

Chapter 10

Configure ATM Interfaces

Asynchronous Transfer Mode (ATM) is a network protocol designed to facilitate the simultaneous handling of various types of traffic streams (voice, data, video) at very high speeds over the same physical connection. By always using 53-byte cells, ATM simplifies the design of hardware which can quickly determine the destination address of each cell. This allows simple switching of network traffic at much higher speeds than are easily accomplished using protocols with variable sizes of transfer units, such as Frame Relay and TCP/IP.

Although it was designed to operate without the requirement of any other networking protocol, other protocols are frequently segmented and encapsulated across multiple, smaller ATM cells; in effect making ATM a transport mechanism for pre-existing technologies such as Frame Relay and the TCP/IP family of protocols.

ATM relies on the concepts of Virtual Paths and Virtual Circuits. A Virtual Path, represented by a specific Virtual Path Identifier (VPI) establishes a route between two devices in a network. Each VPI may contain multiple Virtual Circuits, each represented by a Virtual Circuit Identifier (VCI).

VPIs and VCIs are local to the router, which means that only the two devices connected by the VCI or VPI need know the details of the connection. In a typical ATM network, user data might traverse multiple connections, using many different VPI and VCI connections. Each end device, just as each device in the network, needs to know only the VCI and VPI information for the path to the next device.

This chapter discusses the following ATM-specific interface properties that you can configure.

Configure ATM Physical Interface Properties on page 110

Configure ATM Virtual Circuits on page 111

Configure ATM Interface Encapsulation on page 117

Configure SONET Parameters on ATM Interfaces on page 118

Example: Configure ATM Interfaces on page 118

Configure ATM Physical Interface Properties

For ATM physical interfaces, you can configure two ATM-specific physical device properties: the maximum number of virtual circuits (VCs) allowed on a virtual path (VP) and communication with directly attached ATM switches. You configure these properties by including the `atm-options` statement at the `[edit interfaces interface-name]` hierarchy level:

```
[edit interfaces interface-name]
atm-options {
  vpi vpi-identifier max-vcs maximum-vcs;
  ilmi;
}
```

You can configure the following ATM-specific properties:

Configure the Maximum Number of Virtual Circuits on a Virtual Path on page 110

Configure Communication with Directly Attached ATM Switches on page 111

Configure the Maximum Number of Virtual Circuits on a Virtual Path

You configure the maximum number of virtual circuits allowed on a virtual path so that sufficient memory on the ATM PIC can be allocated for each VC. When configuring ATM interfaces on the router, you must include this statement.

To configure the largest numbered VCs on a VP, include the `vpi` statement at the `[edit interfaces interface-name atm-options]` hierarchy level:

```
[edit interfaces interface-name atm-options]
vpi vpi-identifier max-vcs maximum-vcs;
```

The VP identifier can be a value from 0 through 255. You can configure a maximum number of 4090 VCs per ATM interface and the largest value you can configure is 4089.

All the VPIs that you configure in the `atm-options` statement are stored by the software in a single table. If you modify the VPIs, for example, by editing them in configuration mode or by issuing a load override command, all VCs on the interface are closed and then reopened, resulting in a temporary loss of connectivity for all the VCs on the interface.

You also can include some of the statements in the `sonet-options` statement to set SONET parameters on ATM interfaces as described in “Configure SONET Parameters on ATM Interfaces” on page 118.

Configure Communication with Directly Attached ATM Switches

You configure communication to directly attached ATM switches to enable querying of the IP addresses and port numbers of the switches. You query the switch by entering the `show interfaces interface-name switch-id` command. The router uses VC 0.16 to communicate with the ATM switch.

To configure communication between the router and its directly attached ATM switches, include the `ilmi` statement at the `[edit interfaces interface-name atm-options]` hierarchy level:

```
[edit interfaces interface-name atm-options]
ilmi;
```

Configure ATM Virtual Circuits

When you are using ATM encapsulation on an interface, you must map each logical interface to a virtual circuit identifier (VCI) and optionally to a virtual path identifier (VPI).

An ATM interface can be a point-to-point interface or a point-to-multipoint (also called a multipoint nonbroadcast multi-access [NBMA]) connection.

When configuring ATM virtual circuits, you can do the following:

- Configure the Maximum Number of VCs on a VP on page 111

- Configure a Point-to-Point ATM Connection on page 112

- Configure a Point-to-Multipoint ATM Connection on page 112

- Configure a Multicast-Capable ATM Connection on page 112

- Define the ATM Traffic-Shaping Profile on page 113

- Define the ATM OAM F5 Loopback Cell Period on page 116

- Configure the ATM OAM F5 Loopback Cell Threshold on page 116

Configure the Maximum Number of VCs on a VP

When you are configuring ATM VCs, you must configure the maximum number of virtual circuits (VCs) allowed on a virtual path (VP). You configure this value so that sufficient memory on the ATM PIC can be allocated for each VC.

To configure the largest numbered VCs on a VP, include the `atm-options` statement at the `[edit interfaces interface-name]` hierarchy level:

```
[edit interfaces interface-name]
atm-options {
  vpi vpi-identifier max-vc maximum-vc;
}
```

The VP identifier can be a value from 1 through 255. The maximum number of VCs that can be configured per ATM interface is 4090. The largest VC value that can be configured is 4089.

Configure a Point-to-Point ATM Connection

To configure a VCI and a VPI on a point-to-point ATM interface, include the `vpi` statement at the [edit interfaces *interface-name* unit *logical-unit-number*] hierarchy level:

```
[edit interfaces interface-name unit logical-unit-number]
  vci vpi-identifier.vci-identifier;
```

For each VCI, you configure the VCI and VPI identifiers. The default VPI identifier is 0. The VCI identifier cannot exceed the highest numbered VC configured for the interface with the `vpi` statement, as described in “Configure ATM Physical Interface Properties” on page 110.

When you are configuring point-to-point connections, the MTU sizes on both sides of the connections must be the same.

Configure a Point-to-Multipoint ATM Connection

To configure a point-to-multipoint (NBMA) ATM connection, include the `multipoint-destination` statement within the `address` statement at the [edit interfaces *interface-name* unit *logical-unit-number* family *inet* address *address*] hierarchy level:

```
[edit interfaces interface-name unit logical-unit-number family inet address address]
  multipoint-destination destination-address vci vpi-identifier.vci-identifier;
```

address is the interface’s address. The address must include the destination prefix (for example, /24).

For each destination, include one `multipoint-destination` statement. *destination-address* is the address of the remote side of the connection, and *vci-identifier* and *vpi-identifier* are the VCI and optional VPI identifiers for the connection.

When you are configuring point-to-multipoint connections, all interfaces in the subnet must use the same MTU size.

Configure a Multicast-Capable ATM Connection

By default, ATM connections assume unicast traffic. If your ATM switch performs multicast replication, you can configure the connection to support multicast traffic by including the `mcast-vci` statement at the [edit interfaces *interface-name* unit *logical-unit-number*] hierarchy level:

```
[edit interfaces interface-name unit logical-unit-number]
  mcast-vci vpi-identifier.vci-identifier;
```

vci-identifier and *vpi-identifier* are the VCI and VPI identifiers, which define the ATM VCI over which the switch is expecting to receive multicast packets for replication.

You can configure multicast support only on point-to-multipoint ATM connections.

Configure ATM Inverse ARP

You can configure ATM interfaces to support inverse ATM ARP, as described in RFC 2225. When inverse ATM ARP is enabled, the router responds to received Inverse ATM ARP requests by providing IP address information to the requesting ATM device.

The router does not initiate inverse ATM ARP requests.

By default, inverse ATM ARP is disabled. To configure a VC to respond to inverse ATM ARP requests, include the `inverse-arp` statement at the [edit interfaces *interface-name* unit *logical-unit-number*] or [edit interfaces *interface-name* unit *logical-unit-number* family inet address *address* multipoint-destination *destination*] hierarchy level:

```
[edit interfaces interface-name unit logical-unit-number]
inverse-arp;
```

Define the ATM Traffic-Shaping Profile

When you are using an ATM encapsulation, you can configure a traffic-shaping profile that defines the following:

- Bandwidth utilization, which consists of either a constant rate, or a peak cell rate, with sustained cell rate, and burst tolerance

- Maximum queue length

These values are used in the ATM generic cell-rate algorithm, which is a leaky bucket algorithm that defines the short-term burst rate for ATM cells, the maximum number of cells that can be included in a burst, and the long-term sustained ATM cell traffic rate. Each individual VC has its own independent shaping parameters.

By default, the bandwidth utilization is unlimited; that is, unspecified bit rate (UBR) is used. Also, by default, buffer usage by VCs is unregulated. To define limits to bandwidth utilization on a point-to-point interface or to limit buffer use, include the shaping statement. For point-to-point interfaces, include the shaping statement at the [edit interfaces *interface-name* unit *logical-unit-number*] hierarchy level:

```
[edit interfaces interface-name]
unit logical-unit-number {
  vci vpi-identifier.vci-identifier;
  shaping {
    (cbr rate | vbr peak rate sustained rate burst length);
    queue-length number;
  }
}
```

For virtual circuits that are part of a point-to-multipoint interface, include the shaping statement at the [edit interfaces *interface-name* unit *logical-unit-number* family *family* address *address*] hierarchy level:

```
[edit interfaces interface-name unit logical-unit-number family family]
address address {
  multipoint-destination destination-address {
    vci vpi-identifier.vci-identifier;
    shaping {
      (cbr rate | vbr peak rate sustained rate burst length);
      queue-length number;
    }
  }
}
```

Configure CBR

For traffic which does not require the ability to periodically burst to a higher rate, you can specify a constant bit rate (CBR).

To specify a constant bit rate, include the `cbr` statement at the [edit interfaces *interface-name* unit *logical-unit-number* shaping] or [edit interfaces *interface-name* unit *logical-unit-number* family *family* address *address* multipoint-destination *destination-address* shaping] hierarchy level:

```
cbr rate;
```

Configure VBR

To define variable bandwidth utilization (VBR), include the `vbr` statement at the [edit interfaces *interface-name* unit *logical-unit-number* shaping] or [edit interfaces *interface-name* unit *logical-unit-number* family *family* address *address* multipoint-destination *destination-address* shaping] hierarchy level:

```
vbr peak rate sustained rate burst length;
```

When configuring VBR, you can define the following shaping properties:

Peak rate—Top rate at which traffic can burst.

Sustained rate—Normal traffic rate averaged over time.

Burst length—Maximum number of cells that a burst of traffic can contain. It can be a value from 1 through 255 cells.

Specify the rates in bits per second (bps) or cells per second (cps). For OC-3 interfaces, the highest rate is 135631698 bps (353207.55 cps), which corresponds to 100 percent of the available line rate. For OC-12 interfaces, the highest rate is 271263396 bps (706415.09 cps), which corresponds to 50 percent of the available line rate. Table 7 lists some of the other rates you can specify. If you specify a rate that is not listed, it is rounded to the nearest rate.

In general, the actual packet rate on the interface is calculated with the following formula:

$$\text{actual-rate} = (128 * \text{line-rate}) / (\text{trunc} ((128 * \text{line-rate}) / \text{desired-rate}))$$

line-rate is the maximum available rate on the interface (in bits per second) after factoring out the overhead for SONET and ATM (per-cell) overheads. For OC-3 interfaces, the line rate is calculated as follows:

$$\text{line-rate} = 155,520,000 \text{ bps} \times (26/27) \times (48/53) = 135,631,698.1 \text{ bps}$$

For OC-12 interfaces, the line rate is calculated as follows:

$$\text{line-rate} = 622,080,000 \text{ bps} \times (26/27) \times (48/53) = 542,526,792.45 \text{ bps}$$

desired-rate is the rate you enter in the `vbr` statement, in bits per second.

The `trunc` operator indicates that all digits to the right of the decimal point should be dropped.

For OC-3 interfaces, the maximum available rate is 100 percent of *line-rate*, or 135,631,698 bps. For OC-12 interfaces, the maximum available rate is 50 percent of *line-rate*, or 271,263,396 bps.

The following example shows the calculations for determining the actual rate when the desired rate is 80 percent of the maximum rate:

OC-3:

$135,631,968 \text{ bps} * 0.8 = 108,505,358 \text{ bps}$
 $actual-rate = (128 * 135,631,698.1) / (\text{trunc} ((128 * 135,631,698.1) / 101,723,773.5))$
 $actual-rate = 17,360,857,344 / (\text{trunc} ((17,360,857,344) / 101,723,773.5))$
 $actual-rate = 17,360,857,344 / 160$
 $actual-rate = 108,505,358 \text{ bps}$

OC-12:

$271,263,396 \text{ bps} * 0.8 = 217,010,716.8 \text{ bps}$
 $actual-rate = (128 * 542,526,792.45) / (\text{trunc} ((128 * 542,526,792.45) / 217,010,716.8))$
 $actual-rate = 69,443,429,434 / (\text{trunc} ((69,443,429,434) / 217,010,716.8))$
 $actual-rate = 69,443,429,434 / 320$
 $actual-rate = 217,010,717 \text{ bps}$

Table 7: Traffic-Shaping Rates

Interface Type	Line Rate (bps)	Line Rate (cps)	Percentage of Total Line Rate
OC-3			
	135631698	353207.55	100.00
	134580290	350469.50	99.22
	133545057	347773.58	98.46
	132525629	345118.82	97.71
	131521647	342504.29	96.97
	130532762	339929.07	96.24
	129558637	337392.28	95.52
	128598943	334893.08	94.81
	127653363	332430.63	94.12
	126721587	330004.13	93.43
OC-12			
	271263396	706415.09	50.00
	270207897	703666.40	49.81
	269160579	700939.01	49.61
	268121349	698232.68	49.42
	267090113	695547.17	49.23
	266066779	692882.24	49.04
	265051257	690237.65	48.85
	264043458	687613.17	48.67
	263043293	685008.58	48.48
	262050677	682423.64	48.30

Buffers are shared among all VCs, and by default, there is no limit to the buffer size for a VC. If a VC is particularly slow, it might use all the buffer resources. To limit the queue size of a particular VC, include the `queue-length` statement when configuring the VC:

```
queue-length number;
```

The length can range from 1 through 16383 packets. The default is 16383 packets.

Define the ATM OAM F5 Loopback Cell Period

When you are using an ATM encapsulation, you can configure the OAM F5 loopback cell period on virtual circuits, which is the interval at which OAM F5 loopback cells are transmitted.

By default, no OAM F5 loopback cells are sent. To send OAM F5 loopback cells on a point-to-point interface, include the `oam-period` statement at the [edit interfaces *interface-name* unit *logical-unit-number*] hierarchy level:

```
[edit interfaces interface-name]
unit logical-unit-number {
  vci vpi-identifier.vci-identifier;
  oam-period seconds;
}
```

To send OAM F5 loopback cells on a virtual circuit that is part of a point-to-multipoint interface, include the `oam-period` statement at the [edit interfaces *interface-name* unit *logical-unit-number* family *family* address *address*] hierarchy level:

```
[edit interfaces interface-name unit logical-unit-number family family]
address address {
  multipoint-destination destination-address {
    vci vpi-identifier.vci-identifier;
    oam-period seconds;
  }
}
```

The period can range from 1 through 900 seconds.

OAM VC-AIS (alarm indication signal) and VC-RDI (remote defect indication) defect indication cells are used for identifying and reporting VC defects end-to-end. When a physical link or interface failure occurs, intermediate nodes insert OAM AIS cells into all the downstream VCs affected by the failure. Upon receiving an AIS cell on a VC, the router marks the logical interface down and sends an RDI cell on the same VC to let the remote end know the error status. When an RDI cell is received on a VC, the router sets the logical interface status to down. When no AIS or RDI cells are received for 3 seconds, the router sets the logical interface status to up. You do not need to configure anything to enable defect indication.

Configure the ATM OAM F5 Loopback Cell Threshold

When you are using an ATM encapsulation, you can configure the OAM F5 loopback cell threshold on VCs, which is the minimum number of consecutive OAM F5 loopback cells received before declaring that a VC is up or lost before declaring that a VC is down.

By default, when five consecutive OAM F5 loopback cells are received, the VC is considered to be up, and when five consecutive cells are lost, the VC is considered to be down. To modify these values on a point-to-point interface, include the oam-liveness statement at the [edit interfaces *interface-name* unit *logical-unit-number*] hierarchy level:

```
[edit interfaces interface-name]
unit logical-unit-number {
  vci vpi-identifier.vci-identifier;
  oam-liveness {
    up-count cells;
    down-count cells;
  }
}
```

To modify the OAM F5 loopback cell count threshold on a virtual circuit that is part of a point-to-multipoint interface, include the oam-liveness statement at the [edit interfaces *interface-name* unit *logical-unit-number* family *family* address *address*] hierarchy level:

```
[edit interfaces interface-name unit logical-unit-number family family]
address address {
  multipoint-destination destination-address {
    vci vpi-identifier.vci-identifier;
    oam-liveness {
      up-count cells;
      down-count cells;
    }
  }
}
```

The cell count can be a value from 1 through 255 cells.

Configure ATM Interface Encapsulation

For an ATM interface, the physical interface encapsulation can be:

ATM PVC—ATM Permanent Virtual Circuit (PVC) encapsulation is defined in RFC 1483, *Multiprotocol Encapsulation over ATM Adaptation Layer 5*. ATM interfaces can use ATM PVC encapsulation.

To configure the encapsulation on a physical interface, include the encapsulation statement at the [edit interfaces *interface-name*] hierarchy level:

```
[edit interfaces interface-name]
encapsulation (atm-pvc | cisco-hdlc | frame-relay | ppp);
```

Generally, you configure an interface's encapsulation at the [edit interfaces *interface-name*] hierarchy level. However, for ATM encapsulations, you also can configure the encapsulation type that is used inside the ATM cell itself. To do this, include the encapsulation statement at the [edit interfaces *interface-name* unit *logical-unit-number*] hierarchy level:

```
[edit interfaces interface-name unit logical-unit-number]
encapsulation (atm-nlpid | atm-cisco-nlpid | atm-snap | atm-vc-mux | atm-ccc-vc-mux |
frame-relay-ccc);
```

The ATM encapsulations are defined in RFC 1483, *Multiprotocol Encapsulation over ATM Adaptation Layer 5*.

With the atm-nlpid, atm-cisco-nlpid, and atm-vc-mux encapsulations, you can configure the family inet only. With the circuit cross-connect (CCC) circuit encapsulations, you cannot configure a family on the logical interface.

Configure SONET Parameters on ATM Interfaces

When configuring ATM interfaces, you also can include the following statements in the sonet-options statement to set SONET parameters on ATM interfaces:

```
[edit interfaces at-fpc/pic/port]
sonet-options {
  bytes {
    f2 value;
    s1 value;
    z3 value;
    z4 value;
  }
  loopback (local | remote);
  (payload-scrambler | no-payload-scrambler);
}
```

Example: Configure ATM Interfaces

The following configuration is sufficient to get an ATM OC-3 or OC-12 interface up and running. By default, ATM interfaces use ATM PVC encapsulation.

```
[edit]
user@host# set interfaces at-fpc/pic/port atm-options vpi vpi-identifier
maximum-vcs vcs-value
[edit]
user@host# set interfaces at-fpc/pic/port unit 0 vci vpi-identifier.vci-identifier
[edit]
user@host# set interfaces at-fpc/pic/port unit 0 family inet address local-address
destination remote-address
[edit]
user@host# set interfaces at-fpc/pic/port unit 1 ...
[edit]
user@host# show
interfaces {
  at-fpc/pic/port {
    atm-options {
      vpi vpi-identifier maximum-vcs max-vcs value;
    }
    unit 0 {
      # one unit per VC
      vci vpi-identifier.vci-identifier;
      family inet {
        address local-address {
          destination destination-address;
        }
      }
    }
    unit 1 {
      # second VC
      ...
    }
  }
}
```

The following is a more complex configuration example:

```

interfaces {
  at-0/0/0 {
    encapsulation atm-pvc;
    atm-options {
      vpi 0 maximum-vcs 1200;
    }
    unit 2 {
      encapsulation atm-snap;
      inverse-arp;
      vci 0.80;
      family inet {
        mtu 1500;
        address 192.1.0.3/32 {
          destination 192.1.0.1;
        }
      }
    }
  }
  unit 3 {
    encapsulation atm-snap;
    vci 0.32;
    oam-period 60;
    family inet {
      mtu 1500;
      address 193.123.4.3/32 {
        destination 193.123.4.2;
      }
    }
  }
}
at-0/2/0 {
  encapsulation atm-pvc;
  atm-options {
    vpi 0 maximum-vcs 1200;
  }
  unit 2 {
    encapsulation atm-snap;
    inverse-arp;
    vci 0.82;
    family inet {
      mtu 1500;
      address 192.234.5.3/32 {
        destination 192.234.5.2;
      }
    }
  }
}
at-0/3/0 {
  no-keepalives;
  encapsulation atm-pvc;
  atm-options {
    vpi 0 maximum-vcs 1200;
  }
}

```

```
unit 140 {  
  encapsulation atm-snap;  
  multipoint;  
  family inet {  
    address 194.236.7.4/24 {  
      multipoint-destination 194.236.7.5;  
      vci 0.100;  
      inverse-arp;  
    }  
  }  
}  
at-7/3/0 {  
  encapsulation atm-pvc;  
  atm-options {  
    vpi 0 maximum-vcs 1200;  
  }  
  unit 0 {  
    encapsulation atm-snap;  
    vci 0.32;  
    family inet {  
      address 192.168.12.3/32 {  
        destination 192.168.12.2;  
      }  
    }  
  }  
}
```