CHAPTER 10

Configuring Interfaces

This chapter defines the types of interfaces on the Cisco ME 3400E Ethernet Access switch and describes how to configure them.

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.

UNI, NNI, and ENI Port Types

The Cisco ME switch supports user-network interfaces (UNIs), network node interfaces (NNIs), and enhanced network interfaces (ENIs). UNIs are typically connected to a host, such as a PC or a Cisco IP phone. NNIs are typically connected to a router or to another switch. ENIs have the same functionality as UNIs, but can be configured to support protocol control packets for Cisco Discovery Protocol (CDP), Spanning-Tree Protocol (STP), Link Layer Discovery Protocol (LLDP), and EtherChannel Link Aggregation Control Protocol (LACP) or Port Aggregation Protocol (PAgP).

By default, the 10/100 ports and the dual-purpose ports are configured as UNIs, and the SFP-only module uplink ports are configured as NNIs. No ports are ENIs by default.

Note On the Cisco ME 3400E-24TS-M switch, the dual-purpose ports serve as the uplink ports and are NNIs by default.

If the switch is running the metro access image, only four ports on the switch can be configured as NNIs at one time. If the switch is running the metro IP access image, there is no limit to the number of NNIs that can be configured on the switch. All ports on the switch can be configured as UNIs or ENIs.

The default state for a UNI or ENI is administratively down to prevent unauthorized users from gaining access to other ports as you configure the switch. Traffic is not switched between these ports, and all arriving traffic at UNIs or ENIs must leave on NNIs to prevent a user from gaining access to another user's private network. If it is appropriate for two or more UNIs or ENIs to exchange traffic within the switch, the UNIs and ENIs can be assigned to a community VLAN. See Chapter 12, "Configuring VLANs," for instructions on how to configure community VLANs.

Note Even though the default state for a UNI or ENI is shutdown, entering the **default interface** *interface-id* command changes the port to the enabled state.

The default status for an NNI is administratively up to allow a service provider remote access to the switch during initial configuration.

A port can be reconfigured from UNI to NNI or ENI and the reverse. When a port is reconfigured as another interface type, it inherits all the characteristics of that interface type. When you reconfigure a UNI or ENI to be an NNI, you must enable the port before it becomes active.

Changing the port type from UNI to ENI does not affect the administrative state of the port. If the UNI status is shut down, it remains shut down when reconfigured as an ENI; if the port is in a no shutdown state, it remains in the no shutdown state. At any time, all ports on the Cisco ME switch are either UNI, NNI, or ENI.

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 associated with the VLAN ID or when a user creates te VLAN ID.

To isolate VLANs of different customers in a service-provider network, the Cisco ME switch uses UNI-ENI VLANs. UNI-ENI VLANs isolate user network interfaces (UNIs) or enhanced network interfaces (ENIs) on the switch from UNIs or ENIs that belong to other customer VLANs. There are two types of UNI-ENI VLANs:

- UNI-ENI isolated VLAN—This is the default VLAN state for all VLANs created on the switch. Local switching does not occur among UNIs or ENIs on the switch that belong to the same UNI-ENI isolated VLAN.
- UNI-ENI community VLAN—Local switching is allowed among UNIs and ENIs on the switch that belong to the same UNI community VLAN. If UNIs or ENIs belong to the same customer, and you want to switch packets between the ports, you can configure the common VLAN as a UNI-ENI community VLAN.

Note Local switching takes place between ENIs and UNIs in the same community VLAN. Because you can enable spanning tree on ENIs, but not on UNIs, you should use caution when configuring ENIs and UNIs in the same community VLAN. UNIs are always in the forwarding state.

For more information about UNI VLANs, see the "UNI-ENI VLANs" section on page 12-5.

To configure VLANs, use the **vlan** *vlan-id* global configuration command to enter VLAN configuration mode. The VLAN configurations for VLAN IDs 1 to 1005 are saved in the VLAN database. Extended-range VLANs (VLAN IDs 1006 to 4094) are not added to the VLAN database. VLAN configuration is saved in the switch running configuration, and you can save it in the switch startup configuration file by entering the **copy running-config startup-config** privileged EXEC command.

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 14, "Configuring IEEE 802.1Q Tunneling, VLAN Mapping, 802.1ad, 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, a private-VLAN port, or a tunnel port. You can configure a port as an access port or trunk port. You configure a private VLAN port as a host or promiscuous port that belongs to a private-VLAN primary or secondary VLAN. (Only NNIs can be configured as promiscuous ports.) 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 or bridging.

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 14, "Configuring IEEE 802.1Q Tunneling, VLAN Mapping, 802.1ad, and Layer 2 Protocol Tunneling."

Access Ports

An access port belongs to and carries the traffic of only one VLAN. 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 an IEEE 802.1Q tagged packet, the packet is dropped, and the source address is not learned. IEEE 802.1x can also be used for VLAN assignment.

Two types of access ports are supported:

- Static access ports are manually assigned to a VLAN.
- VLAN membership of dynamic access ports is learned through incoming packets. By default, a dynamic access port is a member of no VLAN, and forwarding to and from the port is enabled only when the VLAN membership of the port is discovered. UNIs begin forwarding packets as soon as they are enabled. 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 ME switch cannot be a VMPS server. Dynamic access ports for VMPS are only supported on UNIs and ENIs.

Trunk Ports

An IEEE 802.1Q trunk port carries the traffic of multiple VLANs and by default is a member of all VLANs in the VLAN database. A 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 multiple VLANs, 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 the VLAN is in the enabled state.

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 14, "Configuring IEEE 802.1Q Tunneling, VLAN Mapping, 802.1ad, 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 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 10-25 for information about what happens when hardware resource limitations are reached.

For more information about IP unicast and multicast routing and routing protocols, see Chapter 36, "Configuring IP Unicast Routing" and Chapter 44, "Configuring IP Multicast Routing."

Note For full Layer 3 routing, you must have the metro IP access image installed on the switch

Ethernet Management Port

The Ethernet management port, also referred to as the *Fa0 or fastethernet0 port*, is a Layer 3 host port to which you can connect a PC. You can use the Ethernet management port instead of the switch console port for network management.

See the "Using the Ethernet Management Port" section on page 10-12 for more information.

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 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 of 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 10-25 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 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. For more information about configuring IP routing, see Chapter 36, "Configuring IP Unicast Routing," and Chapter 44, "Configuring IP Multicast Routing."

Note Routed ports (or SVIs) are supported only when the metro IP access image is installed on the switch.

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, group multiple access ports into one logical access port, group multiple tunnel ports into one logical tunnel port, or group multiple routed ports into one logical routed 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 Cisco Discovery Protocol (CDP), Link Aggregation Control Protocol (LACP), and the Port Aggregation Protocol (PAgP), which operate only on physical NNI or ENI 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 35, "Configuring EtherChannels and Link-State Tracking."

Dual-Purpose Ports

Each dual-purpose port is considered as a single interface with dual front ends (an RJ-45 connector and an SFP module connector). The dual front ends are not redundant interfaces; the switch activates only one connector of the pair.

By default, dual-purpose ports are user-network interfaces (UNIs) and SFP-only module ports are network node interfaces (NNIs). By default, the switch dynamically selects the dual-purpose port media 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 a dual-purpose port, see the "Configuring a Dual-Purpose Port" section on page 10-21.

Each dual-purpose port has two LEDs: one shows the status of the SFP module port, and one shows the status of the RJ-45 port. The port LED is on for whichever connector is active. For more information about the LEDs, see the hardware installation guide.

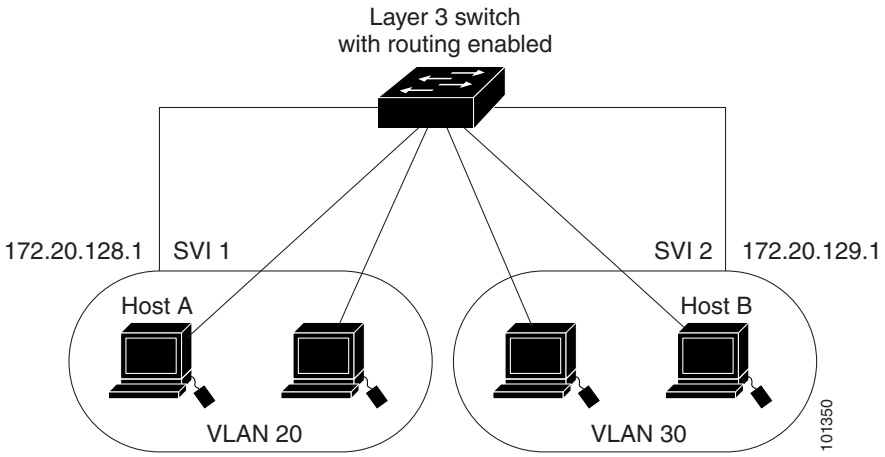
Connecting Interfaces

Devices within a single VLAN can communicate directly through any switch. Ports in different VLANs cannot exchange data without going through a routing device. With a standard Layer 2 switch, ports in different VLANs have to exchange information through a router.

By default, the Cisco ME switch provides VLAN isolation between UNIs or ENIs. UNIs and ENIs cannot exchange traffic unless they are changed to NNIs or assigned to a UNI-ENI community VLAN.

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 (Figure 10-1).

Figure 10-1 Connecting VLANs with the Switch



When the metro IP access image is running on the switch, routing can be enabled on the switch. 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. 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 36, "Configuring IP Unicast Routing," Chapter 44, "Configuring IP Multicast Routing," and Chapter 45, "Configuring MSDP."

Using Interface Configuration Mode

The switch supports these interface types:

- Physical ports—switch ports, routed ports, UNIs, NNIs, and ENIs
- 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 10-9).

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

- Type—Fast Ethernet (fastethernet or fa) for 10/100 Mb/s Ethernet, Gigabit Ethernet (gigabitethernet or gi) for 10/100/1000 Mb/s Ethernet ports, 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 ME switch).
- Port number—The interface number on the switch. The port numbers always begin at 1, starting with the leftmost port when facing the front of the switch, for example, fastethernet 0/1 or gigabitethernet 0/1. If there is more than one interface type (for example, 10/100 ports and SFP module ports), the port numbers restart with the second interface type: gigabitethernet 0/1.

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, Fast Ethernet port 1 is selected:

```
Switch(config)# interface fastethernet0/1
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 **fastethernet 0/1**, **fastethernet0/1**, **fa 0/1**, or **fa0/1**.

Step 3 If you are configuring a UNI or ENI, enter the **no shutdown** interface configuration command to enable the interface:

```
Switch(config-if)# no shutdown
```

Step 4 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 5 After you configure an interface, verify its status by using the **show** privileged EXEC commands listed in the "Monitoring and Maintaining the Interfaces" section on page 10-30.

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*}	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. • 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	**no shutdown**	Enable the port, if necessary. By default, UNIs and ENIs are disabled, and NNIs are enabled.
Step 4		You can now use the normal configuration commands to apply the configuration parameters to all interfaces in the range.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show interfaces** [*interface-id*]	Verify the configuration of the interfaces in the range.
Step 7	**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
  - **fastethernet** module/{first *port*} - {*last port*}, where the module is always 0

  - **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.

- 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 as in a range must be the same type (all Fast Ethernet ports, 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 and 2 to 100 Mb/s:

```
Switch# configure terminal
Switch(config)# interface range fastethernet0/1 - 2
Switch(config-if-range)# no shutdown
Switch(config-if-range)# speed 100
```

This example shows how to use a comma to add different interface type strings to the range to enable Fast Ethernet ports 1 to 3 and Gigabit Ethernet ports 1 and 2 to receive IEEE 802.3x flow control pause frames:

```
Switch# configure terminal
Switch(config)# interface range fastethernet0/1 - 3 , gigabitethernet0/1 - 2
Switch(config-if-range)# flowcontrol receive on
```

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 together 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	**no shutdown**	Enable the port, if necessary. By default, UNIs and ENIs are disabled, and NNIs are enabled.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show running-config | include define**	Show the defined interface range macro configuration.
Step 7	**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
  - **fastethernet** module/{first *port*} - {*last port*}, where the module is always 0
  - **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 - 2** is a valid range; **gigabitethernet0/1-2** 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 Fast Ethernet ports, 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* and assign all of the interfaces in the range to a VLAN:

```
Switch# configure terminal
Switch(config)# define interface-range macro1 fastethernet0/1 - 2, gigabitethernet0/1 - 2
Switch(config)# interface range macro macro1
Switch(config-if-range)# switchport access vlan 20
Switch(config-if-range)# no shut
Switch(config-if-range)# 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#
```

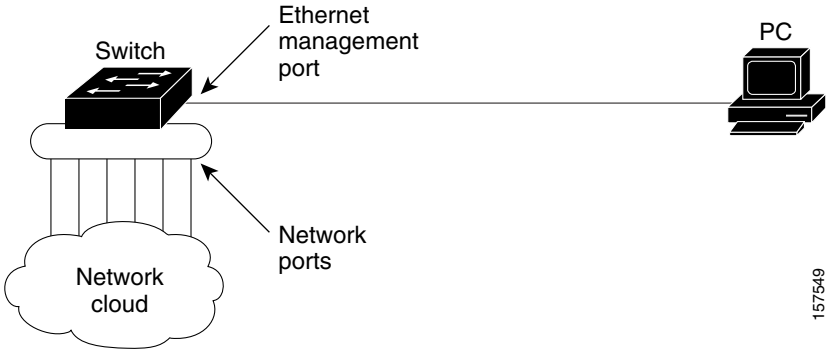
Using the Ethernet Management Port

Understanding the Ethernet Management Port

The Ethernet management port is port *Fa0 or fastethernet0,* a Layer 3 host port to which you can connect a PC. You can use the Ethernet management port instead of the switch console port for network management.

When connecting a PC to the Ethernet management port, you must assign an IP address. Connect the Ethernet management port to the PC as shown in Figure 10-2.

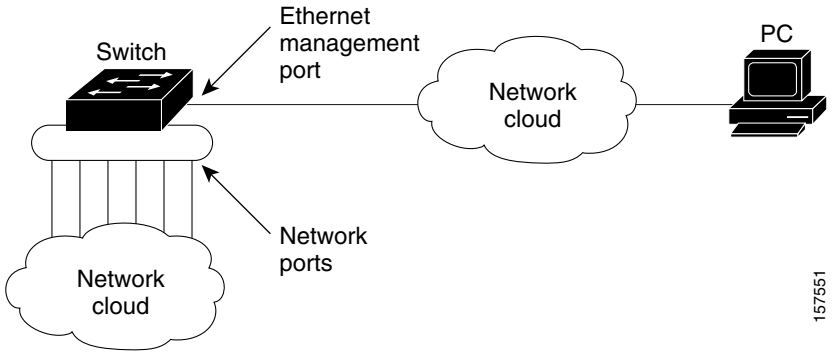
Figure 10-2 Connecting a Switch to a PC



By default, the Ethernet management port is enabled. The switch cannot route packets from the Ethernet management port to a network port or receive routed packets from a network port.

Even though the Ethernet management port does not support routing, you might need to enable routing protocols on the port. For example, in Figure 10-3, you must enable routing protocols on the Ethernet management port when the PC is two or more hops away from the switch and the packets must pass through two or more Layer 3 devices to reach the PC.

Figure 10-3 Network Example with Routing Protocols Enabled



In Figure 10-3, if the Ethernet management port and the network ports are associated with the same routing process, the routes are propagated in this manner:

- The routes from the Ethernet management port are propagated through the network ports to the network.
- The routes from the network ports are propagated through the Ethernet management port to the network.

Because routing is not supported between the Ethernet management port and the network ports, traffic cannot be sent or received between these ports. If traffic is sent or received, data packet loops occur between the ports, which disrupts the switch and network operation. Configure route filters to avoid routes between the Ethernet management port and the network ports and to prevent the loops.

Supported Features on the Ethernet Management Port

The Ethernet management port supports these features:

- Telnet with passwords
- TFTP
- Secure Shell (SSH)
- DHCP-based autoconfiguration
- SMNP (only the ENTITY-MIB and the IF-MIB)
- IP ping
- Interface features
- Speed—10 Mb/s, 100 Mb/s, and autonegotiation
- Duplex mode—Full, half, and autonegotiation
- Loopback detection
- Cisco Discovery Protocol (CDP)
- DHCP relay agent
- IPv4 access control lists (ACLs)
- Routing protocols

Caution Before enabling a feature on the Ethernet management port, make sure that the feature is supported. If you try to configure an unsupported feature on the Ethernet Management port, the feature might not work properly, and the switch might fail.

Configuring the Ethernet Management Port

To specify the Ethernet management port in the CLI, enter **fastethernet0**.

To disable the port, use the **shutdown** interface configuration command. To enable the port, use the **no shutdown** interface configuration command.

You can monitor the Ethernet management port LED to determine the link status to the PC. The LED is green (on) when the link is active, and the LED is off when the link is down. The LED is amber when there is a POST failure.

To display the link status, use the **show interfaces fastethernet 0** privileged EXEC command.

TFTP and the Ethernet Management Port

Use the commands in Table 1 when using TFTP to download or upload a configuration file to the boot loader.

Table 1 Boot Loader Commands

Command	Description
arp [*ip_address*]	Displays the currently cached ARP[1] table when this command is entered without the *ip_address* parameter. Enables ARP to associate a MAC address with the specified IP address when this command is entered with the *ip_address* parameter.
mgmt_clr	Clears the statistics for the Ethernet management port.
mgmt_init	Starts the Ethernet management port.
mgmt_show	Displays the statistics for the Ethernet management port.
ping *host_ip_address*	Sends ICMP ECHO_REQUEST packets to the specified network host.
boot tftp:*/file-url ...*	Loads and boots an executable image from the TFTP server and enters the command-line interface. For more details, see the command reference for this release.
copy tftp:/*source-file-url* *filesystem*:/*destination-file-url*	Copies a Cisco IOS image from the TFTP server to the specified location. For more details, see the command reference for this release.

1. ARP = Address Resolution Protocol.

Configuring Ethernet Interfaces

Default Ethernet Interface Configuration

Table 10-2 shows the Ethernet interface default configuration for NNIs, and Table 10-3 shows the Ethernet interface default configuration for UNIs and ENIs. 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 23, "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 10-2 Default Ethernet Configuration for NNIs

Feature	Default Setting
Operating mode	Layer 2 or switching mode (**switchport** command).
Allowed VLAN range	VLANs 1– 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 access (Layer 2 interfaces only).
Port enable state	Enabled.
Port description	None defined.
Speed	Autonegotiate.
Duplex mode	Autonegotiate.
IEEE 802.3x flow control	Flow control is set to **receive**: **off**. It is always off for sent packets.
EtherChannel	Disabled on all Ethernet ports. See Chapter 35, "Configuring EtherChannels and Link-State Tracking."
Port blocking (unknown multicast and unknown unicast traffic)	Disabled (not blocked) (only Layer 2 interfaces). See the "Configuring Port Blocking" section on page 23-6.
Broadcast, multicast, and unicast storm control	Disabled. See the "Default Storm Control Configuration" section on page 23-3.
Port security	Disabled (only Layer 2 interfaces). See the "Default Port Security Configuration" section on page 23-10.
Port Fast	Disabled. See the "Default Optional Spanning-Tree Configuration" section on page 17-5.
Auto-MDIX	Enabled.
Cisco Discovery Protocol (CDP)	Enabled.
VMPS	Not configured.

Table 10-3 Default Ethernet Configuration for UNIs and ENIs

Feature	Default Setting
Operating mode	Layer 2 or switching mode (**switchport** command).
Allowed VLAN range	VLANs 1– 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 access (Layer 2 interfaces only).
Dynamic VLAN	Enabled.
Port enable state	Disabled when no configuration file exists.
Port description	None defined.
Speed	Autonegotiate.
Duplex mode	Autonegotiate.

Table 10-3 Default Ethernet Configuration for UNIs and ENIs (continued)

Feature	Default Setting
IEEE 802.3x flow control	Flow control is set to **receive**: **off**. It is always off for sent packets.
EtherChannel	Disabled on all Ethernet ports. See Chapter 35, "Configuring EtherChannels and Link-State Tracking."
Port blocking (unknown multicast and unknown unicast traffic)	Disabled (not blocked) (only Layer 2 interfaces). See the "Configuring Port Blocking" section on page 23-6.
Broadcast, multicast, and unicast storm control	Disabled. See the "Default Storm Control Configuration" section on page 23-3.
Port security	Disabled (only Layer 2 interfaces). See the "Default Port Security Configuration" section on page 23-10.
Auto-MDIX	Enabled.

Configuring the Port Type

By default, all the 10/100 ports on the Cisco ME switch are configured as UNIs, and the SFP module ports are configured as NNIs. You can also configure the port type as ENI. An ENI has the same characteristics as a UNI, but it can be configured to support CDP, STP, LLDP, and Etherchannel LACP and PAgP.

You use the **port-type** interface configuration command to change the port types. If the switch is running the metro access image, only four ports on the switch can be configured as NNIs at one time.If the switch is running the metro IP access image, there is no limit to the number of NNIs that can be configured on the switch. All ports on the switch can be configured as UNIs or ENIs.

When a port is changed from an NNI to a UNI or ENI, it inherits the configuration of the assigned VLAN, either in isolated or community mode. For more information about configuring UNI-ENI isolated and UNI-ENI community VLANs, see Chapter 12, "Configuring VLANs."

When you change a port from NNI to UNI or ENI or the reverse, any features exclusive to the port type revert to the default configuration. For Layer 2 protocols, such as STP, CDP, and LLDP, the default for UNIs and ENIs is disabled (although they can be enabled on ENIs) and the default for NNIs is enabled.

Note By default, the switch sends keepalive messages on UNI s and ENIs and does not send keepalive messages on NNIs. Changing the port type from UNI or ENI to NNI or from NNI to UNI or ENI has no effect on the keepalive status. You can change the keepalive state from the default setting by entering the [**no**] **keepalive** interface configuration command. If you enter the **keepalive** command with no arguments, keepalive packets are sent with the default time interval (10 seconds) and number of retries (5). Entering the **no keepalive** command disables keepalive packets on the interface.

Beginning in privileged EXEC mode, follow these steps to configure the port type on an interface:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode
Step 2	**interface** *interface-id*	Specify the interface to configure, and enter interface configuration mode.
Step 3	**no shutdown**	Enable the port, if necessary. By default, UNIs and ENIs are disabled, and NNIs are enabled.

	Command	Purpose
Step 4	**port-type** {**eni** | **nni** | **uni**}	Change a port to an ENI, NNI, or UNI.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show interfaces** *interface-id*	Verify the interface IEEE 802.3x flow control settings.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Entering the **no port-type** or **default port-type** interface configuration command returns the port to the default state: UNI for Fast Ethernet ports and NNI for Gigabit Ethernet ports.

This example shows how to change a port from a UNI to an NNI and save it to the running configuration.

```
Switch# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)# interface fastethernet0/1
Switch(config-if)# port-type nni
Switch(config-if)# no shutdown
5d20h: %SYS-5-CONFIG_I: Configured from console by console
Switch(config-if)# end
Switch# copy running-config startup-config
```

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 combinations of Fast Ethernet (10/100-Mb/s) ports, Gigabit Ethernet (10/100/1000-Mb/s) ports, and small form-factor pluggable (SFP) module slots supporting SFP modules.

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

- Speed and Duplex Configuration Guidelines, page 10-18
- Setting the Interface Speed and Duplex Parameters, page 10-19

Speed and Duplex Configuration Guidelines

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

- You can configure interface speed on Fast Ethernet (10/100-Mb/s) and Gigabit Ethernet (10/100/1000-Mb/s) ports. You can configure Fast Ethernet ports to full-duplex, half-duplex, or to autonegotiate mode. You can configure Gigabit Ethernet ports to full-duplex mode or to autonegotiate. You also can configure Gigabit Ethernet ports to half-duplex mode if the speed is 10 or 100 Mb/s. Half-duplex mode is not supported on Gigabit Ethernet ports operating at 1000 Mb/s.
- With the exception of when 1000BASE-T SFP modules are installed in the SFP module slots, you cannot configure speed on SFP module ports, but you can configure speed to not negotiate (**nonegotiate**) if connected to a device that does not support autonegotiation.

  However, when a 1000BASE-T SFP module is in the SFP module slot, you can configure speed as 10, 100, or 1000 Mb/s, or auto, but not as **nonegotiate**.

  On a 100BASE-FX SFP module, you cannot configure the speed as **nonegotiate**.

- You cannot configure duplex mode on SFP module ports; they operate in full-duplex mode except in these situations:
  - When a Cisco1000BASE-T SFP module is in the SFP module slot, you can configure duplex mode to **auto** or **full**. Half-duplex mode is supported with the **auto** setting.
  - When a Cisco100BASE-FX SFP module is in the SFP module slot, you can configure duplex mode to **half** or **full**. Although the **auto** keyword is available, it puts the interface in half-duplex mode (the default for this SFP module) because the 100BASE-FX SFP module does not support autonegotiation.
- 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. On the Cisco ME switch, STP is supported on NNIs by default and can be enabled on ENIs. UNIs do not support STP.

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.

Note On dual-purpose ports, changing the interface type by entering the **media-type** interface configuration command removes the speed and duplex configurations. See the "Configuring a Dual-Purpose Port" section on page 10-21 for information about speed and duplex setting on these ports.

	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	**no shutdown**	Enable the port, if necessary. By default, UNIs and ENIs are disabled, and NNIs are enabled.

	Command	Purpose
Step 4	**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 or SFP module ports with a 1000BASE-T SFP module. • 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. **Note** When a Cisco1000BASE-T SFP module is in the SFP module slot, the speed can be configured to **10**, **100**, **1000**, or to **auto**, but not to **nonegotiate**.
Step 5	**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. You can configure the duplex setting when the speed is set to **auto**. This command is not available on SFP module ports with these exceptions: • If a Cisco 1000BASE-T SFP module is inserted, you can configure duplex to **auto** or to **full**. • If a Cisco 100BASE-FX SFP module is inserted, you can configure duplex to **full** or to **half**. Although the **auto** keyword is available, it puts the interface in half-duplex mode (the default).
Step 6	**end**	Return to privileged EXEC mode.
Step 7	**show interfaces** *interface-id*	Display the interface speed and duplex mode configuration.
Step 8	**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 10 Mb/s and the duplex mode to half on a 10/100 Mb/s port:

```
Switch# configure terminal
Switch(config)# interface fasttethernet0/3
Switch(config-if)# no shutdown
Switch(config-if)# speed 10
Switch(config-if)# duplex half
```

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 gigabitethernet0/2
Switch(config-if)# speed 100
```

Configuring a Dual-Purpose Port

Some ports on the switches are dual-purpose ports that can be configured as 10/100/100 ports or as small form-factor pluggable (SFP) module ports. Each dual-purpose port is considered as a single interface with dual front ends (an RJ-45 connector and an SFP module connector).

Note Even when operating at 10 or 100 Mb/s, the dual-purpose ports (and the SFP-only module ports) use the frame size that is set with the **system mtu jumbo** global configuration command.

Each dual-purpose port is considered as a single interface with dual front ends (an RJ-45 connector and an SFP module connector). The dual front ends are not redundant interfaces; the switch activates only one connector of the pair.

By default, the dual-purpose ports are user-network interfaces (UNIs) and the SFP-only module ports are network node interfaces (NNIs).

Note An exception is the Cisco ME 3400E-24TS-M switch, where the dual-purpose ports serve as the uplink ports and are NNIs by default.

If the switch is running the metro IP access image, you can configure any number of ports as NNIs. If the switch is running the metro access image, you can configure only four ports as NNIs.

By default, the switch dynamically selects the dual-purpose port media 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. In **auto-select** mode, the switch gives preference to SFP mode if both copper and fiber-optic signals are simultaneously detected.

Beginning in privileged EXEC mode, follow these steps to select which dual-purpose media type to activate. This procedure is optional.

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

	Command	Purpose
Step 3	**media-type** {**auto-select** | **rj45** | **sfp**}	Select the active interface and media type of a dual-purpose port. The keywords have these meanings: • **auto-select**—The switch dynamically selects the media type. This is the default. When a linkup 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). • **rj45**—The switch disables the SFP module interface. If you connect a cable to the SFP 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 the RJ-45 port, it cannot attain a link even if the SFP 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.
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.

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

When you configure **sfp** or **rj45** media type, the non-configured type is disabled, even if there is a connector installed in that interface and no connector in the configured one.

When the media type is **auto-select**, the switch uses these criteria to select the type:

Note An SFP is not *installed* until it has a fiber-optic or copper cable plugged in.

- If only one connector is installed, that interface is active and remains active until the media is removed or the switch is reloaded.
- If you install both types of media in an enabled dual-purpose port, the switch selects the active link based on which type is installed first.
- If both media are installed in the dual-purpose port, and the switch is reloaded or the port is disabled and then reenabled through the **shutdown** and the **no shutdown** interface configuration commands, the switch gives preference to the SFP module interface.

See the **media-type** interface configuration command in the command reference for more information.

Configuring IEEE 802.3x Flow Control

IEEE 802.3x 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 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 IEEE 802.3x 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**: IEEE 802.3x 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 IEEE 802.3x 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 IEEE 802.3x 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	**no shutdown**	Enable the port, if necessary. By default, UNIs and ENIs are disabled, and NNIs are enabled.
Step 4	**flowcontrol** {**receive**} {**on** | **off** | **desired**}	Configure the IEEE 802.3x flow control mode for the port.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show interfaces** *interface-id*	Verify the interface IEEE 802.3x flow control settings.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable IEEE 802.3x flow control, use the **flowcontrol receive off** interface configuration command.

This example shows how to enable IEEE 802.3x 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 speed and duplex on the interface to **auto** so that the feature operates correctly. Auto-MDIX is supported on all 10/100 and 10/100/1000 Mb/s interfaces and on Cisco 10/100/1000 BASE-T/TX SFP module interfaces. It is not supported on 1000 BASE-SX or -LX SFP module interfaces.

Table 10-4 shows the link states that result from auto-MDIX settings and correct and incorrect cabling.

Table 10-4 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	**no shutdown**	Enable the port, if necessary. By default, UNIs and ENIs are disabled, and NNIs are enabled.
Step 4	**speed auto**	Configure the interface to autonegotiate speed with the connected device.
Step 5	**duplex auto**	Configure the interface to autonegotiate duplex mode with the connected device.
Step 6	**mdix auto**	Enable auto-MDIX on the interface.
Step 7	**end**	Return to privileged EXEC mode.
Step 8	**show controllers ethernet-controller** *interface-id* **phy**	Verify the operational state of the auto-MDIX feature on the interface.
Step 9	**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)# no shutdown
```

```
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
Gi 0/2   admin down     down     Connects to Marketing
```

Configuring Layer 3 Interfaces

The switch must be running the metro IP access image to support Layer 3 interfaces. The Cisco ME 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."

- 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 35, "Configuring EtherChannels and Link-State Tracking."

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. 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 switch port.
- If you try to create an extended-range VLAN, an error message is generated, and the extended-range VLAN is rejected.
- 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 shutdown**	Enable the port, if necessary. By default, UNIs and ENIs are disabled, and NNIs are enabled.
Step 4	**no switchport**	For physical ports only, enter Layer 3 mode.
Step 5	**ip address** *ip_address subnet_mask*	Configure the IP address and IP subnet.
Step 6	**no shutdown**	Enable the interface.
Step 7	**end**	Return to privileged EXEC mode.

	Command	Purpose
Step 8	**show interfaces** [*interface-id*] **show ip interface** [*interface-id*] **show running-config interface** [*interface-id*]	Verify the configuration.
Step 9	**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
```

Configuring the System MTU

The default maximum transmission unit (MTU) size for frames received and sent 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. Starting with Cisco IOS Release 12.2(55)SE, you can set an alternate MTU size to be applied so specific interfaces by using the **system mtu** *alternate* global configuration command. You can define only one alternate MTU size on the switch, but you can apply it to multiple interfaces.

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 MTU size is configured by the **system mtu jumbo** command. Fast Ethernet ports are not affected by this command because jumbo frames are not supported on 10/100 interfaces, including 100BASE-FX and 100BASE-BX SFP modules. If you do not configure the **system mtu jumbo** command, the setting of the **system mtu** command applies to all Gigabit Ethernet interfaces.

You do not 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 MTU size, you must reload the switch before the new configuration takes effect. The **system mtu routing** command does not require a switch reset to take effect.

To define an alternate MTU size, enter the **system mtu alternate** *bytes* global configuration command. You then apply the alternate size to specified interfaces by using the **system mtu alternate interface** *interface-id* global configuration command. The range of the alternate MTU is between the configured **system mtu** and **system jumbo mtu** sizes (1500 to 9000 bytes). When you apply an alternate MTU size to an interface, frames received on the interface that are greater than the configured size are dropped. You can configure an alternate MTU size for Fast Ethernet or Gigabit Ethernet interfaces, but you cannot apply an alternate MTU size greater than 1998 bytes on a Fast Ethernet interface. The alternate MTU size has no effect on the routing MTU size.

When you configure an alternate MTU size, you must reload the switch before the configuration takes effect.

Note The system MTU setting is saved in the switch environmental variable in NVRAM and becomes effective when the switch reloads. The MTU settings you enter with the **system mtu** and **system mtu jumbo** commands are not saved in the switch configuration file, even if you enter the **copy running-config startup-config** privileged EXEC command. Therefore, if you use TFTP to configure a new switch by using a backup configuration file and want the system MTU to be other than the default, you must explicitly configure the **system mtu** and **system mtu jumbo** settings on the new switch and then reload the switch.

Frames sizes that can be received by the switch CPU are limited to 1998 bytes, no matter what value you entered with the **system mtu** or **system mtu jumbo** commands. Although frames that are forwarded 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.

Because the switch does not fragment packets, it drops:

- switched packets larger than the packet size supported on the *egress* interface
- routed packets larger than the routing MTU value

For example, if the **system mtu** value is 1998 bytes and the **system mtu jumbo** value is 5000 bytes, packets up to 5000 bytes can be received on interfaces operating at 1000 Mb/s. However, although a packet larger than 1998 bytes can be received on an interface operating at 1000 Mb/s, if its destination interface is operating at 10 or 100 Mb/s, the packet is dropped.

Routed packets are subjected to MTU checks on the sending ports. The MTU value used for routed ports is derived from the configured **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. For example, the Open Shortest Path First (OSPF) protocol uses this MTU value before setting up an adjacency with a peer router. 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 the MTU size for all 10/100 or Gigabit Ethernet interfaces and to set an alternate MTU size for specified 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.

	Command	Purpose
Step 4	**system mtu alternate** *bytes*	(Optional) Set an alternate MTU size. The range is between the configured system MTU size and the configured jumbo MTU size (1500 to 9000 bytes). The default is 1500 bytes.
Step 5	**system mtu alternate interface** {*interface-id* | **range** *interface-range*}	(Optional) Apply the alternate MTU size to the specified interface or range of interfaces.
Step 6	**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 7	**end**	Return to privileged EXEC mode.
Step 8	**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.

After the switch reloads, you can verify your settings by entering the **show system mtu** privileged EXEC command. To verify the MTU setting on an interface, enter the **show interface** *interface-id* **mtu**.

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.
```

This example shows how to set the maximum packet size for Gigabit Ethernet ports to 1800 bytes, to define an alternate MTU size of 1700 bytes and apply it to Gigabit Ethernet port 0/8. Changes are not applied until you reload the switch:

```
Switch(config)# system mtu jumbo 1800
Switch(config)# system mtu alternate 1700
Changes to the Alternate MTU will not take effect until the next reload is done
Switch(config)# system mtu alternate interface gigabitethernet 0/8
Changes to the Alternate MTU on interface will not take effect until the next reload is
done
Switch(config)# exit
Switch# reload
```

This example shows how to apply the alternate MTU to Gigabit Ethernet interfaces 1 to 10. Changes are not applied until you reload the switch:

```
Switch(config)# system mtu alternate interface range gigabitethernet 0/1-10
Changes to the Alternate MTU on interface(s) will not take effect until the next reload is done
Switch(config)# exit
```

Monitoring and Maintaining the Interfaces

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 10-5 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 10-5 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*] **mtu**	Display the MTU setting on an interface.
show interfaces [*interface-id*] **switchport**	Display administrative and operational status of switching mode. 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.

Table 10-5 Show Commands for Interfaces (continued)

Command	Purpose
show interfaces [*interface-id*] **transceiver** [**detail** | **dom-supported-list** | **module** *number* | **properties** | **threshold-table**]	Display these physical and operational status about an SFP module: • *interface-id*–(Optional) Display configuration and status for a specified physical interface. • **detail–**(Optional) Display calibration properties, including high and low numbers and any alarm information for any Digital Optical Monitoring (DoM)-capable transceiver if one is installed in the switch. • **dom-supported-list–**(Optional) List all supported DoM transceivers. • **module** *number***–**(Optional) Limit display to interfaces on module on the switch. The range is 1 to 9. This option is not available if you entered a specific interface ID. • **properties–**(Optional) Display speed, duplex, and inline power settings on an interface • **threshold-table–**(Optional) Display alarm and warning threshold table
show interfaces [*interface-id*] [{**transceiver properties** | **detail**}] *module number*]	Display physical and operational status about an SFP module.
show port-type [**eni** | **nni** | **uni**]	Display interface type information for the Cisco ME switch.
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 boot 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 10-6 lists the privileged EXEC mode **clear** commands that you can use to clear counters and reset interfaces.

Table 10-6 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 enable an 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.