CHAPTER 8

Configuring PNNI Communications

This chapter explains how to configure the RPM-XF to operate as an edge router in a PNNI network. When it is operating as a PNNI edge router, you can configure soft permanent virtual circuits (SPVCs) between the RPM-XF and other switch cards. For example, you can configure an SPVC between two RPM-XFs, or between an RPM-XF and an AXSM card. The SPVC can be configured between cards in the same switch, or between cards on two different switches. When the connection endpoints terminate on different switches, PNNI routes and, if necessary, reroutes connections between the endpoints.

This chapter begins with configuration quickstarts that provide overviews of the tasks required to configure RPM-XF SPVC connections. This chapter contains the following sections:

- Configuration Quickstarts
- Configuring PNNI Connections
- Connection Management
- Connection State Alarms

Configuration Quickstarts

Configuration quickstarts are designed as an overview and quick reference for those who have already configured RPM-XF cards. Use these quickstarts as a guide to configuring your RPM-XF card. If you need additional information on any step, look in the “Purpose” column for a reference to detailed documentation.

Switch and RPM-XF Preparation Quickstart

The following quickstart procedure describes tasks that prepare the switch and the RPM-XF to support multiple PNNI connections through RPM-XF. Follow this procedure whenever you configure a new RPM-XF in a switch.

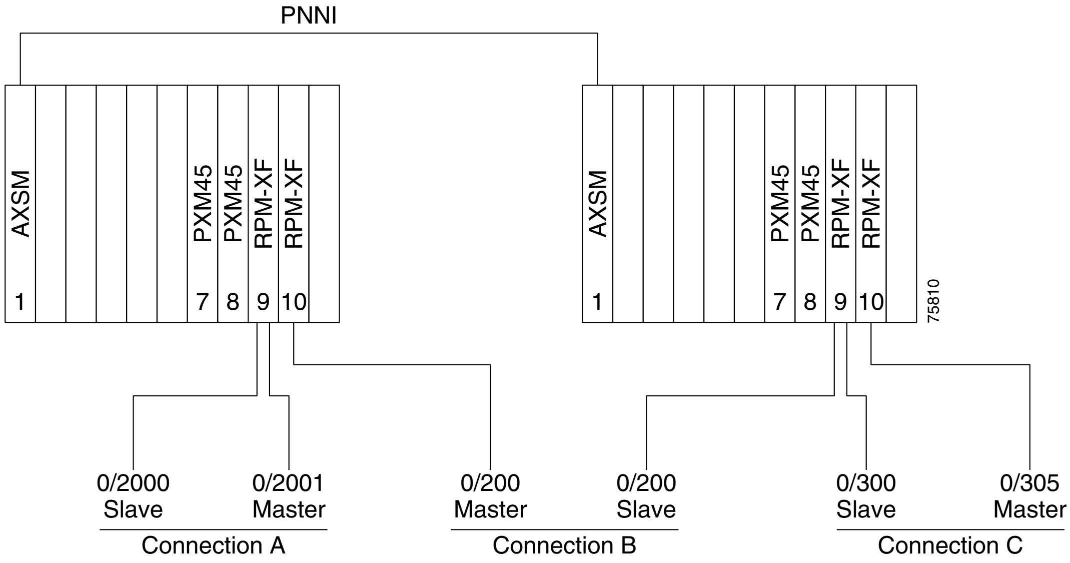
	Command	Purpose
Step 1	**Prompt:** `Switch.7.PXM.a >` **`dspcontrollers`**	Verify that a PNNI controller is defined on the PXM45 card and configured correctly. A PNNI controller is defined once for each switch. See the "Verifying the PNNI Controller Configuration" section later in this chapter.
	addcontroller *<cntrlrId>* **i** *<cntrlrType> <slot> [cntrlrName]*	Adds a controller.

Step 2	**Prompt:** `RPM-XF (config) #`
	`interface switch1`
	`switch partition` `<options>`
	`ingress-percentage-bandwidth` `<options>`
	`egress-percentage-bandwidth` `<options>`
	`vpi` `<options>`
	`vci` `<options>`
	`connection-limit` `<options>`
	Related commands:
	`show switch partitions`
		Assign the switch1 interface resources to the PNNI controller. This procedure must be completed once for each RPM-XF card.
		See the "Assigning Link Resources to a PNNI Controller" section later in this chapter.

RPM-XF to RPM-XF Connection Quickstart

The following quickstart procedure describes how to configure a PNNI SPVC between two RPM-XF subinterfaces. Figure 8-1 illustrates three types of RPM-XF to RPM-XF connections.

Figure 8-1 RPM-XF-to-RPM-XF Connections



Connection endpoints are established on subinterfaces, which are identified by the PVC number in the format VPI/VCI. The subinterfaces can be located on the same RPM-XF card, on different RPM-XF cards within the same switch, or on RPM-XF cards in different switches. You must define the slave end of the connection first, and then the master end.

Connection B is defined between subinterfaces on two different RPM-XF cards in two different switches. Before the connection can operate, the PNNI link between the two switches must be established. Refer to the *Cisco MGX 8850 and MGX 8950 Switch Software Configuration Guide*.

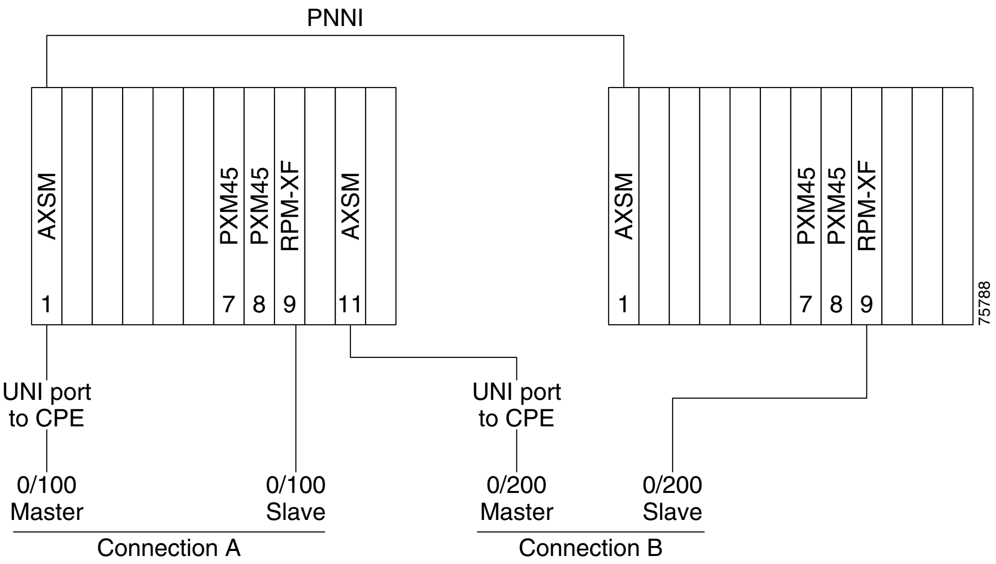
Connection A is defined between two subinterfaces on the same RPM-XF; connection C is configured between two subinterfaces that are on different RPM-XF cards in the same switch. When both ends of a connection terminate on the same switch, there is no need for the PNNI trunk shown in Figure 8-1.

	Command	Purpose
Step 1	**interface Switch 1.xx** <options> **ip address** <options> **pvc** vpi/vci	At the RPM that will host the slave side of the connection, create and configure a subinterface to host the slave side connection. See the "Creating and Configuring a Switch Subinterface" section later in this chapter.
Step 2	**switch connection** <options> Related commands **show switch connection** **show ip int br**	Add the slave end of the new connection to the subinterface PVC. See the "Creating a Slave Connection on the RPM-XF" section later in this chapter.
Step 3	**cc** <slot> **dspcon** <port> <vpi> <vci>	Change to an active PXM45 card and copy or write down the ATM address for the slave endpoint. See the "Creating a Slave Connection on the RPM-XF" section later in this chapter.
Step 4	**interface Switch 1.xx** <options> **ip address** <options> **pvc** vpi/vci	At the RPM that will host the master side of the connection, create and configure a subinterface to host the master side connection. See the "Creating and Configuring a Switch Subinterface" section later in this chapter.
Step 5	**switch connection** <options> Related commands **show switch connection** **show ip int br**	Add the master end of the new connection to the subinterface PVC. See the "Creating a Master Connection on the RPM-XF" section later in this chapter.

RPM-XF Slave to the AXSM Master Connection Quickstart

The following quickstart procedure describes how to configure a PNNI SPVC between an RPM-XF subinterface and an AXSM port. Figure 8-2 illustrates two types of RPM-XF to AXSM connections.

Figure 8-2 RPM-XF Slave to AXSM Master Connections



Connection endpoints are established on subinterfaces, which are identified by the PVC number in the format VPI/VCI. The RPM-XF and AXSM cards can be located on the same switch or on different switches. You must define the RPM-XF as the slave end of the connection first, and then you can define the master end of the connection.

Connection B is defined between the RPM-XF and AXSM cards in two different switches. Before the connection can operate, the PNNI link between the two switches must be established. Refer to the *Cisco MGX 8850 and MGX 8950 Switch Software Configuration Guide*.

Connection A is defined between the RPM-XF and AXSM cards in the same switch. When both ends of a connection terminate on the same switch, there is no need for the PNNI trunk shown in Figure 8-2.

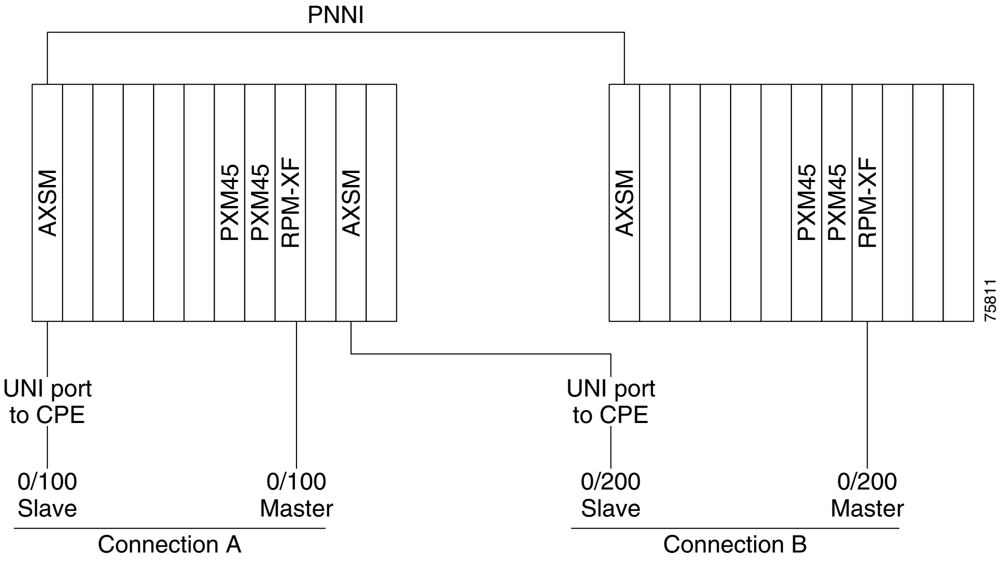
	Command	Purpose
Step 1	**interface Switch 1.xx** <options> **ip address** <options> **pvc** vpi/vci	At the RPM that hosts the slave side of the connection, create and configure a subinterface to host the slave side connection. See the "Creating and Configuring a Switch Subinterface" section later in this chapter.
Step 2	**switch connection** <options> Related commands **show switch connection** **show ip int br**	Add the slave end of the new connection to the subinterface PVC. See the "Creating a Slave Connection on the RPM-XF" section later in this chapter.

	Command	Purpose
Step 3	**cc** <slot> **dspcon** <port> <vpi> <vci>	Change to active PXM45 card and copy or write down the ATM address for the slave endpoint. See the "Creating a Slave Connection on the RPM-XF" section later in this chapter.
Step 4	**cnfcdsct** **upln**	At the AXSM card that will host the master side of the connection, configure the card and the line, if these components are not configured already. Refer to the *Cisco MGX 8850 and MGX 8950 Switch Software Configuration Guide,* Chapter 3, "Preparing AXSM Cards and Lines for Communication."
Step 5	**dsppnni-link** **dsppnni-node-list** **dsppnni-node** **dsppnni-reachable-addr network**	If the RPM-XF card that hosts the slave side of the connection is on a different switch, verify communications between the local and remote switches. Refer to the *Cisco MGX 8850 and MGX 8950 Switch Software Configuration Guide,* Chapter 5, "Provisioning AXSM Communication Links," the "Verifying PNNI Communications" section. **Note** The **dsppnni-link** command provides real-time data. For the other **dsppnni** commands, designate a PNNI topology state element (PTSE) time out.
Step 6	**addport** **addpart**	Add an AXSM UNI port to host the master connection and create a PNNI partition for that port. Refer to the *Cisco MGX 8850 and MGX 8950 Switch Software Configuration Guide,* Chapter 5, "Provisioning AXSM Communication Links," the "MPLS and PNNI UNI Port Configuration Quickstart" section.
Step 7	**addcon** <options> Related commands **dspcons** **dspcon** <port> <vpi> <vci>	Add the master side of the connection to the UNI port. Refer to the *Cisco MGX 8850 and MGX 8950 Switch Software Configuration Guide,* Chapter 5, "Provisioning AXSM Communication Links," the "Configuring the Master Side of SPVCs and SPVPs" section.

AXSM Slave to RPM-XF Master Connection Quickstart

The following quickstart procedure describes how to configure a PNNI SPVC between an RPM-XF subinterface and an AXSM port. Figure 8-3 illustrates two types of RPM-XF to AXSM connections.

Figure 8-3 RPM-XF Master to AXSM Slave Connections



Connection endpoints are established on subinterfaces, which are identified by the PVC number in the format VPI/VCI. The RPM-XF and AXSM cards can be located on the same switch or on different switches. You must define the slave end of the connection first, and then define the master end.

Connection B is defined between RPM-XF and AXSM cards in two different switches. Before the connection can operate, the PNNI link between the two switches must be established. Refer to the *Cisco MGX 8850 and MGX 8950 Switch Software Configuration Guide*.

Connection A is defined between RPM-XF and AXSM cards in the same switch. When both ends of a connection terminate on the same switch, there is no need for the PNNI trunk shown in Figure 8-3.

	Command	Purpose

Step 1	`dnln`
	`cnfcdsct`
	`upln`
		At the AXSM card that will host the slave side of the connection, configure the card and the line, if these components have not been configured already.
		Refer to the *Cisco MGX 8850 and MGX 8950 Switch Software Configuration Guide,* Chapter 3, “Preparing AXSM Cards and Lines for Communication.”

Step 2	`dsppnni-link`
	`dsppnni-node-list`
	`dsppnni-node`
	`dsppnni-reachable-addr network`
		If the RPM-XF card that hosts the master side of the connection is on a different switch, verify communications between the local and remote switches.
		Refer to the *Cisco MGX 8850 and MGX 8950 Switch Software Configuration Guide,* Chapter 5, “Provisioning AXSM Communication Links,” the “Verifying PNNI Communications” section.

	Command	Purpose
Step 3	**addport** **addpart**	Add an AXSM UNI port to host the slave connection and create a PNNI partition for that port. Refer to the *Cisco MGX 8850 and MGX 8950 Switch Software Configuration Guide,* Chapter 5, "Provisioning AXSM Communication Links," the "MPLS and PNNI UNI Port Configuration Quickstart" section.
Step 4	**addcon** <options> Related commands **dspcons** **dspcon** <port> <vpi> <vci>	Add the slave side of the connection to the UNI port. Be sure to copy or write down the ATM address for the slave endpoint. Refer to the *Cisco MGX 8850 and MGX 8950 Switch Software Configuration Guide,* Chapter 5, "Provisioning AXSM Communication Links," the "Configuring the Slave Side of SPVCs and SPVPs" section.
Step 5	**interface Switch 1.xx** <options> **ip address** <options> **pvc** vpi/vci	At the RPM that will host the master side of the connection, create and configure a subinterface to host the master side connection. See the "Creating and Configuring a Switch Subinterface" section later in this chapter.
Step 6	**switch connection** <options> Related commands **show switch connection** **show ip int br**	Add the master end of the new connection to the subinterface PVC. See the "Creating a Master Connection on the RPM-XF" section later in this chapter.

Configuring PNNI Connections

The following sections describe the steps listed in the configuration quickstarts.

Verifying the PNNI Controller Configuration

A PNNI controller must be added to the switch before PNNI can route communications for RPM-XF connections. The PNNI controller is added just once, usually during general switch configuration. To verify that the PNNI controller is added and configured correctly, enter the **dspcontrollers** command on the PXM45 as shown in the following example.

```
Switch.7.PXM.a > dspcontrollers
Switch                           System Rev: 02.01   Mar. 22, 2001 11:25:29 PST
MGX8850                                              Node Alarm: CRITICAL
Number of Controllers:        1
Controller Name:              PNNI Controller
Controller Id:                2
Controller Location:          Internal
Controller Type:              PNNI
Controller Logical Slot:      7
Controller Bay Number:        0
Controller Line Number:       0
Controller VPI:               0
Controller VCI:               0
Controller In Alarm:          NO
Controller Error:
```

The Controller ID, Controller Location, and Controller Type must match the values shown in the example above. The controller name is defined by the person that creates the controller and can be different from what is shown above.

If the **dspcontrollers** command does not display a PNNI controller, enter the **addcontroller** command. (Refer to the *Cisco MGX 8850 and MGX 8950 Switch Software Configuration Guide*.)

Assigning Link Resources to a PNNI Controller

Link resources must be defined on the RPM-XF before you can create a connection or configure a PNNI. Link resources include the following features.

- Ingress bandwidth
- Egress bandwidth
- Virtual path identifier (VPI) range
- Virtual channel identifier (VCI) range
- Number of connections

The **switch partition** command is used to add or modify the resource partitioning on the RPM-XF. Enter the **switch partition** command before you add any connections to the RPM-XF.

Note PAR is not supported.

To assign link resources to a controller, use the following switch partition routine.

Step 1 Enter the **switch partition** command.

```
(config-if)# switch partition <partId> <ctrlrId>
```

Parameter	Description
partId	Range is 1 to 10.
ctrlrId	Range is 2 to 20; **2** is reserved for PNNI.

Step 2 Enter the **ingress-percentage-bandwidth** command at the swpart prompt to specify the minimum and maximum ingress percentage bandwidth.

```
(config-if-swpart)# ingress-percentage-bandwidth <ingMinPctBw> <ingMaxPctBw>
```

Step 3 Enter the **egress-percentage-bandwidth** command to specify the minimum and maximum egress percentage bandwidth.

```
(config-if-swpart)# egress-percentage-bandwidth <egrMinPctBw> <egrMaxPctBw>
```

Step 4 Enter the **vpi** command to specify the minimum and maximum vpi.

```
(config-if-swpart)# vpi <min_vpi> <max_vpi>
```

Step 5 Enter the **vci** command to specify the minimum and maximum vci.

```
(config-if-swpart)# vci <min_vci> <max_vci>
```

Step 6 Enter the **connection-limit** command to specify the minimum and maximum number of connections.

```
(config-if-swpart)# connection-limit <min_con> <max_con>
```

The following is an example of the switch partition commands.

```
Router(config-if)#switch part 1 2
Router(config-if-swpart)#ingress-percentage-bandwidth 1 100
Router(config-if-swpart)#egress-percentage-bandwidth 1 100
Router(config-if-swpart)#vpi 0 0
Router(config-if-swpart)#vci 2000 3000
Router(config-if-swpart)#connection-limit 1000 4000
```

partId = **1** for PNNI.

ctrlrId = **2** for PNNI.

Table 8-1 describes **switch partition** command parameters.

Table 8-1 Switch Partition Parameter Description

Parameter	Description
ingress-percent	The percentage of the ingress bandwidth on the ATM switch interface that can be allocated by the controller type. The aggregate of the ingress bandwidth across all three controllers can exceed 100 percent.
egress-percent	The percentage of the egress bandwidth on the ATM switch interface that can be allocated by the controller type. The aggregate of the egress bandwidth across all three controllers can exceed 100 percent.
min-vpi	The minimum VPI value that can be assigned on SPVCs on this controller.
max-vpi	The maximum VPI value that can be assigned on SPVCs on this controller.
min-vci	The minimum VCI value that can be assigned on SPVCs on this controller.
max-vci	The maximum VCI value that can be assigned on SPVCs on this controller.
min-connection-limit	The minimum number of connections that can be added on this controller.
max-connection-limit	The maximum number of connections that can be added on this controller.

VPI and VCI Assignments

The following list shows how VPI/VCI resources on the RPM-XF can be partitioned for the PNNI controller as follows.

- Partition 1 (Partition ID 5; Interface # 1)—VPI range: 0 to 100; VCI range: 32 to 65535
- Partition 2 (Partition ID 6; Interface # 1)—VPI range: 101 to 200; VCI range: 32 to 65535
- Partition 3 (Partition ID 1; Interface # 1)—VPI range: 201 to 240; VCI range: 32 to 65535

VPI/VCI ranges cannot overlap between partitions, and a partition can only be used by one controller. The VPI/VCI range can be expanded or reduced as long as the VPI/VCIs are not in use. Existing connections will remain unaffected.

The VPI and VCI partitioning parameters can be configured from CWM or via the CLI command **switch partition.** Refer to the *Cisco MGX 8850 Routing Switch Command Reference* for the command syntax and usage.

Bandwidth Allocations

Bandwidth is also configured by using the **switch partition** command. The bandwidth allocated to each controller is managed by the following parameters:

- Minimum bandwidth is the guaranteed minimum bandwidth that will be reserved for use by the controller.
- Maximum bandwidth is the maximum bandwidth that can be used by the controller.

Bandwidth partitioning for the ingress direction and the egress direction are managed separately. As with VPI/VCI resources, the bandwidth partition can be expanded or reduced as long as the resource is not in use.

The bandwidth partitioning parameters can be configured through the **switch partition** CLI command as shown in this example.

```
Router(config-if-swpart)#ingress-percentage-bandwidth 1 100
Router(config-if-swpart)#egress-percentage-bandwidth 1 100
```

Number of Connections

The number of connections or **connection-limit** is also configured by entering the **switch partition** command. The number of connections or lvcs that each controller can add is managed by the following parameters:

- min-connection-limit is the minimum number of connections reserved for a controller.
- max-connection-limit is the maximum number of connections reserved for a controller.

Bandwidth partitioning for the ingress direction and the egress direction are managed separately. As with VPI/VCI resources, the bandwidth partition can be expanded or reduced as long as the resource is not in use.

The number of connections can be configured through the **switch partition** CLI command as shown in this example.

```
Router(config-if-swpart)#connection-limit 100 1000
```

Switch Partition Provisioning

Use the switch partition routine to set the percentage of ingress and egress bandwidth, as shown in the following procedure. (See the "Assigning Link Resources to a PNNI Controller" section earlier in this chapter.)

Step 1 Enter the following **switch partition** commands to configure the resource partition and set the percentage of ingress and egress bandwidth on the RPM.

```
Router(config-if)# switch partition <partId> <ctrlrId>
Router(config-if-swpart)# ingress-percentage-bandwidth <ingMinPctBw> <ingMaxPctBw>
Router(config-if-swpart)# egress-percentage-bandwidth <egrMinPctBw> <egrMaxPctBw>
Router(config-if-swpart)# vpi <min_vpi> <max_vpi>
Router(config-if-swpart)# vci <min_vci> <max_vci>
Router(config-if-swpart)# connection-limit <min_con> <max_con>
```

partId = **1** for PNNI

ctrlrId = **2** for PNNI

Step 2 Enter the **copy run start** command to save the configuration to the RPM's memory.

```
Router#config terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#interface switch1
Router(config-if)#switch part 1 2
Router(config-if-swpart)#ingress-percentage-bandwidth 1 100
Router(config-if-swpart)#egress-percentage-bandwidth 1 100
Router(config-if-swpart)#vpi 0 0
Router(config-if-swpart)#vci 2000 3000
Router(config-if-swpart)#connection-limit 100 1000
Router(config-if-swpart)#end
Router#copy run start
Building configuration...
[OK]
```

Step 3 Enter the **show switch partition** commands to verify your configuration.

```
Router#show switch partition
Part Ctrlr  Guar    Max     Guar    Max
Id   Id     Ing%Bw  Ing%Bw  Egr%Bw  Egr%Bw  minVpi maxVpi minVci maxVci MaxCons
1    2      1       100     1       100     0      0      2000   3000   1001

Router#show switch partition vcc 1
-------------------------------------------------------
Shelf                    : 1
Pxm Slot                 : 7
Slot                     : 9
IfType                   : 3
IfNum                    : 1
Partition ID             : 1
Controller ID            : 2
Guaranteed Ingress Pct BW: 1
Max Ingress Pct BW       : 100
Guaranteed Egress Pct BW : 1
Max Egress Pct BW        : 100
VPI Low                  : 0
VPI High                 : 0
VCI Low                  : 2000
VCI High                 : 3000
Maximum # of Connections : 1001
```

Configuring Switch Interface Signaling

The procedure in this section describes how to configure the signaling for the switch interface so it can support PNNI connections. This configuration is performed on the PXM45 card.

On the PXM45 card, the switch interface appears as a PNNI port, as shown in the following procedure.

Step 1 Enter the **dsppnports** command to determine which PNNI ports represent RPM-XF switch interfaces.

```
Switch.7.PXM.a > dsppnports
Summary of total connections
(p2p=point to point,p2mp=point to multipoint,SpvcD=DAX spvc,SpvcR=Routed spvc)
Type   #Svcc:   #Svpc:   #SpvcD:  #SpvpD:  #SpvcR:  #SpvpR:  #Total:
p2p:   0        0        3        0        0        0        3
p2mp:  0        0        0        0        0        0        0
                                                        Total=3
Summary of total configured SPVC endpoints
Type   #SpvcCfg: #SpvpCfg:
p2p:   7         0
```

```
p2mp:  0         0

Per-port status summary

PortId         IF status         Admin status       ILMI state       #Conns

7.35           up                up                 Undefined        0

7.36           up                up                 Undefined        0

7.37           up                up                 Undefined        0

7.38           up                up                 Undefined        0

Type <CR> to continue, Q<CR> to stop:

9.1.2.2        up                up                 Undefined        4

1:1.1:1        provisioning      up                 Undefined        0

1:1.2:2        down              up                 Undefined        0

2:1.1:1        up                up                 Disable          1

2:2.1:1        provisioning      down               Undefined        0

2:2.2:2        provisioning      down               Undefined        0

3:2.1:5        up                up                 UpAndNormal      1

3:2.2:4        building vc       up                 Disable          0
```

In the example above, port 9.1.2.2 represents an RPM-XF switch interface.

Step 2 To display the port signaling type, enter the **dsppnportsig** command.

```
Switch.7.PXM.a > dsppnportsig 9.1.2.2

provisioned IF-type: uni   version:     uni3.1
sigType:  private          side:        network
addrPlan:  aesa
sigVpi:          0         sigVci:            5
rccVpi:        n/a         rccVci:          n/a
```

The *provisioned IF-type* field in the example above indicates this port is a UNI port, and the version field indicates the signaling is configured to the default value, *uni3.1*. The correct value for a PNNI RPM-XF port is *none*. If the signaling is configured correctly, the **dsppnportsig** command displays the following information.

```
Switch.7.PXM.a > dsppnportsig 9.1.2.2

provisioned IF-type: uni   version:     none
sigType:  private          side:        network
addrPlan:  aesa
sigVpi:          0         sigVci:            5
rccVpi:        n/a         rccVci:          n/a
```

Step 3 To change the port signaling configuration, enter the **dnpnport** command to bring down the port on which you want to configure signaling. For example,

```
Switch.7.PXM.a > dnpnport 9.1.2.2
```

Step 4 Configure the RPM-XF port UNI signaling to *self* by entering the **cnfpnportsig** command.

```
Switch.7.PXM.a > cnfpnportsig <portid> -univer none
```

Replace portid with the port number as shown in the following example:

```
Switch.7.PXM.a > cnfpnportsig 9.1.2.2 -univer none
```

Step 5 Bring up the port you just configured using the **uppnport** command. For example,

```
Switch.7.PXM.a > uppnport 9.1.2.2
```

Step 6 To verify the configuration change, reenter the **dsppnportsig** command.

Creating and Configuring a Switch Subinterface

The switch interface on the RPM-XF does not directly support connection endpoints. Before you can create a connection endpoint, you must define a switch subinterface and define a PVC on that interface. The connection endpoint is configured using the VPI and VCI of the PVC.

Some subinterfaces support multiple PVCs (multipoint) that do the equivalent of broadcasting. Others support only one PVC (point-to-point). If a point-to-multipoint PVC exists, then that PVC can be used as the sole broadcast PVC for all multicast requests.

Each subinterface is identified using the notation **interface switch 1.<***subinterface***>.** The **interface switch 1**is the switch interface number, which is always 1, and the <*subinterface*> is a number that identifies the subinterface. You can choose the subinterface number when you create the subinterface. The subinterface number has to be unique on the RPM-XF card, but it does not have to match any other number.

To create a switch subinterface, configure the following features, as described in the procedure below:

- IP address for the subinterface
- PVC for the subinterface
- PVC configuration parameters

Step 1 To create the subinterface, enter the **interface** command.

```
Router(config)# interface switch 1.<subinterface> <multipoint | point-to-point | mpls |
tag-switching>
```

The following example adds subinterface **1** to the switch **1** interface and defines the subinterface as a point-to-point connection.

```
Router(config)#interface switch 1.1 point-to-point
```

Step 2 Enter the **ip** command to add an IP address to the subinterface.

```
Router(config-subif)# ip address <ip_addr> <subnet_mask>
```

The following example adds IP address 1.1.1.1 to subinterface 1 and defines the network mask as 255.255.255.0

```
Router(config-subif)#ip address 1.1.1.1 255.255.255.0
```

Note The IP addresses for the subinterfaces at the slave and master ends of a connection should share the same subnet.

Step 3 Enter the **pvc** command to add a PVC to the subinterface.

```
Router(config-subif)# pvc <vpi>/<vci>
```

The following example creates a PVC on the subinterface and assigns it VPI 0 and VCI 2000.

```
Router(config-subif)#pvc 0/2000
```

After you enter this command, the switch enters virtual circuit configuration mode for this PVC.

Note The VPI and VCI values you enter for the PVC must be within the ranges set for the PNNI controller when the PNNI partition was defined for the switch interface. For more information, see the "Assigning Link Resources to a PNNI Controller" section earlier in this chapter.

Step 4 Enter a question mark to list the commands available for configuring the PVC.

```
Router(config-if-atm-vc)#?
ATM virtual circuit configuration commands:
  atm                      atm pvc commands
  broadcast                Pseudo-broadcast
  class-vc                 Configure default vc-class name
  default                  Set a command to its defaults
  dialer                   set dialer pool this pvc belongs to
  encapsulation            Select ATM Encapsulation for VC
  exit-vc                  Exit from ATM VC configuration mode
  ilmi                     Configure ILMI management
  inarp                    Change the inverse arp timer on the PVC
  ip addr inarp            Assign an ip address to the atm interface through
                           ATMInarp
  max-reserved-bandwidth   Maximum Reservable Bandwidth on a vc
  no                       Negate a command or set its defaults
  oam                      Configure oam parameters
  oam-pvc                  Send oam cells on this pvc
  pppoe                    PPPoE options
  pppoe-client             pppoe client
  protocol                 Map an upper layer protocol to this connection.
  random-detect            Configure WRED
  service-policy           Attach a policy-map to a VC
  transmit-priority        set the transmit priority for this VC
  tx-ring-limit            Configure PA level transmit ring limit
  ubr                      Enter Unspecified Peak Cell Rate (pcr) in Kbps.
  vbr-nrt                  Enter Variable Bit Rate (pcr)(scr)(bcs)
  vbr-rt                   Enter Variable Bit Rate (pcr)(average)
  vc-hold-queue            Configure hold queue size
  vcci                     VCC Identifier
```

The following example shows some commands you might want to use to configure the PVC.

```
Router(config-if-atm-vc)#oam-pvc manage
Router(config-if-atm-vc)#encapsulation aal5snap
```

Step 5 When you have finished configuring the PVC, enter the **exit-vc** command to return to subinterface configuration mode.

```
Router(config-if-atm-vc)#exit-vc
Router(config-subif)#
```

Creating a Slave Connection on the RPM-XF

When you create a slave connection on an RPM-XF card, that connection endpoint does not route or reroute connections. Connection routing is the responsibility of the master connection endpoint.

To perform routing and rerouting, the master connection endpoint requires the ATM address of the slave endpoint, so the slave endpoint must be defined first. The following procedure describes how to create a slave connection endpoint.

Note You must configure both the slave and master connection endpoints before the connection can operate.

If you have not already done so, create a subinterface and PVC to host the slave connection endpoint. See the "Creating and Configuring a Switch Subinterface" section earlier in this chapter.

Step 1 If the switch is not in subinterface configuration mode, change to that mode.

```
Router>enable
Password:
Router#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#interface switch 1.1
Router(config-subif)#
```

Step 2 To create a VCC, define the slave connection endpoint with the **switch connection** command as follows:

```
Router(config-subif)# switch connection vcc <localVPI> <localVCI> master remote
```

The VPI and VCI that you enter must match the VPI and VCI you used when you configured the PVC that hosts this connection. The following example creates a slave connection for the PVC labeled VPI 0, VCI 2000.

```
Router(config-subif)#switch connection vcc 0 2000 master remote
```

Step 3 To create a VPC, define the slave connection endpoint with the **switch connection** command as follows:

```
Router(config-subif)# switch connection vpc <localVPI> master remote
```

The VPI that you enter must match the VPI you used when you configured the PVC that hosts this connection.

You must also add the atm pvp tunnel using the same VPI, before adding the PVC, as follows:

```
Router(config-subif)# atm pvp <vpi> <PCR>
```

After you create the slave connection endpoint, the RPM-XF enters switch connection configuration mode. The following prompt displays:

```
Router(config-if-swconn)#
```

Step 4 To display a list of configuration commands, enter a question mark at the switch connection prompt. For example,

```
Router(config-if-swconn)#?
Switch connection configuration commands:
  auto_synch    enable auto synch
  cost          Maximum connection cost
  default       Set a command to its defaults
  exit-swconn   Exit from switch connection configuration mode
  no            Negate a command or set its defaults
  priority      Routing Priority
  reroute       reroute the connection
  rmbs          remote MBS value
  rpcr          remote PCR value
  rscr          remote SCR value
  rutil         Connection remote percent utilization
  shutdown      down the connection
  util          Connection local percent utilization
```

Step 5 Configure the switch connection using the switch connection configuration commands.

Note Local traffic parameters of an RPM endpoint are in kilobits per second (kbps) while remote traffic parameters are in cells per second (cps). 2824661 cells per second equal 1197656 kilobits per second.

Step 6 Press **Ctrl-Z** to exit configuration mode. Then save your configuration change.

```
Router(config-subif)#^Z
Router#copy run start
Building configuration...
[OK]
```

Step 7 Enter the **show switch connection** command to view the slave endpoint connection. For example,

```
Router#show switch connection
                                                                         Synch
lVpi lVci  NSAP Address                                        rVpi rVci Status

0    2000  default                                             0    0    inSynch:
```

Parameter	Description
lVpi	Represents the local VPI you specified when creating the connection.
lVci	Represents the local VCI you specified when creating the connection.
NSAP Address	Displays *default* as the address, identifying the connection endpoint as a slave endpoint. Master connection endpoints display an ATM address.
rVpi	Represents the remote VPI. A zero (**0**) value designates that it is a slave endpoint.
rVci	Represents the remote VCI. A zero (**0**) value designates that it is a slave endpoint.

Step 8 Enter the **show ip interface brief** command to view the IP interfaces on the RPM-XF.

```
Router#show ip int br
Interface                  IP-Address      OK? Method Status                Protocol
FastEthernet1/1            172.29.52.3     YES manual administratively down down
Switch1                    unassigned      YES NVRAM  up                    up
Switch1.1                  1.1.1.1         YES manual up                    up
```

Step 9 To enable IP communications over the slave connection endpoint, configure the router for IP routing.

The ATM connection acts as an intermediate IP network between the IP routers connected at the master and slave endpoints.

Before you can configure a master endpoint, you must locate and note the ATM address for this interface.

Step 10 To display the ATM address assigned to the slave connection, switch to the active PXM45 card and enter the **dspcon** command to display connection information. For example,

```
Router#cc 7

(session redirected)
```

```
Switch.7.PXM.a > dspcon 9.1.2.2 0 2000
Port                   Vpi Vci                        Owner      State
-------------------------------------------------------------------------
Local  9:-1.1:-1       0.2000                         SLAVE      FAIL
       Address: 47.00918100000000036b5e2bb2.000001074b01.00
Remote Routed          0.0                            MASTER      --
       Address: 00.000000000000000000000000.000000000000.00

-------------------- Provisioning Parameters --------------------
Connection Type: VCC          Cast Type: Point-to-Point
Service Category: UBR         Conformance: UBR.1
Bearer Class: BCOB-X
Last Fail Cause: N/A                              Attempts: 0
Continuity Check: Disabled    Frame Discard: Disabled
L-Utils: 0     R-Utils: 0     Max Cost: 0     Routing Cost: 0
OAM Segment Ep: Enabled

---------- Traffic Parameters ----------
Tx PCR:  353208         Rx PCR:  353208
Tx CDV:  N/A            Rx CDV:  N/A
Tx CTD:  N/A            Rx CTD:  N/A
```

The slave endpoint ATM address appears below the *Local* port identification. This is the address you need to enter when you create a master connection endpoint at either an RPM-XF card or an AXSM card. The connection state is FAIL because the master endpoint has not been created.

Step 11 Copy or write down the slave endpoint ATM address for later use. (See the "Creating a Master Connection on the RPM-XF" section below.)

You are now ready to create the master endpoint on either an RPM-XF card or an AXSM card.

Creating a Master Connection on the RPM-XF

When creating a master connection on an RPM-XF card, that connection endpoint is responsible for routing and rerouting connections. Before you can create the master endpoint, you must create a slave endpoint on either an RPM-XF or AXSM card. The following procedure describes how to create a master endpoint.

If you have not done so already, create a subinterface and PVC to host the master connection endpoint. See the "Creating and Configuring a Switch Subinterface" section earlier in this chapter.

Note The master and slave endpoints can be on the same RPM-XF card as shown in the following example. This example configuration can be used for testing and configuration practice. However, it has no practical application because you can still configure the RPM-XF to route between two Ethernet interfaces. If you do configure master and slave endpoints on the same RPM-XF card, each endpoint must use a different subinterface.

Step 1 Enter subinterface configuration mode. The following example shows how to do this from the user exec mode:

```
Router>enable
Password:
Router#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#interface switch 1.2 point-to-point
Router(config-subif)#
```

Note When configuring the subinterface, you must specify **point-to-point** or **p**, **mu**, **tag** or **mpls**.

Step 2 To create a VCC, enter the **switch connection** command to define the master connection endpoint.

```
Router(config-subif)# switch connection vcc <localVPI> <localVCI> master local raddr
<ATMaddr> <remoteVPI> <remoteVCI>
```

The local VPI and VCI that you enter must match the VPI and VCI you used when you configured the PVC that hosts this connection. The ATM address is the address you copied or wrote down when you created the slave endpoint, and the remote VPI and VCI must match the values set for the slave endpoint. (See the “Creating a Slave Connection on the RPM-XF” section above.)

The following example creates a master connection for the PVC labeled VPI 0, VCI 2000:

```
Router(config-subif)#switch connection vcc 0 2001 master local raddr
47.00918100000000036b5e2bb2.000001074b01.00 0 2000
```

Step 3 To create a VPC, enter the **switch connection** command to define the master connection endpoint.

```
Router(config-subif)# switch connection vpc <localVPI> master local raddr
<ATMaddr> <remoteVPI>
```

The VPI that you enter must match the VPI used to configure the PVC and the atm PVP that hosts this connection.

After you create the master connection endpoint, the RPM-XF enters the switch connection configuration mode and displays the following prompt:

```
Router(config-if-swconn)#
```

Step 4 To display a list of configuration commands, enter a question mark at the switch connection prompt.

```
Router(config-if-swconn)#?
Switch connection configuration commands:
  auto_synch    enable auto synch
  cost          Maximum connection cost
  default       Set a command to its defaults
  exit-swconn   Exit from switch connection configuration mode
  no            Negate a command or set its defaults
  priority      Routing Priority
  reroute       reroute the connection
  rmbs          remote MBS value
  rpcr          remote PCR value
  rscr          remote SCR value
  rutil         Connection remote percent utilization
  shutdown      down the connection
  util          Connection local percent utilization
```

Step 5 Configure the switch connection using the switch connection configuration commands.

Note Local traffic parameters of an RPM endpoint are in kilobits per second (kbps) while remote traffic parameters are in cells per second (cps). (353208 cps equal 149760 kbps.)

Step 6 Press **Ctrl-Z** to exit configuration mode and then save your configuration change.

```
Router(config-subif)#^Z
Router#copy run start
Building configuration...
```

```
[OK]
```

Step 7 Enter the **show switch connection** command to view the master endpoint connection.

```
Router#show switch connection
                                                                      Synch
lVpi lVci  NSAP Address                                     rVpi rVci Status

0    2000  default                                          0    0    inSynch
0    2001  47.0091.8100.0000.0003.6b5e.2bb2.0000.0107.4b01.00  0    2000 inSynch
```

Parameter	Description
lVpi	Represents the local VPI you specified when creating the connection.
lVci	Represents the local VCI you specified when creating the connection.
NSAP Address	Displays *default* as the address, identifying the connection endpoint as a slave endpoint. Master connection endpoints display an ATM address.
rVpi	Represents the remote VPI. A zero (**0**) value designates that it is a slave endpoint.
rVci	Represents the remote VCI. A zero (**0**) value designates that it is a slave endpoint.

In the example above, both master and slave endpoints are on the same RPM-XF card, so both appear in the connection display. If the master and slave endpoints were on different switches, the display would show only an entry for the local endpoint. If the local endpoint is the master endpoint, the slave ATM address is shown.

Step 8 To verify that the new connection is operating properly, switch to the active PXM45 card and enter the **dspcon** command to display connection information.

```
Switch.7.PXM.a > dspcon 9.1.2.2 0 2001
Port                   Vpi Vci                         Owner      State
-------------------------------------------------------------------------
Local  9:-1.1:-1       0.2001                          MASTER     OK
       Address: 47.00918100000000036b5e2bb2.000001074b01.00
Remote 9:-1.1:-1       0.2000                          SLAVE      OK
       Address: 47.00918100000000036b5e2bb2.000001074b01.00

-------------------- Provisioning Parameters --------------------
Connection Type: VCC          Cast Type: Point-to-Point
Service Category: UBR         Conformance: UBR.1
Bearer Class: BCOB-X
Last Fail Cause: No Fail                           Attempts: 0
Continuity Check: Disabled    Frame Discard: Disabled
L-Utils: 100   R-Utils: 100   Max Cost: -1    Routing Cost: 0
OAM Segment Ep: Enabled

---------- Traffic Parameters ----------
Tx PCR:  353208         Rx PCR:  353208
Tx CDV:  N/A            Rx CDV:  N/A
Tx CTD:  N/A            Rx CTD:  N/A
```

Note that the ATM addresses for both ends of the connection are displayed. The connection state is OK. The connection configuration is complete.

Step 9 To view the IP interfaces on the RPM-XF, switch back to the RPM-XF card and enter the **show ip interface brief** command.

```
Switch.7.PXM.a > cc 9

(session redirected)
```

```
Router>enable
Password:
Router#show ip int br
Interface                  IP-Address      OK? Method Status                Protocol
FastEthernet1/1            172.29.52.3     YES manual administratively down down
Switch1                    unassigned      YES NVRAM  up                    up
Switch1.1                  1.1.1.1         YES manual up                    up
Switch1.2                  1.1.2.1         YES manual up                    up
```

Step 10 To validate that a local connection is operating correctly, **ping** the local IP address.

```
Router#ping 1.1.2.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 1.1.2.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
```

To enable IP communications over the master connection endpoint, configure the router for IP routing. The ATM connection will act as an intermediate IP network between the IP routers connected at the master and slave endpoints.

To validate communications to remote devices at the connection endpoints, **ping** devices on those networks. For example, ping a device connected to an RPM-XF interface, or ping an IP address on the ATM end station connected to an AXSM port.

Connection Management

This section describes connection management tasks for the RPM-XF.

Deleting a Connection

To delete a connection, you must delete both ends of the connection. The connection stops working when you delete either end, but you must delete both ends to remove the entire connection configuration.

To delete a connection endpoint on an RPM-XF card, enter the **no** form of the **switch connection** command.

```
(config-if)# no switch connection vcc <localVPI> <localVCI> master <local | remote>
           [raddr <remoteNsapAddress> <remoteVPI> <remoteVCI>]
```

For example:

```
Router(config-subif)# no switch connection vcc 0 2000 master remote
```

To delete a connection endpoint on an AXSM card, refer to the *Cisco MGX 8850 and MGX 8950 Switch Software Configuration Guide,* Chapter 5, “Provisioning AXSM Communication Links,” the “Deleting SPVCs and SPVPs” section.

Modifying Traffic Parameters

The following traffic connection parameters can be modified:

- Service Type
- PCR/SCR/MBS

- Connection Cost
- Channel Utilization
- Encapsulation Type
- Virtual Template ID
- Inarp Timer
- OAM Loopback Frequency
- Enable/Disable OAM Management
- OAM Retry Up Count/Down Count Interval
- Routing Priority

There is a limitation on the changing of the service type. Although the service type can be modified using the Cisco IOS CLI on the RPM-XF, the new service type is not effective in PNNI. PNNI does not support the changing of the service type. The service type cannot be modified via Cisco WAN Manager (CWM).

Changes to the PCR/MCR/SCR, Connection Cost, and Channel Utilization parameters result in connection reroutes, while changes to the remaining parameters result in database updates.

Enter the **switch connection** command to modify an existing connection that terminates on an RPM-XF.

Downing and Upping the Connection

A connection that terminates on an RPM-XF can be manually downed and upped. When a connection is down, its respective PNNI will be derouted and will remain derouted until the connection is upped again by the user. The master endpoint will attempt to reestablish the PNNI when the connection is upped.

Enter the **shutdown** command at the swconn configuration level to down a connection.

```
Router(config-subif-swconn)#shutdown
```

```
Enter the no shutdown command at the swconn configuration level to up a connection.
```

```
Router(config-subif-swconn)#no shutdown
```

Rerouting the Connection

A connection that terminates on an RPM-XF can be manually released and rerouted. Enter the **reroute** command and the respective SPVC will be released and automatically rerouted to the best available path.

Enter the **reroute** command at the swconn configuration level.

```
Router(config-subif-swconn)#reroute
```

Connection Synchronization

RPM-XF connection management data base synchronization refers to the synchronization between the RPM-XF and the PNNI database and is built from the IOS config file. The persistence of connection database on the RPM-XF card across card resets depends upon the user's execution of the **write mem** command on the RPM-XF card. Due to this, there is a possibility that the RPM-XF connection data base is out of synch with the PNNI connection database.

A connection can be in the following synch states:

- **inSynch**—The SPVC parameter values at the RPM-XF card match with the values at the PNNI controller. The connection is programmed in the RPM-XF hardware.
- **mismatch**—There is a mismatch in the SPVC parameter values between the RPM-XF and the PNNI controller's database. However the connection is programmed in the RPM-XF hardware.

  Delete and re-add or simply modify the connection with the correct parameters.
- **onlyOnRpm** —Connection exists only in RPM-XF database but does not exist in the PNNI controller's database on the PXM. The connection is not programmed in the RPM-XF hardware.

  Delete and re-add the connection.
- **notOnRpm**—Connection does not exist in the RPM-XF database but it exists in the PNNI controller's database on the PXM. However the connection is programmed on the RPM-XF hardware.

  Delete and re-add the connection.
- **onlyOnRpm** (NoRsrc)—Connection exists on both the RPM-XF database and on the PNNI controller's database, but cannot be programmed on the RPM-XF hardware because of insufficient resources in the resource partition (for example, number of connections or bandwidth).

  Modify the resource partition for the PNNI controller to adjust the connection-limit or ingress/egress bandwidth.
- **notOnRpm** (NoRsrc)—Connection does not exist on the RPM-XF database, but exists on the PNNI controller's database. Connection can not be programmed on the RPM-XF hardware because of insufficient resources in the RPM-XF partition (for example, VPI/VCI).

  Modify the resource partition for the PNNI controller to adjust the VPI or VCI ranges and then readd the connection.

Manually Resynchronizing Connections

You can manually resynchronize connections. However, out of synchronization conditions may be triggered by

- Periodic kickoffs
- Individual connection provisioning
- RPM reset

You can force resynchronization by entering the **start_resynch** command at the configure interface level.

```
Router# config terminal
Router(config)# interface sw1
Router(config-if)#switch start_resynch
```

Automatically Resynchronizing Connections

The **auto_synch** command corrects mismatches between the PXM and RPM databases. If your network is highly unstable, do not turn on **auto_synch**.

The commands that are used to enable and disable the **auto_synch** feature are moved under the new **switch** command. Here is an example of how you use this command on the config level.

```
Router# conf t
Router(config)# int sw1
Router(config-if)#switch auto_synch on <off|manual> "default is off"
```

Connection State Alarms

This section describes the alarm state of each PNNI, how alarms occur, and what they mean.

Endpoint status indicators reported by RPM-XF and their meanings include:

- **egrAisRdi**—The endpoint is receiving AIS or RDI cells in the egress direction (from the network).
- **oamLpbkFail**—An OAM loopback failure has occurred.
- **mismatch**—There is a mismatch between the RPM-XF and PNNI controller's connection database.
- **conditioned**—There is a routing failure.
- **onlyOnRpm—**The connection exists only on the RPM-XF card.

These alarms are triggered when

- There is a change in the endpoint status.
- A failure is detected by the Connection Manager during a routine routing status check.