



Features for Any Interface

Use the information in this chapter to understand the types of interfaces supported on Cisco routers and access servers and to locate configuration information for various types of interfaces.

To identify the hardware platform or software image information associated with a feature, use Cisco Feature Navigator on Cisco.com to search for information about the feature.

For a complete description of the interface commands used in this and other chapters that describe interface configuration, refer to the *Cisco IOS Interface and Hardware Component Command Reference*. To locate documentation of other commands that appear in this chapter, use the master commands list or search online.

This chapter contains general information that applies to all interface types; it includes the following sections:

- [Understanding Interface Configuration, page 5](#)
- [Understanding Subinterfaces, page 6](#)
- [Configuring Features Available on Any Interface, page 7](#)
- [Understanding OIR, page 11](#)
- [Understanding Fast Switching Support, page 11](#)
- [Monitoring and Maintaining the Interface, page 12](#)

For examples of configuration commands shown in this chapter, see the “[Interface Configuration Examples](#)” section on page 27.

Understanding Interface Configuration

These general instructions apply to all interface configuration processes. Begin interface configuration in global configuration mode. To configure an interface, follow these steps:

1. Use the **configure** EXEC command at the privileged EXEC prompt to enter global configuration mode.
2. Once in the global configuration mode, start configuring the interface by using the **interface** command. Identify the interface type followed by the number of the connector or interface card. These numbers are assigned at the factory at the time of installation or when cards are added to a system and can be displayed with the **show interfaces** EXEC command. A report is provided for each interface that the device supports, as seen in the following partial sample display.

```
Router# show interfaces
```

```
Serial 0 is administratively down, line protocol is down
Hardware is MCI Serial
MTU 1500 bytes, BW 1544 Kbit, DLY 20000 usec, rely 255/255, load 1/255
Encapsulation HDLC, loopback not set, keepalive set (10 sec)
```

Use the **show hardware EXEC** command to see a list of the system software and hardware.

To begin configuring serial interface 0, add the following line to the configuration file:

```
interface serial 0
```



Note It is not necessary to add a space between the interface type and interface number. For example, in the preceding line you can specify either *serial 0* or *serial0*. The command will work either way.

3. Follow each **interface** command with the interface configuration commands that your particular interface requires. The commands that you use define the protocols and applications that will run on the interface. The commands are collected and applied to the **interface** command until you use another **interface** command, a command that is not an interface configuration command, or you type the Ctrl-Z sequence to get out of configuration mode and return to privileged EXEC mode.
4. Once an interface is configured, you can check its status by using the EXEC **show** commands listed in the tables in the [“Monitoring and Maintaining the Interface” section on page 12](#).



Note Configuring channelized T1 and E1 interfaces requires additional steps. When you configure channelized T1 or channelized E1, you must first define the channels and the time slots that comprise the channels by using the **controller t1** and the **channel-group** controller configuration commands. Then configure the virtual serial interfaces using the **interface serial** command in global configuration mode. Refer to the *Cisco IOS Dial Technologies Configuration Guide* for instructions on configuring channelized E1 or channelized T1 interfaces.

Understanding Subinterfaces

Configuring multiple virtual interfaces, or subinterfaces, on a single physical interface allows greater flexibility and connectivity on the network. A subinterface is a mechanism that allows a single physical interface to support multiple logical interfaces or networks. That is, several logical interfaces or networks can be associated with a single hardware interface. Subinterfaces are implemented in various WAN and LAN protocols, including ATM, Frame Relay, Switched Multimegabit Data Service (SMDS), X.25, and Novell IPX (Internetwork Packet Exchange). For more information about using subinterfaces, refer to the appropriate protocol chapter.

IDB Scalability

Cisco IOS software uses interface descriptor blocks (IDBs) to store interface-specific information, such as protocols configured and timers, so that Cisco IOS device drivers can interact efficiently with various types of interfaces. IDBs are an exhaustible resource tied to the memory available on the router. Each physical interface comprises a hardware IDB and at least one software IDB, although more than one software IDB may be supported and mapped to the same physical interface.

An IDB is used for each of these types of interfaces:

- Physical
- Dialer
- Virtual
- Hidden
- Subinterface
- Tunnel
- Loopback

Hardware IDBs

The hardware IDB contains physical state information about the interface. Hardware IDBs are allocated from the fast memory pool if it exists on the platform. If there is no fast memory pool, they are allocated from process memory.

Software IDBs

The software IDB contains application-specific information for the router. New software IDBs can be allocated after system initialization to create subinterfaces and virtual interfaces such as loopback and tunnel interfaces. Software IDBs are allocated from process memory.

The number of interfaces supported depends on the platform, the cards installed in the device, and auto configuration for platform and memory configuration.

Configuring Features Available on Any Interface

The following sections describe optional tasks that you can perform on any type of interface:

- [Adding a Description for an Interface, page 8](#)
- [Configuring MOP, page 8](#)
- [Controlling Interface Hold-Queue Limits, page 8](#)
- [Setting Bandwidth, page 9](#)
- [Setting Interface Delay, page 9](#)
- [Adjusting Timers, page 9](#)
- [Limiting Transmit Queue Size, page 10](#)
- [Adjusting Maximum Packet Size or MTU Size, page 10](#)

Adding a Description for an Interface

You can add a description about an interface to help you remember what is attached to it. This description is meant solely as a comment to help identify what the interface is being used for. The description will appear in the output of the following commands: **show configuration**, **show system:running-config**, and **show interfaces**. When you add a description for a T1 controller interface, it will appear in the output of the **show controllers t1** and **show system:running-config** commands.

To add a description for any interface except a T1 or E1 controller interface, use the following command in interface configuration mode. To add a description for a T1 or E1 controller in a Cisco 7200 series, or Cisco 7500 series router, use the following command in controller configuration mode.

Command	Purpose
Router(config-if)# description <i>string</i>	Adds a comment to help identify an interface.

For examples of adding interface descriptions, see the section “Interface Description Examples” at the end of this chapter.

Configuring MOP

To enable Maintenance Operation Protocol (MOP) on an interface, use the following command in interface configuration mode.

Command	Purpose
Router(config-if)# mop enabled	Enables MOP.

To enable an interface to send out periodic MOP system identification messages, use the following command in interface configuration mode.

Command	Purpose
Router(config-if)# mop sysid	Enables MOP message support.

Controlling Interface Hold-Queue Limits

Each interface has a hold-queue limit. This limit is the number of data packets that the interface can store in its hold queue before rejecting new packets. When the interface empties one or more packets from the hold queue, it can accept new packets again. To specify the hold-queue limit of an interface, use the following command in interface configuration mode.

Command	Purpose
Router(config-if)# hold-queue <i>length</i> {in out}	Specifies the maximum number of packets allowed in the hold queue.

Setting Bandwidth

Higher-level protocols use bandwidth information to make operating decisions. For example, the Interior Gateway Routing Protocol (IGRP) uses the minimum path bandwidth to determine a routing metric. TCP adjusts initial retransmission parameters on the basis of the apparent bandwidth of the outgoing interface. To set a bandwidth value for an interface, use the following command in interface configuration mode.

Command	Purpose
Router(config-if) # bandwidth <i>kilobits</i>	Sets a bandwidth value.

The bandwidth setting is a routing parameter only; it does not affect the physical interface.

Setting Interface Delay

Higher-level protocols might use delay information to make operating decisions. For example, IGRP can use delay information to differentiate between a satellite link and a land link. To set a delay value for an interface, use the following command in interface configuration mode.

Command	Purpose
Router(config-if) # delay <i>tens-of-microseconds</i>	Sets a delay value for an interface.

Setting the delay value sets an informational parameter only; you cannot adjust the actual delay of an interface using this configuration command.

Adjusting Timers

To adjust the frequency of update messages, use the following command in interface configuration mode.

Command	Purpose
Router(config-if) # keepalive <i>[seconds]</i>	Adjusts the frequency with which the Cisco IOS software sends messages to itself (Ethernet and Token Ring) or to the other end (HDLC-serial and PPP-serial links) to ensure that a network interface is alive for a specified interface.

The interval is adjustable in 1-second increments down to 1 second. An interface is declared down after three update intervals have passed without receiving a keepalive packet.

When adjusting the interval for a very low bandwidth serial interface, large packets can delay the smaller keepalive packets long enough to cause the line protocol to go down. You might need to experiment to determine the best value.

Limiting Transmit Queue Size

You can control the size of the transmit queue available to a specified interface on the Multiport Communications Interface (MCI) and Serial Communication Interface (SCI) cards. To limit the size, use the following command in interface configuration mode.

Command	Purpose
Router(config-if)# tx-queue-limit <i>number</i>	Limits the size of the transmit queue.

Adjusting Maximum Packet Size or MTU Size

Each interface has a default maximum packet size or maximum transmission unit (MTU) size. This number generally defaults to 1500 bytes. On serial interfaces, the MTU size varies, but cannot be set smaller than 64 bytes. To adjust the maximum packet size or MTU size, use the following command in interface configuration mode.

Command	Purpose
Router(config-if)# mtu <i>bytes</i>	Adjusts the maximum packet size or MTU size.



Note

The maximum MTU size that can be configured on the native Gigabit Ethernet ports on the Cisco 7200 series router is 9216. The range of configurable MTU value is from 1500 to 9216.



Caution

Changing an MTU size on a Cisco 7500 series router results in resizing and reassigning buffers and resetting all interfaces. The following message is displayed:

```
%RSP-3-Restart:cbus complex.
```

Using Protocol-Specific Versions of the mtu Command

Changing the MTU value with the **mtu** interface configuration command can affect values for the protocol-specific versions of the command (the **ip mtu** command, for example). If the value specified with the **ip mtu** interface configuration command is the same as the value specified with the **mtu** interface configuration command, and you change the value for the **mtu** interface configuration command, the **ip mtu** value automatically matches the new **mtu** interface configuration command value. However, changing the values for the **ip mtu** configuration commands has no effect on the value for the **mtu** interface configuration command.

Using the mtu Command on ATM and LANE Interfaces

ATM interfaces are not bound by what is configured on the major interface. By default, MTU on a subinterface is equal to the default MTU (4490); if a client is configured, the default is 1500. MTU can be changed on subinterfaces, but it may result in recarving of buffers to accommodate the new maximum MTU on the interface.

Understanding OIR

The online insertion and removal (OIR) feature allows you to remove and replace interface processors while the system is online. You can shut down the interface processor before removal and restart it after insertion without causing other software or interfaces to shut down. This feature is not available on all platforms. Refer to the appropriate platform specifications for details.

**Note**

Do not remove or install more than one interface processor at one time. After a removal or installation, ensure that the router is functioning properly before continuing.

You do not need to notify the software that you are going to remove or install an interface processor. When the Route Processor (RP) is notified by the system that an interface processor has been removed or installed, it stops routing and scans the system for a configuration change. All interface processors are initialized, and each interface type is verified against the system configuration; then the system runs diagnostics on the new interface. There is no apparent disruption to normal operation of the device during interface processor insertion or removal.

Only an interface of a type that has been configured previously will be brought online; others require configuration. If a newly installed interface processor does not match the system configuration, the interface is left in an administratively down state until the system operator configures the system with the new interfaces.

Hardware (MAC-level) addresses for all interfaces on the Cisco 7500 series routers are stored on an EEPROM component in the RP instead of on the individual interface boards. On the Cisco 7500 series routers, an address allocator in the EEPROM contains a sequential block of 40 addresses (5 interface slots times a maximum of 8 possible ports per slot); each address is assigned to a specific slot and port address in the chassis, regardless of how the interfaces are configured. On the Cisco 7200 series, hardware addresses are stored in a midplane EEPROM that supports 1024 addresses per box.

Storage of hardware addresses in EEPROM allows interfaces to be replaced online without requiring the system to update switching tables and data structures. Regardless of the types of interfaces installed, the hardware addresses do not change unless you replace the system RP. If you do replace the RP, the hardware addresses of *all* ports change to those specified in the address allocator on the new RP.

Understanding Fast Switching Support

Switching is the process by which packets are forwarded. The Cisco IOS software supports multiple methods of switching. Cisco routers fast-switch Layer 2 Forwarding (L2F) traffic. In stack group environments in which some L2F traffic is offloaded to a powerful router, fast switching provides improved scalability.

For information about switching features, refer to the *Cisco IOS IP Switching Configuration Guide*. For documentation of commands used to configure switching features, refer to the *Cisco IOS IP Switching Command Reference*.

Monitoring and Maintaining the Interface

To monitor and maintain the interfaces, you can perform the tasks in the following sections:

- [Monitoring Interface and Controller Status, page 12](#)
- [Monitoring the T1 or E1 Controller, page 14](#)
- [Monitoring and Maintaining CSU/DSU Service Modules, page 14](#)
- [Monitoring and Maintaining a Hub, page 17](#)
- [Clearing and Resetting the Interface, page 18](#)
- [Shutting Down and Restarting an Interface, page 19](#)
- [Configuring Interface Index Persistence, page 20](#)
- [Configuring Loopback Detection, page 23](#)
- [Running Interface Loopback Diagnostics, page 23](#)
- [Enabling Loopback Testing of Fractional T1/T1, page 25](#)
- [Reloading a Cisco 7500 Single Line Card, page 25](#)

Monitoring Interface and Controller Status

Cisco IOS software contains commands that you can use at the privileged EXEC or user EXEC prompt to display information about an interface including the version of the software and the hardware, the controller status, and statistics about the interfaces. The following table lists some of the interface monitoring commands. (You can display the full list of **show** commands by using the **show ?** command at the privileged EXEC or user EXEC prompt.) These commands are fully described in the *Cisco IOS Interface and Hardware Component Command Reference*.

To display information about an interface, use the following commands in privileged EXEC or user EXEC mode, as indicated.

Command	Mode	Purpose
Router# show async status show interfaces async	Privileged EXEC	Displays the status of the asynchronous interface.
Router> show compress	User EXEC	Displays compression statistics on a serial interface.
Router# show controllers [bri cbus fddi lance mci serial token]	Privileged EXEC	Displays current internal status information for the interface controller cards.
Router# show controllers cbus	Privileged EXEC	Displays information about the Switch Processor (SP) controller on the Cisco 7500 series routers.
Router> show controllers [e1 ethernet fastethernet gigabitethernet fddi serial t1 token]	User EXEC	Displays current internal status information for the interface controller cards.
Router> show controllers [ethernet fastethernet gigabitethernet fddi serial token]	User EXEC	Displays current internal status information for the interface controller cards on the Cisco 7200 series and Cisco 7500 series routers.

Command	Mode	Purpose
Router# show derived-config [interface type number]	Privileged EXEC	Displays the composite results of all the configuration commands that apply to an interface, including commands that come from sources such as static templates, dynamic templates, dialer interfaces, and AAA per-user attributes.
Router# show diagbus [slot]	Privileged EXEC	Displays diagnostic information about the controller, interface processor, and port adapters associated with a specified slot of a Cisco 7200 series or Cisco 7500 series router.
Router# show interfaces [type number] [first] [last] [accounting]	Privileged EXEC	If accounting is configured, displays the number of packets of each protocol type that have been sent through the interface.
Router# show interfaces [type slot/port] [accounting]		For Cisco 7500 series routers with a Packet over SONET Interface Processor.
Router# show interfaces [type slot/port-adapter/port] [accounting]		For Cisco 7500 series routers with VIP or VIP2 cards.
Router# show interfaces ctunnel interface-number	Privileged EXEC	Displays information about an IP over CLNS tunnel.
Router> show interfaces pos [slot/port]	User EXEC	Displays information about Cisco 7500 series with a Packet over SONET Interface Processor.
Router# show interfaces async [number] [accounting]	Privileged EXEC	Displays the number of packets of each protocol type that have been sent through the asynchronous serial line.
Router# show system:running-config	Privileged EXEC	Displays the currently running configuration in RAM.
Router# show rif	Privileged EXEC	Displays the current contents of the Routing Information Field (RIF) cache.
Router# show protocols	Privileged EXEC	Displays the global (system-wide) and interface-specific status of any configured Level 3 protocol.
Router# show version	Privileged EXEC	Displays the hardware configuration, software version, the names and sources of configuration files, and the boot images.

Monitoring the T1 or E1 Controller

This section applies to channelized T1 or E1 interfaces. Because the T1 or E1 link itself is viewed as the controller, use the following commands in privileged EXEC mode to display information about activity on the T1 or E1 line.

Command	Purpose
Router# show controllers t1	Displays information about the T1 link.
Router# show controllers e1	Displays information about the E1 link.

Alarms, line conditions, and other errors are displayed. The data is updated every 10 seconds. Every 15 minutes, the cumulative data is stored and retained for 24 hours. This means at any one time, up to 96 15-minute accumulations are counted in the data display.

Monitoring and Maintaining CSU/DSU Service Modules

This section describes how to monitor and maintain service modules. Tasks involved to monitor and maintain service modules are described in these sections:

- [Performing a Self-Test, page 14](#)
- [Displaying a Performance Report, page 15](#)
- [Performing Loopback Tests, page 15](#)
- [Resetting the CSU/DSU, page 17](#)

Performing a Self-Test

To perform a self-test on the integrated channel service unit/data service unit (CSU/DSU), use the following command in privileged EXEC mode.

Command	Purpose
Router# test service-module interface	Performs a self-test. Specifies the interface type and number.

This command cannot be used if a DTE, line, or remote loopback is in progress. A series of tests are performed on the CSU/DSU, which include a ROM checksum test, a RAM test, an EEPROM checksum test, a flash checksum test, and a DTE loopback with an internal pattern test. This self-test is also performed at power on.

Data transmission is interrupted for 5 seconds when you issue this command. To view the output of the most recent self-test, enable the **show service-module** command.

Displaying a Performance Report

To display the performance report for an integrated CSU/DSU, use one of the following commands in privileged EXEC mode.

Command	Purpose
Router# show service-module interface	Displays a performance report. Choose either serial interface 1 or serial interface 0.
Router# show service-module interface performance-statistics [<i>interval-range</i>]	Displays the CSU/DSU performance statistics for the past 24 hours. This command applies only to the FT1/T1 module.

The *interval-range* value specifies the number of 15-minute intervals displayed in the report. You can choose a range from 1 to 96, where each value represents the CSU/DSU activity performed in that 15-minute interval. For example, a range of 2-3 displays the performance statistics for the intervals two and three.

Performing Loopback Tests

You can loop packets back to the network from the integrated CSU/DSU and loop packets through a local CSU/DSU to a remote CSU/DSU.

Performing Loopback Line Test

To loop data received from the line at the integrated CSU/DSU and loop packets back to the line, use the following commands in interface configuration mode.

	Command	Purpose
Step 1	Router(config-if)# loopback line	Performs loopback on the network at a point physically near the CSU/DSU interface.
Step 2	Router(config-if)# loopback line payload	Performs loopback on the network at a point physically near the interface between the CSU/DSU and the router.

Packets are looped from an incoming network transmission back into the network at a CSU or DSU loopback point.

When the **loopback line** command is configured on the 2-wire, 56-kbps CSU/DSU module or the 4-wire, 56/64-kbps CSU/DSU modules installed on a Cisco 2524 or Cisco 2525 router, the network data loops back at the CSU and the router data loops back at the DSU. If the CSU/DSU is configured for switched mode, you must have an established connection to perform a payload-line loopback. When the **loopback line payload** command is configured, the CSU/DSU module loops the data through the DSU portion of the module. Data is not looped back to the serial interface.

If you enable the **loopback line** command on the fractional T1/T1 module, the CSU/DSU performs a full-bandwidth loopback through the CSU portion of the module and data transmission through the serial interface is interrupted for the duration of the loopback. No reframing or corrections of bipolar violation errors or cyclic redundancy check (CRC) errors are performed. When you configure the **line loopback payload** command on the FT1/T1 module, the CSU/DSU performs a loopback through the DSU portion of the module. The **line loopback payload** command reframes the data link, regenerates the signal, and corrects bipolar violations and Extended Super Frame (ESF) CRC errors.

When performing a T1-line loopback with Extended Super Frame, communication over the facilities data link is interrupted but performance statistics are still updated. To show interfaces currently in loopback operation, use the **show service-module** privileged EXEC command.

Performing Loopback DTE

To loop packets back to DTE from within the local CSU/DSU, use the following command in interface configuration mode.

Command	Purpose
Router(config-if)# loopback dte	Loops packets to DTE.

Packets are looped from within the CSU/DSU back to the serial interface of the router. Send a test ping to see if the packets successfully looped back. To cancel the loopback test, use the **no loopback dte** command.

When using the 4-wire, 56/64-kbps CSU/DSU module, an out-of-service signal is transmitted to the remote CSU/DSU.

Performing a Remote Loopback Test Using the FT1/T1 CSU/DSU Module

The **loopback** command applies only when the remote CSU/DSU device is configured for this function. It is used for testing the data communication channels along with or without remote CSU/DSU circuitry. The loopback is usually performed at the line port, rather than the DTE port, of the remote CSU/DSU.

On the integrated FT1/T1 CSU/DSU module installed on a Cisco 2524 and Cisco 2525 router, the **loopback remote full** command sends the loopup code to the remote CSU/DSU. The remote CSU/DSU should perform a full-bandwidth loopback through the CSU portion of the module. The **loopback remote payload** command sends the loopup code on the configured time slots, while maintaining the D4-Extended Super Frame. The remote CSU/DSU performs the equivalent of a loopback line payload request. The remote CSU/DSU loops back only those time slots that are configured on the remote end. This loopback reframes the data link, regenerates the signal, and corrects bipolar violations and Extended Super Frame CRC errors. The **loopback remote smart-jack** command sends a loopup code to the remote smart jack. You cannot put the local smart jack into loopback.

To loop packets on the integrated FT1/T1 CSU/DSU module, use the following command in interface configuration mode.

Command	Purpose
Router(config-if)# loopback remote {full payload smart-jack} [0in1 1in1 1in2 1in5 1in8 3in24 qrw user-pattern 24bit-binary value]	Loops packets at a remote CSU/DSU using the fractional FT1/T1 CSU/DSU module.

Failure to loop up or initiate a remote loopback request could be caused by enabling the **no service-module t1 remote-loopback** command or having an alternate remote-loopback code configured on the remote end. When the loopback is terminated, the result of the pattern test is displayed.



Note

If the FT1/T1 CSU/DSU module is configured to provide internal clocking, the module ceases to generate clocking when it is placed into loopback.

2- and 4-Wire, 56/64-kbps CSU/DSU Modules

The **loopback** command applies only when the remote CSU/DSU device is configured for this function. It is used for testing the data communication channels along with or without remote CSU/DSU circuitry. The loopback is usually performed at the line port, rather than the DTE port, of the remote CSU/DSU.

On the 2- and 4-wire, 56/64-kbps CSU/DSU modules, an active connection is required before a loopup can be initiated while in switched mode. When transmitting V.54 loopbacks, loopback is initiated for the remote device using V.54 messages. Failure to loop up or initiate a remote loopback request could be caused by enabling the **no service-module 56k remote-loopback** command.

To loop packets at the remote CSU/DSU, use the following command in interface configuration mode.

Command	Purpose
Router(config-if)# loopback remote [2047 511 stress-pattern <i>pattern</i> <i>number</i>]	Loops packets at a remote CSU/DSU using the 2- and 4-wire, 56/64-kbps CSU/DSU modules.

To display loopback interfaces, use the **show interfaces loopback EXEC** command.

Resetting the CSU/DSU

To reset the CSU/DSU, use the following command in privileged EXEC mode.

Command	Purpose
Router# clear service-module <i>interface</i>	Resets the CSU/DSU. Specifies the interface type and number.

Use this command only in severe circumstances (for example, when the router is not responding to a CSU/DSU configuration command).

This command terminates all DTE and line loopbacks that are locally or remotely configured. It also interrupts data transmission through the router for up to 15 seconds. The software performs an automatic software reset in the case of two consecutive configuration failures.

The CSU/DSU module is not reset with the **clear interface** command.



Caution

If you experience technical difficulties with your router and intend to contact customer support, do not use this command. The command erases the past CSU/DSU performance statistics of the router. To clear only the CSU/DSU performance statistics, issue the **clear counters** command.

Monitoring and Maintaining a Hub

To monitor and maintain a hub, you can perform the tasks in the following sections:

- [Shutting Down the Hub Port, page 18](#)
- [Resetting the Hub or Clearing the Hub Counters, page 18](#)
- [Monitoring the Hub, page 18](#)

Shutting Down the Hub Port

To shut down or disable a hub port, use the following commands beginning in global configuration mode.

	Command	Purpose
Step 1	<code>Router(config)# hub ethernet number port [end-port]</code>	Specifies the hub number and the hub port (or range of hub ports) and enters hub configuration mode.
Step 2	<code>Router(config-hub)# shutdown</code>	Shuts down the hub port.

See the examples of shutting down a hub port in the [“Hub Configuration Examples”](#) section of the “Configuring LAN Interfaces” chapter.

Resetting the Hub or Clearing the Hub Counters

To reset the hub or clear the hub counters, use one of the following commands in user EXEC mode.

Command	Purpose
<code>Router> clear hub ethernet number</code>	Resets and reinitializes the hub hardware.
<code>Router> clear hub counters [ethernet number [port [end-port]]]</code>	Clears the hub counters displayed by the show hub command.

Monitoring the Hub

To display hub information, use the following command in user EXEC mode.

Command	Purpose
<code>Router> show hub [ethernet number [port [end-port]]]</code>	Displays hub statistics.

Clearing and Resetting the Interface

To clear the interface counters displayed with the **show interfaces** command, use any of the following commands in user EXEC mode.

Command	Purpose
<code>Router> clear counters [type number] [ethernet serial]</code>	Clears the interface counters.
<code>Router> clear counters fastethernet number</code>	Clears interface counters for the Fast Ethernet NIM on the Cisco 4000 series or Cisco 4500 series routers.
<code>Router> clear counters [type slot/port]</code>	Clears interface counters for the Cisco 7200 series routers.
<code>Router> clear counters [type slot/port-adaptor]</code>	Clears interface counters for the Cisco 7500 series with VIP or VIP2 Interface Processors.

The **clear counters** command clears all the current interface counters from the interface unless the optional arguments are specified to clear only a specific interface type from a specific slot and port number.

**Note**

The **clear counters** command will not clear counters retrieved using SNMP (Simple Network Management Protocol), but only those seen with the **show interfaces** command in user EXEC mode.

To clear and reset interfaces, use the following commands in user EXEC mode. Under normal circumstances, you do not need to clear the hardware logic on interfaces.

	Command	Purpose
Step 1	Router> clear interface <i>type number</i>	Resets the hardware logic on an interface.
Step 2	Router> clear line [<i>number</i>]	Resets the hardware logic on an asynchronous serial line.
Step 3	Router> clear rif-cache	Clears the entire Token Ring RIF cache.

Shutting Down and Restarting an Interface

You can disable an interface by shutting it down. Shutting down an interface disables all functions on the specified interface and marks the interface as unavailable on all monitoring command displays. This information is communicated to other network servers through all dynamic routing protocols. The interface will not be mentioned in any routing updates. On serial interfaces, shutting down an interface causes the dedicated Token Ring (DTR) signal to be dropped. On Token Ring interfaces, shutting down an interface causes the interface to deinsert from the ring. On FDDI interfaces, shutting down an interface causes the optical bypass switch, if present, to go into bypass mode.

To shut down an interface and then restart it, use the following commands in interface configuration mode.

	Command	Purpose
Step 1	Router(config-if)# shutdown	Shuts down an interface.
Step 2	Router(config-if)# no shutdown	Enables an interface that has been disabled.

To check whether an interface is disabled, use the **show interfaces** command in user EXEC mode. An interface that has been shut down is shown as administratively down in the **show interfaces** command display. See the examples in the [“Interface Shutdown Examples”](#) section on page 28.

One reason to shut down an interface is if you want to change the electrical interface type or mode of a Cisco 7500 series port online. You replace the serial adapter cable and use software commands to restart the interface, and if necessary, reconfigure the port for the new interface. At system startup or restart, the Fast Serial Interface Processor (FSIP) polls the interfaces and determines the electrical interface type of each port (according to the type of port adapter cable attached). However, it does not necessarily poll an interface again when you change the adapter cable online. To ensure that the system recognizes the new interface type, shut down using the **shutdown** command, and enable the interface after changing the cable. Refer to your hardware documentation for more details.

Configuring Interface Index Persistence

Interface Index Persistence allows interfaces to be identified with unique values that will remain constant even when a device is rebooted. These interface identification values are used for network monitoring and management using SNMP.

One of the identifiers most commonly used in SNMP-based network management applications is the interface index (ifIndex) value. IfIndex is a unique identifying number associated with a physical or logical interface; as far as most software is concerned, the ifIndex is the “name” of the interface.

Although there is no requirement in the relevant RFCs that the correspondence between particular ifIndex values and their interfaces be maintained across reboots, applications such as device inventory, billing, and fault detection increasingly depend on the maintenance of this correspondence.

IfIndex persistence means that the mapping between the ifDescr object values and the ifIndex object values (generated from the IF-MIB) will be retained across reboots.

Interface Index Persistence allows for greater accuracy when collecting and processing network management data by uniquely identifying input and output interfaces for traffic flows and SNMP statistics. Relating each interface to a known entity, such as an Internet service provider (ISP), allows network management data to be more effectively utilized.

Network data is increasingly being used worldwide for usage-based billing, network planning, policy enforcement, and trend analysis. The ifIndex information is used to identify input and output interfaces for traffic flows and SNMP statistics. Inability to reliably relate each interface to a known entity, such as a customer, invalidates the data.

The interface-specific ifIndex persistence command **snmp ifindex persistence** cannot be used on subinterfaces. A command applied to an interface is automatically applied to all the subinterfaces associated with that interface.

Testing indicates that approximately 25 bytes of NVRAM storage are used by this feature per interface. There may be some boot delay exhibited on platforms with lower CPU speeds.

For more information on configuring and using ifIndex persistence, refer to the following documents:

- The “[Configuring SNMP Support](#)” chapter of the *Cisco IOS Network Management Configuration Guide* (available at Cisco.com)
- The SNMP commands in the *Cisco IOS Network Management Command Reference* (available at Cisco.com)
- *Ethernet-like Interfaces MIB and Interfaces Group MIB Enhancements* feature module, Cisco IOS Release 12.1(2)T (available at Cisco.com)

MIBs Supported for This Feature

- Interfaces MIB (IF-MIB)

Note that this feature does not change any existing MIBs or add any new MIBs.

To obtain lists of MIBs supported by platform and Cisco IOS release, and to download MIB modules, go to the Cisco MIB repository on Cisco.com at the following URL:

<http://www.cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml>

RFC Compliance to Support This Feature

- RFC 2233, *The Interfaces Group MIB Using SMIPv2*

RFCs are available from a variety of internet sources. The primary source is the IETF’s website at <http://www.ietf.org>.

Interface Index Persistence Configuration Task List

The configuration tasks described in this section assume that you have configured SNMP on your routing device and that you are using SNMP to monitor network activity using the Cisco IOS command-line interface and/or a network management system (NMS) application.

See the following sections for configuration tasks for the Interface Index Persistence feature. Each task in the list is identified as required or optional.

- [Enabling and Disabling IfIndex Persistence Globally, page 21](#) (Optional)
- [Enabling and Disabling IfIndex Persistence on Specific Interfaces, page 21](#) (Optional)

Enabling and Disabling IfIndex Persistence Globally

IfIndex persistence is disabled by default. To globally enable ifIndex values that are maintained across reboots, use the following command in global configuration mode.

Command	Purpose
Router(config)# snmp-server ifindex persist	Globally enables ifIndex values that will remain constant across reboots.

To globally disable ifIndex persistence after enabling it, use the following command in global configuration mode.

Command	Purpose
Router(config)# no snmp-server ifindex persist	Disables global ifIndex persistence.



Note

After ifIndex persistence commands have been entered, the configuration must be saved using the **copy running-config startup-config** command in EXEC mode to ensure consistent ifIndex values.

Enabling and Disabling IfIndex Persistence on Specific Interfaces

To enable ifIndex persistence on a specific interface only, use the following commands, beginning in global configuration mode.

	Command	Purpose
Step 1	Router(config)# interface <i>type slot/port</i>	Enters interface configuration mode for the specified interface. Note that the syntax of the interface command will vary depending on the platform that you are using.
Step 2	Router(config-if)# snmp ifindex persist	Enables an ifIndex value that is constant across reboots on the specified interface.
Step 3	Router(config-if)# exit	Exits interface configuration mode.

To disable ifIndex persistence on a specific interface only, use the following commands, beginning in global configuration mode.

	Command	Purpose
Step 1	Router(config)# interface <i>type slot/port</i>	Enters interface configuration mode for the specified interface. Note that the syntax of the interface command will vary depending on the platform that you are using.
Step 2	Router(config-if)# no snmp ifindex persist	Disables an ifIndex value that is constant across reboots on the specified interface.
Step 3	Router(config-if)# exit	Exits interface configuration mode.

To clear the interface-specific ifIndex persistence setting and configure the interface to use the global configuration setting, use the following commands, beginning in global configuration mode.

	Command	Purpose
Step 1	Router(config)# interface <i>type slot/port</i>	Enters interface configuration mode for the specified interface. Note that the syntax of the interface command will vary depending on the platform that you are using.
Step 2	Router(config-if)# snmp ifindex clear	Clears any interface-specific ifIndex persistence configuration for the specified interface. The ifIndex setting (enabled or disabled) will match the global configuration setting.
Step 3	Router(config-if)# exit	Exits interface configuration mode.

When you clear the interface-specific setting, only the global ifIndex persistence setting will apply to the interface. Regardless of whether the specific interface has ifIndex persistence enabled or disabled, the ifIndex persistence setting will default to the global setting after you issue the **snmp ifindex clear** command.

For example, assume that you enabled ifIndex persistence on Ethernet interface 0/1, and then globally enabled ifIndex persistence. Using the **snmp ifindex clear** command in interface configuration mode for Ethernet interface 0/1 would leave that interface with ifIndex enabled, because the global setting is to have ifIndex persistence enabled.

Likewise, if you disabled ifIndex persistence for Ethernet interface 0/1, globally enabled ifIndex persistence, and then issued the **snmp ifindex clear** command on that interface, ifIndex would be enabled (according to the global setting) on that interface.



Tip

Use the **snmp ifindex clear** command on a specific interface when you want that interface to use the global configuration setting for ifIndex persistence. This command clears any ifIndex configuration commands previously entered for that specific interface.

To verify that ifIndex commands have been configured, use the **more system:running-config** command.

Configuring Loopback Detection

When an interface has a backup interface configured, it is often desirable that the backup interface be enabled when the primary interface is either down or in loopback. By default, the backup is only enabled if the primary interface is down. By using the **down-when-looped** command, the backup interface will also be enabled if the primary interface is in loopback. To achieve this condition, use the following command in interface configuration mode.

Command	Purpose
Router(config-if) # down-when-looped	Configures an interface to tell the system it is down when loopback is detected.

If testing an interface using the loopback command, you should not have loopback detection configured, or packets will not be transmitted out the interface that is being tested.

Running Interface Loopback Diagnostics

You can use a loopback test on lines to detect and distinguish equipment malfunctions between line and modem or CSU/DSU problems on the network server. If correct data transmission is not possible when an interface is in loopback mode, the interface is the source of the problem. The DSU might have similar loopback functions that you can use to isolate the problem if the interface loopback test passes. If the device does not support local loopback, this function will have no effect.

You can specify hardware loopback tests on the Ethernet and synchronous serial interfaces and on all Token Ring interfaces that are attached to CSU/DSUs and that support the local loopback signal. The CSU/DSU acts as a DCE device; the router or access server acts as a DTE device. The local loopback test generates a CSU loop—a signal that goes through the CSU/DSU to the line, then back through the CSU/DSU to the router or access server. The **ping** command can also be useful during loopback operation.

The loopback tests are described in the following sections:

- High-Speed Serial Interface (HSSI), including the High-Speed Communications Interface (HSCI) card ribbon cable
- Cisco Multiport Communications Interface (MCI) and Cisco Serial Communication Interface (SCI) synchronous serial interfaces
- MCI and Cisco Multiport Ethernet Controller (MEC) Ethernet interfaces (an Ethernet loopback server is also provided on the Ethernet interfaces.)
- Ethernet loopback server
- Channelized E1 interfaces (local loopback only)
- Channelized T1 interfaces (local and remote loopback)
- Fractional T1/T1 interfaces
- Token Ring interfaces
- Channelized E1 controller and interface (local loopback only)
- Channelized T1 controller and interface (local and remote loopback)
- Troubleshooting channelized E1 and channelized T1

The following sections describe each test.


Note

Loopback does not work on an X.21 DTE because the X.21 interface definition does not include a loopback definition.

Enabling Loopback Testing on the HSSI

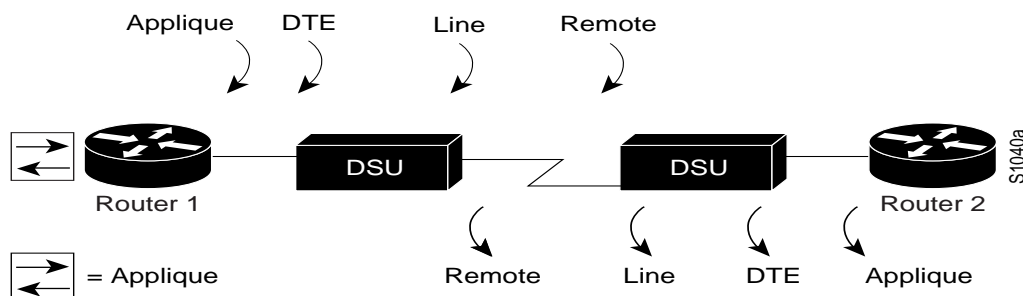
The HSSI allows you to perform the tasks described in these sections:

- [Enabling a Loopback Test to the DTE, page 24](#)
- [Enabling a Loopback Test Through the CSU/DSU, page 24](#)
- [Enabling a Loopback Test over a Remote DS-3 Link, page 25](#)

These tests apply only when the device supports them and are used to check the data communication channels. The tests are usually performed at the line port rather than at the DTE port of the remote CSU/DSU.

The internal loopback concepts are illustrated in [Figure 2](#).

Figure 2 HSSI Loopback Testing



Enabling a Loopback Test to the DTE

You can loop packets to DTE within the CSU/DSU at the DTE interface, when the device supports this function. Doing so is useful for testing the DTE-to-DCE cable. To loop the packets to DTE, use the following command in interface configuration mode.

Command	Purpose
Router(config-if)# loopback dte	Loops packets to DTE internally.

Enabling a Loopback Test Through the CSU/DSU

You can loop packets completely through the CSU/DSU to configure a CSU loop, when the device supports this feature. Doing so is useful for testing the DCE device (CSU/DSU) itself. To configure a CSU loop, use the following command in interface configuration mode.

Command	Purpose
Router(config-if)# loopback line	Loops packets completely through the CSU/DSU.

Enabling a Loopback Test over a Remote DS-3 Link

You can loop packets through the CSU/DSU, over the digital signal level 3 (DS-3) link, and to the remote CSU/DSU and back. To do this, use the following command in interface configuration mode.

Command	Purpose
Router(config-if)# loopback remote	Loops packets through the CSU/DSU to a remote CSU/DSU over the DS-3 link.

This command applies only when the device supports the remote function. It is used for testing the data communication channels. The loopback usually is performed at the line port, rather than the DTE port, of the remote CSU/DSU.

Configuring the Ethernet Loopback Server

The router software provides an Ethernet loopback server that supports Digital Equipment Corporation (Digital), Intel, and Xerox systems specified by the “blue book,” a joint specification written by Digital, Intel, and Xerox that defines the Ethernet protocol. The loopback server responds to forward data loopback messages sent either to the MAC address of the server or to the broadcast address. Currently, the Ethernet loopback server does not respond to the loopback assistance multicast address.

Use the Ethernet loopback server to test communications between your internetworking products and Digital systems that do not support the IP **ping** command, such as DECnet-only VMS systems.

To originate a loop test on your VMS system with a Cisco server, use the Digital Network Control Program (NCP) **loop circuit** command. For more information about the **loop circuit** command, consult the DECnet VAX documentation. Cisco network servers support all options that can be specified by the VMS hosts.

Enabling Loopback on Token Ring Cards

To place all Token Ring interface cards into loopback mode, use the following command in interface configuration mode.

Command	Purpose
Router(config-if)# loopback	Enables loopback and verifies that the Token Ring interface receives back every packet it sends.

Enabling Loopback Testing of Fractional T1/T1

For information, see the [“Performing Loopback Tests”](#) section on page 15.

Reloading a Cisco 7500 Single Line Card

The Cisco IOS software allows users to correct a line card failure on a Cisco 7500 series router by reloading the failed line card without reloading any other line cards on the network backplane. During the single line card reload process, all physical lines and routing protocols on the other line cards of the network backplane remain active.

The Cisco 7500 Single Line Card Reload feature works on all route switch processor (RSP) images for all Cisco IOS releases that support the Cisco 7500 Single Line Card Reload feature.

Improved Line Card Recovery Time

Use this feature to correct a line card hardware failure when the Cisco 7500 Single Line Card Reload feature is enabled. The entire system, which now only reloads one line card instead of every line card, also experiences a dramatic improvement in recovery time.

Network Traffic Flow Improvements

Because the Cisco 7500 Single Line Card Reload feature only reloads the line card with the hardware failure rather than all of the line cards on the Cisco 7500 network backplane, the active line cards can continue to forward network traffic.

Configuring Cisco 7500 Single Line Card Reloading

To enable the Cisco 7500 Single Line Card Reloading feature on the Cisco 7500 series router, use the **service single-slot-reload-enable** command in global configuration mode.

Command	Purpose
Router(config)# service single-slot-reload-enable	Enables single line card reloading for all of the line cards in the Cisco 7500 series router.

Disabling Cisco 7500 Single Line Card Reloading

The Cisco 7500 Single Line Card Reloading feature is disabled by default. Therefore, the process for disabling the Cisco 7500 Single Line Card Reloading feature is only necessary if the Cisco 7500 Single Line Card Reloading feature has been enabled by the user on the Cisco 7500 series router.

To disable the Cisco 7500 Single Line Card feature, enter the **no service single-slot-reload-enable** command in global configuration mode on the Cisco 7500 series router.

Command	Purpose
Router(config)# no service single-slot-reload-enable	Disables single line card reloading for all line cards in the Cisco 7500 series router.

Verifying Cisco 7500 Single Line Card Reloading

Use the **show running-config** command to verify that single line card reloading has been successfully enabled on the Cisco 7500 series router. If the “service single-slot-reload-enable” line appears in the command output, Cisco 7500 Single Line Card Reloading is enabled. If this line does not appear in the command output, Cisco 7500 Single Line Card Reloading is disabled.

Use the **show diag** command to display hardware information on line cards, including the history of line card reloads.

Troubleshooting Tips

The **debug oir** command is used to debug the online insertion and removal (OIR) feature (which is also known as hot-swapping or power-on servicing). The **debug oir** command is often useful in debugging problems related to OIR, including single line card reloading.

Interface Configuration Examples

This section includes the following examples to illustrate configuration tasks described in this chapter:

- [Interface Enablement Configuration Examples, page 27](#)
- [Interface Description Examples, page 27](#)
- [Interface Shutdown Examples, page 28](#)
- [Interface Index Persistence Examples, page 28](#)
- [Cisco 7500 Line Card Reload Examples, page 29](#)

Interface Enablement Configuration Examples

The following example illustrates how to begin interface configuration on a serial interface. It assigns PPP encapsulation to serial interface 0.

```
interface serial 0
 encapsulation ppp
```

series routers Specific IP Addresses Configuration for an Interface Example

This example shows how to configure the access server so that it will use the default address pool on all interfaces except interface 7, on which it will use an address pool called lass:

```
ip address-pool local
ip local-pool lass 172.30.0.1
 async interface
 interface 7
 peer default ip address lass
```

Interface Description Examples

The following example illustrates how to add a description about an interface that will appear in configuration files and monitoring command displays:

```
interface ethernet 0
 description First Ethernet in network 1
 ip address 172.18.15.78 255.255.255.0
```

The following example for a Cisco 7500 series routers describes an administration network attached to the Ethernet processor in slot 2, port 4:

```
interface ethernet 2/4
 description 2nd floor administration net
```

Interface Shutdown Examples

The following example turns off the Ethernet interface in slot 2 at port 4:

```
interface ethernet 2/4
shutdown
```

The following example restarts the interface:

```
interface ethernet 2/4
no shutdown
```

The following example shuts down a Token Ring interface:

```
interface tokenring 0
shutdown
```

The following example shuts down a T1 circuit number 23 that is running on a Cisco 7500 series router:

```
interface serial 4/0:23
shutdown
```

The following example shuts down the entire T1 line physically connected to a Cisco 7500 series router:

```
controller t1 4/0
shutdown
```

Interface Index Persistence Examples

This section provides the following configuration examples:

- [Enabling IfIndex Persistence on All Interfaces Example, page 28](#)
- [Enabling IfIndex Persistence on a Specific Interface Example, page 28](#)
- [Disabling IfIndex Persistence on a Specific Interface Example, page 28](#)
- [Clearing IfIndex Persistence Configuration from a Specific Interface Example, page 29](#)

Enabling IfIndex Persistence on All Interfaces Example

In the following example, ifIndex persistence is enabled for all interfaces:

```
snmp-server ifindex persist
```

Enabling IfIndex Persistence on a Specific Interface Example

In the following example, ifIndex persistence is enabled for Ethernet interface 0/1 only:

```
interface ethernet 0/1
snmp ifindex persist
exit
```

Disabling IfIndex Persistence on a Specific Interface Example

In the following example, ifIndex persistence is disabled for Ethernet interface 0/1 only:

```
interface ethernet 0/1
no snmp ifindex persist
exit
```

Clearing IfIndex Persistence Configuration from a Specific Interface Example

In the following example, any previous setting for ifIndex persistence on Ethernet interface 0/1 is removed from the configuration. If ifIndex persistence is globally enabled, ifIndex persistence will be enabled for Ethernet interface 0/1. If ifIndex persistence is globally disabled, ifIndex persistence will be disabled for Ethernet interface 0/1.

```
interface ethernet 0/1
 snmp ifindex clear
exit
```

Cisco 7500 Line Card Reload Examples

In the following example, single line card reloading is enabled for all lines cards in the Cisco 7500 series router:

```
service single-slot-reload-enable
```

In the following example, single line card reloading is disabled for all line cards in the Cisco 7500 series router:

```
no service single-slot-reload-enable
```

