliseconds (ms). Default: 100 (10 seconds).
Step 4	**interface atm** *slot* / *subslot* / *port* **.** *subinterface* **Example:** `Router(config)# interface atm 2/0/1.1`	Creates or modifies a subinterface and enters subinterface configuration mode.
Step 5	**ip address** *ip-address mask* **Example:** `Router(config-subif)# ip address 10.16.1.2 255.255.0.0`	Assigns an IP address and subnet mask to the subinterface.
Step 6	**pvc** *vpi* / *vci* **Example:**	Enables an ANCP connection over ATM PVC and enters ATM virtual circuit configuration mode.

	Command or Action	Purpose
	`Router(config-subif)# pvc 2/100`	
Step 7	**ancp enable** **Example:** `Router(config-if-atm-vc)# ancp enable`	Enables ANCP on the interface where IP is configured.
Step 8	**exit** **Example:** `Router(config-if-atm-vc)# exit`	Exits ATM virtual circuit configuration mode.

Configuring Loopback Cells to Verify Connectivity

You can optionally configure the PVC to generate end-to-end F5 OAM loopback cells to verify connectivity on the virtual circuit. The remote end must respond by echoing back such cells. If OAM response cells are missed (indicating the lack of connectivity), the PVC state goes down. If all the PVCs on a subinterface go down, the subinterface goes down.

For information about managing PVCs using OAM, see the section "Configuring OAM Management for PVCs, on page 26".

For an example of OAM loopback cell generation, see the section "Configuring Loopback Cells Example, on page 33".

To configure transmission of end-to-end F5 OAM cells on a PVC, use the following commands in interface-ATM-VC configuration mode:

SUMMARY STEPS

1. Router(config-if-atm-vc)# **oam-pvc** [**manage**] *frequency*
2. Router(config-if-atm-vc)# **oam retry** *up-count down-count retry-frequency*

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	Router(config-if-atm-vc)# **oam-pvc** [**manage**] *frequency*	Configures transmission of end-to-end F5 OAM loopback cells on a PVC, specifies how often loopback cells should be sent, and optionally enables OAM management of the connection.
Step 2	Router(config-if-atm-vc)# **oam retry** *up-count down-count retry-frequency*	(Optional) Specifies OAM management parameters for verifying connectivity of a PVC connection. This command is only supported if OAM management is enabled. • Use the *up-count*argument to specify the number of consecutive end-to-end F5 OAM loopback cell

	Command or Action	Purpose
		responses that must be received in order to change a PVC connection state to up. Use the *down-count*argument to specify the number of consecutive end-to-end F5 OAM loopback cell responses that are not received in order to tear down a PVC. Use the *retry-frequency* argument to specify the frequency (in seconds) that end-to-end F5 OAM loopback cells should be transmitted when a change in UP/DOWN state is being verified. For example, if a PVC is up and a loopback cell response is not received after the *frequency* (in seconds) specified using the **oam-pvc** command, then loopback cells are sent at the *retry-frequency* to verify whether or not the PVC is down.

Configuring Broadcast on a PVC

To send duplicate broadcast packets for all protocols configured on a PVC, use the following command in interface-ATM-VC configuration mode:

Note If you enable or disable broadcasting directly on a PVC using the **protocol** command, this configuration will take precedence over any direct configuration using the **broadcast** command.

Command	Purpose
`Router(config-if-atm-vc)# broadcast`	Sends duplicate broadcast packets for all protocols configured on a PVC.

Configuring a PVC on a Multipoint Subinterface

Creating a multipoint subinterface allows you to create a point-to-multipoint PVC that can be used as a broadcast PVC for all multicast requests. To create a PVC on a multipoint subinterface, use the following procedure beginning in global configuration mode:

SUMMARY STEPS

1. Router(config)# **interface atm** *slot/subslot/port.subinterface* **multipoint**
2. Router(config-subif)# **ip address** *address mask*
3. Router(config-subif)# **no ip directed-broadcast**
4. Router(config-subif)# **pvc** [*name*] *vpi /vci*
5. Router(config-if-atm-vc)# **protocol** *protocol* {*protocol-address* | **inarp**} **broadcast**
6. Router(config-if-atm-vc)# **inarp** *minutes*
7. Router(config-if-atm-vc)# **encapsulation**{**aal5mux***protocol*|**aal5nlpidaal5snap**}
8. Router(config-if-atm-vc)# **end**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	Router(config)# **interface atm** *slot/subslot/port.subinterface* **multipoint**	Creates the specified point-to-multipoint subinterface on the given port on the specified ATM SPA, and enters subinterface configuration mode, where: • *slot* —Specifies the chassis slot number where the SIP is installed. • *subslot* —Specifies the secondary slot of the SIP where the SPA is installed. • *port* —Specifies the number of the individual interface port on a SPA. • *subinterface* —Specifies the number of the subinterface.
Step 2	Router(config-subif)# **ip address** *address mask*	Assigns the specified IP address and subnet mask to this subinterface.
Step 3	Router(config-subif)# **no ip directed-broadcast**	(Optional) Disables the forwarding of IP directed broadcasts, which are sometimes used in denial of service (DOS) attacks.
Step 4	Router(config-subif)# **pvc** [*name*] *vpi* /*vci*	Configures a new ATM PVC by assigning its VPI/VCI numbers and enters ATM VC configuration mode. The valid values for *vpi* /*vci* are: • *name* —(Optional) An arbitrary string that identifies this PVC. • *vpi* —Specifies the VPI ID. The valid range is 0 to 255. • *vci* —Specifies the VCI ID. The valid range is 32 to 65535. Values 1 to 31 are reserved and should not be used, except for 5 for the QSAAL PVC and 16 for the ILMI PVC. ILMI is not supported. **Note** When using the **pvc** command, remember that the *vpi* /*vci* combination forms a unique identifier for the interface and all of its subinterfaces. If you specify a *vpi* /*vci* combination that has been used on another subinterface, the Cisco IOS XE software assumes that you want to modify that PVC's configuration and automatically switches to its parent subinterface.

	Command or Action	Purpose
Step 5	Router(config-if-atm-vc)# **protocol** *protocol* {*protocol-address* | **inarp**} **broadcast**	Configures the PVC for a particular protocol and maps it to a specific *protocol-address* . • *protocol* —Typically set to **ip**or **pppoe**, but other values are possible. **Note** PPP is not supported • *protocol-address* —Destination address or virtual template interface for this PVC (if appropriate for the *protocol*). • **inarp** —Specifies that the PVC uses Inverse ARP to determine its address. • **broadcast** — Specifies that this mapping should be used for multicast packets.
Step 6	Router(config-if-atm-vc)# **inarp** *minutes*	(Optional) If using Inverse ARP, configures how often the PVC transmits Inverse ARP requests to confirm its address mapping. The valid range is 1 to 60 minutes, with a default of 15 minutes.
Step 7	Router(config-if-atm-vc)# **encapsulation**{**aal5mux***protocol*|**aal5nlpidaal5snap**}	(Optional) Configures the ATM adaptation layer (AAL) and encapsulation type, where: • **aal5mux** *protocol* —AAL and encapsulation type for multiplex (MUX)-type VCs. A protocol must be specified when you use this encapsulation type. • **aal5nlpid** —(IP only) AAL and encapsulation type that allows ATM interfaces to interoperate with High-Speed Serial Interfaces (HSSIs) that are using an ATM data service unit (ADSU) and running ATM-Data Exchange Interface (DXI). Supported on ATM PVCs only. • **aal5snap** —AAL and encapsulation type that supports Inverse Address Resolution Protocol (ARP). Logical link control/Subnetwork Access Protocol (LLC/SNAP) precedes the protocol datagram. This is the default. **Note** Repeat Step 1 through Step 7 for each multipoint subinterface to be configured on this ATM SPA.
Step 8	Router(config-if-atm-vc)# **end**	Exits interface configuration mode and returns to privileged EXEC mode.

Assigning a VC Class to a PVC

By creating a VC class, you can preconfigure a set of default parameters that you may apply to a PVC. To create a VC class, refer to the section "Configuring VC Classes, on page 24".

Once you have created a VC class, use the following command in interface-ATM-VC configuration mode to apply the VC class to a PVC:

Command	Purpose
`Router(config-if-atm-vc)#` **`class-vc`** *`vc-class-name`*	Applies a VC class to a PVC. • The *vc-class-name* argument is the same as the *name* argument you specified when you created a VC class using the **vc-class atm** command. Refer to the section "Configuring VC Classes, on page 24" for a description of how to create a VC class.

Configuring PVC Trap Support

You can configure the PVC to provide failure notification by sending a trap when a PVC on an ATM interface fails or leaves the UP operational state.

PVC Failure Notification

Only one trap is generated per hardware interface, within the specified interval defined by the interval "atmIntPvcNotificationInterval". If other PVCs on the same interface go DOWN during this interval, traps are generated and held until the interval has elapsed. Once the interval has elapsed, the traps are sent if the PVCs are still DOWN.

No trap is generated when a PVC returns to the UP state after having been in the DOWN state. If you need to detect the recovery of PVCs, you must use the SNMP management application to regularly poll your router.

PVC Status Tables

When PVC trap support is enabled, the SNMP manager can poll the SNMP agent to get PCV status information. The table "atmInterfaceExtTable" provides PVC status on an ATM interface. The table "atmCurrentlyFailingPVclTable" provides currently failing and previously failed PVC time-stamp information.

Note PVC traps are only supported on permanent virtual circuit links (PVCLs), not permanent virtual path links (PVPLs).

Prerequisites

Before you enable PVC trap support, you must configure SNMP support and an IP routing protocol on your router. See the "ATM Configuration Examples, on page 30" section.

For more information about configuring SNMP support, refer to the chapter "Configuring SNMP Support" in the *Cisco IOS Configuration Fundamentals Configuration Guide*.

For information about configuring IP routing protocols, refer to the *Cisco IOS IP Routing Protocols Configuration Guide*.

To receive PVC failure notification and access to PVC status tables on your router, you must have the Cisco PVC trap MIB called CISCO-IETF-ATM2-PVCTRAP-MIB.my compiled in your NMS application. You can find this MIB on the Web at Cisco's MIB websiteat the URL: http://www.cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml.

Enabling PVC Trap Support

When you configure PVC trap support, you must also enable OAM management on the PVC.

For more information on OAM management, see the section "Configuring OAM Management for PVCs, on page 26" later in this chapter.

The new objects in this feature are defined in the IETF draft The Definitions of Managed Objects for ATM Management , which is an extension to the AToM MIB (RFC 1695).

For an example of configuring PVC trap support, see the section "Configuring PVC Trap Support Example, on page 33".

To enable PVC trap support and OAM management, use the following commands beginning in global configuration mode:

SUMMARY STEPS

1. Router(config)# **snmp-server enable traps atm pvc interval** *seconds***fail-interval** *seconds*
2. Do one of the following:
   - Router(config)# **interface atm** *slot* **/0**[**.** *subinterface-number* {**multipoint** | **point-to-point**}]
   -
   -
   - Router(config)# **interface atm** *slot* **/** *port-adapter* **/0**[**.** *subinterface-number* {**multipoint** | **point-to-point**}]
3. Router(config-if)# **pvc** [*name*] *vpi* **/** *vci*
4. Router(config-if-atm-vc)# **oam-pvc manage**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	Router(config)# **snmp-server enable traps atm pvc interval** *seconds***fail-interval** *seconds*	Enables PVC trap support.
Step 2	Do one of the following: • Router(config)# **interface atm** *slot* **/0**[**.** *subinterface-number* {**multipoint** | **point-to-point**}] • • • Router(config)# **interface atm** *slot* **/** *port-adapter* **/0**[**.** *subinterface-number* {**multipoint** | **point-to-point**}] **Example:**	Specifies the ATM interface using the appropriate form of the **interface atm** command.[2]

	Command or Action	Purpose
	`Router(config)# interface atm number[. subinterface-number {multipoint | point-to-point}]`	
Step 3	Router(config-if)# **pvc** [*name*] *vpi* / *vci*	Enables the PVC.
Step 4	Router(config-if-atm-vc)# **oam-pvc manage**	Enables end-to-end OAM management for an ATM PVC.

Configuring VC Classes

A VC class is a set of preconfigured VC parameters that you configure and apply to a particular VC or ATM interface. You may apply a VC class to an ATM main interface, subinterface, or PVC. For example, you can create a VC class that contains VC parameter configurations that you will apply to a particular PVC. You might create another VC class that contains VC parameter configurations that you will apply to all VCs configured on a particular ATM main interface or subinterface. Refer to the "ATM Configuration Examples, on page 30" section for examples of VC class configurations.

Creating a VC Class

To create a VC class, use the following command in global configuration mode:

For examples of creating VC classes, see the section "Creating a VC Class Example, on page 33".

Command	Purpose
`Router(config)# vc-class atm name`	Creates a VC class and enters vc-class configuration mode.

Configuring VC Parameters

After you create a VC class and enter vc-class configuration mode, configure VC parameters using one or more of the following commands:

- **broadcast**
- **cbr**
- **encapsulation aal5**
- **idle-timeout**
- **inarp**
- **oam-pvc**
- **oam retry**
- **protocol**
- **ubr**
- **vbr-nrt**

Refer to the sections "Configuring PVCs, on page 8" and "Configuring PVC Trap Support, on page 22" for descriptions of how to configure these commands for PVCs.

For examples of creating VC classes, see the section "Creating a VC Class Example, on page 33".

Applying a VC Class on an ATM PVC

Once you have created and configured a VC class, you can apply it directly on an ATM PVC, or you can apply it on an ATM interface or subinterface.

To apply a VC class directly on an ATM PVC use the following commands beginning in interface configuration mode:

SUMMARY STEPS

1. Router(config-if)# **pvc** [*name*] *vpi* / *vci*
2. Router(config-if-atm-vc)# **class-vc***vc-class-name*

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	Router(config-if)# **pvc** [*name*] *vpi* / *vci*	Specifies an ATM PVC
Step 2	Router(config-if-atm-vc)# **class-vc***vc-class-name*	Applies a VC class directly on the PVC.

Applying a VC Class on an ATM Interface

To apply a VC class on an ATM main interface or subinterface, use the following commands beginning in global configuration mode:

SUMMARY STEPS

1. Do one of the following:
   - Router(config)# **interface atm** *slot* **/0[.** *subinterface-number* {**multipoint** | **point-to-point**}]
   -
   -
   - Router(config)# **interface atm** *slot* / *port-adapter* **/0[.** *subinterface-number* {**multipoint** | **point-to-point**}]
2. Router(config-if)# **class-int** *vc-class-name*

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	Do one of the following: • Router(config)# **interface atm** *slot* **/0**[**.** *subinterface-number* {**multipoint** | **point-to-point**}] • • • Router(config)# **interface atm** *slot* **/** *port-adapter* **/0**[**.** *subinterface-number* {**multipoint** | **point-to-point**}] **Example:** **Example:** **Example:** `Router(config)# interface atm number[. subinterface-number {multipoint | point-to-point}]`	Specifies the ATM interface using the appropriate format of the **interface atm** command.[3]
Step 2	Router(config-if)# **class-int** *vc-class-name*	Applies a VC class on an the ATM main interface or subinterface.

Configuring VC Management

When you configure VC management, you enable the router to detect VC connections and disconnections automatically. This notifies protocols to reroute packets immediately, preventing protocols from waiting for unpredictable and relatively long timeout periods.

You may use operation, administration, and maintenance (OAM) to manage your PVCs, and decide which method is reliable in your particular network.

When a PVC goes down, route caches for protocols configured on that PVC are cleared (or flushed) so that new routes may be learned. The route cache flush is applied on the PVC's interface. When all PVCs on a subinterface go down, VC management shuts down the subinterface in addition to flushing route caches. ATM hardware must keep the PVC active, however, so that OAM cells may flow. When any PVC on a subinterface comes up, the subinterface is brought up.

VC management using OAM is referred to as OAM management.

Configuring OAM Management for PVCs

OAM management may be enabled for PVCs.

By default, end-to-end F5 OAM loopback cell generation is turned off for each PVC. A PVC is determined as down when any of the following is true on that PVC:

- The router does not receive a loopback reply after a configured number of retries of sending end-to-end F5 OAM loopback cells.
- The router receives a Virtual Circuit-Alarm Indication Signals (VC-AIS) cell.
- The router receives a Virtual Circuit-Remote Detect Indicator (VC-RDI) cell.

A PVC is determined as up when all of the following are true on that PVC:

- The router receives a configured number of successive end-to-end F5 OAM loopback cell replies.
- The router does not receive VC-AIS cell for 3 seconds.
- The router does not receive VC-RDI cell for 3 seconds.

To configure OAM management for an ATM PVC, use the following commands beginning in global configuration mode:

SUMMARY STEPS

1. Do one of the following:
   - Router(config)# **interface atm** *slot* **/0[.** *subinterface-number* {**multipoint** | **point-to-point**}]
   -
   -
   - Router(config)# **interface atm** *slot* **/** *port-adapter* **/0[.** *subinterface-number* {**multipoint** | **point-to-point**}]
2. Router(config-if)# **pvc** [*name*] *vpi* **/** *vci*
3. Router(config-if-atm-vc)# **oam-pvc manage** [*frequency*]
4. Router(config-if-atm-vc)# **oam retry** *up-count down-count retry-frequency*

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	Do one of the following: • Router(config)# **interface atm** *slot* **/0[.** *subinterface-number* {**multipoint** | **point-to-point**}] • • • Router(config)# **interface atm** *slot* **/** *port-adapter* **/0[.** *subinterface-number* {**multipoint** | **point-to-point**}] **Example:** **Example:** **Example:**	Specifies the ATM interface using the appropriate format of the **interface atm** command.[4]

	Command or Action	Purpose
	`Router(config)# Router(config)# interface atm number[. subinterface-number {multipoint | point-to-point}]`	
Step 2	Router(config-if)# **pvc** [*name*] *vpi* / *vci*	Specifies the ATM PVC.
Step 3	Router(config-if-atm-vc)# **oam-pvc manage** [*frequency*]	Enables OAM management on the PVC.
Step 4	Router(config-if-atm-vc)# **oam retry** *up-count down-count retry-frequency*	(Optional) Specifies OAM management parameters for re-establishing and removing a PVC connection. • Use the *up-count*argument to specify the number of consecutive end-to-end F5 OAM loopback cell responses that must be received in order to change a PVC connection state to up. Use the *down-count*argument to specify the number of consecutive end-to-end F5 OAM loopback cell responses that are not received in order to tear down a PVC. Use the *retry-frequency* argument to specify the frequency (in seconds) that end-to-end F5 OAM loopback cells should be transmitted when a change in UP/DOWN state is being verified. For example, if a PVC is up and a loopback cell response is not received after the *frequency* (in seconds) specified using the **oam-pvc** command, then loopback cells are sent at the *retry-frequency* to verify whether or not the PVC is down.

Configuring Classical IP and Inverse ARP in a PVC Environment

The ATM Inverse ARP 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.

By default, Inverse ARP datagrams will be sent on this virtual circuit every 15 minutes. To adjust the Inverse ARP time period, use the **inarp** *minutes* command in interface-ATM-VC configuration mode.

Note The ATM ARP mechanism works with IP only. The Inverse ATM ARP mechanism works with IP and IPX only. For all other protocols, the destination address must be specified.

In a PVC environment, the ATM Inverse ARP mechanism is enabled by default for IP and IPX when you use the following commands beginning in global configuration mode:

SUMMARY STEPS

1. Do one of the following:
   - Router(config)# **interface atm** *slot* **/0**

-
-
- Router(config)# **interface atm** *slot* / *port-adapter* /**0**

2. Router(config-if)# **ip address** *address mask*
3. Router(config-if)# **pvc** [*name*] *vpi* / *vci*
4. Router(config-if-atm-vc)# **no shutdown**
5.

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	Do one of the following: • Router(config)# **interface atm** *slot* /**0** • • • Router(config)# **interface atm** *slot* / *port-adapter* /**0** **Example:** **Example:** **Example:** `Router(config)# interface atm number`	Specifies the ATM interface using the appropriate format of the **interface atm** command.[5]
Step 2	Router(config-if)# **ip address** *address mask*	Specifies the IP address of the interface.
Step 3	Router(config-if)# **pvc** [*name*] *vpi* / *vci*	Creates a PVC.
Step 4	Router(config-if-atm-vc)# **no shutdown**	Enables the ATM interface.
Step 5		Repeat Step 3 for each PVC you want to create.

Customizing the ATM Interface

You can customize the ATM interface. The features you can customize have default values that will most likely suit your environment and probably need not be changed. However, you might need to enter configuration commands, depending upon the requirements for your system configuration and the protocols you plan to route on the interface.

Configuring MTU Size

Each interface has a default maximum packet size or maximum transmission unit (MTU) size. For ATM interfaces, this number defaults to 4470 bytes. The MTU can be set on a per-sub-interface basis as long as the interface MTU is as large or larger than the largest subinterface MTU.

To set the maximum MTU size, use the following command in interface configuration mode:

Command	Purpose
`Router(config-if)# mtu bytes`	Sets the maximum MTU size.

ATM Configuration Examples

The examples in the following sections illustrate how to configure ATM for the features described in this chapter. The examples below are presented in the same order as the corresponding configuration task sections:

Creating a PVC Example

The following example shows how to create a PVC on an ATM main interface with AAL5/MUX encapsulation configured and a VBR-NRT QOS specified. For further information, refer to sections “Creating a PVC” and “Configuring PVC Traffic Parameters”.

```
interface 2/0
 pvc cisco 1/40
 encapsulation aal5mux ip
 vbr-nrt 100000 50000 20
 exit
```

PVC with AAL5 and LLC SNAP Encapsulation Examples

The following example shows how to create a PVC 0/50 on ATM interface 3/0. It uses the global default LLC/SNAP encapsulation over AAL5. The interface is at IP address 1.1.1.1 with 1.1.1.5 at the other end of the connection. For further information, refer to the sections “Creating a PVC” and “Mapping a Protocol Address to a PVC".

```
interface atm 3/0
 ip address 1.1.1.1 255.255.255.0
 pvc 0/50
 protocol ip 1.1.1.5 broadcast
 exit
!
 ip route-cache cbus
```

The following example is a typical ATM configuration for a PVC:

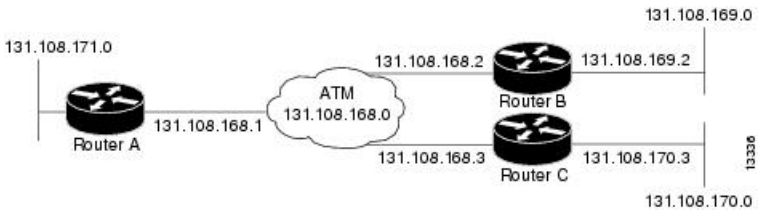
```
interface atm 4/0
 ip address 172.21.168.112 255.255.255.0
 atm maxvc 512
 pvc 1/51
 protocol ip 171.21.168.110
 exit
!
```

```
 pvc 2/52
 protocol decnet 10.1 broadcast
 exit
!
 pvc 3/53
 protocol clns 47.004.001.0000.0c00.6e26.00 broadcast
 exit
!
 decnet cost 1
 clns router iso-igrp comet
 exit
!
router iso-igrp comet
 net 47.0004.0001.0000.0c00.6666.00
 exit
!
router igrp 109
 network 172.21.0.0
 exit
!
ip domain-name CISCO.COM
```

PVCs in a Fully Meshed Network Example

The figure below illustrates a fully meshed network. The configurations for routers A, B, and C follow the figure. In this example, the routers are configured to use PVCs. Fully meshed indicates that any workstation can communicate with any other workstation. Note that the two **protocol** statements configured in router A identify the ATM addresses of routers B and C. The two **protocol**statements in router B identify the ATM addresses of routers A and C. The two **protocol** statements in router C identify the ATM addresses of routers A and B. For further information, refer to the sections “Creating a PVC” and “Mapping a Protocol Address to a PVC”.

Figure 3: Fully Meshed ATM Configuration Example



Router A

```
ip routing
!
interface atm 4/0
 ip address 131.108.168.1 255.255.255.0
 pvc 0/32
 protocol ip 131.108.168.2 broadcast
 exit
!
 pvc 0/33
 protocol ip 131.108.168.3 broadcast
 exit
```

Router B

```
ip routing
!
interface atm 2/0
 ip address 131.108.168.2 255.255.255.0
 pvc 0/32
 protocol ip 131.108.168.1 broadcast
 exit
!
 pvc 0/34
 protocol ip 131.108.168.3 broadcast
 exit
```

Router C

```
ip routing
!
interface atm 4/0
 ip address 131.108.168.3 255.255.255.0
 pvc 0/33
 protocol ip 131.108.168.1 broadcast
 exit
!
 pvc 0/34
 protocol ip 131.108.168.2 broadcast
 exit
```

Enabling Inverse ARP Example

The following example shows how to enable Inverse ARP on an ATM interface and specifies an Inverse ARP time period of 10 minutes.

```
interface atm 2/0/0.1
 pvc 1/32
 inarp 10
 exit
```

Enabling ANCP on ATM Interfaces Example

The following example shows how to enable ANCP on an ATM subinterface. In the example, ANCP is enabled on ATM subinterface 2/0/1.1.

```
interface ATM2/0/0.1 point-to-point
      description ANCP Link to one DSLAM
      no ip mroute-cache
      ip address 192.168.0.2 255.255.255.252
      pvc 254/32
        protocol ip 192.168.0.1
        ancp enable
        no snmp trap link-status
```

Configuring Loopback Cells Example

The following example shows how to enable OAM management on an ATM PVC. The PVC is assigned the name routerA and the VPI and VCI are 0 and 32, respectively. OAM management is enabled with a frequency of 3 seconds between OAM cell transmissions. For further information, refer to the section "Enabling ANCP on an ATM Interface, on page 16".

```
interface atm 2/0
 pvc routerA 0/32
 oam-pvc manage 3
 oam retry 5 5 10
```

Configuring PVC Trap Support Example

The following example shows how to configure PVC trap support on your Cisco router:

```
!For PVC trap support to work on your router, you must first have SNMP support and
!an IP routing protocol configured on your router:
Router(config)# snmp-server community public ro
Router(config)# snmp-server host 171.69.61.90 public

Router(config)# ip routing
Router(config)# router igrp 109
Router(config-router)# network 172.21.0.0
!
!Enable PVC trap support and OAM management:
Router(config)# snmp-server enable traps atm pvc interval 40 fail-interval 10
Router(config)# interface atm 1/0.1
Router(config-if)# pvc 0/1
Router(config-if-atm-vc)# oam-pvc manage
!
! Now if PVC 0/1 goes down, host 171.69.61.90 will receive traps.
```

For further information, refer to the "Configuring PVC Trap Support, on page 22" section.

Creating a VC Class Example

The following example shows how to create a VC class named main and how to configure UBR and encapsulation parameters. For further information, refer to the sections "Creating a VC Class, on page 24" and "Configuring VC Parameters, on page 24".

```
vc-class atm main
 ubr 10000
 encapsulation aal5mux ip
```

The following example shows how to create a VC class named sub and how to configure UBR and PVC management parameters. For further information, refer to the sections "Creating a VC Class, on page 24" and "Configuring VC Parameters, on page 24".

```
vc-class atm sub
 ubr 15000
 oam-pvc manage 3
```

The following example shows how to create a VC class named pvc and how to configure VBR-NRT and encapsulation parameters. For further information, refer to the sections "Creating a VC Class, on page 24" and "Configuring VC Parameters, on page 24"".

```
vc-class atm pvc
 vbr-nrt 10000 5000 64
 encapsulation aal5snap
```

Applying a VC Class Example

The following example shows how to apply the VC class named main to the ATM main interface 4/0. For further information, refer to the section "Applying a VC Class on an ATM PVC, on page 25".

```
interface atm 4/0
 class-int main
 exit
```

The following example shows how to apply the VC class named sub to the ATM subinterface 4/0.5:

```
interface atm 4/0.5 multipoint
 class-int sub
 exit
```

The following example shows how to apply the VC class named pvc directly on the PVC 0/56:

```
interface atm 4/0.5 multipoint
 pvc 0/56
 class-vc pvc
 exit
```

OAM Management on an ATM PVC Example

The following example shows how to enable OAM management on an ATM PVC. The PVC is assigned the name routerA and the VPI and VCI are 0 and 32, respectively. OAM management is enabled with a frequency of 3 seconds between OAM cell transmissions. For further information, refer to the section "Configuring OAM Management for PVCs, on page 26".

```
interface atm 2/0
 pvc routerA 0/32
 oam-pvc manage 3
 oam retry 5 5 10
!
interface atm 4/0.1 point-to-point
 pvc 0/35
 exit
```

No **map-group** and **map-list** commands are needed for IP.

```
interface atm 4/0
 ip address 1.1.1.1 255.0.0.0
 pvc 1/33
 pvc 1/34
 pvc 1/35
 bridge-group 1
!
bridge 1 protocol dec
```

Example Configuring a PVC Bundle

```
Device1:
interface ATM0/0/0.1 point-to-point
ip address 10.0.0.1 255.255.255.0
bundle test
  encapsulation aal5snap
  oam-bundle manage
  pvc-bundle 0/32
   vbr-rt 19000 15000 5000
   precedence 7
  pvc-bundle 1/33
   ubr 2480
   precedence 6
  pvc-bundle 1/34
   ubr 4890
   precedence 3-5
  pvc-bundle 1/35
!
Device2:

interface ATM0/1/0.1 point-to-point
ip address 10.0.0.2 255.255.255.0
bundle test
  encapsulation aal5snap
  oam-bundle manage
  pvc-bundle 0/32
   vbr-rt 19000 15000 5000
   precedence 7
  pvc-bundle 1/33
   ubr 2480
   precedence 6
  pvc-bundle 1/34
   ubr 4890
   precedence 3-5
  pvc-bundle 1/35
!
```

PVC on a Point-to-Point Subinterface Configuration Example

```
interface ATM5/0/0.9 point-to-point
 mtu 4474
 bandwidth 34000
 ip vrf forwarding vrfexample
 ip address 192.0.2.1 255.255.255.0
 ip mtu 4470
 pvc 11/105
  ubr 38
  oam-pvc manage
  encapsulation aal5snap
 !
interface ATM5/0/0.11 point-to-point
 mtu 4474
 bandwidth 7000
 ip address 192.0.2.2 255.255.255.0
 ip mtu 4470
 pvc 100/50
  cbr 7000
  encapsulation aal5snap
```

```
service-policy input Leased_Line_Ingress
max-reserved-bandwidth 100
```

Monitoring and Maintaining the ATM Interface

After configuring an ATM interface, you can display its status. You can also display the current state of the ATM network and connected virtual circuits. To show current virtual circuits and traffic information, use the following commands in EXEC mode:

Command	Purpose
`Router# show arp`	Displays entries in the ARP table.
`Router# show atm class-links {vpi / vci | name}`	Displays PVC parameter configurations and where the parameter values are inherited from.

`Router# show atm interface atm slot /0`
`Router# show atm interface atm slot / port-adapter /0`
`Router# show atm interface atm number`
	Displays ATM-specific information about the ATM interface using the appropriate format of the **show atm interface atm** command.[6]

`Router# show atm map`	Displays the list of all configured ATM static maps to remote hosts on an ATM network.
`Router# show atm pvc [vpi / vci | name | interface atm interface_number]`	Displays all active ATM PVCs and traffic information.
`Router# show atm traffic`	Displays global traffic information to and from all ATM networks connected to the router, OAM statistics, and a list of counters of all ATM traffic on this router.
`Router# show atm vc [vcd-number | [range lower-limit-vcd upper-limit-vcd] [interface ATM interface-number] [detail [prefix {vpi/vci | vcd | interface | vc_name}]] [connection-name] | signalling [freed-svcs | [cast-type {p2mp | p2p}] [detail] [interface ATM interface-number]] | summary ATM interface-number]`	Displays all active ATM virtual circuits (PVCs) and traffic information. **Note** The SVCs and the **signalling** keyword are not supported.
`Router# show interfaces atm controller.port-channels.subinterface`	Displays statistics for the ATM interface using the appropriate format of the **show interfaces atm** command.

Command	Purpose
`Router# show network-clocks`	Displays the clock signal sources and priorities that you established on the router.

[6] To determine the correct form of the interface atm command, consult your ATM network module, port adapter, or router documentation.

Feature Information for Configuring ATM

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 1: Feature Information for Configuring ATM

Feature Name	Releases	Feature Information
Configuring ATM, ATM Sub-interface Multipoint	Cisco IOS Release 2.5.0	

Feature Name	Releases	Feature Information
		ATM supports two types of interfaces: point-to-point and multipoint. • Point-to-point subinterface--With point-to-point subinterfaces, each pair of devices has its own subnet. If you put the PVC on a point-to-point subinterface, the device assumes that there is only one point-to-point PVC configured on the subinterface. Therefore, any IP packets with a destination IP address in the same subnet are forwarded on this virtual circuit (VC). This is the simplest way to configure the mapping and is therefore the recommended method. • Multipoint networks--Multipoint networks have three or more devices in the same subnet. If you put the PVC in a point-to-multipoint subinterface or in the main interface (which is multipoint by default), you need to either configure a static mapping or enable inverse Address Resolution Protocol (ARP) for dynamic mapping. ANCP needs to be enabled on ATM interface when message is sent from the DSLAM. The following commands were introduced or modified: **pvc**, **protocol**, **cbr**, **ubr**, **encapsulation aal5snap**, **interface atm**, **ip address**, **inarp**, **oam-pvc manage**, **oam-retry**, **class-vc**, **snmp-server enable traps**, **vc-class atm**, **class-int**, **mtu**, **show atm class-links**, **show atm interface**

Feature Name	Releases	Feature Information
		atm, **show atm map**, **show atm pvc**, **show atm traffic**, **show atm vc**, **show network-clocks**, ancp enable, ancp neighbor.
IP Multicast over ATM PVC Bundle	Cisco IOS Release XE 3.7S	ATM VC bundle management allows you to configure multiple VCs that have different QoS characteristics between any pair of ATM-connected devices. No new commands were introduced or modified.

Additional References

Related Documents

Related Topic	Document Title
ATM commands	*Cisco IOS Asynchronous Transfer Mode Command Reference*
ATM SPAs	"Overview of ATM SPAs" in the *Cisco ASR 1000 Series Aggregation Services Routers SIP and SPA Software Configuration Guide*
Configuring IP to ATM class of service (CoS)	• "IP to ATM CoS Overview" • "Configuring IP to ATM CoS"
Configuring PPP over Ethernet (PPPoE) over ATM	"Configuring PPPoE over ATM"

Standards

Standard	Title
No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.	--

MIBs

MIB	MIBs Link
• Cisco PVC trap MIB - CISCO-IETF-ATM2-PVCTRAP-MIB	To locate and download MIBs for selected platforms, Cisco IOS XE software releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

RFCs

RFC	Title
No new or modified RFCs are supported by this feature, and support for existing RFCs has not been modified by this feature.	--

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/techsupport