C H A P T E R 10

Configuring Interface Characteristics

This chapter defines the types of interfaces on the switch and describes how to configure them. The chapter consists of these sections:

Note For complete syntax and usage information for the commands used in this chapter, see the switch command reference for this release and the online *Cisco IOS Interface Command Reference, Release 12.2*.

Understanding Interface Types

This section describes the different types of interfaces supported by the switch with references to chapters that contain more detailed information about configuring these interface types. The rest of the chapter describes configuration procedures for physical interface characteristics.

These sections describe the interface types:

Port-Based VLANs

A VLAN is a switched network that is logically segmented by function, team, or application, without regard to the physical location of the users. For more information about VLANs, see Chapter 12, "Configuring VLANs." Packets received on a port are forwarded only to ports that belong to the same VLAN as the receiving port. Network devices in different VLANs cannot communicate with one another without a Layer 3 device to route traffic between the VLANs.

VLAN partitions provide hard firewalls for traffic in the VLAN, and each VLAN has its own MAC address table. A VLAN comes into existence when a local port is configured to be associated with the VLAN, when the VLAN Trunking Protocol (VTP) learns of its existence from a neighbor on a trunk, or when a user creates a VLAN.

To configure VLANs, use the **vlan** *vlan-id* global configuration command to enter VLAN configuration mode. The VLAN configurations for normal-range VLANs (VLAN IDs 1 to 1005) are saved in the VLAN database. If VTP is version 1 or 2, to configure extended-range VLANs (VLAN IDs 1006 to 4094), you must first set VTP mode to transparent. Extended-range VLANs created in transparent mode are not added to the VLAN database but are saved in the switch running configuration. With VTP version 3, you can create extended-range VLANs in client or server mode. These VLANs are saved in the VLAN database.

Add ports to a VLAN by using the **switchport** interface configuration commands:

- Identify the interface.
- For a trunk port, set trunk characteristics, and if desired, define the VLANs to which it can belong.
- For an access port, set and define the VLAN to which it belongs.
- For a tunnel port, set and define the VLAN ID for the customer-specific VLAN tag. See Chapter 16, "Configuring IEEE 802.1Q and Layer 2 Protocol Tunneling."

Switch Ports

Switch ports are Layer 2-only interfaces associated with a physical port. Switch ports belong to one or more VLANs. A switch port can be an access port, a trunk port, or a tunnel port. You can configure a port as an access port or trunk port or let the Dynamic Trunking Protocol (DTP) operate on a per-port basis to set the switchport mode by negotiating with the port on the other end of the link. You must manually configure tunnel ports as part of an asymmetric link connected to an IEEE 802.1Q trunk port. Switch ports are used for managing the physical interface and associated Layer 2 protocols and do not handle routing.

Configure switch ports by using the **switchport** interface configuration commands. Use the **switchport** command with no keywords to put an interface that is in Layer 3 mode into Layer 2 mode.

Note When you put an interface that is in Layer 3 mode into Layer 2 mode, the previous configuration information related to the affected interface might be lost, and the interface is returned to its default configuration.

For detailed information about configuring access port and trunk port characteristics, see Chapter 12, "Configuring VLANs." For more information about tunnel ports, see Chapter 16, "Configuring IEEE 802.1Q and Layer 2 Protocol Tunneling."

Internal Gigabit Ethernet Ports

The Gigabit Ethernet ports 1 to 16 (gibabitethernet0/1 to gibabitethernet0/16) are internal interfaces that provide communication between the switch and the blade server. These interfaces operate at 1000 Mbps, full-duplex, and use the 1000BASE-X protocol.

If the Onboard Administrator detects a physical problem between the blade server and the switch, the Onboard Administrator changes these interfaces to the EKEY error-disabled state. You must use the Onboard Administrator to find the root cause of the problem, and to recover from the error-disabled state. See the HP BladeSystem documentation at http://www.hp.com/go/bladesystem/documentation for more information.

Access Ports

An access port belongs to and carries the traffic of only one VLAN (unless it is configured as a voice VLAN port). Traffic is received and sent in native formats with no VLAN tagging. Traffic arriving on an access port is assumed to belong to the VLAN assigned to the port. If an access port receives a tagged packet (Inter-Switch Link [ISL] or IEEE 802.1Q tagged), the packet is dropped, and the source address is not learned.

Two types of access ports are supported:

- Static access ports are manually assigned to a VLAN (or through a RADIUS server for use with IEEE 802.1x. For more information, see the "802.1x Authentication with VLAN Assignment" section on page 8-14.)

- VLAN membership of dynamic access ports is learned through incoming packets. By default, a dynamic access port is not a member of any VLAN, and forwarding to and from the port is enabled only when the VLAN membership of the port is discovered. Dynamic access ports on the switch are assigned to a VLAN by a VLAN Membership Policy Server (VMPS). The VMPS can be a Catalyst 6500 series switch; the Cisco Catalyst Blade Switch 3020 for HP cannot be a VMPS server.

You can also configure an access port with an attached Cisco IP Phone to use one VLAN for voice traffic and another VLAN for data traffic from a device attached to the phone. For more information about voice VLAN ports, see Chapter 14, "Configuring Voice VLAN."

Trunk Ports

A trunk port carries the traffic of multiple VLANs and by default is a member of all VLANs in the VLAN database. These trunk port types are supported:

- In an ISL trunk port, all received packets are expected to be encapsulated with an ISL header, and all transmitted packets are sent with an ISL header. Native (non-tagged) frames received from an ISL trunk port are dropped.

- An IEEE 802.1Q trunk port supports simultaneous tagged and untagged traffic. An IEEE 802.1Q trunk port is assigned a default port VLAN ID (PVID), and all untagged traffic travels on the port default PVID. All untagged traffic and tagged traffic with a NULL VLAN ID are assumed to belong to the port default PVID. A packet with a VLAN ID equal to the outgoing port default PVID is sent untagged. All other traffic is sent with a VLAN tag.

Although by default, a trunk port is a member of every VLAN known to the VTP, you can limit VLAN membership by configuring an allowed list of VLANs for each trunk port. The list of allowed VLANs does not affect any other port but the associated trunk port. By default, all possible VLANs (VLAN ID 1 to 4094) are in the allowed list. A trunk port can become a member of a VLAN only if VTP knows of the VLAN and if the VLAN is in the enabled state. If VTP learns of a new, enabled VLAN and the VLAN

is in the allowed list for a trunk port, the trunk port automatically becomes a member of that VLAN and traffic is forwarded to and from the trunk port for that VLAN. If VTP learns of a new, enabled VLAN that is not in the allowed list for a trunk port, the port does not become a member of the VLAN, and no traffic for the VLAN is forwarded to or from the port.

For more information about trunk ports, see Chapter 12, "Configuring VLANs."

Tunnel Ports

Tunnel ports are used in IEEE 802.1Q tunneling to segregate the traffic of customers in a service-provider network from other customers who are using the same VLAN number. You configure an asymmetric link from a tunnel port on a service-provider edge switch to an IEEE 802.1Q trunk port on the customer switch. Packets entering the tunnel port on the edge switch, already IEEE 802.1Q-tagged with the customer VLANs, are encapsulated with another layer of an IEEE 802.1Q tag (called the metro tag), containing a VLAN ID unique in the service-provider network, for each customer. The double-tagged packets go through the service-provider network keeping the original customer VLANs separate from those of other customers. At the outbound interface, also a tunnel port, the metro tag is removed, and the original VLAN numbers from the customer network are retrieved.

Tunnel ports cannot be trunk ports or access ports and must belong to a VLAN unique to each customer.

For more information about tunnel ports, see Chapter 16, "Configuring IEEE 802.1Q and Layer 2 Protocol Tunneling."

Routed Ports

A routed port is a physical port that acts like a port on a router; it does not have to be connected to a router. A routed port is not associated with a particular VLAN, as is an access port. A routed port behaves like a regular router interface, except that it does not support VLAN subinterfaces. Routed ports can be configured with a Layer 3 routing protocol. A routed port is a Layer 3 interface only and does not support Layer 2 protocols, such as DTP and STP.

Configure routed ports by putting the interface into Layer 3 mode with the **no switchport** interface configuration command. Then assign an IP address to the port, enable routing, and assign routing protocol characteristics by using the **ip routing** and **router** *protocol* global configuration commands.

✎
Note   Entering a **no switchport** interface configuration command shuts down the interface and then re-enables it, which might generate messages on the device to which the interface is connected. When you put an interface that is in Layer 2 mode into Layer 3 mode, the previous configuration information related to the affected interface might be lost.

The number of routed ports that you can configure is not limited by software. However, the interrelationship between this number and the number of other features being configured might impact CPU performance because of hardware limitations. See the "Configuring Layer 3 Interfaces" section on page 10-21 for information about what happens when hardware resource limitations are reached.

For more information about IP unicast routing and routing protocols, see Chapter 35, "Configuring IP Unicast Routing."

Switch Virtual Interfaces

A switch virtual interface (SVI) represents a VLAN of switch ports as one interface to the routing or bridging function in the system. Only one SVI can be associated with a VLAN, but you need to configure an SVI for a VLAN only when you wish to route between VLANs, to fallback-bridge nonroutable protocols between VLANs, or to provide IP host connectivity to the switch. By default, an SVI is created for the default VLAN (VLAN 1) to permit remote switch administration. Additional SVIs must be explicitly configured.

> **Note**   You cannot delete interface VLAN 1.

SVIs provide IP host connectivity only to the system; in Layer 3 mode, you can configure routing across SVIs.

Although the switch supports a total or 1005 VLANs (and SVIs), the interrelationship between the number of SVIs and routed ports and the number of other features being configured might impact CPU performance because of hardware limitations. See the "Configuring Layer 3 Interfaces" section on page 10-21 for information about what happens when hardware resource limitations are reached.

SVIs are created the first time that you enter the **vlan** interface configuration command for a VLAN interface. The VLAN corresponds to the VLAN tag associated with data frames on an ISL or IEEE 802.1Q encapsulated trunk or the VLAN ID configured for an access port. Configure a VLAN interface for each VLAN for which you want to route traffic, and assign it an IP address. For more information, see the "Manually Assigning IP Information" section on page 3-14.

> **Note**   When you create an SVI, it does not become active until it is associated with a physical port.

SVIs support routing protocols and bridging configurations. For more information about configuring IP routing, see Chapter 35, "Configuring IP Unicast Routing."

SVI Autostate Exclude

The line state of an SVI with multiple ports on a VLAN is in the *up* state when it meets these conditions:

- The VLAN exists and is active in the VLAN database on the switch.
- The VLAN interface exists and is not administratively down.
- At least one Layer 2 (access or trunk) port exists, has a link in the *up* state on this VLAN, and is in the spanning-tree forwarding state on the VLAN.

> **Note**   The protocol link state for VLAN interfaces come up when the first switchport belonging to the corresponding VLAN link comes up and is in STP forwarding state.

The default action, when a VLAN has multiple ports, is that the SVI goes down when all ports in the VLAN go down. You can use the SVI autostate exclude feature to configure a port so that it is not included in the SVI line-state up-an- down calculation. For example, if the only active port on the VLAN is a monitoring port, you might configure autostate exclude on that port so that the VLAN goes down when all other ports go down. When enabled on a port, **autostate exclude** applies to all VLANs that are enabled on that port.

The VLAN interface is brought up when one Layer 2 port in the VLAN has had time to converge (transition from STP listening-learning state to forwarding state). This prevents features such as routing protocols from using the VLAN interface as if it were fully operational and minimizes other problems, such as routing black holes. For information about configuring autostate exclude, see the "Configuring SVI Autostate Exclude" section on page 10-22.

EtherChannel Port Groups

EtherChannel port groups treat multiple switch ports as one switch port. These port groups act as a single logical port for high-bandwidth connections between switches or between switches and servers. An EtherChannel balances the traffic load across the links in the channel. If a link within the EtherChannel fails, traffic previously carried over the failed link changes to the remaining links. You can group multiple trunk ports into one logical trunk port or multiple access ports into one logical access port.

Most protocols operate over either single ports or aggregated switch ports and do not recognize the physical ports within the port group. Exceptions are the DTP, the Cisco Discovery Protocol (CDP), and the Port Aggregation Protocol (PAgP), which operate only on physical ports.

When you configure an EtherChannel, you create a port-channel logical interface and assign an interface to the EtherChannel. For Layer 3 interfaces, you manually create the logical interface by using the **interface port-channel** global configuration command. Then you manually assign an interface to the EtherChannel by using the **channel-group** interface configuration command. For Layer 2 interfaces, use the **channel-group** interface configuration command to dynamically create the port-channel logical interface. This command binds the physical and logical ports together. For more information, see Chapter 34, "Configuring EtherChannels and Layer 2 Trunk Failover."

Dual-Purpose Uplink Ports

The Cisco Catalyst Blade Switch 3020 for HP supports dual-purpose uplink ports on six of the eight uplink ports. Four of the uplink ports, 17 to 20, are considered as a single interface with dual front ends (an RJ-45 connector and an SFP module connector). The dual front ends on ports 17 to 20 are not redundant interfaces, and the switch activates only one connector of the pair.

By default, the switch dynamically selects the interface type that first links up. However, you can use the **media-type** interface configuration command to manually select the RJ-45 connector or the SFP module connector. For information about configuring speed and duplex settings for a dual-purpose uplink, see the "Setting the Interface Speed and Duplex Parameters" section on page 10-17.

Uplink ports 17 to 20 have two LEDs: one shows the status of the RJ-45 port, and one shows the status of the SFP module port. The port LED is on for whichever connector is active. For more information about the LEDs, see the hardware installation guide.

Ports 23x and 24x are different from the other dual-purpose ports. When operating in external mode, these ports are single, uplink 10/100/1000BASE-T copper Gigabit Ethernet ports. When operating in internal mode, they use the 1000BASE-X mode, and they form a cross-connection with a switch that is installed in a corresponding module bay in the blade server. The default operation mode for ports 23x and 24x is external, set by using the **rj45** keyword of the **media-type** interface configuration command.

Connecting Interfaces

Devices within a single VLAN can communicate directly through any switch. Ports in different VLANs cannot exchange data without going through a routing device. In the configuration shown in Figure 10-1, when Blade Server A in VLAN 20 sends data to Blade Server B in VLAN 30, the data must go from Blade Server A to the switch, to the router, back to the switch, and then to Blade Server B.

Figure 10-1   Connecting VLANs with Layer 2 Switches
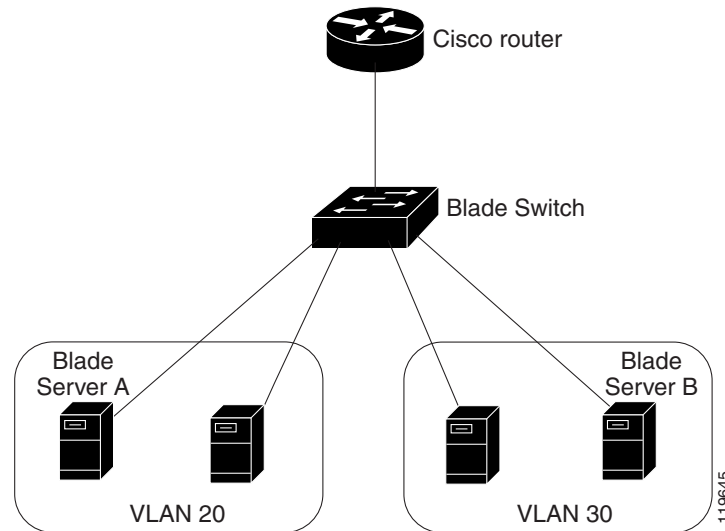
By using the switch with routing enabled, when you configure both VLAN 20 and VLAN 30 with an SVI to which an IP address is assigned, packets can be sent from Host A to Host B directly through the switch with no need for an external router.

The switch supports basic routing (static routing and RIP). Whenever possible, to maintain high performance, forwarding is done by the switch hardware. However, only IP Version 4 packets with Ethernet II encapsulation can be routed in hardware. Non-IP traffic and traffic with other encapsulation methods can be fallback-bridged by hardware.

The routing function can be enabled on all SVIs and routed ports. The switch routes only IP traffic. When IP routing protocol parameters and address configuration are added to an SVI or routed port, any IP traffic received from these ports is routed. For more information, see Chapter 35, "Configuring IP Unicast Routing."

Management-Only Interface

The Fast Ethernet 0 (fa0) interface is an internal connection to the HP Onboard Administrator and is only used for switch management traffic, not for data traffic. It is connected to the Onboard Administrator through the blade server backplane connector. Management information that is sent to or received from this interface is not sent to the other Ethernet interfaces on the switch. This interface cannot send or receive data traffic from the servers that are connected to Gigabit interfaces 0/1 to 0/16. The speed and duplex settings for this interface are fixed at 100 Mpbs and full duplex.

The fa0 interface is a routed interface. You can use the IP addresses that are assigned to this interface to manage the switch through the HP Onboard Administrator module. By default, the fa0 interface is assigned an IP address through a DHCP server. You can also statically configure the IP address. You can see the IP address that is assigned to the fa0 interface from the Onboard Administrator GUI, through which you can manage the switch through the HP Onboard Administrator module.

We recommend that you set up your network so that you can communicate with the assigned fa0 IP address from the same external network in which the HP Onboard Administrator is located. For more information on the IP routing and IP forwarding capabilities of the HP Onboard Administrator module, see the HP BladeSystem documentation at http://www.hp.com/go/bladesystem/documentation.

If you do not want to manage the switch through the HP Onboard Administrator module, you can disable the fa0 interface by using the **shutdown** interface configuration command.

The fa0 interface does not route its received IP packets to the IP Address that is assigned to other VLAN interfaces on the switch. IP packets that are received by the VLAN interfaces are not routed to the fa0 interface. The fa0 interface is a routed interface, but the switch does not route IP data packets.

Using Interface Configuration Mode

The switch supports these interface types:

- Physical ports—switch ports and routed ports
- VLANs—switch virtual interfaces
- Port channels—EtherChannel interfaces

You can also configure a range of interfaces (see the "Configuring a Range of Interfaces" section on page 10-10).

To configure a physical interface (port), specify the interface type, module number, and switch port number, and enter interface configuration mode.

- Type—Gigabit Ethernet (gigabitethernet or gi) for 10/100/1000 Mb/s Ethernet port or small form-factor pluggable (SFP) module Gigabit Ethernet interfaces.
- Module number—The module or slot number on the switch (always 0 on the Cisco Catalyst Blade Switch 3020for HP).
- Port number— the interface number on the switch. The port numbers always begin at 1, starting with the internal blade server-facing interfaces. The external interfaces on the front panel of the switch start at 17. The SFP module ports are numbered left to right. The external RJ-45 ports are numbered from top to bottom and from left to right. The first RJ-45 port, 17, is on the top left. RJ-45 port 18 is on the bottom left.

You can identify physical interfaces by physically checking the interface location on the switch. You can also use the **show** privileged EXEC commands to display information about a specific interface or all the interfaces on the switch. The remainder of this chapter primarily provides physical interface configuration procedures.

Procedures for Configuring Interfaces

These general instructions apply to all interface configuration processes.

Step 1   Enter the **configure terminal** command at the privileged EXEC prompt:

```
Switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)#
```

Step 2   Enter the **interface** global configuration command. Identify the interface type and the number of the connector. In this example, Gigabit Ethernet port 1 is selected:

```
Switch(config)# interface gigabitethernet0/1
Switch(config-if)#
```

> **Note**   You do not need to add a space between the interface type and interface number. For example, in the preceding line, you can specify either **gigabitethernet 0/1**, **gigabitethernet0/1**, **gi 0/1**, or **gi0/1**.

Step 3   Follow each **interface** command with the interface configuration commands that the interface requires. The commands that you enter define the protocols and applications that will run on the interface. The commands are collected and applied to the interface when you enter another interface command or enter **end** to return to privileged EXEC mode.

You can also configure a range of interfaces by using the **interface range** or **interface range macro** global configuration commands. Interfaces configured in a range must be the same type and must be configured with the same feature options.

Step 4   After you configure an interface, verify its status by using the **show** privileged EXEC commands listed in the .

Enter the **show interfaces** privileged EXEC command to see a list of all interfaces on or configured for the switch. A report is provided for each interface that the device supports or for the specified interface.

Configuring a Range of Interfaces

You can use the **interface range** global configuration command to configure multiple interfaces with the same configuration parameters. When you enter the interface-range configuration mode, all command parameters that you enter are attributed to all interfaces within that range until you exit this mode.

Beginning in privileged EXEC mode, follow these steps to configure a range of interfaces with the same parameters:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface range** {*port-range*	**macro** *macro_name*}	Specify the range of interfaces (VLANs or physical ports) to be configured, and enter interface-range configuration mode. • You can use the **interface range** command to configure up to five port ranges or a previously defined macro. • The **macro** variable is explained in the "Configuring and Using Interface Range Macros" section on page 10-11. • In a comma-separated *port-range*, you must enter the interface type for each entry and enter spaces before and after the comma. • In a hyphen-separated *port-range*, you do not need to re-enter the interface type, but you must enter a space before the hyphen.
Step 3		Use the normal configuration commands to apply the configuration parameters to all interfaces in the range. Each command is executed as it is entered.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show interfaces** [*interface-id*]	Verify the configuration of the interfaces in the range.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

When using the **interface range** global configuration command, note these guidelines:

• Valid entries for *port-range*:

– **vlan** *vlan-ID - vlan-ID*, where the VLAN ID is 1 to 4094

– **gigabitethernet** module/{*first port*} - {*last port*}, where the module is always 0

– **port-channel** *port-channel-number - port-channel-number*, where the *port-channel-number* is 1 to 48

✎

Note   When you use the **interface range** command with port channels, the first and last port-channel number must be active port channels.

- You must add a space between the first interface number and the hyphen when using the **interface range** command. For example, the command **interface range gigabitethernet0/1 - 4** is a valid range; the command **interface range gigabitethernet0/1-4** is not a valid range.

- The **interface range** command only works with VLAN interfaces that have been configured with the **interface vlan** command. The **show running-config** privileged EXEC command displays the configured VLAN interfaces. VLAN interfaces not displayed by the **show running-config** command cannot be used with the **interface range** command.

- All interfaces defined in a range must be the same type (all Gigabit Ethernet ports, all EtherChannel ports, or all VLANs), but you can enter multiple ranges in a command.

This example shows how to use the **interface range** global configuration command to set the speed on ports 1 to 4 to 100 Mb/s:

```
Switch# configure terminal
Switch(config)# interface range gigabitethernet0/1 - 4
Switch(config-if-range)# speed 100
```

If you enter multiple configuration commands while you are in interface-range mode, each command is executed as it is entered. The commands are not batched and executed after you exit interface-range mode. If you exit interface-range configuration mode while the commands are being executed, some commands might not be executed on all interfaces in the range. Wait until the command prompt reappears before exiting interface-range configuration mode.

Configuring and Using Interface Range Macros

You can create an interface range macro to automatically select a range of interfaces for configuration. Before you can use the **macro** keyword in the **interface range macro** global configuration command string, you must use the **define interface-range** global configuration command to define the macro.

Beginning in privileged EXEC mode, follow these steps to define an interface range macro:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**define interface-range** *macro_name* *interface-range*	Define the interface-range macro, and save it in NVRAM. • The *macro_name* is a 32-character maximum character string. • A macro can contain up to five comma-separated interface ranges. • Each *interface-range* must consist of the same port type.
Step 3	**interface range macro** *macro_name*	Select the interface range to be configured using the values saved in the interface-range macro called *macro_name.* You can now use the normal configuration commands to apply the configuration to all interfaces in the defined macro.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show running-config	include define**	Show the defined interface range macro configuration.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no define interface-range** *macro_name* global configuration command to delete a macro.

When using the **define interface-range** global configuration command, note these guidelines:

- Valid entries for *interface-range*:

    – **vlan** *vlan-ID- vlan-ID*, where the VLAN ID is 1 to 4094

    – **gigabitethernet** module/{*first port*} - {*last port*}, where the module is always 0

    – **port-channel** *port-channel-number - port-channel-number*, where the *port-channel-number* is 1 to 48.

    > ✎
    >
    > **Note**   When you use the interface ranges with port channels, the first and last port-channel number must be active port channels.

- You must add a space between the first interface number and the hyphen when entering an *interface-range*. For example, **gigabitethernet0/1 - 4** is a valid range; **gigabitethernet0/1-4** is not a valid range.

- The VLAN interfaces must have been configured with the **interface vlan** command. The **show running-config** privileged EXEC command displays the configured VLAN interfaces. VLAN interfaces not displayed by the **show running-config** command cannot be used as *interface-ranges*.

- All interfaces defined as in a range must be the same type (all Gigabit Ethernet ports, all EtherChannel ports, or all VLANs), but you can combine multiple interface types in a macro.

This example shows how to define an interface-range named *enet_list* to include ports 1 and 2 and to verify the macro configuration:

```
Switch# configure terminal
Switch(config)# define interface-range enet_list gigabitethernet0/1 - 2
Switch(config)# end
Switch# show running-config | include define
define interface-range enet_list GigabitEthernet0/1 - 2
```

This example shows how to create a multiple-interface macro named *macro1*:

```
Switch# configure terminal
Switch(config)# define interface-range macro1 gigabitethernet0/1 - 2,
gigabitethernet0/5 - 7
Switch(config)# end
```

This example shows how to enter interface-range configuration mode for the interface-range macro *enet_list*:

```
Switch# configure terminal
Switch(config)# interface range macro enet_list
Switch(config-if-range)#
```

This example shows how to delete the interface-range macro *enet_list* and to verify that it was deleted.

```
Switch# configure terminal
Switch(config)# no define interface-range enet_list
Switch(config)# end
Switch# show run | include define
Switch#
```

Configuring Ethernet Interfaces

These sections contain this configuration information:

Default Ethernet Interface Configuration

Table 10-1 shows the Ethernet interface default configuration, including some features that apply only to Layer 2 interfaces. For more details on the VLAN parameters listed in the table, see Chapter 12, "Configuring VLANs." For details on controlling traffic to the port, see Chapter 24, "Configuring Port-Based Traffic Control."

> **Note** To configure Layer 2 parameters, if the interface is in Layer 3 mode, you must enter the **switchport** interface configuration command without any parameters to put the interface into Layer 2 mode. This shuts down the interface and then re-enables it, which might generate messages on the device to which the interface is connected. When you put an interface that is in Layer 3 mode into Layer 2 mode, the previous configuration information related to the affected interface might be lost, and the interface is returned to its default configuration.

Table 10-1   Default Layer 2 Ethernet Interface Configuration

Feature	Default Setting
Operating mode	Layer 2 or switching mode (**switchport** command).
Allowed VLAN range	VLANs 1 to 4094.
Default VLAN (for access ports)	VLAN 1 (Layer 2 interfaces only).
Native VLAN (for IEEE 802.1Q trunks)	VLAN 1 (Layer 2 interfaces only).
VLAN trunking	Switchport mode dynamic auto (supports DTP) (Layer 2 interfaces only).
Port enable state	All ports are enabled.
Port description	None defined.
Speed	Autonegotiate.
Duplex mode	Autonegotiate.
Flow control	Flow control is set to **receive**: **off**. It is always off for sent packets.
EtherChannel (PAgP)	Disabled on all Ethernet ports. See Chapter 34, "Configuring EtherChannels and Layer 2 Trunk Failover."

Table 10-1   Default Layer 2 Ethernet Interface Configuration (continued)

Feature	Default Setting
Port blocking (unknown multicast and unknown unicast traffic)	Disabled (not blocked) (Layer 2 interfaces only). See the "Configuring Port Blocking" section on page 24-8.
Broadcast, multicast, and unicast storm control	Disabled. See the "Default Storm Control Configuration" section on page 24-3.
Protected port	Disabled (Layer 2 interfaces only). See the "Configuring Protected Ports" section on page 24-6.
Port security	Disabled (Layer 2 interfaces only). See the "Default Port Security Configuration" section on page 24-11.
Port Fast	Disabled. Enabled by default on Gigabit Ethernet interfaces 0/1 to 0/16. See the "Default Optional Spanning-Tree Configuration" section on page 19-9.
Auto-MDIX	Enabled. Note The switch might not support a pre-standard powered device—such as Cisco IP phones and access points that do not fully support IEEE 802.3af—if that powered device is connected to the switch through a crossover cable. This is regardless of whether auto-MIDX is enabled on the switch port.
Keepalive messages	Disabled on SFP module ports; enabled on all other ports.

Setting the Type of a Dual-Purpose Uplink Port

The Cisco Catalyst Blade Switch 3020 for HP supports dual-purpose uplink ports. For more information, see the "Dual-Purpose Uplink Ports" section on page 10-6.

Beginning in privileged EXEC mode, follow these steps to select which dual-purpose uplink to activate so that you can set the speed and duplex. This procedure is optional.

	Command	Purpose
Step 1	configure terminal	Enter global configuration mode.
Step 2	interface *interface-id*	Specify the dual-purpose uplink port to be configured, and enter interface configuration mode.

	Command	Purpose
Step 3	media-type {auto-select	rj45	sfp	internal}	Select the interface and type of a dual-purpose uplink port. These keyword meanings apply on Gigabit Ethernet interfaces 0/17 to 0/20 and 0/23 to 0/24; they do not apply on Gigabit Ethernet interfaces 0/1 to 0/16 or 0/21 to 0/22. The keywords have these meanings: • **auto-select**—The switch dynamically selects the type. When link up is achieved, the switch disables the other type until the active link goes down. When the active link goes down, the switch enables both types until one of them links up. In auto-select mode, the switch configures both types with autonegotiation of speed and duplex (the default). Depending on the type of installed SFP module, the switch might not be able to dynamically select it. For more information, see the information that follows this procedure. **Note** Gigabit Ethernet interfaces gi0/23 and gi0/24 do not support the **media-type** command **auto-select** module option. • **rj45**—The switch disables the SFP module interface. If you connect a cable to this port, it cannot attain a link even if the RJ-45 side is down or is not connected. In this mode, the dual-purpose port behaves like a 10/100/1000BASE-TX interface. You can configure the speed and duplex settings consistent with this interface type. • **sfp**—The switch disables the RJ-45 interface. If you connect a cable to this port, it cannot attain a link even if the SFP module side is down or if the SFP module is not present. Based on the type of installed SFP module, you can configure the speed and duplex settings consistent with this interface type. **Note** Gigabit Ethernet interfaces gi0/23 and gi0/24 do not support the **media-type** command **sfp** module option. • **internal**—This option enables the switch to establish a cross-connection to a switch in an adjoining bay only for Gigabit Ethernet interfaces gi0/23 and gi0/24. For information about setting the speed and duplex, see the "Speed and Duplex Configuration Guidelines" section on page 10-16.
Step 4	end	Return to privileged EXEC mode.
Step 5	show interfaces *interface-id* transceiver properties	Verify your setting.
Step 6	copy running-config startup-config	(Optional) Save your entries in the configuration file.

To return to the default setting, use the **no media-type** interface configuration command.

When you change the interface type, the speed and duplex configurations are removed. The switch configures both types to autonegotiate speed and duplex (the default). If you configure **auto-select**, you cannot configure the **speed** and **duplex** interface configuration commands.

When the switch powers on or when you enable a dual-purpose uplink port through the **shutdown** and the **no shutdown** interface configuration commands, the switch gives preference to the SFP module interface. In all other situations, the switch selects the active link based on which type first links up.

Configuring Interface Speed and Duplex Mode

Ethernet interfaces on the switch operate at 10, 100, or 1000 Mb/s and in either full- or half-duplex mode. In full-duplex mode, two stations can send and receive traffic at the same time. Normally, 10-Mb/s ports operate in half-duplex mode, which means that stations can either receive or send traffic.

Switch models include Gigabit Ethernet (10/100/1000-Mb/s) ports and small form-factor pluggable (SFP) module slots supporting SFP modules.

Gigabit interfaces 0/1 to 0/16 are internal downlink ports to the blade server. The speed for these interfaces are set at 1000 Mbps, and the duplex is set to full; these settings cannot be changed.

These sections describe how to configure the interface speed and duplex mode:

- Speed and Duplex Configuration Guidelines, page 10-16
- Setting the Interface Speed and Duplex Parameters, page 10-17

Speed and Duplex Configuration Guidelines

When configuring an interface speed and duplex mode, note these guidelines:

- Gigabit Ethernet (10/100/1000-Mb/s) ports support all speed options and all duplex options (auto, half, and full). However, Gigabit Ethernet ports operating at 1000 Mb/s do not support half-duplex mode.
- The 1000BASE-SX SFP module ports support the **nonegotiate** keyword in the **speed** interface configuration command. Duplex options are not supported.
- You cannot configure duplex mode on SFP module ports; they operate in full-duplex mode except in these situations:
    - You can configure Cisco 1000BASE-T SFP modules for auto, full, or half-duplex mode.
    - Cisco 1000BASE-SX SFP modules can operate only in full-duplex mode.
- If you are connected to a device that does not support autonegotiation, you can configure speed on copper SFP module ports; however, you can only configure fiber SFP module ports to not negotiate (**nonegotiate**).
- If both ends of the line support autonegotiation, we highly recommend the default setting of **auto** negotiation.
- If one interface supports autonegotiation and the other end does not, configure duplex and speed on both interfaces; do not use the **auto** setting on the supported side.
- When STP is enabled and a port is reconfigured, the switch can take up to 30 seconds to check for loops. The port LED is amber while STP reconfigures.

⚠️

Caution   Changing the interface speed and duplex mode configuration might shut down and re-enable the interface during the reconfiguration.

Setting the Interface Speed and Duplex Parameters

Beginning in privileged EXEC mode, follow these steps to set the speed and duplex mode for a physical interface:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the physical interface to be configured, and enter interface configuration mode.
Step 3	**speed** {**10**	**100**	**1000**	**auto** [**10**	**100**	**1000**]	**nonegotiate**}	Enter the appropriate speed parameter for the interface: • Enter **10**, **100**, or **1000** to set a specific speed for the interface. The **1000** keyword is available only for 10/100/1000 Mb/s ports. • Enter **auto** to enable the interface to autonegotiate speed with the connected device. If you use the **10**, **100**, or the **1000** keywords with the **auto** keyword, the port autonegotiates only at the specified speeds. • The **nonegotiate** keyword is available only for SFP module ports. SFP module ports operate only at 1000 Mb/s but can be configured to not negotiate if connected to a device that does not support autonegotiation. For more information about speed settings, see the "Speed and Duplex Configuration Guidelines" section on page 10-16.
Step 4	**duplex** {**auto**	**full**	**half**}	Enter the duplex parameter for the interface. Enable half-duplex mode (for interfaces operating only at 10 or 100 Mb/s). You cannot configure half-duplex mode for interfaces operating at 1000 Mb/s. For more information about duplex settings, see the "Speed and Duplex Configuration Guidelines" section on page 10-16.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show interfaces** *interface-id*	Display the interface speed and duplex mode configuration.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no speed** and **no duplex** interface configuration commands to return the interface to the default speed and duplex settings (autonegotiate). To return all interface settings to the defaults, use the **default interface** *interface-id* interface configuration command.

This example shows how to set the interface speed to 100 Mb/s on a 10/100/1000 Mb/s port:

```
Switch# configure terminal
Switch(config)# interface gigabitethernet0/21
Switch(config-if)# speed 100
```

Note For interfaces gi0/1 to gi0/16, speed and duplex settings do not apply, as they are only internal server-facing interfaces. For interfaces 17 to 20, speed and duplex do not apply when they are operating in SFP module mode. For interfaces gi0/23 and gi0/24, speed and duplex do not apply when configured for media-type internal. For more information, see the "Internal Gigabit Ethernet Ports" section on page 10-3.

Configuring IEEE 802.3x Flow Control

Flow control enables connected Ethernet ports to control traffic rates during congestion by allowing congested nodes to pause link operation at the other end. If one port experiences congestion and cannot receive any more traffic, it notifies the other port by sending a pause frame to stop sending until the condition clears. Upon receipt of a pause frame, the sending device stops sending any data packets, which prevents any loss of data packets during the congestion period.

Note   Cisco Catalyst Blade Switch 3020 for HP ports can receive, but not send, pause frames.

You use the **flowcontrol** interface configuration command to set the interface's ability to **receive** pause frames to **on**, **off**, or **desired**. The default state is **off**.

When set to **desired**, an interface can operate with an attached device that is required to send flow-control packets or with an attached device that is not required to but can send flow-control packets.

These rules apply to flow control settings on the device:

- **receive on** (or **desired**): The port cannot send pause frames but can operate with an attached device that is required to or can send pause frames; the port can receive pause frames.
- **receive off**: Flow control does not operate in either direction. In case of congestion, no indication is given to the link partner, and no pause frames are sent or received by either device.

Note   For details on the command settings and the resulting flow control resolution on local and remote ports, see the **flowcontrol** interface configuration command in the command reference for this release.

Beginning in privileged EXEC mode, follow these steps to configure flow control on an interface:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode
Step 2	**interface** *interface-id*	Specify the physical interface to be configured, and enter interface configuration mode.
Step 3	**flowcontrol** {**receive**} {**on**	**off**	**desired**}	Configure the flow control mode for the port.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show interfaces** *interface-id*	Verify the interface flow control settings.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable flow control, use the **flowcontrol receive off** interface configuration command.

This example shows how to turn on flow control on a port:

```
Switch# configure terminal
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# flowcontrol receive on
Switch(config-if)# end
```

Configuring Auto-MDIX on an Interface

When automatic medium-dependent interface crossover (auto-MDIX) is enabled on an interface, the interface automatically detects the required cable connection type (straight through or crossover) and configures the connection appropriately. When connecting switches without the auto-MDIX feature, you must use straight-through cables to connect to devices such as servers, workstations, or routers and crossover cables to connect to other switches or repeaters. With auto-MDIX enabled, you can use either type of cable to connect to other devices, and the interface automatically corrects for any incorrect cabling. For more information about cabling requirements, see the hardware installation guide.

Auto-MDIX is enabled by default. When you enable auto-MDIX, you must also set the interface speed and duplex to **auto** so that the feature operates correctly. Auto-MDIX is supported on all 10/100/1000-Mb/s interfaces. It is not supported on 1000BASE-SX SFP module interfaces.

Table 10-2 shows the link states that result from auto-MDIX settings and correct and incorrect cabling.

Table 10-2   Link Conditions and Auto-MDIX Settings

Local Side Auto-MDIX	Remote Side Auto-MDIX	With Correct Cabling	With Incorrect Cabling
On	On	Link up	Link up
On	Off	Link up	Link up
Off	On	Link up	Link up
Off	Off	Link up	Link down

Beginning in privileged EXEC mode, follow these steps to configure auto-MDIX on an interface:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode
Step 2	**interface** *interface-id*	Specify the physical interface to be configured, and enter interface configuration mode.
Step 3	**speed auto**	Configure the interface to autonegotiate speed with the connected device.
Step 4	**duplex auto**	Configure the interface to autonegotiate duplex mode with the connected device.
Step 5	**mdix auto**	Enable auto-MDIX on the interface.
Step 6	**end**	Return to privileged EXEC mode.
Step 7	**show controllers ethernet-controller** *interface-id* **phy**	Verify the operational state of the auto-MDIX feature on the interface.
Step 8	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable auto-MDIX, use the **no mdix auto** interface configuration command.

This example shows how to enable auto-MDIX on a port:

```
Switch# configure terminal
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# speed auto
Switch(config-if)# duplex auto
Switch(config-if)# mdix auto
Switch(config-if)# end
```

Adding a Description for an Interface

You can add a description about an interface to help you remember its function. The description appears in the output of these privileged EXEC commands: **show configuration**, **show running-config**, and **show interfaces**.

Beginning in privileged EXEC mode, follow these steps to add a description for an interface:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the interface for which you are adding a description, and enter interface configuration mode.
Step 3	**description** *string*	Add a description (up to 240 characters) for an interface.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show interfaces** *interface-id* **description** or **show running-config**	Verify your entry.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no description** interface configuration command to delete the description.

This example shows how to add a description on a port and how to verify the description:

```
Switch# config terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)# interface gigabitethernet0/2
Switch(config-if)# description Connects to Marketing
Switch(config-if)# end
Switch# show interfaces gigabitethernet0/2 description
Interface Status        Protocol Description
Gi0/2   admin down     down     Connects to Marketing
```

Configuring Layer 3 Interfaces

The switch supports these types of Layer 3 interfaces:

- SVIs: You should configure SVIs for any VLANs for which you want to route traffic. SVIs are created when you enter a VLAN ID following the **interface vlan** global configuration command. To delete an SVI, use the **no interface vlan** global configuration command. You cannot delete interface VLAN 1.

> **Note** When you create an SVI, it does not become active until it is associated with a physical port. For information about assigning Layer 2 ports to VLANs, see Chapter 12, "Configuring VLANs."

When configuring SVIs, you can also configure SVI autostate exclude on a port in the SVI to exclude that port from being included in determining SVI line-state status. See the "Configuring SVI Autostate Exclude" section on page 10-22.

- Routed ports: Routed ports are physical ports configured to be in Layer 3 mode by using the **no switchport** interface configuration command.

- Layer 3 EtherChannel ports: EtherChannel interfaces made up of routed ports.

  EtherChannel port interfaces are described in Chapter 34, "Configuring EtherChannels and Layer 2 Trunk Failover."

A Layer 3 switch can have an IP address assigned to each routed port and SVI.

There is no defined limit to the number of SVIs and routed ports that can be configured in a switch stack. However, the interrelationship between the number of SVIs and routed ports and the number of other features being configured might have an impact on CPU usage because of hardware limitations. If the switch is using maximum hardware resources, attempts to create a routed port or SVI have these results:

- If you try to create a new routed port, the switch generates a message that there are not enough resources to convert the interface to a routed port, and the interface remains as a switchport.

- If you try to create an extended-range VLAN, an error message is generated, and the extended-range VLAN is rejected.

- If the switch is notified by VLAN Trunking Protocol (VTP) of a new VLAN, it sends a message that there are not enough hardware resources available and shuts down the VLAN. The output of the **show vlan** user EXEC command shows the VLAN in a suspended state.

- If the switch attempts to boot up with a configuration that has more VLANs and routed ports than hardware can support, the VLANs are created, but the routed ports are shut down, and the switch sends a message that this was due to insufficient hardware resources.

All Layer 3 interfaces require an IP address to route traffic. This procedure shows how to configure an interface as a Layer 3 interface and how to assign an IP address to an interface.

> **Note** If the physical port is in Layer 2 mode (the default), you must enter the **no switchport** interface configuration command to put the interface into Layer 3 mode. Entering a **no switchport** command disables and then re-enables the interface, which might generate messages on the device to which the interface is connected. Furthermore, when you put an interface that is in Layer 2 mode into Layer 3 mode, the previous configuration information related to the affected interface might be lost, and the interface is returned to its default configuration

Beginning in privileged EXEC mode, follow these steps to configure a Layer 3 interface:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** {{**fastethernet** | **gigabitethernet**} *interface-id*} | {**vlan** *vlan-id*} | {**port-channel** *port-channel-number*}	Specify the interface to be configured as a Layer 3 interface, and enter interface configuration mode.
Step 3	**no switchport**	For physical ports only, enter Layer 3 mode.
Step 4	**ip address** *ip_address subnet_mask*	Configure the IP address and IP subnet.
Step 5	**no shutdown**	Enable the interface.
Step 6	**end**	Return to privileged EXEC mode.
Step 7	**show interfaces** [*interface-id*] **show ip interface** [*interface-id*] **show running-config interface** [*interface-id*]	Verify the configuration.
Step 8	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove an IP address from an interface, use the **no ip address** interface configuration command.

This example shows how to configure a port as a routed port and to assign it an IP address:

```
Switch# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)# interface gigabitethernet0/2
Switch(config-if)# no switchport
Switch(config-if)# ip address 192.20.135.21 255.255.255.0
Switch(config-if)# no shutdown
```

Configuring SVI Autostate Exclude

Configuring SVI autostate exclude on an access or trunk port in an SVI excludes that port in the calculation of the status of the SVI line state (up or down) even if it belongs to the same VLAN. When the excluded port is in the up state, and all other ports in the VLAN are in the down state, the SVI state is changed to down.

At least one port in the VLAN should be up and not excluded to keep the SVI state up. You can use this command to exclude the monitoring port status when determining the status of the SVI.

Beginning in privileged EXEC mode, follow these steps to exclude a port from SVI state-change calculations:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify a Layer 2 interface (physical port or port channel), and enter interface configuration mode.
Step 3	**switchport autostate exclude**	Exclude the access or trunk port when defining the status of an SVI line state (up or down)
Step 4	**end**	Return to privileged EXEC mode.

	Command	Purpose
Step 5	show running config interface *interface-id*	(Optional) Show the running configuration.
	show interface *interface-id* switchport	Verify the configuration.
Step 6	copy running-config startup-config	(Optional) Save your entries in the configuration file.

This example shows how to configure an access or trunk port in an SVI to be excluded from the status calculation:

```
Switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)# interface gigabitethernet0/2
Switch(config-if)# switchport autostate exclude
Switch(config-if)# exit
```

Configuring the System MTU

The default maximum transmission unit (MTU) size for frames received and transmitted on all interfaces on the switch is 1500 bytes. You can increase the MTU size for all interfaces operating at 10 or 100 Mb/s by using the **system mtu** global configuration command. You can increase the MTU size to support jumbo frames on all Gigabit Ethernet interfaces by using the **system mtu jumbo** global configuration command. You can change the MTU size for routed ports by using the **system mtu routing** global configuration command.

> **Note** You cannot configure a routing MTU size that exceeds the system MTU size. If you change the system MTU size to a value smaller than the currently configured routing MTU size, the configuration change is accepted, but not applied until the next switch reset. When the configuration change takes effect, the routing MTU size automatically defaults to the new system MTU size.

Gigabit Ethernet ports are not affected by the **system mtu** command; 10/100 ports are not affected by the **system mtu jumbo** command. If you do not configure the **system mtu jumbo** command, the setting of the **system mtu** command applies to all Gigabit Ethernet interfaces.

You cannot set the MTU size for an individual interface; you set it for all 10/100 or all Gigabit Ethernet interfaces on the switch. When you change the system or jumbo MTU size, you must reset the switch before the new configuration takes effect.The **system mtu routing** command does not require a switch reset to take effect.

Frames sizes that can be received by the switch CPU are limited to 1998 bytes, no matter what value was entered with the **system mtu** or **system mtu jumbo** commands. Although frames that are forwarded or routed are typically not received by the CPU, in some cases packets are sent to the CPU, such as traffic sent to control traffic, SNMP, Telnet, or routing protocols.

Routed packets are subjected to MTU checks on the output ports. The MTU value used for routed ports is derived from the applied **system mtu** value (not the **system mtu jumbo** value). That is, the routed MTU is never greater than the system MTU for any VLAN. The routing protocols use the system MTU value when negotiating adjacencies and the MTU of the link.To view the MTU value for routed packets for a specific VLAN, use the **show platform port-asic mvid** privileged EXEC command.

> **Note**   If Layer 2 Gigabit Ethernet interfaces are configured to accept frames greater than the 10/100 interfaces, jumbo frames received on a Layer 2 Gigabit Ethernet interface and sent on a Layer 2 10/100 interface are dropped.

Beginning in privileged EXEC mode, follow these steps to change MTU size for all 10/100 or Gigabit Ethernet interfaces:

	Command	**Purpose**
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**system mtu** *bytes*	(Optional) Change the MTU size for all interfaces on the switch that are operating at 10 or 100 Mb/s. The range is 1500 to 1998 bytes; the default is 1500 bytes.
Step 3	**system mtu jumbo** *bytes*	(Optional) Change the MTU size for all Gigabit Ethernet interfaces on the switch. The range is 1500 to 9000 bytes; the default is 1500 bytes.
Step 4	**system mtu routing** *bytes*	(Optional) Change the system MTU for routed ports. The range is 1500 to the system MTU value, the maximum MTU that can be routed for all ports. Although larger packets can be accepted, they cannot be routed.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**copy running-config startup-config**	Save your entries in the configuration file.
Step 7	**reload**	Reload the operating system.

If you enter a value that is outside the allowed range for the specific type of interface, the value is not accepted.

Once the switch reloads, you can verify your settings by entering the **show system mtu** privileged EXEC command.

This example shows how to set the maximum packet size for a Gigabit Ethernet port to 1800 bytes:

```
Switch(config)# system mtu jumbo 1800
Switch(config)# exit
Switch# reload
```

This example shows the response when you try to set Gigabit Ethernet interfaces to an out-of-range number:

```
Switch(config)# system mtu jumbo 25000
                                  ^
% Invalid input detected at '^' marker.
```

Monitoring and Maintaining the Interfaces

These sections contain interface monitoring and maintenance information:

- Monitoring Interface Status, page 10-25
- Clearing and Resetting Interfaces and Counters, page 10-26
- Shutting Down and Restarting the Interface, page 10-26

Monitoring Interface Status

Commands entered at the privileged EXEC prompt display information about the interface, including the versions of the software and the hardware, the configuration, and statistics about the interfaces. Table 10-3 lists some of these interface monitoring commands. (You can display the full list of **show** commands by using the **show ?** command at the privileged EXEC prompt.) These commands are fully described in the *Cisco IOS Interface Command Reference, Release 12.2*.

Table 10-3   Show Commands for Interfaces

Command	Purpose
show interfaces [*interface-id*]	Display the status and configuration of all interfaces or a specific interface.
show interfaces *interface-id* **status** [**err-disabled**]	Display interface status or a list of interfaces in an error-disabled state.
show interfaces [*interface-id*] **switchport**	Display administrative and operational status of switching (nonrouting) ports. You can use this command to find out if a port is in routing or in switching mode.
show interfaces [*interface-id*] **description**	Display the description configured on an interface or all interfaces and the interface status.
show ip interface [*interface-id*]	Display the usability status of all interfaces configured for IP routing or the specified interface.
show interface [*interface-id*] **stats**	Display the input and output packets by the switching path for the interface.
show interfaces transceiver properties	(Optional) Display speed and duplex settings on the interface.
show interfaces [*interface-id*] [{**transceiver properties**	**detail**}] *module number*	Display physical and operational status about an SFP module.
show running-config interface [*interface-id*]	Display the running configuration in RAM for the interface.
show version	Display the hardware configuration, software version, the names and sources of configuration files, and the bootup images.
show controllers ethernet-controller *interface-id* **phy**	Display the operational state of the auto-MDIX feature on the interface.

Clearing and Resetting Interfaces and Counters

Table 10-4 lists the privileged EXEC mode **clear** commands that you can use to clear counters and reset interfaces.

Table 10-4   Clear Commands for Interfaces

Command	Purpose
clear counters [*interface-id*]	Clear interface counters.
clear interface *interface-id*	Reset the hardware logic on an interface.
clear line [*number* | **console 0** | **vty** *number*]	Reset the hardware logic on an asynchronous serial line.

To clear the interface counters shown by the **show interfaces** privileged EXEC command, use the **clear counters** privileged EXEC command. The **clear counters** command clears all current interface counters from the interface unless you specify optional arguments that clear only a specific interface type from a specific interface number.

Note The **clear counters** privileged EXEC command does not clear counters retrieved by using Simple Network Management Protocol (SNMP), but only those seen with the **show interface** privileged EXEC command.

Shutting Down and Restarting the Interface

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 is not mentioned in any routing updates.

Beginning in privileged EXEC mode, follow these steps to shut down an interface:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** {**vlan** *vlan-id*} | {{**fastethernet** | **gigabitethernet**} *interface-id*} | {**port-channel** *port-channel-number*}	Select the interface to be configured.
Step 3	**shutdown**	Shut down an interface.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show running-config**	Verify your entry.

Use the **no shutdown** interface configuration command to restart the interface.

To verify that an interface is disabled, enter the **show interfaces** privileged EXEC command. A disabled interface is shown as *administratively down* in the display.