MPLS Label Switch Controller

This feature module describes the Cisco MPLS Label Switch Controller (LSC) feature. It includes information about the benefits of the MPLS LSC, supported platforms, configuration examples, and related commands.

Feature Overview

The label switch controller (LSC), combined with the Cisco BPX 8650 IP+ATM switch, delivers scalable integration of IP services over an ATM network.

The LSC enables the BPX 8650 to:

- Participate in a MPLS network
- Directly peer with IP edge routers
- Support the full suite of IP features available in Cisco IOS

The LSC creates MPLS highly scalable IP+ATM integration by using a direct peer relationship between the BPX 8650 and IP edge routers. This direct peer relationship removes the limit placed on the number of IP edge routers (seen in traditional IP-over-ATM networks) allowing service providers to keep pace with the growing demand for IP services. The LSC also supports the quick and direct implementation of advanced IP services over ATM networks with BPX 8650s.

MPLS combines the performance and virtual circuit capabilities of Layer 2 (data link layer) switching with the proven scalability of Layer 3 (network layer) routing to deliver a solution to service providers that meets the challenge of managing explosive growth and providing differentiated services while leveraging their existing infrastructure.

The MPLS architecture provides the flexibility to:

- Run over any combination of Layer 2 technologies
- Support any Layer 3 protocol while scaling beyond today's current solutions.

By deploying MPLS across the Internet or large enterprise networks, customers can:

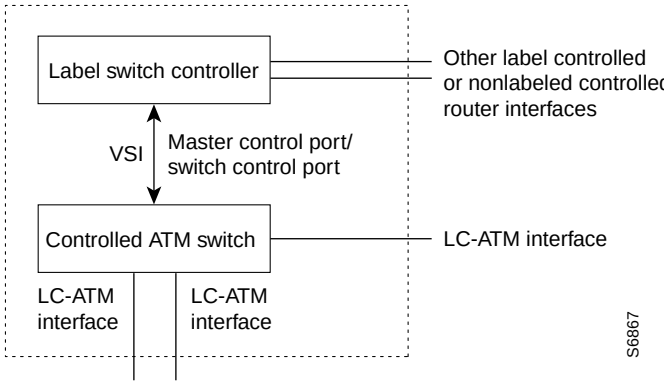
- Save money by using existing ATM and routing infrastructures
- Grow revenue with tag-enabled services
- Increase productivity with enhanced network scalability and performance

Functional Description

The label switch controller (LSC) is a label switch router (LSR) that controls the operation of a separate ATM switch. Together, the router and ATM switch function as a single ATM MPLS router (ATM-LSR). A Cisco 7200 or 7500 series router acts as the LSC, and a Cisco BPX 8600 Service Node or a partner's switch acts as the VSI-controlled ATM switch. The LSC controls the ATM switch using the Cisco Virtual Switch Interface (VSI), which runs over an ATM link connecting the two.

The combination of a LSC and the ATM switch it controls is shown in Figure 1.

Figure 1 Label Switch Controller and Controlled ATM Switch



In Figure 1, the dotted line represents the external interface of the LSC and controlled switch as seen in the IP routing topology. The controlled ATM switch shows one or more LC-ATM interfaces at this external interface and the LSC may include additional interfaces that may or may not be label controlled.

Controlled Switch Ports Represented as Router Interfaces

On the LSC, the LC-ATM ports on the controlled switch are represented as an IOS interface type called extended Label ATM (XTagATM). You associate XTagATM interfaces with particular physical interfaces on the controlled switch through the **extended-port** interface configuration command.

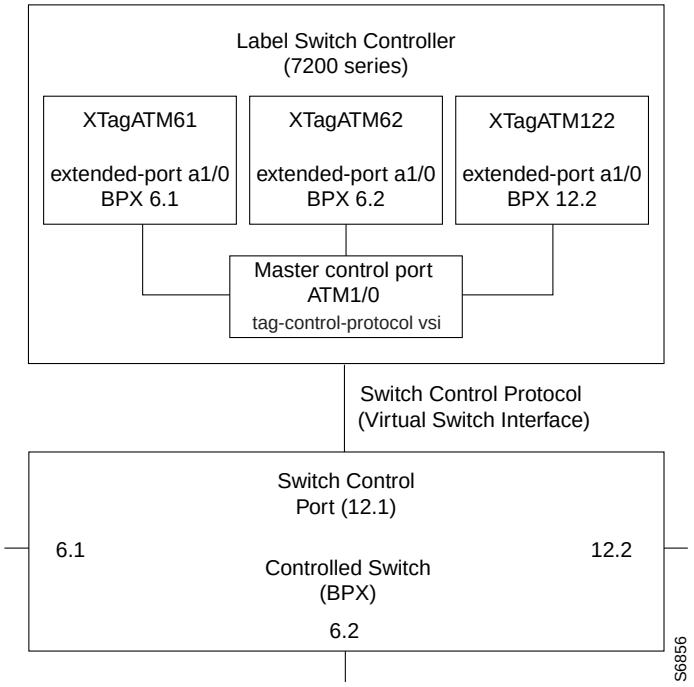
Figure 2 illustrates a configuration in which a LSC controls three ports on a BPX—6.1, 6.2, and 12.2.

These corresponding XTagATM interfaces were created on the LSC and associated with the corresponding ATM ports using the **extended-port** interface configuration command. Note that:

- An additional port on the BPX (12.1) acts as the switch control port
- An ATM interface (ATM1/0) on the LSC acts as the master control port

Figure 2 shows a typical LSC configuration in which the LSC and BPX switch function together as an ATM-LSR.

Figure 2 Typical LSC and BPX Configuration



LSC as Label Edge Device

The LSC can:

- Function simultaneously as a controller for an ATM switch and as a label edge device. Traffic can be forwarded between a router interface and a LC-ATM interface on the controlled switch as well as between two LC-ATM interfaces on the controlled switch.
- Perform the imposition and removal of labels and can serve as the head or tail of a label-switched path tunnel. However, when the LSC acts as a label edge device the LSC is limited by the capabilities of its control link with the switch as follows:
  - Total throughput between all other router interfaces and switch interfaces is limited by the bandwidth of the control link (that is, OC-3, 155 Mbps).
  - Label space for LSC-terminated VCs is limited by the number of VCs supported on the control link.

Support for ATM Forum Protocols

You can connect the LSC to a network running ATM Forum protocols while the LSC simultaneously performs its LSC function. However, you must connect the ATM Forum network through a separate ATM interface (that is, not through the master control port).

Tag Switching/MPLS Terminology

The following table lists old tag switching terms and new MPLS terms used in this document.

Old Designation	New Designation
Tag Switching	MPLS, Multiprotocol Label Switching
Tag (short for Tag Switching)	MPLS
Tag (item or packet)	Label
TDP (Tag Distribution Protocol)	LDP (Label Distribution Protocol) Cisco TDP and LDP (MPLS Label Distribution Protocol) are nearly identical in function, but use incompatible message formats and some different procedures. Cisco is changing from TDP to a fully compliant LDP.
Tag Switched	Label Switched
TFIB (Tag Forwarding Information Base)	LFIB (Label Forwarding Information Base)
TSR (Tag Switching Router)	LSR (Label Switching Router)
TSC (Tag Switch Controller)	LSC (Label Switch Controller)
ATM-TSR	ATM-LSR (ATM Label Switch Router, for example, Cisco BPX 8650 switch.)
TVC (Tag VC, Tag Virtual Circuit)	LVC (Label VC, Label Virtual Circuit)
TSP (Tag Switch Protocol)	LSP (Label Switch Protocol)
XTag ATM (extended Tag ATM port)	XmplsATM (extended MPLS ATM port)

Benefits

IP-ATM Integration

Enables ATM switches, including the Cisco BPX 8650 and 8680 switches to directly support advanced IP services and protocols, thereby reducing operational costs and bandwidth, and decreasing time to market for new services.

Explicit Routing

Provides Layer 2 VCs to gigabit router backbones and integrated IP+ATM environments, including support for explicit routing and provisioning of IP VPN services.

Virtual Private Networks

Supports IP-based VPNs on either a Frame Relay/ATM backbone, integrated IP-ATM backbone, or a gigabit router backbone.

Supported Platforms

- Cisco 7500 series routers—The supported interfaces are the ATM Interface Processor (AIP), Virtual Interface Processor (VIP) and ATM port adapter (PA-A1 and PA-A3).
- Cisco 7200 series routers—The supported interface is the ATM port adapter (PA-A1 and PA-A3).

Supported Standards, MIBs, and RFCs

MIB

No new or modified MIBs are supported by this feature.

RFC

No new or modified RFCs are supported by this feature.

Standards

No new or modified standards are supported by this feature.

Configuration Tasks

This section provides an example of a configuration task for enabling MPLS on a label switch controller (LSC).

Refer to the *Cisco BPX 8600 Series* documentation for the BPX Service Node configuration examples.

Configuring MPLS on a LSC-Controlled BPX Port

Step	Command	Purpose
1	`Router(config)# interface loopback0` `Router(config-if)# ip address 192.103.210.5`	Enable a loopback interface. A loopback interface provides stable router and LDP identifiers.
1	`Router(config)# interface atm1/0` `Router(config-if)# tag-control-protocol vsi`	Enable the VSI protocol on the control interface ATM1/0.
2	`Router(config-if)# interface XTagATM61` `Router(config-if)# extended-port atm1/0 bpx 6.1`	Configure MPLS on the extended label ATM interface by creating an extended label ATM (XTagATM) virtual interface and bind it to BPX port 6.1.

3	`Router(config-if)# ip unnumbered loopback0`
	`Router(config-if)# tag-switching atm vpi 2-5`
	`Router(config-if)# tag-switching ip`
	`Router(config-if)# exit`
		Configure MPLS on the extended label ATM interface.
		The range selected should be limited such that the total number of VPIs does not exceed 4. For example:
		tag-switching atm vpi 2-5
		tag-switching atm vpi 10-13

4	`Router(config)# ip cef switch`	Enable Cisco Express Forwarding (CEF) switching.

Note For Release 12.0(5)T, the XTagATM interfaces must be configured with the **no ip route-cache cef** command.

Verifying LSC Configuration

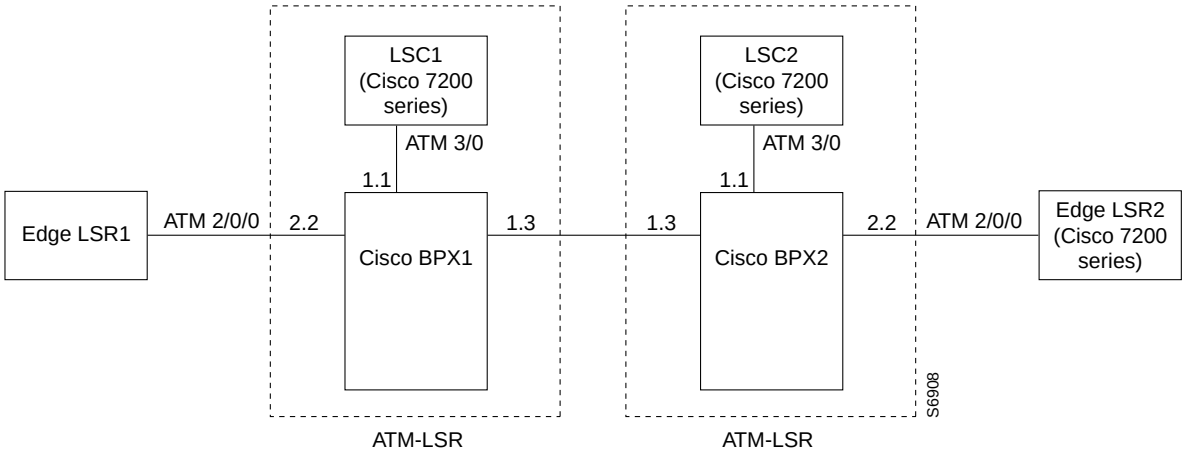
Step	Command	Purpose
1	`Router# show controller vsi session`	Displays the VSI session state.
2	`Router# show tag-switching interfaces`	Displays the MPLS-enabled interface states.
3	`Router# show controllers vsi control-interface`	Displays information about an ATM interface that controls an external ATM switch or VSI control interface.
4	`Router# show interface XTagATM`	Displays information about an extended MPLS ATM interface.

Configuration Example

Configuring ATM-LSRs

In Figure 3, the network topology includes ATM-LSRs in a MPLS network. The network topology provides configurations for two LSCs (Cisco 7200 routers), two BPX service nodes and two edge LSRs (Cisco 7500 routers).

Figure 3 ATM-LSR Network Configuration Example



This section shows examples for the following configurations:

- LSC1
- BPX1 and BPX2
- LSC2
- Edge LSR1
- Edge LSR2

Configuration for LSC1

```
7200 LSC1:
   ip cef switch
   !
   interface loopback0
     ip address 192.103.210.5 255.255.255.255
   !
     interface ATM3/0
     no ip address
     tag-control-protocol vsi
   !
   interface XTagATM13
     extended-port ATM3/0 bpx 1.3
   !
     ip unnumbered loopback0
     tag-switching atm vpi 2-5
     no ip route-cache cef
     tag-switching ip
   !
   interface XTagATM22
     extended-port ATM3/0 bpx 2.2
   !
     ip unnumbered loopback0
     tag-switching atm vpi 2-5
     no ip route-cache cef
     tag-switching ip
   !
```

Configuration for BPX1 and BPX2

```
BPX1 and BPX2:
   uptrk 1.1
   cnfrsrc 1.1 256 0 1 e 0 2000 1 255 0 353000
   uptrk 1.3
   cnfrsrc 1.3 256 0 1 e 0 2000 1 255 0 353000
   uptrk 2.2
   cnfrsrc 2.2 256 0 1 e 0 2000 1 255 0 353000
   addshelf 1.1 v 1 1
```

Configuration for LSC2

```
7200 LSC2:
   ip cef switch
   !
   interface loopback0
     ip address 142.2.143.22 255.255.255.255
   !
   interface ATM3/0
     no ip address
     tag-control-protocol vsi slaves 2
   !
   interface XTagATM13
     extended-port ATM3/0 bpx 1.3
   !
     ip unnumbered loopback0
     tag-switching atm vpi 2-5
     no ip route-cache cef
     tag-switching ip
   !
   interface XTagATM22
     extended-port ATM3/0 bpx 2.2
   !
     ip unnumbered loopback0
     tag-switching atm vpi 2-5
     no ip route-cache cef
```

```
   tag-switching ip
   !
```

Configuration for Edge LSR1

```
7500 LSR1:
   ip cef distributed switch
   !
   interface ATM2/0/0
     no ip address
   !
   interface ATM2/0/0.5 tag-switching
     ip address 142.6.132.2 255.255.0.0
     tag-switching atm vpi 2-5
     tag-switching ip
   !
```

Configuration for Edge LSR2

```
7500 LSR2:
   ip cef distributed switch
   !
   interface ATM2/0/0
     no ip address
   !
   interface ATM2/0/0.9 tag-switching
     ip address 142.2.142.2 255.255.0.0
     tag-switching atm vpi 2-5
     tag-switching ip
```

Configuring Multi-VCs

When configuring Multi-VC support, 4 label VCs to each destination are created by default. These 4 VCs are called standard, available, premium, and control. By default class 0 and class 4 traffic take the standard VC, class 1 and class 5 take the available VC, class 2 and class 6 take the premium VC and class 3 and class 7 take the control VC.

This section shows examples for the following configurations:

- LSC1
- BPX1 and BPX2
- LSC2
- Edge LSR1
- Edge LSR2

Configuration for LSC1

```
7200 LSC1:
   ip cef switch
   !
   interface loopback0
     ip address 192.103.210.5 255.255.255.255
   !
     interface ATM3/0
     no ip address
     tag-control-protocol vsi
   !
   interface XTagATM13
     extended-port ATM3/0 bpx 1.3
```

```
  tag-switching atm cos available 25
  tag-switching atm cos standard 25
  tag-switching atm cos premium 25
  tag-switching atm cos control 25
!
  ip unnumbered loopback0
  tag-switching atm vpi 2-5
  no ip route-cache cef
!
  tag-switching ip
!
interface XTagATM23
  extended-port ATM3/0 bpx 2.2
  tag-switching atm cos available 20
  tag-switching atm cos standard 30
  tag-switching atm cos premium 25
  tag-switching atm cos control 25
!
  ip unnumbered loopback0
  tag-switching atm vpi 2-5
  no ip route-cache cef
!
  tag-switching ip
!
```

Configuration for BPX1 and BPX2

```
BPX1 and BPX2:
   uptrk 1.1
   cnfrsrc 1.1 256 0 1 e 0 2000 1 255 0 353000
   uptrk 1.3
   cnfrsrc 1.3 256 0 1 e 0 2000 1 255 0 353000
   uptrk 2.2
   cnfrsrc 2.2 256 0 1 e 0 2000 1 255 0 353000
   addshelf 1.1 v 1 1
```

Configuration for LSC2

```
7200 LSC2:
   ip cef switch
   !
   interface loopback0
     ip address 142.2.143.22 255.255.255.255
   !
     interface ATM3/0
     no ip address
     tag-control-protocol vsi slaves 2
   !
   interface XTagATM13
     extended-port ATM3/0 bpx 1.3
     tag-switching atm cos available 25
     tag-switching atm cos standard 25
     tag-switching atm cos premium 25
     tag-switching atm cos control 25
   !
     ip unnumbered loopback0
     tag-switching atm vpi 2-5
     no ip route-cache cef
   !
     tag-switching ip
   !
   interface XTagATM22
     extended-port ATM3/0 bpx 2.2
```

```
  tag-switching atm cos available 10
  tag-switching atm cos standard 40
  tag-switching atm cos premium 25
  tag-switching atm cos control 25
!
  ip unnumbered loopback0
  tag-switching atm vpi 2-5
  no ip route-cache cef
!
  tag-switching ip
!
```

Configuration for Edge LSR1

```
7500 LSR1:
  ip cef distributed switch
  !
  interface ATM2/0/0
    no ip address
  !
  interface ATM2/0/0.5 tag-switching
    ip address 142.6.132.2 255.255.0.0
    tag-switching atm vpi 2-5
    tag-switching atm multi-vc
    tag-switching ip
  !
7500 LSR2:
  ip cef distributed switch
  !
  interface ATM2/0/0
    no ip address
  !
  interface ATM2/0/0.9 tag-switching
    ip address 142.2.142.2 255.255.0.0
    tag-switching atm vpi 2-5
    tag-switching atm multi-vc
    tag-switching ip
  !
```

QoS Support

If LSC1 supports QoS and the LSC2 does not support QoS, for example, LSC1 makes VC requests for the following default classes (control=CoS3, standard= CoS1). LSC2 will ignore the call field in the request and allocate two UBR label VCs.

If LSR1 supports QoS and the LSR2 does not support QoS, LSR2 will receive the request to create multiple label VCs, but by default, it creates class 0 only (UBR).

Command Reference

This section documents new commands. All other commands used with this feature are documented in the Cisco IOS Release 12.0 command reference publications.

- extended-port
- interface XTagATM
- show atm vc
- show interface XTagATM
- show controllers XTagATM
- show controllers vsi control-interface
- show controllers vsi descriptor
- show controllers vsi session
- show controllers vsi status
- show controllers vsi traffic
- show tag-switching atm-tdp bindings
- show tag-switching atm-tdp bindwait
- show xtagatm cos-bandwidth-allocation XTagATM
- show xtagatm cross-connect
- show xtagatm vc
- tag-control-protocol vsi
- tag-switching atm control-vc
- tag-switching atm cos
- tag-switching atm vpi
- tag-switching atm vp-tunnel

In Cisco IOS Release 12.0(1)T or later, you can search and filter the output for **show** and **more** commands. This functionality helps you to sort through large amounts of output, or to exclude output that you do not need to see.

To use this functionality, enter a **show** or **more** command followed by the "pipe" character (|), one of the keywords **begin**, **include**, or **exclude**, and an expression that you want to search or filter on:

command **|** {**begin** | **include** | **exclude**} *regular-expression*

Following is an example of the **show atm vc** command in which you want the command output to begin with the first line where the expression "PeakRate" appears:

show atm vc | begin PeakRate

For more information on the search and filter functionality, refer to the Cisco IOS Release 12.0(1)T feature module titled *CLI String Search*.

Command Conventions

boldface font	Commands and keywords are in **boldface**.
italic font	Arguments for which you supply values are in *italics*. In contexts that do not allow *italics*, arguments are enclosed in **angle brackets** < >.
[ ]	Elements in square **brackets** are optional.
{ **x** | **y** | **z** }	Required alternative keywords are grouped in **braces** and separated by vertical bars.
{ **x** | **y** | **z**]	Required alternative keywords are grouped in **brackets** and separated by vertical bars.

extended-port

To associate the currently selected extended MPLS ATM (XTagATM) interface with a particular external interface on the remotely controlled ATM switch, use the **extended-port** interface configuration command.

extended-port *ctrl-if* {**bpx** *bpx-port-number* | **descriptor** *vsi-descriptor* | **vsi** *vsi-port-number*}

Syntax Description

ctrl-if	Identifies the ATM interface used to control the remote ATM switch. You must configure VSI on this interface using the **tag-control-protocol** interface configuration command.
bpx *bpx-port-number*	Specifies the associated BPX interface using the native BPX syntax. *slot.port* [*.virtual port*] You may only use this form of the command when the controlled switch is a BPX.
descriptor *vsi-descriptor*	Specifies the associated port by its VSI physical descriptor. Note that the *vsi-descriptor* string must exactly match the corresponding VSI physical descriptor.
vsi *vsi-port-number*	Specifies the associated port by its VSI logical interface number (integer).

Default

No default behavior or values.

Command Mode

Interface configuration

Command History

Release	Modification
12.0(5)T	This command was introduced.

Usage Guidelines

The **extended-port** interface configuration command associates an XTagATM interface with a particular external interface on the remotely controlled ATM switch. The three alternate forms of the command permit the external interface on the controlled ATM switch to be specified in three different ways.

Example

The following example shows you how to create an extended MPLS ATM interface and bind it to the BPX port 2.3.

```
interface XTagATM0
extended-port atm0/0 bpx 2.3
```

Related Command

Command	Description
interface XTagATM	Enters configuration mode for an extended MPLS ATM (XTagATM) interface.

interface XTagATM

To enter interface configuration mode for the extended MPLS ATM (XTagATM) interface, use the **interface XTagATM** global configuration command. The interface is created the first time this command is issued for a particular interface number.

interface XTagATM *if-num*

Syntax Description

if-num	Specifies the interface number.

Default

No default behavior or values.

Command Mode

Global configuration

Command History

Release	Modification
12.0(5)T	This command was introduced.

Usage Guidelines

Extended MPLS ATM interfaces are virtual interfaces that are created on first reference-like tunnel interfaces. They are similar to ATM interfaces except that they only support LC-ATM encapsulation.

Example

The following example shows you how to create the extended MPLS ATM interface with the interface number 62:

```
(config)# interface XTagATM62
```

Related Command

Command	Description
extended-port	Associates the currently selected extended MPLS ATM (XTagATM) interface with a remotely controlled switch.

show atm vc

To display information about private ATM virtual circuits (VCs), use the **show atm vc** privileged EXEC command.

show atm vc [*vcd*]

Private VCs exist on the control interface of a LSC to support corresponding VCs on an extended MPLS ATM interface.

Syntax Description

vcd	(Optional) Specifies the virtual circuit to display information about.

Default

No default behavior or values.

Command Mode

EXEC

Command History

Release	Modification
12.0(5)T	This command was introduced.

Usage Guidelines

VCs on the extended MPLS ATM interfaces do not appear in the **show atm vc** command output. Instead, the **show xtagatm vc** command provides a similar output which shows information only on extended MPLS ATM VCs.

Examples

In the following example, no VCD is specified and private VCs are present.

```
Router# show atm vc
AAL /         Peak   Avg.  Burst
Interface     VCD   VPI   VCI Type  Encapsulation  Kbps   Kbps  Cells Status
ATM1/0          1     0    40  PVC  AAL5-SNAP          0      0     0 ACTIVE
ATM1/0          2     0    41  PVC  AAL5-SNAP          0      0     0 ACTIVE
ATM1/0          3     0    42  PVC  AAL5-SNAP          0      0     0 ACTIVE
ATM1/0          4     0    43  PVC  AAL5-SNAP          0      0     0 ACTIVE
ATM1/0          5     0    44  PVC  AAL5-SNAP          0      0     0 ACTIVE
ATM1/0         15     1    32  PVC  AAL5-XTAGATM       0      0     0 ACTIVE
ATM1/0         17     1    34  TVC  AAL5-XTAGATM       0      0     0 ACTIVE
ATM1/0         26     1    43  TVC  AAL5-XTAGATM       0      0     0 ACTIVE
ATM1/0         28     1    45  TVC  AAL5-XTAGATM       0      0     0 ACTIVE
ATM1/0         29     1    46  TVC  AAL5-XTAGATM       0      0     0 ACTIVE
ATM1/0         33     1    50  TVC  AAL5-XTAGATM       0      0     0 ACTIVE
```

When you specify a VCD value and the VCD corresponds to that of a private VC on a control interface, the display output appears as follows:

```
Router# show atm vc 15

ATM1/0 33     1    50  TVC  AAL5-XTAGATM       0      0     0 ACTIVE
ATM1/0: VCD: 15, VPI: 1, VCI: 32, etype:0x8, AAL5 - XTAGATM, Flags: 0xD38
PeakRate: 0, Average Rate: 0, Burst Cells: 0, VCmode: 0x0
XTagATM1, VCD: 1, VPI: 0, VCI: 32
OAM DISABLED, InARP DISABLED
InPkts: 38811, OutPkts: 38813, InBytes: 2911240, OutBytes: 2968834
InPRoc: 0, OutPRoc: 0, Broadcasts: 0
InFast: 0, OutFast: 0, InAS: 0, OutAS: 0
OAM F5 cells sent: 0, OAM cells received: 0
Status: ACTIVE
```

Table 1 defines the fields displayed in this example.

Table 1 Show ATM VC Command Field Descriptions

Field	Description
ATM1/0	Interface slot and number.
VCD	Virtual circuit descriptor (virtual circuit number).
VPI	Virtual path identifier.
VCI	Virtual channel identifier.
etype	Encapsulation type.
AAL5 - XTAGATM	Type of ATM adaptation layer (AAL) and encapsulation. A private VC has AAL5 and encapsulation XTAGATM.
Flags	Bit mask describing virtual circuit information. The flag values are summed to result in the displayed value. 0x10000 ABR VC 0x20000 CES VC 0x40000 TVC 0x100 TEMP (automatically created) 0x200 MULTIPOINT 0x400 DEFAULT_RATE 0x800 DEFAULT_BURST 0x10 ACTIVE 0x20 PVC 0x40 SVC 0x0 AAL5-SNAP 0x1 AAL5-NLPID 0x2 AAL5-FRNLPID 0x3 AAL5-MUX 0x4 AAL3/4-SMDS 0x5 QSAAL 0x6 AAL5-ILMI 0x7 AAL5-LANE 0x8 AAL5-XTAGATM 0x9 CES-AAL1 0xA F4-OAM
PeakRate	Number of packets transmitted at the peak rate.
Average Rate	Number of packets transmitted at the average rate.
Burst Cells	Value that, when multiplied by 32, equals the maximum number of ATM cells the virtual circuit can transmit at the peak rate of the virtual circuit.

Field	Description
VCmode	AIP-specific or NPM-specific register describing the usage of the virtual circuit. Contains values such as rate queue, peak rate, and AAL mode, which are also displayed in other fields.
XTAGATM1	Interface of corresponding extended MPLS ATM VC.
VCD	Virtual circuit descriptor (virtual circuit number) of the corresponding extended MPLS ATM VC.
VPI	Virtual path identifier of the corresponding extended MPLS ATM VC.
VCI	Virtual channel identifier of the corresponding extended MPLS ATM VC.
OAM frequency	Seconds between OAM loopback messages or DISABLED if OAM is not in use on this VC.
InARP frequency	Minutes between InARP messages, or DISABLED if InARP is not in use on this VC.
InPkts	Total number of packets received on this virtual circuit. This number includes all silicon-switched, fast-switched, autonomous-switched, and process-switched packets.
OutPkts	Total number of packets sent on this virtual circuit. This number includes all silicon-switched, fast-switched, autonomous-switched, and process-switched packets.
InBytes	Total number of bytes received on this virtual circuit. This number includes all silicon-switched, fast-switched, autonomous-switched, and process-switched packets.
OutBytes	Total number of bytes sent on this virtual circuit. This number includes all silicon-switched, fast-switched, autonomous-switched, and process-switched packets.
InPRoc	Number of process-switched input packets.
OutPRoc	Number of process-switched output packets.
Broadcasts	Number of process-switched broadcast packets.
InFast	Number of fast-switched input packets.
OutFast	Number of fast-switched output packets.
InAS	Number of autonomous-switched or silicon-switched input packets.
OutAS	Number of autonomous-switched or silicon-switched output packets.
OAM F5 cells sent	Number of OAM cells sent on this virtual circuit.
OAM cells received	Number of OAM cells received on this virtual circuit.
Status	Displays the current state of the specified ATM interface.

show interface XTagATM

To display information about an extended MPLS ATM interface, use the **show interface XTagATM** EXEC command.

show interface XTagATM *if-num*

Syntax Description

if-num	Specifies the MPLS ATM interface number.

Default

No default behavior or values.

Command Mode

EXEC

Command History

Release	Modification
12.0(5)T	This command was introduced.

Usage Guidelines

Extended MPLS ATM interfaces are virtual interfaces that are created on first reference like tunnel interfaces. They are similar to ATM interfaces except that they only support LC-ATM encapsulation.

Example

The following is sample output from the **show interface XTagATM** command:

```
Router# show interface XTagATM0

XTagATM0 is up, line protocol is up
  Hardware is Tag-Controlled Switch Port
  Interface is unnumbered.  Using address of Loopback0 (12.0.0.17)
  MTU 4470 bytes, BW 156250 Kbit, DLY 80 usec, rely 255/255, load 1/255
  Encapsulation ATM Tagswitching, loopback not set
  Encapsulation(s): AAL5
  Control interface: ATM1/0, switch port: bpx 10.2
  9 terminating VCs, 16 switch cross-connects
  Switch port traffic:
     129302 cells input, 127559 cells output
  Last input 00:00:04, output never, output hang never
  Last clearing of "show interface" counters never
  Queueing strategy: fifo
  Output queue 0/0, 0 drops; input queue 0/75, 0 drops
  Terminating traffic:
  5 minute input rate 1000 bits/sec, 1 packets/sec
  5 minute output rate 0 bits/sec, 1 packets/sec
     61643 packets input, 4571695 bytes, 0 no buffer
     Received 0 broadcasts, 0 runts, 0 giants
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
     53799 packets output, 4079127 bytes, 0 underruns
```

```
     0 output errors, 0 collisions, 0 interface resets
     0 output buffers copied, 0 interrupts, 0 failures
```

Table 2 defines the significant fields in this display.

Table 2 Show Interface XTagATM Command Field Descriptions

Field	Description
XTagATM0 is up	Interface is currently active.
line protocol is up	Shows line protocol is up.
Hardware is Tag-Controlled Switch Port	Specifies the hardware type.
Interface is unnumbered	Specifies that this is an unnumbered interface.
MTU	Maximum transmission unit of the extended MPLS ATM interface.
BW	Bandwidth of the interface in kilobits per second.
DLY	Delay of the interface in microseconds.
rely	Reliability of the interface as a fraction of 255/ (255/255 is 100% reliability), calculated as an exponential average over 5 minutes.
load	Load on the interface as a fraction of 255 (255/255 is completely saturated), calculated as an exponential average over 5 minutes.
Encapsulation ATM Tagswitching	Encapsulation method.
loopback not set	Indicates that loopback is not set.
Encapsulation(s)	Identifies the ATM adaptation layer.
Control interface	Identifies the control port switch port with which the extended MPLS ATM interface has been associated through the **extended-port** interface configuration command.
9 terminating VCs	Number of terminating VCs with an endpoint on this extended MPLS ATM interface. Packets are transmitted and/or received by the LSC on a terminating VC, or are forwarded between a LSC-controlled switch port and a router interface.
16 switch cross-connects	Number of switch cross-connects on the external switch with an endpoint on the switch port that corresponds to this interface. This includes cross-connects to terminating VCs that carry data to and from the LSC, as well as cross-connects that bypass the LSC and switch cells directly to other ports.
Switch port traffic	Number of cells received and transmitted on all cross-connects associated with this interface.
Terminating traffic counts	Indicates that counters below this line apply only to packets transmitted or received on terminating VCs.
5-minute input rate, 5-minute output rate	Average number of bits and packets transmitted per second in the last 5 minutes.
packets input	Total number of error-free packets received by the system.
bytes	Total number of bytes, including data and MAC encapsulation, in the error-free packets received by the system.
no buffer	Number of received packets discarded because there was no buffer space in the main system. Compare with ignored count. Broadcast storms on Ethernets and bursts on noise on serial lines are often responsible for no input buffer events.

Field	Description
broadcasts	Total number of broadcast or multicast packets received by the interface.
runts	Number of packets that are discarded because they are smaller than the medium's minimum packet size.
giants	Number of packets that are discarded because they exceed the medium's maximum packet size.
input errors	Total number of no buffer, runts, giants, CRCs, frame, overrun, ignored and abort counts. Other input-related errors can also increment the count, so that this sum may not balance with the other counts.
CRC	Cyclic redundancy checksum generated by the originating LAN station or far end device does not match the checksum calculated from the data received. On a LAN, this usually indicates noise or transmission problems on the LAN interface or the LAN bus. A high number of CRCs is usually the result of collisions or a station transmitting bad data. On a serial link, CRCs usually indicate noise, gain hits or other transmission problems on the data link.
frame	Number of packets received incorrectly having a CRC error and a noninteger number of octets.
overrun	Number of times the serial receiver hardware was unable to hand received data to a hardware buffer because the input rate exceeded the receiver's ability to handle the data.
ignored	Number of received packets ignored by the interface because the interface hardware ran low on internal buffers. These buffers are different from the system buffers mentioned previously in the buffer description. Broadcast storms and bursts of noise can cause the ignored count to be incremented.
abort	Illegal sequence of one bits on the interface. This usually indicates a clocking problem between the interface and the data link equipment.
packets output	Total number of messages transmitted by the system.
bytes	Total number of bytes, including data and MAC encapsulation, transmitted by the system.
underruns	Number of times that the transmitter has been running faster than the router can handle. This may never be reported on some interfaces.
output errors	Sum of all errors that prevented the final transmission of datagrams out of the interface being examined. Note that this may not balance with the sum of the enumerated output errors, as some datagrams may have more than one error, and others may have errors that do not fall into any of the specifically tabulated categories.
collisions	Number of messages retransmitted due to an Ethernet collision. This is usually the result of an overextended LAN (Ethernet or transceiver cable too long, more than two repeaters between stations, or too many cascaded multiport transceivers). A packet that collides is counted only one time in output packets.

Field	Description
interface resets	Number of times an interface has been completely reset. Resets occur if packets queued for transmission were not sent within several seconds. On a serial line, this can be caused by a malfunctioning modem that is not supplying the transmit clock signal, or by a cable problem. If the system notices that the carrier detect line of a serial interface is up, but the line protocol is down, it periodically resets the interface in an effort to restart it. Interface resets can also occur when an interface is looped back or shut down.
output buffers copied	Number of packets copied from a MEMD buffer into a system buffer before being placed on the output hold queue.
interrupts	Displays the value of hwidb to tx_restarts.
failures	Number of packets discarded because no MEMD buffer was available.

Related Command

Command	Description
interface XTagATM	Enters configuration mode for an extended MPLS ATM (XTagATM) interface.

show controllers XTagATM

To display information about an extended MPLS ATM interface or, if an interface is not specified, about all extended MPLS ATM interfaces, that are controlled through the VSI protocol, use the **show controllers XTagATM** EXEC command.

show controllers XTagATM *if-num*

Syntax Description

if-num	Specifies the interface number.

Default

No default behavior or values.

Command Mode

EXEC

Command History

Release	Modification
12.0(5)T	This command was introduced.

Usage Guidelines

Per-interface information includes the interface name, the physical descriptor, the interface status, the physical interface state (supplied by the switch), acceptable VPI and VCI ranges, maximum cell rate, available cell rate (forward/backward), and available channels.

Similar information appears if you enter the **show controllers vsi descriptor** command However, you must specify an interface by its (switch-supplied) physical descriptor, instead of its IOS interface name. For the BPX, the physical descriptor has the form:

slot.port.0

Example

In this example, the sample output is from the **show controllers XTagATM** command specifying interface 0.

```
Router# show controllers XTagATM 0

Interface XTagATM0 is up
Hardware is Tag-Controlled ATM Port (on BPX switch BPX-VSI1)
Control interface ATM1/0 is up
Physical descriptor is 10.2.0
Logical interface 0x000A0200 (0.10.2.0)
Oper state ACTIVE, admin state UP
VPI range 1-255, VCI range 32-65535
VPI is not translated at end of link
Tag control VC need not be strictly in VPI/VCI range
Available channels: ingress 30, egress 30
Maximum cell rate: ingress 300000, egress 300000
```

```
Available cell rate: ingress 300000, egress 300000
Endpoints in use: ingress 7, egress 8, ingress/egress 1
Rx cells 134747
rx cells discarded 0, rx header errors 0
rx invalid addresses (per card): 52994
last invalid address 0/32
Tx cells 132564
tx cells discarded: 0
```

Table 3 defines the significant fields in this display.

Table 3 Show Controllers XTagATM Command Field Descriptions

Field	Description
Interface XTagATM is up	Indicates the overall status of the interface. May be “up,” “down,” or “administratively down.”
Hardware is Tag-Controlled ATM Port	Indicates the hardware type. If the XTagATM was successfully associated with a switch port, a description of the form "(on <switch_type> switch <name>)" follows this field, where <switch_type> indicates the type of switch (for example, BPX), and "name" is an identifying string learned from the switch. If the XTagATM interface was not bound to a switch interface (with the **extended-port** interface configuration command), then the label "Not bound to a control interface and switch port" appears. If the interface has been bound, but the target switch interface has not been discovered by the LSC, then the label "Bound to undiscovered switch port (id <number>)" appears, where <number> is the logical interface ID, in hexadecimal notation.
Control interface ATM1/0 is up	Indicates that the XTagATM interface was bound (with the extended-port interface configuration command) to the VSI master whose control interface is ATM1/0 and that this control interface is up.
Physical descriptor is...	A string identifying the interface which was learned from the switch.
Logical interface	This 32-bit quantity, learned from the switch uniquely identifies the interface. It appears in both hexadecimal and dotted quad notation
Oper state	Operational state of the interface, according to the switch. One of: • ACTIVE • FAILED_EXT (that is, in external alarm) • FAILED_INT (indicates the inability of the LSC to communicate with the VSI slave controlling the interface, or another internal failure) • REMOVED (administratively removed on the switch)
admin state	Administrative state of the interface, according to the switch—Up or Down.
VPI range 1 to 255	Indicates the allowable VPI range for the interface which was configured on the switch.
VCI range 32 to 65535	Indicates the allowable VCI range for the interface which was configured on, or determined by the switch.
LSC control VC need not be strictly in VPI or VCI range	Indicates that the label control VC does not need to be within the range specified by VPI range but may be on VPI 0 instead.

Field	Description
Available channels	Indicates the number of channels (endpoints) which are currently free to be used for cross-connects.
Maximum cell rate	Maximum cell rate for the interface, which was configured on the switch.
Available cell rate	Cell rate which is currently available for new cross-connects on the interface.
Endpoints in use	Number of endpoints (channels) in use on the interface, broken down by anticipated traffic flow: • ingress—endpoints carry traffic into the switch • egress— endpoints carry traffic away from the switch • ingress/egress—endpoints carry traffic in both directions
Rx cells	Number of cells received on the interface.
rx cells discarded	Number of cells received on the interface which were discarded due to traffic management actions. rx header errors.
rx header errors	Number of cells received on the interface with cell header errors.
rx invalid addresses (per card)	Number of cells received with invalid addresses (that is, unexpected VPI or VCI.) On the BPX, this counter is maintained per port group (not per interface.)
last invalid address	Address of the last cell received on the interface with an invalid address (for example, 0/32).
Tx cells	Number of cells transmitted out the interface.
tx cells discarded	Number of cells intended for transmission out the interface that were discarded due to traffic management actions.

Related Command

Command	Description
show controllers vsi descriptor	Displays information about a switch interface discovered by the LSC through VSI.

show controllers vsi control-interface

To display information about an ATM interface that is configured with the **tag-control-protocol vsi** EXEC command to control an external switch, or if an interface is not specified, about all VSI control interfaces, use the **show controllers vsi control-interface** command.

show controllers vsi control-interface [*interface*]

Syntax Description

interface	(Optional) Specifies the interface number.

Default

No default behavior or values.

Command Mode

EXEC

Command History

Release	Modification
12.0(5)T	This command was introduced.

Examples

The following is sample output from the **show controllers vsi control-interface** command:

```
Router# show controllers vsi control-interface

Interface:            ATM2/0        Connections:          14
```

The display shows the number of cross-connects currently on the switch that were established by the LSC through VSI over the control interface.

Related Command

Command	Description
tag-control-protocol vsi	Configures the use of VSI on a control port.

show controllers vsi descriptor

To display information about a switch interface discovered by the LSC through VSI, or if no descriptor is specified, about all such discovered interfaces, use the **show controllers vsi descriptor** EXEC command. You specify an interface by its (switch-supplied) physical descriptor.

show controllers vsi descriptor [*descriptor*]

Syntax Description

descriptor	Optional) Physical descriptor. For the BPX, the physical descriptor has the following form: *slot.port.0*

Default

No default behavior or values.

Command Mode

EXEC

Command History

Release	Modification
12.0(5)T	This command was introduced.

Usage Guidelines

Per-interface information includes the interface name, the physical descriptor, the interface status, the physical interface state (supplied by the switch), acceptable VPI and VCI ranges, maximum cell rate, available cell rate (forward/backward), and available channels.

Similar information is displayed when you enter the **show controllers XTagATM** command However, you must specify an IOS interface name instead of a physical descriptor.

Example

The following is sample output from the **show controllers vsi descriptor** command:

```
Router# show controllers vsi descriptor 12.2.0

Phys desc: 12.2.0
Log intf:  0x000C0200 (0.12.2.0)
Interface: XTagATM0
IF status: up                  IFC state: ACTIVE
Min VPI:   1                   Maximum cell rate:  10000
Max VPI:   259                 Available channels: 2000
Min VCI:   32                  Available cell rate (forward):  10000
Max VCI:   65535               Available cell rate (backward): 10000
```

Table 4 defines the significant fields in this display.

Table 4 Show Controllers VSI Descriptor Command Field Description

Field	Description
Phys desc	Physical descriptor. A string learned from the switch which identifies the interface.
Log intf	Logical interface ID. This 32-bit quantity, learned from the switch, uniquely identifies the interface.
Interface	The (IOS) interface name.
IF status	Overall interface status. May be "up," “down," or "administratively down."
Min VPI	Minimum virtual path identifier. Indicates the low end of the VPI range configured on the switch.
Max VPI	Maximum virtual path identifier. Indicates the high end of the VPI range configured on the switch.
Min VCI	Minimum virtual path identifier. Indicates the high end of the VPI range configured on the switch.
Max VCI	Maximum virtual channel identifier. Indicates the high end of the VCI range configured on, or determined by, the switch.
IFC state	Operational state of the interface, according to the switch. One of: • FAILED_EXT (that is, in external alarm) • FAILED_INT (indicates the inability of the LSC to communicate with the VSI slave controlling the interface, or another internal failure) • REMOVED administratively removed on the switch.
Maximum cell rate	Maximum cell rate for the interface, which has been configured on the switch, in cells per second.
Available channels	Indicates the number of channels (endpoints) which are currently free to be used for cross-connects.
Available cell rate (forward)	Cell rate which is currently available in the forward (that is, ingress) direction for new cross-connects on the interface.
Available cell rate (backward)	Cell rate which is currently available in the backward (that is, egress) direction, for new cross-connects on the interface.

Related Command

Command	Description
show controllers XTagATM	Displays information about an extended MPLS ATM interface.

show controllers vsi session

To display information about all sessions with VSI slaves, use the **show controllers vsi session** EXEC command.

show controllers vsi session [*session-num* [interface *interface*]]

Note A session consists of an exchange of VSI messages between the VSI master (the LSC) and a VSI slave (an entity on the switch). There may be multiple VSI slaves for a switch. On the BPX, each port or trunk card assumes the role of a VSI slave.

Syntax Description

session-num	Specifies the session number.
interface *interface*	Specifies the VSI control interface.

Default

No default behavior or values.

Command Mode

EXEC

Command History

Release	Modification
12.0(5)T	This command was introduced.

Usage Guidelines

If a session number and an interface are specified, detailed information on the individual session is presented. If the session number is specified but the interface is omitted, detailed information on all sessions with that number is presented. (Only one session can contain a given number in the first release, as multiple control interfaces are not supported.)

Examples

The following is sample output from the **show controllers vsi session** command:

```
Router# show controllers vsi session

Interface    Session  VCD    VPI/VCI    Switch/Slave Ids   Session State

ATM0/0       0        1      0/40       0/1                ESTABLISHED
ATM0/0       1        2      0/41       0/2                ESTABLISHED
ATM0/0       2        3      0/42       0/3                DISCOVERY
ATM0/0       3        4      0/43       0/4                RESYNC-STARTING
ATM0/0       4        5      0/44       0/5                RESYNC-STOPPING
ATM0/0       5        6      0/45       0/6                RESYNC-UNDERWAY
ATM0/0       6        7      0/46       0/7                UNKNOWN
ATM0/0       7        8      0/47       0/8                UNKNOWN
```

```
ATM0/0       8        9       0/48       0/9               CLOSING
ATM0/0       9        10      0/49       0/10              ESTABLISHED
ATM0/0       10       11      0/50       0/11              ESTABLISHED
ATM0/0       11       12      0/51       0/12              ESTABLISHED
```

Table 5 defines the significant fields in this display.

Table 5 Show Controllers VSI Session Command Field Descriptions

Field	Description
Interface	Control interface name.
Session	Session number (from 0 to <n-1>), where n is the number of sessions on the control interface.
VCD	Virtual circuit descriptor (virtual circuit number). Identifies the VC carrying the VSI protocol between the master and the slave for this session.
VPI/VCI	Virtual path identifier/virtual channel identifier (for the VC used for this session.)
Switch/Slave Ids	Switch and slave identifiers supplied by the switch.
Session State	Indicates the status of the session between the master and the slave. ESTABLISHED is the fully operational steady state; UNKNOWN indicates that the slave is not responding. Other possible states include: CONFIGURING RESYNC_STARTING RESYNC_UNDERWAY RESYNC_ENDING DISCOVERY SHUTDOWN_STARTING SHUTDOWN_ENDING INACTIVE

In this example, session number 9 is specified with the **show controllers vsi session** command:

```
Router# show controllers vsi session 9

Interface:            ATM1/0        Session number:       9
VCD:                  10            VPI/VCI:              0/49
Switch type:          BPX           Switch id:            0
Controller id:        1             Slave id:             10
Keepalive timer:      15            Powerup session id:   0x0000000A
Cfg/act retry timer:  8/8           Active session id:    0x0000000A
Max retries:          10            Ctrl port log intf:   0x000A0100
Trap window:          50            Max/actual cmd wndw:  21/21
Trap filter:          all           Max checksums:        19
Current VSI version:  1             Min/max VSI version:  1/1
Messages sent:        2502          Inter-slave timer:    4.000
Messages received:    2502          Messages outstanding: 0
```

Table 6 defines the significant fields in this display.

Table 6 Show Controllers VSI Session (With Session Number 9 Specified) Command Field Descriptions

Field	Description
Interface	Name of the control interface on which this session is configured.
Session number	A number from 0 to <n-1>, where n is the number of slaves. Configured on the LSC with the **slaves** option of the **tag-control-protocol vsi** command.
VCD	Virtual circuit descriptor (virtual circuit number). Identifies the VC which carries VSI protocol messages for this session.
VPI/VCI	Virtual path identifier or virtual channel identifier, for the VC used for this session.
Switch type	Switch device. For example, the BPX.
Switch id	Switch identifier (supplied by the switch).
Controller id	Controller identifier. Configured on the LSC with the **id** option of the tag-control-protocol vsi command, and also configured on the switch.
Slave id	Slave identifier (supplied by the switch).
Keepalive timer	VSI master keepalive timeout period, in seconds. Configured on the LSC through the **keepalive** option of the **tag-control-protocol-vsi** command.If no valid message is received by the LSC within this period of time, the LSC sends a keepalive message to the slave.
Powerup session id	Session id (supplied by the slave) which it used at powerup time.
cfg/act retry timer	Configured and actual message retry timeout period, in seconds. If no response is received for a command sent by the master within the actual retry timeout period, the message is resent. This applies to most message transmissions. The configured retry timeout value is specified through the **retry** option of the **tag-control-protocol vsi** command. The actual retry timeout value is the larger of the configured value and the minimum retry timeout value permitted by the switch.
Active session id	Session ID for the currently active session (supplied by the slave.)
Max retries	Maximum number of times that a particular command transmission will be retried by the master. That is, a message may be sent up to <max_retiries+1> times. Configured on the LSC through the **retry** option of the **tag-control-protocol vsi** command.
Ctrl port log intf	Logical interface identifier for the control port, as supplied by the switch.
Trap window	Maximum number of outstanding trap messages permitted by the master. This is advertised, but not enforced, by the LSC.
Max/actual cmd wndw	Maximum command window is the maximum number of outstanding (that is, unacknowledged) commands that may be sent by the master before waiting for acknowledgments. This number is communicated to the master by the slave. The command window is the maximum number of outstanding commands that are permitted by the master, before it waits for acknowledgments. This is always less than the maximum command window.

Field	Description
Trap filter	This is always “all” for the LSC, indicating that it wants to receive all traps from the slave. This is communicated to the slave by the master.
Max checksums	Maximum number of checksum blocks supported by the slave. (In this release, the LSC uses only one checksum block.)
Current VSI version	VSI protocol version currently in use by the master for this session. (In the first release, this is always 1.)
Min/max VSI version	Minimum and maximum VSI versions supported by the slave, as last reported by the slave. If both are zero, the slave has not yet responded to the master.
Messages sent	Number of commands sent to the slave.
Inter-Slave timer	Timeout value associated by the slave for messages it sends to other slaves. On a VSI-controlled switch with a distributed slave implementation (such as the BPX), VSI messages may be sent between slaves to complete their processing. For the LSC VSI implementation to function properly, the value of its retry timer is forced to be at least two times the value of the inter-slave timer. (See "Cfg/act retry timer" in this table).
Messages received	Number of responses and traps received by the master from the slave for this session.
Messages outstanding	Current number of outstanding messages (that is, commands sent by the master for which responses have not yet been received.)

Related Command

Command	Description
tag-control-protocol vsi	Configures the use of VSI on a control port.

show controllers vsi status

To display a one-line summary of each VSI-controlled interface, use the **show controllers vsi status** EXEC command.

show controllers vsi status

Syntax Description

This command has no arguments or keywords.

Default

No default behavior or values.

Command Mode

EXEC

Command History

Release	Modification
12.0(5)T	This command was introduced.

Usage Guidelines

If an interface has been discovered by the LSC, but no extended MPLS ATM interface has been associated with it through the **extended-port** interface configuration command, then the interface name is marked <unknown>, and interface status is marked n/a.

Example

The following is sample output from the **show controllers vsi status** command:

```
Router# show controllers vsi status

Interface Name                  IF Status   IFC State  Physical Descriptor
switch control port                   n/a      ACTIVE  12.1.0
XTagATM0                               up      ACTIVE  12.2.0
XTagATM1                               up      ACTIVE  12.3.0
<unknown>                             n/a  FAILED-EXT  12.4.0
```

Table 7 defines the significant fields in this display.

Table 7 Show Controllers VSI Status Command Field Descriptions

Field	Description
Interface Name	The (IOS) interface name.
IF Status	Overall interface status. May be "up," "down," or "administratively down."
IFC State	The operational state of the interface, according to the switch. One of: • FAILED_EXT (that is, in external alarm) • FAILED_INT (indicates the inability of the LSC to communicate with the VSI slave controlling the interface, or another internal failure) • REMOVED (administratively removed on the switch)
Physical Descriptor	A string learned from the switch which identifies the interface.

show controllers vsi traffic

To display traffic information about VSI-controlled interfaces, VSI sessions, or VCs on VSI-controlled interfaces, use the **show controllers vsi traffic** EXEC command.

show controllers vsi traffic [{ **descriptor** *descriptor* **| session** *session-num* **|**
vc [**descriptor** *descriptor* [*vpi vci*]]}]

Syntax Description

descriptor *descriptor*	Specifies the interface.
session *session-num*	Specifies a session number.
vpi	Virtual path identifier.
vpi	Virtual circuit identifier.

Default

No default behavior or values.

Command Mode

EXEC

Command History

Release	Modification
12.0(5)T	This command was introduced.

Usage Guidelines

If none of the optional command parameters is specified, traffic for all interfaces is displayed. You can specify a single interface by its (switch-supplied) physical descriptor. For the BPX, the physical descriptor has the form:

slot.port. 0

If a session number is specified, VSI protocol traffic counts by message type are displayed. The VC traffic display is the same as the one produced by the **show xtagatm vc cross-connect traffic descriptor** command.

Examples

The following is sample output from the **show controllers vsi traffic** command:

```
Router# show controllers vsi traffic

Phys desc: 10.1.0
Interface: switch control port
IF status: n/a
Rx cells: 304250             Rx cells discarded: 0
Tx cells: 361186             Tx cells discarded: 0
Rx header errors: 4294967254 Rx invalid addresses (per card): 80360
```

```
Last invalid address: 0/53


Phys desc: 10.2.0
Interface: XTagATM0
IF status: up
Rx cells: 202637             Rx cells discarded: 0
Tx cells: 194979             Tx cells discarded: 0
Rx header errors: 4294967258 Rx invalid addresses (per card): 80385
Last invalid address: 0/32


Phys desc: 10.3.0
Interface: XTagATM1
IF status: up
Rx cells: 182295             Rx cells discarded: 0
Tx cells: 136369             Tx cells discarded: 0
Rx header errors: 4294967262 Rx invalid addresses (per card): 80372
Last invalid address: 0/32
```

Table 8 defines the significant fields in this display.

Table 8 Show Controllers VSI Traffic Command Field Descriptions

Field	Description
Phys desc	Physical descriptor of the interface.
Interface	The (IOS) interface name.
Rx cells	Number of cells received on the interface.
Tx cells	Number of cells transmitted on the interface.
Tx cells discarded	Number of cells which could not be transmitted on the interface due to traffic management and which were therefore discarded.
Rx header errors	Number of cells which were discarded due to ATM header errors.
Rx cells discarded	Number of cells received on the interface which were discarded due to traffic management.
Rx invalid addresses	Number of cells received with an invalid address (that is, an unexpected VPI/VCI combination). With the BPX, this count is of all such cells received on all interfaces in the port group of this interface.
Last invalid address	Number of cells received on this interface with ATM cell header errors.

The following sample output is displayed when you enter the **show controllers vsi traffic session 9** command:

```
Router# show controllers vsi traffic session 9
                          Sent                                Received
Sw Get Cnfg Cmd:           3656       Sw Get Cnfg Rsp:         3656
Sw Cnfg Trap Rsp:          0          Sw Cnfg Trap:            0
Sw Set Cnfg Cmd:           1          Sw Set Cnfg Rsp:         1
Sw Start Resync Cmd:       1          Sw Start Resync Rsp:     1
Sw End Resync Cmd:         1          Sw End Resync Rsp:       1
Ifc Getmore Cnfg Cmd:      1          Ifc Getmore Cnfg Rsp:    1
Ifc Cnfg Trap Rsp:         4          Ifc Cnfg Trap:           4
Ifc Get Stats Cmd:         8          Ifc Get Stats Rsp:       8
Conn Cmt Cmd:              73         Conn Cmt Rsp:            73
```

```
Conn Del Cmd:            50         Conn Del Rsp:            0
Conn Get Stats Cmd:      0          Conn Get Stats Rsp:      0
Conn Cnfg Trap Rsp:      0          Conn Cnfg Trap:          0
Conn Bulk Clr Stats Cmd: 0          Conn Bulk Clr Stats Rsp: 0
Gen Err Rsp:             0          Gen Err Rsp:             0
unused:                  0          unused:                  0
unknown:                 0          unknown:                 0
TOTAL:                   3795       TOTAL:                   3795
```

Table 9 defines the significant fields in this display.

Table 9 Show Controllers VSI Traffic Session Command Field Descriptions

Field	Description
Sw Get Cnfg Cmd	Number of VSI "get switch configuration command" messages sent.
Sw Cnfg Trap Rsp	Number of VSI switch configuration asynchronous trap response messages sent.
Sw Set Cnfg Cmd	Number of VSI "set switch configuration command" messages sent.
Sw Start Resync Cmd	Number of VSI "set resynchronization start command" messages sent.
Sw End Resync Cmd	Number of VSI "set resynchronization end command" messages sent.
Ifc Getmore Cnfg Cmd	Number of VSI "get more interfaces configuration command" messages sent.
Ifc Cnfg Trap Rsp	Number of VSI "interface configuration asynchronous trap response" messages sent.
Ifc Get Stats Cmd	Number of VSI "get interface statistics command" messages sent.
Conn Cmt Cmd	Number of VSI "set connection committed command" messages sent.
Conn Del Cmd	Number of VSI "delete connection command" messages sent.
Conn Get Stats Cmd	Number of VSI "get connection statistics command" messages sent.
Conn Cnfg Trap Rsp	Number of VSI "connection configuration asynchronous trap response" messages sent.
Conn Bulk Clr Stats Cmd	Number of VSI "bulk clear connection statistics command" messages sent.
Gen Err Rsp	Number of VSI "generic error response" messages sent or received.
Sw Get Cnfg Rsp	Number of VSI "get connection configuration command response" messages received.
Sw Cnfg Trap	Number of VSI "switch configuration asynchronous trap" messages received.
Sw Set Cnfg Rsp	Number of VSI "set switch configuration response" messages received.
Sw Start Resync Rsp	Number of VSI "set resynchronization start response" messages received.
Sw End Resync Rsp	Number of VSI "set resynchronization end response" messages received.
Ifc Getmore Cnfg Rsp	Number of VSI "get more interfaces configuration response" messages received.

Field	Description
Ifc Cnfg Trap	Number of VSI "interface configuration asynchronous trap" messages received.
Ifc Get Stats Rsp	Number of VSI "get interface statistics response" messages received.
Conn Cmt Rsp	Number of VSI "set connection committed response" messages received.
Conn Del Rsp	Number of VSI "delete connection response" messages received.
Conn Get Stats Rsp	Number of VSI "get connection statistics response" messages received.
Conn Cnfg Trap	Number of VSI "connection configuration asynchronous trap" messages received.
Conn Bulk Clr Stats Rsp	Number of VSI "bulk clear connection statistics response" messages received.
unused, unknown	"Unused" messages are those whose function codes are recognized as being part of the VSI protocol, but which are not used by the LSC, and consequently are not expected to be received or sent. "Unknown" messages have function codes which the LSC does not recognize as part of the VSI protocol.
TOTAL	Total number of VSI messages sent or received.

show tag-switching atm-tdp bindings

To display the requested entries from the ATM LDP label bindings database, use the **show tag-switching atm-tdp bindings** EXEC command.

show tag-switching atm-tdp bindings [*A.B.C.D* {*mask* | *length*}]
[**local-tag** | **remote-tag** *vpi vci*] [**neighbor atm** *slot/subslot/port*] [**remote-tag** *vpi vci*]

Syntax Description

A.B.C.D	Destination of prefix.
mask	Destination netmask prefix.
length	Netmask length, in the range of 1 to 32.
local-tag *vpi vci*	Matches locally assigned label values.
neighbor atm *slot/subslot/port*	Matches labels assigned by a neighbor on the specified ATM interface.
remote-tag *vpi vci*	Matches remotely assigned label values.

Default

Displays all database entries.

Command Mode

EXEC

Command History

Release	Modification
12.0(5)T	This command was introduced.

Usage Guidelines

The display output can show the entire database or a subset of entries based on the prefix, the VC label value, or an assigning interface.

Example

The following example shows a sample display using this command.

```
Switch#  show tag atm-tdp bindings
Destination: 13.13.13.6/32
 Headend Router ATM1/0.1 (2 hops) 1/33 Active, VCD=8, CoS=available
 Headend Router ATM1/0.1 (2 hops) 1/34 Active, VCD=9, CoS=standard
 Headend Router ATM1/0.1 (2 hops) 1/35 Active, VCD=10, CoS=premium
 Headend Router ATM1/0.1 (2 hops) 1/36 Active, VCD=11, CoS=control

Destination: 102.0.0.0/8
 Headend Router ATM1/0.1 (1 hop) 1/37 Active, VCD=4, CoS=available
 Headend Router ATM1/0.1 (1 hop) 1/34 Active, VCD=5, CoS=standard
 Headend Router ATM1/0.1 (1 hop) 1/35 Active, VCD=6, CoS=premium
 Headend Router ATM1/0.1 (1 hop) 1/36 Active, VCD=7, CoS=control

Destination: 13.0.0.18/32
 Tailend Router ATM1/0.1 1/33 Active, VCD=8
```

Table 10 describes each of the fields displayed when you use this command.

Table 10 show tag-switching atm-tdp bindings Field Descriptions

Field	Description
Destination:	Destination IP address/length of netmask.
Headend Router	VC type: • Headend—VC that originates at this router • Tailend—VC that terminates at this router
ATM1/0.1	ATM interface.
1/33	VPI/VCI
Active	LVC state: • Active—Set up and working • Bindwait—Waiting for response

Related Command

Command	Description
show tag-switching atm-tdp bindwait	Displays the number of bindings waiting for label assignments for a remote MPLS ATM switch.

show tag-switching atm-tdp bindwait

To display the number of bindings waiting for label assignments from a remote MPLS ATM switch, use the **show tag-switching atm-tdp bindwait** EXEC command.

show tag-switching atm-tdp bindwait

Syntax Description

This command has no keywords or arguments.

Default

No default behavior or values.

Command Mode

EXEC

Command History

Release	Modification
12.0(5)T	This command was introduced.

Examples

The following example shows a sample display using this command:

```
Router# show tag-switching atm-tdp bindwait
```

Related Command

Command	Description
show tag-switching atm-tdp bindings	Displays requested entries from the ATM LDP label binding database.

show xtagatm cos-bandwidth-allocation XTagATM

To display information about CoS bandwidth allocation on extended MPLS ATM interfaces, use the **show xtagatm cos-bandwidth-allocation XTagATM** EXEC command.

show xtagatm, cos-bandwidth-allocation XTagATM [**XTagATM** *interface number***]**

Syntax Description

XTagATM *interface number*	Specifies the XTagATM interface number.

Default

Available 50%, control 50%

Command Mode

EXEC

Command History

Release	Modification
12.0(5)T	This command was introduced.

Usage Guidelines

Use this command to display CoS bandwidth allocation information for various categories including: available, standard, premium, and control.

Example

The following example shows output from this command:

```
Router# show xtagatm cos-bandwidth-allocation XTagATM 123

CoS           Bandwidth allocation
available             25%
standard              25%
premium               25%
control               25%
```

show xtagatm cross-connect

To display information about the LSC's view of the cross-connect table on the remotely controlled ATM switch, use the **show xtagatm cross-connect** EXEC command.

show xtagatm cross-connect [traffic] [{**interface** *interface* [*vpi vci*] |
descriptor *descriptor* [*vpi vci*]]

Syntax Description

traffic	Displays receive and transmit cell counts for each connection.
interface *interface*	Displays only connections with an endpoint of the specified interface.
vpi vci	Displays only detailed information on the endpoint with the specified VPI/VCI on the specified interface.
descriptor *descriptor*	Displays only connections with an endpoint on the interface with the specified physical descriptor.

Default

No default behavior or values.

Command Mode

EXEC

Command History

Release	Modification
12.0(5)T	This command was introduced.

Examples

Each connection is listed twice in the sample output from the **show xtagatm vc cross-connect** command under each interface that is linked by the connection. Connections are marked as "->" (unidirectional traffic flow, into the first interface), "<-" (unidirectional traffic flow, away from the interface) or "<->" (bidirectional).

The following is sample output from the **show xtagatm cross-connect** command:

```
Router# show xtagatm cross-connect

Phys Desc    VPI/VCI     Type   X-Phys Desc  X-VPI/VCI   State

10.1.0       1/37        ->     10.3.0       1/35        UP
10.1.0       1/34        ->     10.3.0       1/33        UP
10.1.0       1/33        <->    10.2.0       0/32        UP
10.1.0       1/32        <->    10.3.0       0/32        UP
10.1.0       1/35        <-     10.3.0       1/34        UP
10.2.0       1/57        ->     10.3.0       1/49        UP
10.2.0       1/53        ->     10.3.0       1/47        UP
10.2.0       1/48        <-     10.1.0       1/50        UP
```

```
10.2.0       0/32        <->    10.1.0       1/33        UP
10.3.0       1/34        ->     10.1.0       1/35        UP
10.3.0       1/49        <-     10.2.0       1/57        UP
10.3.0       1/47        <-     10.2.0       1/53        UP
10.3.0       1/37        <-     10.1.0       1/38        UP
10.3.0       1/35        <-     10.1.0       1/37        UP
10.3.0       1/33        <-     10.1.0       1/34        UP
10.3.0       0/32        <->    10.1.0       1/32        UP
```

Table 11 defines the significant fields in this display.

Table 11 Show XTagATM Cross-Connect Command Field Descriptions

Field	Description
Phys desc	Physical descriptor. A switch-supplied string identifying the interface on which the endpoint exists.
VPI/VCI	Virtual path identifier and virtual channel identifier for this endpoint.
Type	"->" indicates an ingress endpoint, where traffic is only expected to be received into the switch; "<-" indicates an egress endpoint, where traffic is only expected to be transmitted out the interface; "<->" indicates that traffic is expected to be both transmitted and received at this endpoint.
X-Phys desc	Physical descriptor for the interface of the other endpoint belonging to the cross-connect.
X-VPI/VCI	Virtual path identifier and virtual channel identifier of the other endpoint belonging to the cross-connect.
State	Indicates the status of the cross-connect to which this endpoint belongs. Is typically "UP;" other values, all of which are transient, include: DOWN ABOUT_TO_DOWN ABOUT_TO_CONNECT CONNECTING ABOUT_TO_RECONNECT RECONNECTING ABOUT_TO_RESYNC RESYNCING NEED_RESYNC_RETRY ABOUT_TO_RESYNC_RETRY RETRYING_RESYNC ABOUT_TO_DISCONNECT DISCONNECTING

A sample of the detailed information provided for a single endpoint is:

```
Router# show xtagatm cross-connect descriptor 12.1.0 1 42

Phys desc:   12.1.0
Interface:   n/a
Intf type:   switch control port
VPI/VCI:     1/42
```

```
X-Phys desc: 12.2.0
X-Interface: XTagATM0
X-Intf type: extended tag ATM
X-VPI/VCI:   2/38
Conn-state:  UP
Conn-type:   input/output
Cast-type:   point-to-point
Rx service type:   Tag COS 0
Rx cell rate:      n/a
Rx peak cell rate: 10000
Tx service type:   Tag COS 0
Tx cell rate:      n/a
Tx peak cell rate: 10000
```

Table 12 defines the significant fields in this display.

Table 12 Show XTagATM Cross-Connect Descriptor Field Descriptions

Field	Description
Phys desc	Physical descriptor. A switch-supplied string identifying the interface on which the endpoint exists.
Interface	The (IOS) interface name.
Intf type	Interface type. Either extended MPLS ATM or switch control port.
VPI/VCI	Virtual path identifier and virtual channel identifier for this endpoint.
X-Phys desc	Physical descriptor for the interface of the other endpoint belonging to the cross-connect.
X-Interface	The (IOS) name for the interface of the other endpoint belonging to the cross-connect.
X-Intf type	Interface type for the interface of the other endpoint belonging to the cross-connect.
X-VPI/VCI	Virtual path identifier and virtual channel identifier of the other endpoint belonging to the cross-connect.
Conn-state	Indicates the status of the cross-connect to which this endpoint belongs. Is typically UP; other values, all of which are transient, include: DOWN ABOUT_TO_DOWN ABOUT_TO_CONNECT CONNECTING ABOUT_TO_RECONNECT RECONNECTING ABOUT_TO_RESYNC RESYNCING NEED_RESYNC_RETRY ABOUT_TO_RESYNC_RETRY RETRYING_RESYNC ABOUT_TO_DISCONNECT DISCONNECTING

Field	Description
Conn-type	input–Indicates an ingress endpoint where traffic is only expected to be received into the switch output–Indicates an egress endpoint, where traffic is only expected to be transmitted out the interface input/output–Indicates that traffic is expected to be both transmitted and received at this endpoint
Cast-type	Indicates whether or not the cross-connect is multicast. In the first release, this is always point-to-point.
Rx service type	Class of service type for the receive, or ingress, direction. This is MPLS COS <n>, (MPLS Class of Service <n>), where n is in the range 0-7, for input and input/output endpoints; this will be n/a for output endpoints. (In the first release, this is either 0 or 7).
Rx cell rate	(guaranteed) cell rate in the receive, or ingress, direction. In the first release, this is always n/a.
Rx peak cell rate	Peak cell rate in the receive, or ingress, direction, in cells per second. This is n/a for an output endpoint.
Tx service type	Class of service type for the transmit, or egress, direction. This is MPLS COS <n>, (MPLS Class of Service <n>), where n is in the range 0-7, for output and input/output endpoints; this will be n/a for input endpoints. (In the first release, n will be either 0 or 7).
Tx cell rate	(guaranteed) cell rate in the transmit, or egress, direction. In the first release, this is always n/a.
Tx peak cell rate	Peak cell rate in the transmit, or egress, direction, in cells per second. This is n/a for an input endpoint.

show xtagatm vc

To display information about terminating VCs on extended MPLS ATM (XTagATM) interfaces, use the **show xtagatm vc** EXEC command.

show xtagatm vc [*vcd* [*interface*]]

Syntax Description

vcd	(Optional) Virtual circuit descriptor (virtual circuit number). If you specify the *vcd* argument, then detailed information about all VCs with that *vcd* appears. If you do not specify the *vcd* argument, then a summary description of all VCs on all XTagATM interfaces appears.
interface	(Optional) Interface number. If you specify the *interface* and the *vcd* arguments, then the single VC with the specified *vcd* on the specified *interface* is selected.

Default

No default behavior or values.

Command Mode

EXEC

Command History

Release	Modification
12.0(5)T	This command was introduced.

Usage Guidelines

The columns marked VCD, VPI and VCI display information for the corresponding private VC on the control interface. The private VC connects the XTagATM VC to the external switch. It is termed private because its VPI and VCI are only used for communication between the LSC and the switch, and is different from the VPI and VCI seen on the XTagATM interface and the corresponding switch port.

Example

Each connection is listed twice in the sample output from the **show xtagatm vc cross-connect** command under each interface that is linked by the connection. Connections are marked as input (unidirectional traffic flow, into the interface), output (unidirectional traffic flow, away from the interface) or in/out (bidirectional).

The following is sample output from the **show xtagatm vc** command:

```
Router# show xtagatm vc
                                                AAL / Control Interface
Interface     VCD   VPI   VCI Type  Encapsulation  VCD   VPI   VCI Status
XTagATM0        1     0    32  PVC  AAL5-SNAP        2     0    33 ACTIVE
XTagATM0        2     1    33  TVC  AAL5-MUX         4     0    37 ACTIVE
XTagATM0        3     1    34  TVC  AAL5-MUX         6     0    39 ACTIVE
```

Table 13 defines the significant fields in this display.

Table 13 Show XTagATM vc Command Field Descriptions

Field	Description
VCD	Virtual circuit descriptor (virtual circuit number).
VPI	Virtual path identifier.
VCI	Virtual circuit identifier.
Control Interf. VCD	VCD for the corresponding private VC on the control interface.
Control Interf. VPI	VPI for the corresponding private VC on the control interface.
Control Interf. VCI	VCI for the corresponding private VC on the control interface.
Encapsulation	Displays the type of connection on the interface.
Status	Displays the current state of the specified ATM interface.

Related Commands

Command	Description
show atm vc	Displays information about private ATM VCs.
show xtagatm cross-connect	Displays information about remotely connected ATM switches.

tag-control-protocol vsi

To configure the use of VSI on a particular master control port, use the **tag-control-protocol vsi** interface configuration command. To disable VSI, use the **no** form of this command.

tag-control-protocol vsi [**id** *controller-id*] [**base-vc** *vpi vci*] [**slaves** *slave-count*] [**keepalive** *timeout*] [**retry** *timeout count*]

no tag-control-protocol vsi [**id** *controller-id*] [**base-vc** *vpi vci*] [**slaves** *slave-count*] [**keepalive** *timeout*] [**retry** *timeout count*]

Syntax Description

id *controller-id*	Determines the value of the controller-id field present in the header of each VSI message. The default is 1.
base-vc *vpi vci*	Determines the VPI/VCI value for the channel to the first slave. Together with the slaves value, this determines the VPI/VCI values for the channels to all the slaves, which are *vpi/vci* *vpi/vci+1, and so on.* *vpi/vci+slave_count-1.* The default is 0/40.
slaves *slave-count*	Determines the number of slaves reachable through this master control port. The default is 14 (suitable for the BPX). In the first release, at most twelve sessions will be established with the BPX. The default of 14 will attempt sessions with cards 7 and 8, but such sessions are not used in this release, and is always marked as UNKNOWN.
keepalive *timeout*	Determines the value of the keepalive timer (in seconds). Note that the keepalive timer value should be greater than the value of the *retry_timer* times the *retry_count+1*. The default is 15 seconds.
retry *timeout count*	Determines the value of the message retry timer (in seconds) and the maximum number of retries. The default is 8 seconds, 10 retries.

Default

No default behavior or values.

Command Mode

Interface configuration

Command History

Release	Modification
12.0(5)T	This command was introduced.

Usage Guidelines

The command is only available on interfaces that can serve as a VSI master control port. It is recommended that all options to the **tag-control-protocol** command be entered at once.

Once VSI is active on the control interface (through an earlier **tag-control-protocol vsi** command), re-entering the command may cause all associated XTagATM interfaces to go down and come back up. In particular, re-entering the **tag-control-protocol vsi** command with any of the following options causes VSI to be shut down and re-activated on the control interface:

- **id**
- **base-vc**
- **slaves**

VSI remains continuously active (that is, will not be shut down and re-activated) if **tag-control-protocol vsi** command is re-entered with only one or more of the following options:

- **keepalive**
- **retry**

In either case, re-entering the **tag-control-protocol vsi** command causes the specified options to take on the newly specified values; the other options retain their previous values. To restore default values to all the options, enter the **no tag-control-protocol** command, followed by the **tag-control-protocol vsi** command.

Example

The following example shows you how to configure the VSI driver on the control interface:

```
interface atm 0/0
tag-control-protocol vsi 0 51
```

tag-switching atm control-vc

To configure the VPI and VCI values to be used for the initial link to the MPLS peer, use the **tag-switching atm control-vc** interface configuration command. Use this link to establish the LDP session and to carry non-IP traffic.

tag-switching atm control-vc *vpi vci*
no tag-switching atm control-vc *vpi vci*

Syntax Description

vpi	Virtual path identifier, in the range of 0 to 255.
vci	Virtual circuit identifier, in the range of 1 to 65535.

Default

0/32

Command Mode

Interface configuration

Command History

Release	Modification
12.0(5)T	This command was introduced.

Usage Guidelines

On an extended MPLS ATM (XTagATM) interface, the default VPI range to be used for tagged VCs is the configured VPI range that is learned from the switch. This default range is sufficient for most applications. Use the **tag-switching vpi** command on an XTagATM interface only when it is necessary to override the default.

For the **tag-switching atm vpi** command, the VPI range specified must lie within the range that was configured on the BPX for the corresponding BPX interface.

Example

The following example shows you how to create a MPLS subinterface on a router and how to select VPI 1 and VCI 34 as the control VC.

```
interface atm4/0.1 tag-switching
tag-switching ip
tag-switching atm control-vc 1 34
```

Related Command

Command	Description
tag-switching ip (**interface**)	Enables label switching of IPv4 packets on an interface.

tag-switching atm cos

To change the value of configured bandwidth allocation for CoS, use the **tag-switching atm cos** global configuration command.

tag-switching atm cos [**available** | **standard** | **premium** | **control**] *weight*

Syntax Description

weight	Specifies the total weight for all classes. The range of this value is between 0 and 100.
available	Specifies the weight for the class **available**. This is the lowest class priority.
standard	Specifies the weight for the class **standard**. This is the next lowest class priority.
premium	Specifies the weight for the class **premium**. This is the next highest class priority.
control	Specifies the weight for the class **control**. This is the highest class priority.

Default

Available 50%, control 50%

Command Mode

Global configuration

Command History

Release	Modification
12.0(5)T	This command was introduced.

Example

The following example shows output from this command:

```
tag-switching atm cos
interface XTagATM 0
 ip unnumbered loopback0
 no ip directed-broadcast
 no ip route-cache cef
 extended-port ATM1/0 bpx 10.2
 tag-switching atm cos available 50
 tag-switching atm cos control 50
 tag-switching atm vpi 2-5
 tag-switching ip
```

tag-switching atm vpi

To configure the range of values to use in the VPI field for label VCs, use the **tag-switching atm vpi** interface configuration command. To clear the interface configuration, use the **no** form of this command.

tag-switching atm vpi *vpi* [- *vpi*]
no tag-switching atm vpi *vpi* [- *vpi*]

Syntax Description

vpi	Virtual path identifier, low end of range (1 to 255).
- *vpi*	(Optional) Virtual path identifier, high end of range (1 to 255).

Default

1-1

Command Mode

Interface configuration

Command History

Release	Modification
12.0(5)T	This command was introduced.

Usage Guidelines

To configure ATM MPLS on a router interface (for example, an ATM Interface Processor), you must enable a MPLS subinterface.

Note The **tag-switching atm control-vc and tag-switching atm vpi** subinterface level configuration commands are available on any interface that can support ATM labeling.

Use this command to select an alternate range of VPI values for ATM label assignment on this interface. The two ends of the link negotiate a range defined by the intersection of the range configured at each end.

To configure the VPI range for a label edge router (LER) subinterface connected to another router or to an LSC, the range selected should be limited to 4 VPIs.

Example

The following example shows you how to create a subinterface and how to select a VPI range from VPI 1 to VPI 3:

```
interface atm4/0.1 tag-switching
tag-switching ip
tag-switching atm vpi 1-3
```

Related Command

Command	Description
tag-switching atm control-vc	Configures VPI and VCI values for the initial link to an MPLS peer.

tag-switching atm vp-tunnel

To specify an interface or a subinterface as a VP tunnel, use the **tag-switching atm vp-tunnel** interface configuration command.

tag-switching atm vp-tunnel *vpi*

Syntax Description

vpi	Provides VPI value for the local end of the tunnel.

Default

No default behavior or values.

Command Mode

Interface configuration

Command History

Release	Modification
12.0(5)T	This command was introduced.

Usage Guidelines

The **tag-switching atm vp-tunnel** and **tag-switching atm vpi** commands are mutually exclusive.

This command is available on both extended MPLS ATM interfaces and on LC-ATM subinterfaces of ordinary router ATM interfaces. The command is not available on the 1010, where all subinterfaces are automatically VP tunnels.

On an XTagATM interface, the tunnel/non-tunnel status and the VPI value to be used in case the XTagATM interface is a tunnel, are normally learned from the switch through VSI interface discovery. Therefore, it is not necessary to use the **tag-switching atm vp-tunnel** command on an XTagATM interface in most applications.

Example

The following example shows you how to specify a MPLS subinterface VP tunnel, with a VPI value 4.

```
tag-switching atm vp-tunnel 4
```

Debug Commands

This section documents the new **debug** command related to the MPLS LSC feature.

- **debug tag-switching xtagatm cross-connect**
- **debug tag-switching xtagatm vc**
- **debug tag-switching xtagatm vc**
- **debug tag-switching xtagatm errors**
- **debug tag-switching xtagatm events**
- **debug vsi errors**
- **debug vsi events**
- **debug vsi packets**
- **debug vsi param-groups**

debug tag-switching xtagatm cross-connect

Use the **debug tag-switching xtagatm cross-connect** command to display requests and responses for establishing and removing cross-connects on the controlled ATM switch. The **no** form of this command disables debugging output.

debug tag-switching xtagatm cross-connect
no debug tag-switching xtagatm cross-connect

Syntax Description

This command has no arguments or keywords.

Default

No default behavior or values.

Command History

Release	Modification
12.0(5)T	This command was introduced.

Usage Guidelines

You can use the debug tag-switching xtagatm cross-connect command to monitor requests to establish or remove cross-connects from XTagATM interfaces to the VSI master, and the VSI master's responses to these requests.

Note Use this command with care, because it generates output for each cross-connect operation performed by the Label Switch Controller. In a network configuration with a large number of label virtual circuits (LVCs) the volume of output generated may interface with system timing and the correct operation of other router functions. Use this command only in situations where the LVC setup or teardown rate is low.

Example

The following is an example of the display you see when you enter **debug tag-switching xtagatm cross-connect**:

```
Router# debug tag-switching xtagatm cross-connect

XTagATM: cross-conn request; SETUP, userdata 0x17, userbits 0x1, prec 7
        0xC0100 (Ctl-If) 1/32 <-> 0xC0200 (XTagATM0) 0/32
XTagATM: cross-conn response; DOWN, userdata 0x60CDCB5C, userbits 0x2, result
OK
        0xC0200 1/37 --> 0xC0300 1/37
```

Table 14 defines the significant fields shown in this display.

Table 14 Debug Tag-Switching XTagATM Cross-Connect Command Field Description

Field	Description
XTagATM	Identifies the source of the debug message as an XTagATM interface.
cross-conn	Indicates that the debug message pertains to cross-connect setup or teardown operation.
request	Request from an XTagATM interface to the VSI master to set up or teardown a cross-connect.
response	Response from the VSI master to an XTagATM interface that a cross-connect was set up or removed.
SETUP	The request is for the setup of a cross-connect.
TEARDOWN	The request is for the teardown of a cross-connect.
UP	The cross-connect is established.
DOWN	The cross-connect is not established.
userdata, userbits	Values passed with the request which is returned in the corresponding fields in the matching response.
prec	The precedence for the cross-connect.
result	Indicates the status of the completed request.
0xC0100 (Ctl-If) 1/32	Indicates that one endpoint of the cross-connect is on the interface whose logical interface number is 0xC0100, that this interface is the VSI control interface, that the VPI value at this endpoint is 1, and that the VCI value at this end of the cross-connect is 32.
<->	Indicates that this is a bidirectional cross-connect.
0xC0200 (XTagATM0) 0/32	Indicates that the other endpoint of the cross-connect is on the interface whose logical interface number is 0xC0200, that this interface is associated with XTagATM interface 0, that the VPI value at this endpoint is 0 and that the VCI value at this end of the cross-connect is 32.
->	Indicates that this response pertains to a unidirectional cross-connect.

Related Command

Command	Description
show xtagatm cross-connect	Displays information about remotely connected ATM switches.

debug tag-switching xtagatm vc

Use the **debug tag-switching xtagatm vc** command to display information about events that affect individual XTagATM terminating VCs. The **no** form of this command disables debugging output.

debug tag-switching xtagatm vc
no debug tag-switching xtagatm vc

Syntax Description

This command has no arguments or keywords.

Default

No default behavior or values.

Command History

Release	Modification
12.0(5)T	This command was introduced.

Usage Guidelines

You can use the **debug tag-switching xtagatm vc** command to display detailed information about all events that affect individual XTagATM terminating VCs.

Note Use this command with care, because it results in extensive output when the number of XTagATM VCs being set up or torn down is large, and this output may interfere with system timing and normal operation of other router functions. Use the **debug tag-switching xtagatm vc** command only in situations where the number of XTagATM VCs being created or removed is small.

Example

The following is an example of the display you see when you enter **debug**:

```
Router# debug tag-switching xtagatm vc

XTagATM VC: XTagATM1 18 0/32 (ATM1/0 0 0/0):  Setup,  Down --> UpPend
XTagATM VC: XTagATM1 18 0/32 (ATM1/0 88 1/32):  Complete,  UpPend --> Up
XTagATM VC: XTagATM1 19 1/33 (ATM1/0 0 0/0):  Setup,  Down --> UpPend
XTagATM VC: XTagATM0 43 0/32 (ATM1/0 67 1/84):  Teardown,  Up --> DownPend
```

Table 15 defines the significant fields shown in this display.

Table 15 Debug Tag-Switching XTagATM VC Command Field Description

Field	Description
XTagATM VC	Identifies the source of the debug message as the XTagATM interface terminating VC facility.
XTagATM <ifnum>	Identifies the particular XTagATM interface for the terminating VC.
vcd vpi/vci	Indicates the VCD and VPI and VCI value for the terminating VC.
(ctl-if vcd vpi/vci)	Shows the control interface, and the VCD and VPI and VCI value for the private VC corresponding the to XTagATM vc on the control interface.
Setup, Complete, Teardown	Name of the particular event that has occurred for the indicated VC.
oldstate -> newstate	Indicates the state of the terminating VC before and after the processing the indicated event.

debug tag-switching xtagatm errors

Use the **debug tag-switching xtagatm errors** command to display information about error and abnormal conditions that occur on XTagATM interfaces. The **no** form of this command disables debugging output.

debug tag-switching xtagatm errors
no debug tag-switching xtagatm errors

Syntax Description

This command has no arguments or keywords.

Default

No default behavior or values.

Command History

Release	Modification
12.0(5)T	This command was introduced.

Usage Guidelines

You can use the **debug tag-switching xtagatm errors** command to display information about abnormal conditions and events that occur on XTagATM interfaces.

Example

The following is an example of the display you see when you enter **debug tag-switching xtagatm errors**:

```
Router# debug tag-switching xtagatm errors

XTagATM VC: XTagATM0 1707 2/352 (ATM1/0 1769 3/915): Cross-connect setup
failed NO_RESOURCES
```

This message indicates that an attempt to set up a cross-connect for the a terminating VC on XTagATM0 failed, and that the reason for the failure was a lack of resources on the controlled ATM switch.

debug tag-switching xtagatm events

Use the **debug tag-switching xtagatm events** command to display information about major events that occur on XTagATM interfaces, not including events for specific XTagATM VCs and switch cross-connects. The **no** form of this command disables debugging output.

debug tag-switching xtagatm events
no debug tag-switching xtagatm events

Syntax Description

This command has no arguments or keywords.

Default

No default behavior or values.

Command History

Release	Modification
12.0(5)T	This command was introduced.

Usage Guidelines

You can use the debug tag-switching xtagatm events command to monitor the major events that occur on XTagATM interfaces. The command only monitors events that pertain to XTagATM interfaces as a whole and does not include any events which pertain to individual XTagATM VCs or individual switch cross-connects. The specific events monitored when debug tag-switching xtagatm events is in effect include:

- Receipt of asynchronous notifications sent by the VSI master, through the external ATM API (ExATM API) to an XTagATM interface.
- Resizing of the table that is used to store switch cross-connect information. This table is resized automatically as the number of cross-connects increases.
- Marking of XTagATM VCs as stale when an XTagATM interface goes down to ensure that they are cleaned up before new XTagATM VCs can be created on the interface.

Example

The following is an example of the display you see when you enter **debug tag-switching xtagatm events**:

```
Router# debug tag-switching xtagatm events

XTagATM: desired cross-connect table size set to 256
XTagATM: ExATM API intf event Up, port 0xA0100 (None)
XTagATM: ExATM API intf event Down, port 0xA0100 (None)
XTagATM: marking all VCs stale on XTagATM0
```

Table 16 defines the significant fields shown in this display.

Table 16 Debug Tag-Switching XTagATM Events Command Field Description

Field	Description
XTagATM	Identifies the source of the debug message as the XTagATM as an XTagATM interface.
desired cross-connect table size set to 256	Indicates that the table of cross-connect information has been set to hold 256 entries. A single cross-connect table is shared among all XTagATM interfaces. The cross-connect table is automatically resized as the number of cross-connects increases.
ExATM API	Indicates that the information in the debug output pertains to an asynchronous notification sent by the VSI master to the XTagATM driver.
event Up/Down	Indicates the specific event that was sent by the VSI master to the XTagATM driver.
port 0xA0100 (None)	Indicates that the event pertains to the VSI interface whose logical interface number is 0xA0100, and that this logical interface is not bound (through the extended-port interface configuration command) to any XTagATM interface.
marking all VCs stale on XTagATM0	Indicates that all existing XTagATM VCs on XTagATM0 are marked as stale, and that XTagATM0 remains down until all of these VCs are cleaned up.

debug vsi api

Use the **debug vsi api** command to display information on events associated with the external ATM API interface to the VSI master. The **no** form of this command disables debugging output.

debug vsi api
no debug vsi api

Syntax Description

This command has no arguments or keywords.

Default

No default behavior or values.

Command History

Release	Modification
12.0(5)T	This command was introduced.

Usage Guidelines

You can use **debug vsi api** command to monitor the communication between the VSI master and the XTagATM component about interface changes and cross-connect requests.

Example

The following is an example of the display you see when you enter **debug vsi api**:

```
Router# debug vsi api

VSI_M: vsi_exatm_conn_req: 0x000C0200/1/35 -> 0x000C0100/1/50
       desired state up, status OK
VSI_M: vsi_exatm_conn_resp: 0x000C0200/1/33 -> 0x000C0100/1/49
       curr state up, status OK
```

Table 17 defines the significant fields shown in this display.

Table 17 Debug VSI API Command Field Description

Field	Description
vsi_exatm_conn_req	Indicates that a connect or disconnect request was submitted to the VSI Master.
0x000C0200	The logical interface identifier of the primary endpoint, in hexadecimal form.
1/35	VPI and VCI of the primary endpoint.
->	Indicates that the expected traffic flow is unidirectional (from the primary endpoint to the secondary endpoint). The other value for this field is "<->," which indicates bidirectional traffic flow.
0x000C0100	Logical interface identifier of the secondary endpoint.
1/50	VPI and VCI of the secondary endpoint.

Field	Description
desired state	Up indicates a connect request; Down indicates a disconnect request.
status (in vsi_exatm_conn_req output)	A mnemonic indicating the success or failure of the initial processing of the request. One of: • OK • INVALID_ARGS • NONEXIST_INTF • TIMEOUT • NO_RESOURCES • FAIL OK means only that the request was successfully queued for transmission to the switch; it does not indicate completion of the request.

debug vsi errors

Use the **debug vsi errors** command to display information on errors encountered by the VSI Master. The **no** form of this command disables debugging output.

debug vsi errors [**interface** *interface* [**slave** *number*]]
no debug vsi errors [**interface** *interface* [**slave** *number*]]

Syntax Description

interface *interface*	Specifies the interface number.
slave *number*	Specifies the slave number (beginning with 0).

Default

No default behavior or values.

Command History

Release	Modification
12.0(5)T	This command was introduced.

Usage Guidelines

You can use the **debug vsi errors** command to display information on errors encountered by the VSI master when parsing received messages, as well as information on unexpected conditions encountered by the VSI Master.

If the interface parameter is specified, output is restricted to errors associated with the indicated VSI control interface. If the slave number is specified, output is further restricted to errors associated with the session with the indicated slave.

Note Slave numbers are the same as the session numbers discussed under the **show controllers vsi session** command.

Multiple uses of the form of the command which specifies slave number allows multiple slaves to be debugged immediately. For example, the following commands restrict output to that for errors associated with sessions 0 and 1 on control interface atm2/0 (but for no other sessions).

```
Router# debug vsi errors interface atm2/0 slave 0
Router# debug vsi errors interface atm2/0 slave 1
```

Some errors are not associated with any particular control interface or session. Messages associated with these errors are printed regardless of the **interface** or **slave** options which are in effect.

Example

The following is an example of the display you see when you enter **debug vsi errors**:

```
Router# debug vsi errors

VSI Master: parse error (unexpected param-group contents) in GEN ERROR RSP rcvd on
ATM2/0:0/51 (slave 0)
            errored section is at offset 16, for 2 bytes:
 01.01.00.a0 00.00.00.00 00.12.00.38 00.10.00.34
*00.01*00.69 00.2c.00.00 01.01.00.80 00.00.00.08
 00.00.00.00 00.00.00.00 00.00.00.00 0f.a2.00.0a
 00.01.00.00 00.00.00.00 00.00.00.00 00.00.00.00
 00.00.00.00
```

Table 18 defines the significant fields shown in this display.

Table 18 Debug VSI Errors Command Field Description

Field	Description
parse error	Indicates that an error has been encountered while parsing a message received by the VSI master.
unexpected param-group contents	Indicates the type of parsing error encountered. In this case, a parameter group within the message contained invalid data.
GEN ERROR RSP	A mnemonic for the function code in the header of the errored message.
ATM2/0	The control interface on which the errored message was received.
0/51	VPI or VCI of the VC (on the control interface) on which the errored message was received.
slave	Number of the session on which the errored message was received.
offset <n>	Indicates the number of bytes between the start of the VSI header the start of the errored portion of the message.
<n> bytes	Length of the errored section.
00.01.00.a0 [...]	The entire errored message, as a series of hexadecimal bytes. Note that the errored section is between asterisks (*).

debug vsi events

Use the **debug vsi events** command to display information on events that affect entire sessions as well as events that affect only individual connections. The **no** form of this command disables debugging output.

debug vsi events [**interface** *interface* [**slave** *number*]]
no debug vsi events [**interface** *interface* [**slave** *number*]]

Syntax Description

interface *interface*	Specifies the interface number.
slave *number*	Specifies the slave number (beginning with zero).

Default

No default behavior or values.

Command History

Release	Modification
12.0(5)T	This command was introduced.

Usage Guidelines

You can use the **debug vsi events** command to display information on events associated with the per-session state machines of the VSI master, as well as the per-connection state machines. If the interface parameter is specified, output is restricted to events associated with the indicated VSI control interface. If the slave number is specified, output is further restricted to events associated with the session with the indicated slave.

Note Slave numbers are the same as the session numbers discussed under the **show controllers vsi session** command.

Multiple uses of the form of the command which specifies slave number allows multiple slaves to be debugged at once. For example, the following commands restrict output to that for events associated with sessions 0 and 1 on control interface atm2/0 (but for no other sessions). Output associated with all per-connection events are displayed regardless of the **interface** or **slave** options which are in effect.

```
Router# debug vsi events interface atm2/0 slave 0
Router# debug vsi events interface atm2/0 slave 1
```

Example

The following is an example of the display you see when you enter **debug vsi events**:

```
Router# debug vsi events

VSI Master: conn 0xC0200/1/37->0xC0100/1/51:
            CONNECTING -> UP
VSI Master(session 0 on ATM2/0):
    event CONN_CMT_RSP, state ESTABLISHED -> ESTABLISHED
VSI Master(session 0 on ATM2/0):
    event KEEPALIVE_TIMEOUT, state ESTABLISHED -> ESTABLISHED
VSI Master(session 0 on ATM2/0):
    event SW_GET_CNFG_RSP, state ESTABLISHED -> ESTABLISHED
debug vsi packets
```

Table 19 defines the significant fields shown in this display.

Table 19 Debug VSI Events Command Field Description

Field	Description
conn	Indicates that the event applies to a particular connection.
0xC0200	Logical interface identifier of the primary endpoint, in hexadecimal form.
1/37	VPI or VCI of the primary endpoint.
->	Indicates the expected traffic flow is unidirectional (from the primary endpoint to the secondary endpoint.) The other value for this field is "<->," indicating bidirectional traffic flow.
0xC0100	Logical interface identifier of the secondary endpoint.
1/51	VPI or VCI of the secondary endpoint.
<state1> -> <state2>	• <state1> is a mnemonic for the state of the connection before the event occurred • <state2> represents the state of the connection after the event occurred
session	Indicates the number of the session with which the event is associated.
ATM2/0	Indicates the control interface associated with the session.
event	A mnemonic for the event that has occurred. This includes mnemonics for the function codes of received messages (for example, CONN_CMT_RSP), as well as mnemonics for other sorts of events (for example, KEEPALIVE_TIMEOUT).
state <state1> -> <state2>	Mnemonics for the session states associated with the transition triggered by the event. <state1> is a mnemonic for the state of the session before the event occurred; <state2> is a mnemonic for the state of the session after the event occurred.

debug vsi packets

Use the **debug vsi packets** command to display a one-line summary of each VSI message sent and received by the LSC. The **no** form of this command disables debugging output.

debug vsi packets [**interface** *interface* [**slave** *number*]]
no debug vsi packets [**interface** *interface* [**slave** *number*]]

Syntax Description

interface *interface*	Specifies the interface number.
slave *number*	Specifies the slave number (beginning with zero).

Default

No default behavior or values.

Command History

Release	Modification
12.0(5)T	This command was introduced.

Usage Guidelines

If the interface parameter is specified, output is restricted to messages sent and received on the indicated VSI control interface. If the slave number is specified, output is further restricted to messages sent and received on the session with the indicated slave.

Note Slave numbers are the same as the session numbers discussed under the **show controllers vsi session** command.

Multiple uses of the form of the command which specifies slave number allows multiple slaves to be debugged at once. For example, the following commands restrict output to that for messages received on atm2/0 for sessions 0 and 1, (but for no other sessions).

```
Router# debug vsi packets interface atm2/0 slave 0
Router# debug vsi packets interface atm2/0 slave 1
```

Example

The following is an example of the display you see when you enter **debug vsi packets**:

```
Router# debug vsi packets

VSI master(session 0 on ATM2/0): sent msg SW GET CNFG CMD on 0/51
VSI master(session 0 on ATM2/0): rcvd msg SW GET CNFG RSP on 0/51
VSI master(session 0 on ATM2/0): sent msg SW GET CNFG CMD on 0/51
VSI master(session 0 on ATM2/0): rcvd msg SW GET CNFG RSP on 0/51
```

Table 20 defines the significant fields shown in this display.

Table 20 Debug VSI Packets Command Field Description

Field	Description
session	Session number identifying a particular VSI slave. Numbers begin with zero. See the **show controllers vsi session** command.
ATM2/0	Identifier for the control interface on which the message was sent or received.
sent	Message was sent by the VSI master.
rcvd	Message was received by the VSI master.
msg	A mnemonic for the function code from the message header.
0/51	VPI or VCI of the VC (on the control interface) on which the message was sent or received.

debug vsi param-groups

Use the **debug vsi param-groups** command to display the first 128 bytes of each VSI message sent and received by the LSC (in hexadecimal form). The **no** form of this command disables debugging output.

debug vsi param-groups [**interface** *interface* [**slave** *number*]]
no debug vsi param-groups [**interface** *interface* [**slave** *number*]]

Note **param-groups** stands for parameter groups. A parameter group is a component of a VSI message.

Syntax Description

interface *interface*	Specifies the interface number.
slave *number*	Specifies the slave number (beginning with zero).

Default

No default behavior or values.

Command History

Release	Modification
12.0(5)T	This command was introduced.

Usage Guidelines

This command is most commonly used with the **debug vsi packets** command to monitor incoming and outgoing VSI messages.

If the:

- Interface parameter is specified—Output is restricted to messages sent and received on the indicated VSI control interface
- Slave number is specified—Output is further restricted to messages sent and received on the session with the indicated slave.

Note Slave numbers are the same as the session numbers discussed under the **show controllers vsi session** command.

Multiple uses of the form of the command, which specifies slave number, allows multiple slaves to be debugged at once. For example, the following commands restrict output to that for messages received on atm2/0 for sessions 0 and 1, (but for no other sessions).

```
Router# debug vsi param-groups interface atm2/0 slave 0
Router# debug vsi param-groups interface atm2/0 slave 1
```

Examples

Sample Display

The following is an example of the display you see when you enter **debug vsi param-groups**:

```
Router# debug vsi param-groups

Outgoing VSI msg of 12 bytes (not including encap):
 01.02.00.80 00.00.95.c2 00.00.00.00
Incoming VSI msg of 72 bytes (not including encap):
 01.02.00.81 00.00.95.c2 00.0f.00.3c 00.10.00.08
 00.01.00.00 00.00.00.00 01.00.00.08 00.00.00.09
 00.00.00.09 01.10.00.20 01.01.01.00 0c.08.80.00
 00.01.0f.a0 00.13.00.15 00.0c.01.00 00.00.00.00
 42.50.58.2d 56.53.49.31
Outgoing VSI msg of 12 bytes (not including encap):
 01.02.00.80 00.00.95.c3 00.00.00.00
Incoming VSI msg of 72 bytes (not including encap):
 01.02.00.81 00.00.95.c3 00.0f.00.3c 00.10.00.08
 00.01.00.00 00.00.00.00 01.00.00.08 00.00.00.09
 00.00.00.09 01.10.00.20 01.01.01.00 0c.08.80.00
 00.01.0f.a0 00.13.00.15 00.0c.01.00 00.00.00.00
 42.50.58.2d 56.53.49.31
```

Table 21 defines the significant fields shown in this display.

Table 21 Debug VSI Param-Groups Command Field Description

Field	Description
Outgoing	Message was sent by the VSI master.
Incoming	Message was received by the VSI master.
bytes	Number of bytes in the message, starting at the VSI header, and excluding the link layer encapsulation.
01.02...	Up to the first 128 bytes of the message, in hexadecimal form.

Glossary

The following terms are defined for a MPLS context only, not for general situations.

ATM-LSR—A MPLS router with a number of LC-ATM interfaces. The router forwards the cells from these interfaces using labels carried in the VPI or VCI field.

ATM edge LSR—A MPLS router that is connected to the ATM-LSR cloud through LC-ATM interfaces. The ATM edge LSR adds labels to untagged packets and strips labels from labeled packets.

BPX—Broadband Packet Exchange (BPX). The BPX is a carrier quality switch, with trunk and CPU hot standby redundancy.

BXM—Broadband Switch Module. ATM port card for the BPX switch.

VSI—Virtual Switch Interface (VSI). The protocol that enables a LSC to control an ATM switch over an ATM link.

VSI master—In a hardware context, a device that controls a VSI switch (for example, a VSI Label Switch Controller). In a software context, a process that implements the master side of the VSI protocol.

VSI slave—In a hardware context, a switch or a port card that implements the VSI. In a software context, a process that implements the slave side of the VSI protocol.

extended label ATM interface—A new type of interface supported by the remote ATM switch driver and a particular switch-specific driver that supports MPLS over an ATM interface on a remotely controlled switch.

external ATM interface—One of the interfaces on the controlled ATM switch other than the switch control port. It is also referred to as an exposed ATM interface, because it is available for connections outside of the label controlled switch.

master control port—A physical interface on a LSC that is connected to one end of a slave control link.

remote ATM switch driver—A set of interfaces that allow IOS software to control the operation of a remote ATM switch through a control protocol, such as VSI.

Ships in the Night (SIN)—The ability to support both MPLS procedures and ATM Forum protocols on the same physical interface, or on the same router or switch platform. In this mode, the two protocol stacks operate independently.

controlled ATM switch—An ATM switch that is being controlled by a LSC.

switch control port—An interface that uses a LSC to control the operation of a controlled ATM switch (for example, VSI). The protocol runs on the ATM link.

label controlled switch—The Label Switch Controller and controlled ATM switch that it controls, viewed together as a unit.

Label switch controller (LSC)—An IOS platform that runs the generic MPLS software and is capable of controlling of the operation of an external ATM (or other type of) switch, making the interfaces of the latter appear externally as LC-ATM interfaces.

Label switched path (LSP) tunnel—A configured connection between two routers, using MPLS to carry the packets.

Label switch router (LSR)—A Layer 3 router that forwards packets based on the value of a label encapsulated in the packets.

LC-ATM interface—An MPLS interface where labels are carried in the VPI or VCI bits of ATM cells and where VC connections are established under the control of MPLS control software.

LFIB—Label Forwarding Information Base (LFIB). A data structure and way of managing forwarding in which destinations and incoming labels are associated with outgoing interfaces and labels.

LVC—Label switched controlled virtual circuit (LVC). A virtual circuit (VC) established under the control of MPLS. An LVC is not a PVC or an SVC. It must traverse only a single hop in a tag-switched path (LSP), but may traverse several ATM hops only if it exists within a VP tunnel.