16- and 36-Port Cisco EtherSwitch Network Module for Cisco 2600 Series, Cisco 3600 Series, and Cisco 3700 Series

Feature History

Release	Modification
12.2(2)XT	This feature was introduced on the Cisco 2600 series, Cisco 3600 series and Cisco 3700 series router.
12.2(8)T	This feature was integrated into Cisco IOS Release 12.2(8)T.
12.2(11)T	This feature was supported in Cisco IOS Release 12.2(11)T.

This feature module describes the 16- and 36-Port Cisco EtherSwitch Network Module (NM-16ESW and NM-36ESW) for Cisco 2600 series, Cisco 3600 series, and Cisco 3700 series routers in Cisco IOS Release 12.2(2)XT and Cisco IOS Release 12.2(8)T and above.

This document includes the following sections:

Americas Headquarters:
Cisco Systems, Inc., 170 West Tasman Drive, San Jose, CA 95134-1706 USA

Feature Overview

This document explains how to configure the 16- and 36-port Cisco EtherSwitch network modules. This network module is supported on Cisco 2600 series, Cisco 3600 series, and Cisco 3700 series routers. The Cisco EtherSwitch network module is a modular, high density voice network module that provides layer 2 switching across Ethernet ports. The 16-port Cisco EtherSwitch network module has 16 10/100BASE-TX ports and an optional 10/100/1000BASE-T Gigabit Ethernet port. The 36-port Cisco EtherSwitch network module has 36 10/100BASE-TX ports and two optional 10/100/1000BASE-T Gigabit Ethernet ports. The gigabit Ethernet can be used as an uplink port to a server or, as a stacking link to another 16- or 36-port Cisco EtherSwitch network modules in the same system. The 36-port Cisco EtherSwitch network module requires a double-wide slot. An optional power module can also be added to provide inline power for IP telephones.

The 16- and 36-port Cisco EtherSwitch network modules support the following:

- Layer 2 Ethernet Interfaces, page 2
- Switch Virtual Interfaces (SVI), page 4
- VLAN Trunk Protocol, page 5
- EtherChannel, page 7
- Spanning Tree Protocol, page 8
- Cisco Discovery Protocol, page 18
- Switched Port Analyzer (SPAN), page 18
- Quality of Service, page 20
- Maximum Number of VLAN and Multicast Groups, page 22
- IP Multicast Support, page 22
- Storm-Control, page 23
- Port Security, page 24
- Ethernet Switching in Cisco AVVID Architecture, page 25
- Stacking, page 25
- Flow Control, page 26

Layer 2 Ethernet Interfaces

Layer 2 Ethernet Switching

Cisco EtherSwitch network modules support simultaneous, parallel connections between Layer 2 Ethernet segments. Switched connections between Ethernet segments last only for the duration of the packet. New connections can be made between different segments for the next packet.

The Cisco EtherSwitch network module solves congestion problems caused by high-bandwidth devices and a large number of users by assigning each device (for example, a server) to its own 10-, 100-, or 1000-Mbps segment. Because each Ethernet interface on the switch represents a separate Ethernet segment, servers in a properly configured switched environment achieve full access to the bandwidth.

Because collisions are a major bottleneck in Ethernet networks, an effective solution is full-duplex communication. Normally, Ethernet operates in half-duplex mode, which means that stations can either receive or transmit. In full-duplex mode, two stations can transmit and receive at the same time. When packets can flow in both directions simultaneously, effective Ethernet bandwidth doubles to 20 Mbps for 10-Mbps interfaces and to 200 Mbps for Fast Ethernet interfaces.

Switching Frames Between Segments

Each Ethernet interface on an Cisco EtherSwitch network module can connect to a single workstation or server, or to a hub through which workstations or servers connect to the network.

On a typical Ethernet hub, all ports connect to a common backplane within the hub, and the bandwidth of the network is shared by all devices attached to the hub. If two stations establish a session that uses a significant level of bandwidth, the network performance of all other stations attached to the hub is degraded.

To reduce degradation, the switch treats each interface as an individual segment. When stations on different interfaces need to communicate, the switch forwards frames from one interface to the other at wire speed to ensure that each session receives full bandwidth.

To switch frames between interfaces efficiently, the switch maintains an address table. When a frame enters the switch, it associates the MAC address of the sending station with the interface on which it was received.

Building the Address Table

The Cisco EtherSwitch network module builds the address table by using the source address of the frames received. When the switch receives a frame for a destination address not listed in its address table, it floods the frame to all interfaces of the same virtual local area network (VLAN) except the interface that received the frame. When the destination station replies, the switch adds its relevant source address and interface ID to the address table. The switch then forwards subsequent frames to a single interface without flooding to all interfaces. The address table can store at least 8,191 address entries without flooding any entries. The switch uses an aging mechanism, defined by a configurable aging timer; so if an address remains inactive for a specified number of seconds, it is removed from the address table.

Note Default parameters on the aging timer are recommended.

VLAN Trunks

A trunk is a point-to-point link between one or more Ethernet switch interfaces and another networking device such as a router or a switch. Trunks carry the traffic of multiple VLANs over a single link and allow you to extend VLANs across an entire network and supports only one encapsulation on all Ethernet interfaces:

802.1Q-802.1Q is an industry-standard trunking encapsulation.

You can configure a trunk on a single Ethernet interface or on an EtherChannel bundle. For more information about EtherChannel, see the "Configuring Layer 2 EtherChannels (Port-Channel Logical Interfaces)" section on page 40.

Layer 2 Interface Modes

Switchport mode access puts the interface into nontrunking mode. The interface will stay in access mode regardless of what the connected port mode is. Only access VLAN traffic will travel on the access port and untagged (802.3).

Switchport mode trunk puts the interface into permanent trunking mode.

Table 1 Default Layer 2 Ethernet Interface Configuration

Feature	Default Value
Interface mode	switchport mode access / trunk
Trunk encapsulation	switchport trunk encapsulation dot1q
Allowed VLAN range	VLANs 1-1005
Default VLAN (for access ports)	VLAN 1
Native VLAN (for 802.1Q trunks)	VLAN 1
Spanning Tree Protocol (STP)	Enabled for all VLANs
STP port priority	128
STP port cost	100 for 10-Mbps Ethernet interfaces 19 for 10/100-Mbps Fast Ethernet interfaces 19 for 1000-Mbps Fast Ethernet interfaces

When you connect a Cisco switch to a device other than a Cisco device through an 802.1Q trunk, the Cisco switch combines the spanning tree instance of the VLAN trunk with the spanning tree instance of the other 802.1Q switch. However, spanning tree information for each VLAN is maintained by Cisco switches separated by a cloud of 802.1Q switches that are not Cisco switches. The 802.1Q cloud separating the Cisco switches that is not Cisco devised, is treated as a single trunk link between the switches.

Make sure that the native VLAN for an 802.1Q trunk is the same on both ends of the trunk link. If the VLAN on one end of the trunk is different from the VLAN on the other end, spanning tree loops might result. Inconsistencies detected by a Cisco switch mark the line as broken and block traffic for the specific VLAN.

Disabling spanning tree on the VLAN of an 802.1Q trunk without disabling spanning tree on every VLAN in the network can potentially cause spanning tree loops. Cisco recommends that you leave spanning tree enabled on the VLAN of an 802.1Q trunk or that you disable spanning tree on every VLAN in the network. Make sure that your network is loop-free before disabling spanning tree.

Layer 2 Interface Configuration Guidelines and Restrictions

Follow these guidelines and restrictions when configuring Layer 2 interfaces:

In a network of Cisco switches connected through 802.1Q trunks, the switches maintain one instance of spanning tree for each VLAN allowed on the trunks. 802.1Q switches that are not Cisco switches, maintain only one instance of spanning tree for all VLANs allowed on the trunks.

Switch Virtual Interfaces (SVI)

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 it is necessary to configure an SVI for a VLAN only when you wish to route between VLANs, 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. In Layer 2 mode, SVIs provide IP host connectivity only to the system; in Layer 3 mode, you can configure routing across SVIs.

To use SVIs in Layer 3 mode, you must have the enhanced multilayer switch image installed on your switch.

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 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.

SVIs support routing protocol and bridging configurations. For more information about configuring IP routing, see the "Configuring IP Multicast Layer 3 Switching" section on page 55.

VLAN Trunk Protocol

VLAN Protocol (VTP) is a Layer 2 messaging protocol that maintains VLAN configuration consistency by managing the addition, deletion, and renaming of VLANs within a VTP domain. A VTP domain (also called a VLAN management domain) is made up of one or more switches that share the same VTP domain name and that are interconnected with trunks. VTP minimizes misconfigurations and configuration inconsistencies that can result in a number of problems, such as duplicate VLAN names, incorrect VLAN-type specifications, and security violations. Before you create VLANs, you must decide whether to use VTP in your network. With VTP, you can make configuration changes centrally on one or more switches and have those changes automatically communicated to all the other switches in the network.

VTP Domain

A VTP domain (also called a VLAN management domain) is made up of one or more interconnected switches that share the same VTP domain name. A switch can be configured to be in one and only one VTP domain. You make global VLAN configuration changes for the domain using either the command-line interface (CLI) or Simple Network Management Protocol (SNMP).

By default, the switch is in VTP server mode and is in an un-named domain state until the switch receives an advertisement for a domain over a trunk link or until you configure a management domain. You cannot create or modify VLANs on a VTP server until the management domain name is specified or learned.

If the switch receives a VTP advertisement over a trunk link, it inherits the management domain name and the VTP configuration revision number. The switch ignores advertisements with a different management domain name or an earlier configuration revision number.

If you configure the switch as VTP transparent, you can create and modify VLANs but the changes affect only the individual switch.

When you make a change to the VLAN configuration on a VTP server, the change is propagated to all switches in the VTP domain. VTP advertisements are transmitted out all trunk connections using IEEE 802.1Q encapsulation.

VTP maps VLANs dynamically across multiple LAN types with unique names and internal index associations. Mapping eliminates excessive device administration required from network administrators.

VTP Modes

You can configure a switch to operate in any one of these VTP modes:

- Server-In VTP server mode, you can create, modify, and delete VLANs and specify other configuration parameters (such as VTP version) for the entire VTP domain. VTP servers advertise their VLAN configuration to other switches in the same VTP domain and synchronize their VLAN configuration with other switches based on advertisements received over trunk links. VTP server is the default mode.
- Client-VTP clients behave the same way as VTP servers, but you cannot create, change, or delete VLANs on a VTP client.

- Transparent-VTP transparent switches do not participate in VTP. A VTP transparent switch does not advertise its VLAN configuration and does not synchronize its VLAN configuration based on received advertisements. However, in VTP version 2, transparent switches do forward VTP advertisements that they receive out their trunk interfaces.

VTP Advertisements

Each switch in the VTP domain sends periodic advertisements out each trunk interface to a reserved multicast address. VTP advertisements are received by neighboring switches, which update their VTP and VLAN configurations as necessary.

The following global configuration information is distributed in VTP advertisements:

- VLAN IDs (801.Q)
- VTP domain name
- VTP configuration revision number
- VLAN configuration, including maximum transmission unit (MTU) size for each VLAN
- Frame format

VTP Version 2

If you use VTP in your network, you must decide whether to use VTP version 1 or version 2. VTP version 2 supports the following features not supported in version 1:

Unrecognized Type-Length-Value (TLV) Support—A VTP server or client propagates configuration changes to its other trunks, even for TLVs it is not able to parse. The unrecognized TLV is saved in NVRAM.

Version-Dependent Transparent Mode—In VTP version 1, a VTP transparent switch inspects VTP messages for the domain name and version, and forwards a message only if the version and domain name match. Since only one domain is supported in the NM-16ESW software, VTP version 2 forwards VTP messages in transparent mode, without checking the version.

Consistency Checks—In VTP version 2, VLAN consistency checks (such as VLAN names and values) are performed only when you enter new information through the CLI or SNMP. Consistency checks are not performed when new information is obtained from a VTP message, or when information is read from NVRAM. If the digest on a received VTP message is correct, its information is accepted without consistency checks.

VTP Configuration Guidelines and Restrictions

Follow these guidelines and restrictions when implementing VTP in your network:

- All switches in a VTP domain must run the same VTP version.
- You must configure a password on each switch in the management domain when in secure mode.
- A VTP version 2-capable switch can operate in the same VTP domain as a switch running VTP version 1, provided that VTP version 2 is disabled on the VTP version 2-capable switch. (VTP version 2 is disabled by default).
- Do not enable VTP version 2 on a switch unless all switches in the same VTP domain are version 2-capable. When you enable VTP version 2 on a switch, all version 2-capable switches in the domain enable VTP version 2
- The Cisco IOS **end** and **Ctrl-Z** commands are not supported in VLAN database mode.
- The VLAN database stored on internal flash is supported.
- Use the **squeeze flash** command to remove old copies of overwritten VLAN databases.

EtherChannel

EtherChannel bundles up to eight individual Ethernet links into a single logical link that provides bandwidth of up to 1600 Mbps (Fast EtherChannel full duplex) between the network module and another switch or host.

A Cisco EtherSwitch network module system supports a maximum of six EtherChannels. All interfaces in each EtherChannel must have the same speed duplex and mode.

Load Balancing

EtherChannel balances traffic load across the links in a channel by reducing part of the binary pattern formed from the addresses in the frame to a numerical value that selects one of the links in the channel.

EtherChannel load balancing can use MAC addresses, or IP addresses; either source or destination or both source and destination. The selected mode applies to all EtherChannels configured on the switch.

Use the option that provides the greatest variety in your configuration. For example, if the traffic on a channel is going only to a single MAC address, using the destination MAC address always chooses the same link in the channel; using source addresses or IP addresses may result in better load balancing.

EtherChannel Configuration Guidelines and Restrictions

If improperly configured, some EtherChannel interfaces are disabled automatically to avoid network loops and other problems. Follow these guidelines and restrictions to avoid configuration problems:

- All Ethernet interfaces on all modules support EtherChannel (maximum of eight interfaces) with no requirement that interfaces be physically contiguous or on the same module.

Configure all interfaces in an EtherChannel to operate at the same speed and duplex mode.

- Enable all interfaces in an EtherChannel. If you shut down an interface in an EtherChannel, it is treated as a link failure and its traffic is transferred to one of the remaining interfaces in the EtherChannel.
- An EtherChannel will not form if one of the interfaces is a Switched Port Analyzer (SPAN) destination port.

For Layer 2 EtherChannels:

- Assign all interfaces in the EtherChannel to the same VLAN, or configure them as trunks.

An EtherChannel supports the same allowed range of VLANs on all interfaces in a trunking Layer 2 EtherChannel. If the allowed range of VLANs is not the same, the interfaces do not form an EtherChannel.

Interfaces with different Spanning Tree Protocol (STP) port path costs can form an EtherChannel as long they are otherwise compatibly configured. Setting different STP port path costs does not, by itself, make interfaces incompatible for the formation of an EtherChannel.

After you configure an EtherChannel, configuration that you apply to the port-channel interface affects the EtherChannel.

Spanning Tree Protocol

This section describes how to configure the Spanning Tree Protocol (STP) on Cisco EtherSwitch network module systems.

Spanning tree is a Layer 2 link management protocol that provides path redundancy while preventing undesirable loops in the network. For a Layer 2 Ethernet network to function properly, only one active path can exist between any two stations. Spanning tree operation is transparent to end stations, which cannot detect whether they are connected to a single LAN segment or to a switched LAN of multiple segments.

The Cisco EtherSwitch network module uses STP (the IEEE 802.1D bridge protocol) on all VLANs. By default, a single instance of STP runs on each configured VLAN (provided that you do not manually disable STP). You can enable and disable STP on a per-VLAN basis.

When you create fault-tolerant internetworks, you must have a loop-free path between all nodes in a network. The spanning tree algorithm calculates the best loop-free path throughout a switched Layer 2 network. Switches send and receive spanning tree frames at regular intervals. The switches do not forward these frames, but use the frames to construct a loop-free path.

Multiple active paths between end stations cause loops in the network. If a loop exists in the network, end stations might receive duplicate messages and switches might learn endstation MAC addresses on multiple Layer 2 interfaces. These conditions result in an unstable network.

Spanning Tree Protocol defines a tree with a root switch and a loop-free path from the root to all switches in the Layer 2 network. Spanning tree forces redundant data paths into a standby (blocked) state. If a network segment in the spanning tree fails and a redundant path exists, the spanning tree algorithm recalculates the spanning tree topology and activates the standby path.

When two ports on a switch are part of a loop, the spanning tree port priority and port path cost setting determine which port is put in the forwarding state and which port is put in the blocking state. The spanning tree port priority value represents the location of an interface in the network topology and how well located it is to pass traffic. The spanning tree port path cost value represents media speed.

Bridge Protocol Data Units

The stable active spanning tree topology of a switched network is determined by the following:

- The unique bridge ID (bridge priority and MAC address) associated with each VLAN on each switch
- The spanning tree path cost to the root bridge
- The port identifier (port priority and MAC address) associated with each Layer 2 interface

The Bridge Protocol Data Units (BPDU) are transmitted in one direction from the root switch, and each switch sends configuration BPDUs to communicate and compute the spanning tree topology. Each configuration BPDU contains the following minimal information:

- The unique bridge ID of the switch that the transmitting switch believes to be the root switch
- The spanning tree path cost to the root
- The bridge ID of the transmitting bridge
- Message age
- The identifier of the transmitting port
- Values for the hello, forward delay, and max-age protocol timers

When a switch transmits a bridge packet data unit (BPDU) frame, all switches connected to the LAN on which the frame is transmitted receive the BPDU. When a switch receives a BPDU, it does not forward the frame but instead uses the information in the frame to calculate a BPDU, and, if the topology changes, initiate a BPDU transmission.

A BPDU exchange results in the following:

- One switch is elected as the root switch.
- The shortest distance to the root switch is calculated for each switch based on the path cost.
- A designated bridge for each LAN segment is selected. This is the switch closest to the root bridge through which frames is forwarded to the root.
- A root port is selected. This is the port providing the best path from the bridge to the root bridge.
- Ports included in the spanning tree are selected.
- Election of the Root Bridge

For each VLAN, the switch with the highest bridge priority (the lowest numerical priority value) is elected as the root switch. If all switches are configured with the default priority (32768), the switch with the lowest MAC address in the VLAN becomes the root switch.

The spanning tree root switch is the logical center of the spanning tree topology in a switched network. All paths that are not needed to reach the root switch from anywhere in the switched network are placed in spanning tree blocking mode.

BPDUs contain information about the transmitting bridge and its ports, including bridge and MAC addresses, bridge priority, port priority, and path cost. Spanning tree uses this information to elect the root bridge and root port for the switched network, as well as the root port and designated port for each switched segment.

STP Timers

The following describe the STP timers that affect the entire spanning tree performance:

Timer	Purpose
Hello timer	Determines how often the switch broadcasts hello messages to other switches.
Forward delay timer	Determines how long each of the listening and learning states will last before the port begins forwarding
Maximum age timer	Determines the amount of time protocol information received on a port is stored by the switch.

Spanning Tree Port States

Propagation delays can occur when protocol information passes through a switched LAN. As a result, topology changes can take place at different times and at different places in a switched network. When a Layer 2 interface transitions directly from nonparticipation in the spanning tree topology to the forwarding state, it can create temporary data loops. Ports must wait for new topology information to propagate through the switched LAN before starting to forward frames. They must allow the frame lifetime to expire for frames that have been forwarded using the old topology.

Each Layer 2 interface on a switch using spanning tree exists in one of the following five states:

- Blocking—The Layer 2 interface does not participate in frame forwarding.
- Listening—First transitional state after the blocking state when spanning tree determines that the Layer 2 interface should participate in frame forwarding.

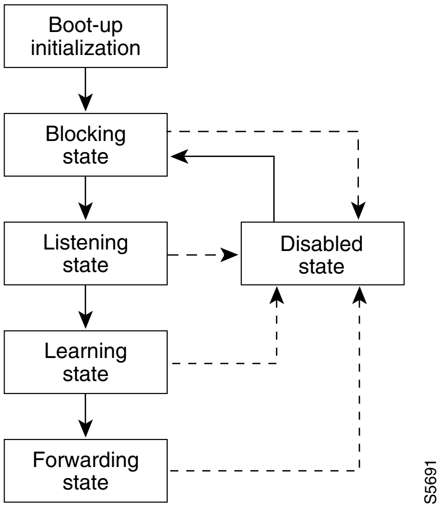
- Learning—The Layer 2 interface prepares to participate in frame forwarding.
- Forwarding—The Layer 2 interface forwards frames.
- Disabled—The Layer 2 interface does not participate in spanning tree and is not forwarding frames.

A Layer 2 interface moves through these five states as follows:

- From initialization to blocking
- From blocking to listening or to disabled
- From listening to learning or to disabled
- From learning to forwarding or to disabled
- From forwarding to disabled

Figure 1 illustrates how a port moves through the five stages.

Figure 1 STP Port States



When you enable spanning tree, every port in the switch, VLAN, or network goes through the blocking state and the transitory states of listening and learning at power up. If properly configured, each Layer 2 interface stabilizes to the forwarding or blocking state.

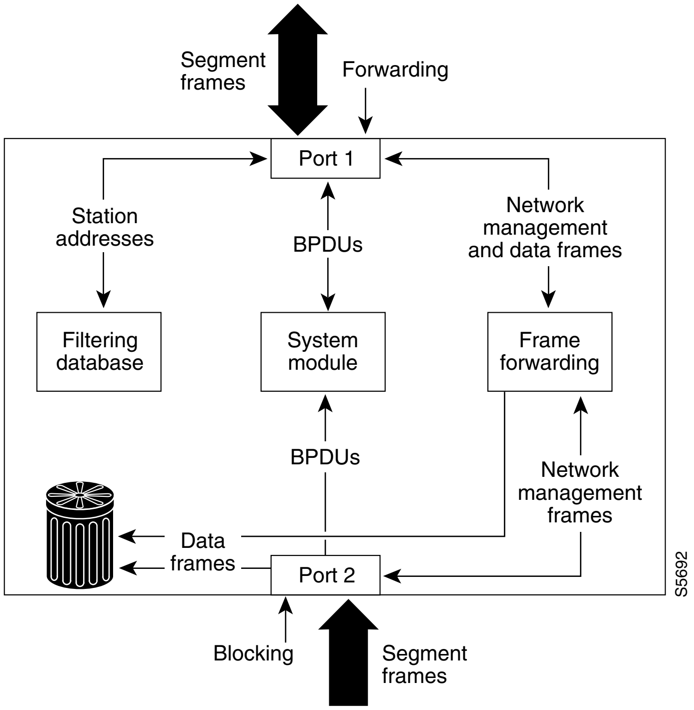
When the spanning tree algorithm places a Layer 2 interface in the forwarding state, the following process occurs:

1. The Layer 2 interface is put into the listening state while it waits for protocol information that suggests that it should go to the blocking state.
2. The Layer 2 interface waits for the forward delay timer to expire, moves the Layer 2 interface to the learning state, and resets the forward delay timer.
3. In the learning state, the Layer 2 interface continues to block frame forwarding as it learns end station location information for the forwarding database.
4. The Layer 2 interface waits for the forward delay timer to expire and then moves the Layer 2 interface to the forwarding state, where both learning and frame forwarding are enabled.

Blocking State

A Layer 2 interface in the blocking state does not participate in frame forwarding, as shown in Figure 2. After initialization, a BPDU is sent out to each Layer 2 interface in the switch. A switch initially assumes it is the root until it exchanges BPDUs with other switches. This exchange establishes which switch in the network is the root or root bridge. If only one switch is in the network, no exchange occurs, the forward delay timer expires, and the ports move to the listening state. A port always enters the blocking state following switch initialization.

Figure 2 Interface 2 in Blocking State



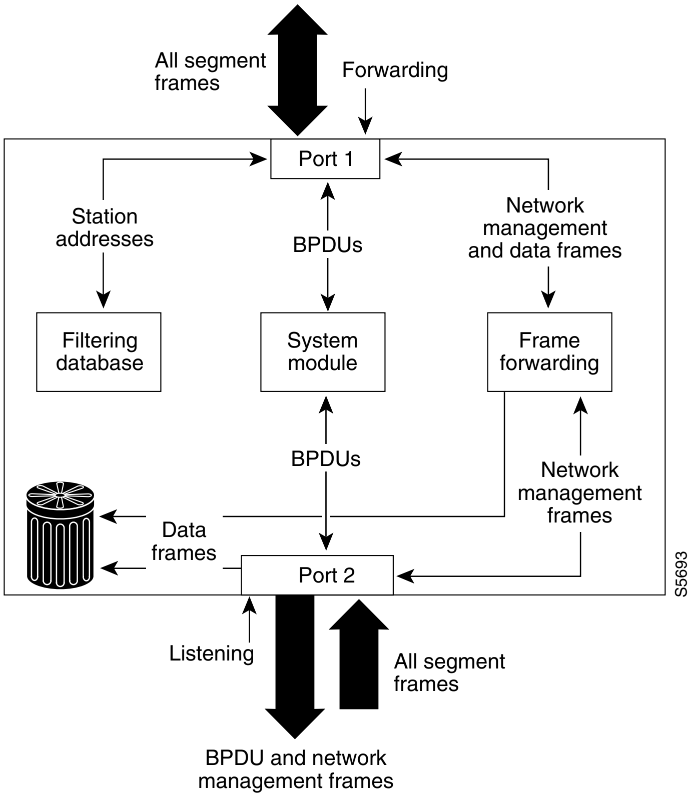
A Layer 2 interface in the blocking state performs as follows:

- Discards frames received from the attached segment.
- Discards frames switched from another interface for forwarding.
- Does not incorporate end station location into its address database. (There is no learning on a blocking Layer 2 interface, so there is no address database update.)
- Receives BPDUs and directs them to the system module.
- Does not transmit BPDUs received from the system module.
- Receives and responds to network management messages.

Listening State

The listening state is the first transitional state a Layer 2 interface enters after the blocking state. The Layer 2 interface enters this state when STP determines that the Layer 2 interface should participate in frame forwarding. Figure 3 shows a Layer 2 interface in the listening state.

Figure 3 Interface 2 in Listening State



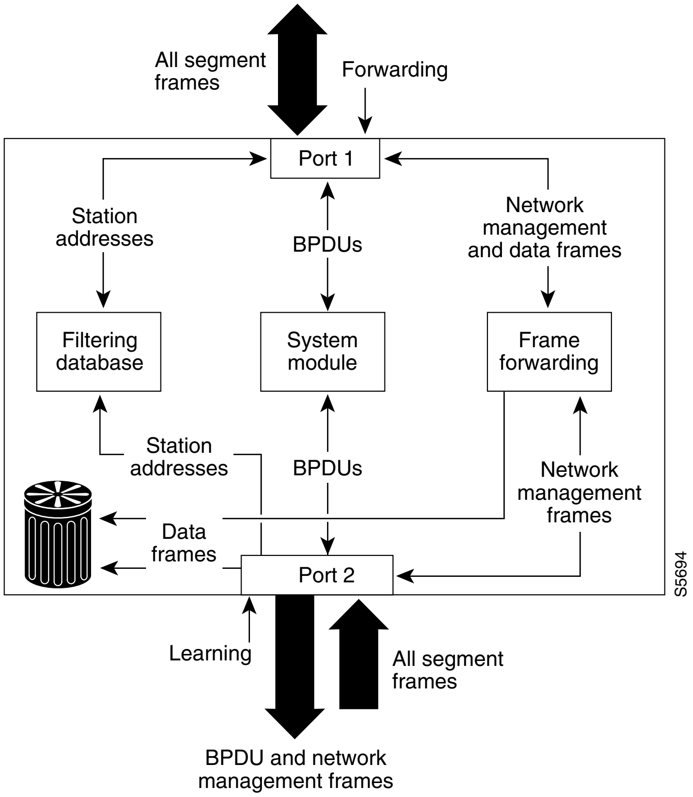
A Layer 2 interface in the listening state performs as follows:

- Discards frames received from the attached segment.
- Discards frames switched from another interface for forwarding.
- Does not incorporate end station location into its address database. (There is no learning at this point, so there is no address database update.)
- Receives BPDUs and directs them to the system module.
- Receives, processes, and transmits BPDUs received from the system module.
- Receives and responds to network management messages.

Learning State

A Layer 2 interface in the learning state prepares to participate in frame forwarding. The Layer 2 interface enters the learning state from the listening state. Figure 4 shows a Layer 2 interface in the learning state.

Figure 4 Interface 2 in Learning State



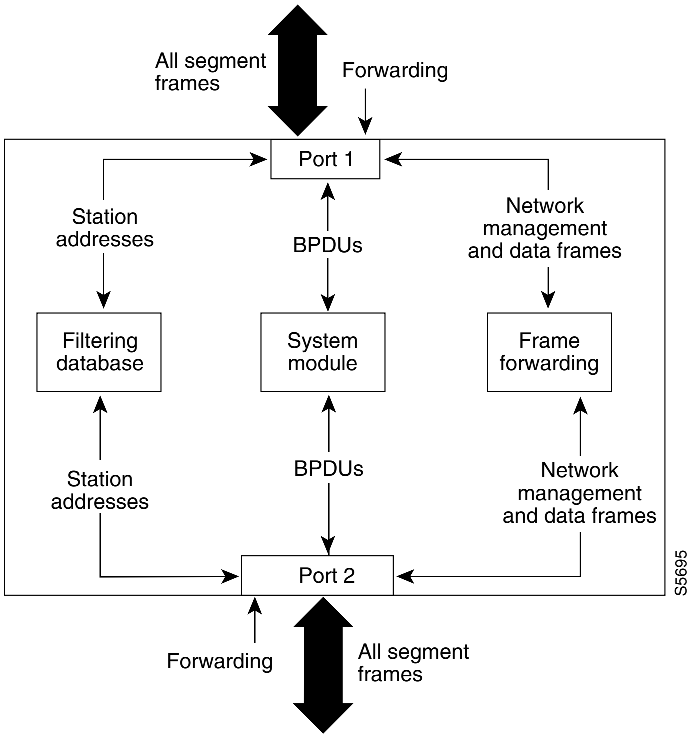
A Layer 2 interface in the learning state performs as follows:

- Discards frames received from the attached segment.
- Discards frames switched from another interface for forwarding.
- Incorporates end station location into its address database.
- Receives BPDUs and directs them to the system module.
- Receives, processes, and transmits BPDUs received from the system module.
- Receives and responds to network management messages.

Forwarding State

A Layer 2 interface in the forwarding state forwards frames, as shown in Figure 5. The Layer 2 interface enters the forwarding state from the learning state.

Figure 5 Interface 2 in Forwarding State



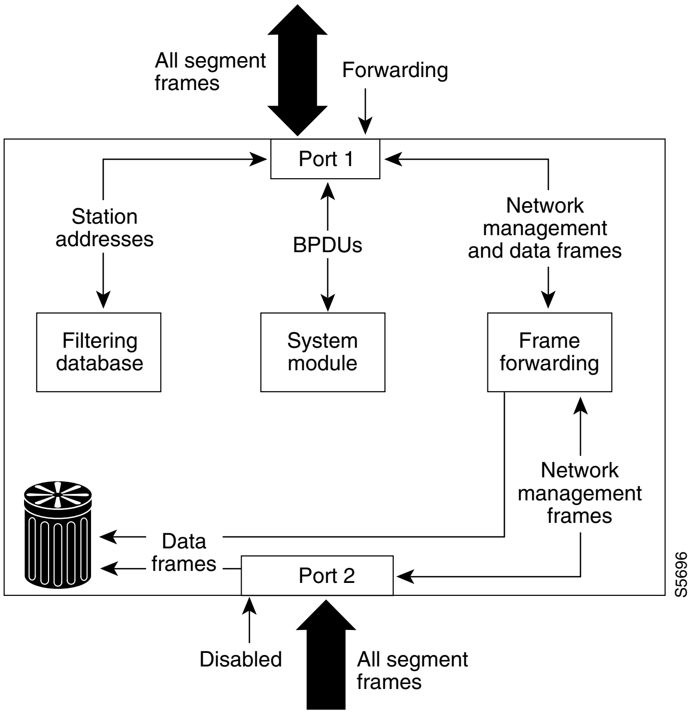
A Layer 2 interface in the forwarding state performs as follows:

- Forwards frames received from the attached segment.
- Forwards frames switched from another Layer 2 interface for forwarding.
- Incorporates end station location information into its address database.
- Receives BPDUs and directs them to the system module.
- Processes BPDUs received from the system module.
- Receives and responds to network management messages.

Disabled State

A Layer 2 interface in the disabled state does not participate in frame forwarding or spanning tree, as shown in Figure 6. A Layer 2 interface in the disabled state is virtually nonoperational.

Figure 6 Interface 2 in Disabled State



A disabled Layer 2 interface performs as follows:

- Discards frames received from the attached segment.
- Discards frames switched from another Layer 2 interface for forwarding.
- Does not incorporate end station location into its address database. (There is no learning, so there is no address database update.)
- Does not receive BPDUs.
- Does not receive BPDUs for transmission from the system module.

MAC Address Allocation

The mac address allocation manager has a pool of MAC addresses that are used as the bridge IDs for the VLAN spanning trees.

Platform	Maximum number of VLANs allowed
3640 or higher	64 VLANS
3620	32 VLANs
2600	32 VLANs

MAC addresses are allocated sequentially, with the first MAC address in the range assigned to VLAN 1, the second MAC address in the range assigned to VLAN 2, and so forth.

For example, if the MAC address range is 00-e0-1e-9b-2e-00 to 00-e0-1e-9b-31-ff, the VLAN 1 bridge ID is 00-e0-1e-9b-2e-00, the VLAN 2 bridge ID is 00-e0-1e-9b-2e-01, the VLAN 3 bridge ID is 00-e0-1e-9b-2e-02, and so forth.

Default Spanning Tree Configuration

Spanning Tree Default Configuration

Feature	Default Value
Enable state	Spanning tree enabled for all VLANs
Bridge priority	32768
Spanning tree port priority (configurable on a per-interface basis; used on interfaces configured as Layer 2 access ports)	128
Spanning tree port cost (configurable on a per-interface basis; used on interfaces configured as Layer 2 access ports)	Fast Ethernet: 19 Ethernet: 100 Gigabit Ethernet: 19
Spanning tree VLAN port priority (configurable on a per-VLAN basis; used on interfaces configured as Layer 2 trunk ports)	128
Spanning tree VLAN port cost (configurable on a per-VLAN basis; used on interfaces configured as Layer 2 trunk ports)	Fast Ethernet: 10 Ethernet: 10
Hello time	2 seconds
Forward delay time	15 seconds
Maximum aging time	20 seconds

Spanning Tree Port Priority

In the event of a loop, spanning tree considers port priority when selecting an interface to put into the forwarding state. You can assign higher priority values to interfaces that you want spanning tree to select first, and lower priority values to interfaces that you want spanning tree to select last. If all interfaces have the same priority value, spanning tree puts the interface with the lowest interface number in the forwarding state and blocks other interfaces. The possible priority range is 0 through 255, configurable in increments of 4 (the default is 128).

Cisco IOS software uses the port priority value when the interface is configured as an access port and uses VLAN port priority values when the interface is configured as a trunk port.

Spanning Tree Port Cost

The spanning tree port path cost default value is derived from the media speed of an interface. In the event of a loop, spanning tree considers port cost when selecting an interface to put into the forwarding state. You can assign lower cost values to interfaces that you want spanning tree to select first and higher cost values to interfaces that you want spanning tree to select last. If all interfaces have the same cost value, spanning tree puts the interface with the lowest interface number in the forwarding state and blocks other interfaces.

The possible cost range is 0 through 65535 (the default is media-specific).

Spanning tree uses the port cost value when the interface is configured as an access port and uses VLAN port cost values when the interface is configured as a trunk port.

Cisco Discovery Protocol

Cisco Discovery Protocol (CDP) is a protocol that runs over Layer 2 (the data link layer) on all Cisco routers, bridges, access servers, and switches. CDP allows network management applications to discover Cisco devices that are neighbors of already known devices, in particular, neighbors running lower-layer, transparent protocols. With CDP, network management applications can learn the device type and the SNMP agent address of neighboring devices. This feature enables applications to send SNMP queries to neighboring devices.

CDP runs on all LAN and WAN media that support Subnetwork Access Protocol (SNAP). Each CDP-configured device sends periodic messages to a multicast address. Each device advertises at least one address at which it can receive SNMP messages. The advertisements also contain the time-to-live, or hold-time information, which indicates the length of time a receiving device should hold CDP information before discarding it.

Switched Port Analyzer (SPAN)

Switched Port Analyzer (SPAN) Session

A Switched Port Analyzer (SPAN) session is an association of a destination interface with a set of source interfaces. You configure SPAN sessions using parameters that specify the type of network traffic to monitor. SPAN sessions allow you to monitor traffic on one or more interfaces and to send either ingress traffic, egress traffic, or both to one destination interface. You can configure one SPAN session with separate or overlapping sets of SPAN source interfaces or VLANs. Only switched interfaces can be configured as SPAN sources or destinations on the same network module.

SPAN sessions do not interfere with the normal operation of the switch. You can enable or disable SPAN sessions with command-line interface (CLI) or SNMP commands. When enabled, a SPAN session might become active or inactive based on various events or actions, and this would be indicated by a syslog message. The **show monitor session SPAN session number** command displays the operational status of a SPAN session.

A SPAN session remains inactive after system power-up until the destination interface is operational.

Destination Interface

A destination interface (also called a monitor interface) is a switched interface to which SPAN sends packets for analysis. You can have one SPAN destination interface. Once an interface becomes an active destination interface, incoming traffic is disabled. You cannot configure a SPAN destination interface to receive ingress traffic. The interface does not forward any traffic except that required for the SPAN session.

An interface configured as a destination interface cannot be configured as a source interface. EtherChannel interfaces cannot be SPAN destination interfaces.

Specifying a trunk interface as a SPAN destination interface stops trunking on the interface.

Source Interface

A source interface is an interface monitored for network traffic analysis. One or more source interfaces can be monitored in a single SPAN session with user-specified traffic types (ingress, egress, or both) applicable for all the source interfaces.

You can configure source interfaces in any VLAN. You can configure EtherChannel as source interfaces, which means that all interfaces in the specified VLANs are source interfaces for the SPAN session.

Trunk interfaces can be configured as source interfaces and mixed with nontrunk source interfaces; however, the destination interface never encapsulates.

Traffic Types

Ingress SPAN (Rx) copies network traffic received by the source interfaces for analysis at the destination interface. Egress SPAN (Tx) copies network traffic transmitted from the source interfaces. Specifying the configuration option both copies network traffic received and transmitted by the source interfaces to the destination interface.

SPAN Traffic

Network traffic, including multicast, can be monitored using SPAN. Multicast packet monitoring is enabled by default. In some SPAN configurations, multiple copies of the same source packet are sent to the SPAN destination interface. For example, a bidirectional (both ingress and egress) SPAN session is configured for sources a1 and a2 to a destination interface d1. If a packet enters the switch through a1 and gets switched to a2, both incoming and outgoing packets are sent to destination interface d1; both packets would be the same (unless a Layer-3 rewrite had occurred, in which case the packets would be different).

Note Monitoring of VLANs is not supported

SPAN Configuration Guidelines and Restrictions

Follow these guidelines and restrictions when configuring SPAN:

- Enter the **no monitor session** *session number* command with no other parameters to clear the SPAN session number.
- EtherChannel interfaces can be SPAN source interfaces; they cannot be SPAN destination interfaces.
- If you specify multiple SPAN source interfaces, the interfaces can belong to different VLANs.
- Monitoring of VLANs is not supported
- Only one SPAN session may be run at any given time.
- Outgoing CDP and BPDU packets will not be replicated.
- SPAN destinations never participate in any spanning tree instance. SPAN includes BPDUs in the monitored traffic, so any BPDUs seen on the SPAN destination are from the SPAN source.
- Use a network analyzer to monitor interfaces.
- You can have one SPAN destination interface.
- You can mix individual source interfaces within a single SPAN session.
- You cannot configure a SPAN destination interface to receive ingress traffic.
- When enabled, SPAN uses any previously entered configuration.
- When you specify source interfaces and do not specify a traffic type (**Tx**, **Rx**, or **both**), **both** is used by default.

Quality of Service

Understanding Quality of Service (QoS)

Typically, networks operate on a best-effort delivery basis, which means that all traffic has equal priority and an equal chance of being delivered in a timely manner. When congestion occurs, all traffic has an equal chance of being dropped.

With the QoS feature configured on your switch, you can select specific network traffic, prioritize it according to its relative importance, and use congestion-management and congestion-avoidance techniques to provide preferential treatment. Implementing QoS in your network makes network performance more predictable and bandwidth utilization more effective.

The QoS implementation for this release is based on the DiffServ architecture, an emerging standard from the Internet Engineering Task Force (IETF). This architecture specifies that each packet is classified upon entry into the network. The classification is carried in the IP packet header, using 6 bits from the deprecated IP type of service (ToS) field to carry the classification (*class*) information. Classification can also be carried in the Layer 2 frame. These special bits in the Layer 2 frame or a Layer 3 packet are described here and shown in Figure 7:

- Prioritization values in Layer 2 frames:

  Layer 2 802.1Q frame headers have a 2-byte Tag Control Information field that carries the CoS value in the three most-significant bits, which are called the User Priority bits. On interfaces configured as Layer 2 802.1Q trunks, all traffic is in 802.1Q frames except for traffic in the native VLAN.

  Other frame types cannot carry Layer 2 CoS values.

  Layer 2 CoS values range from 0 for low priority to 7 for high priority.

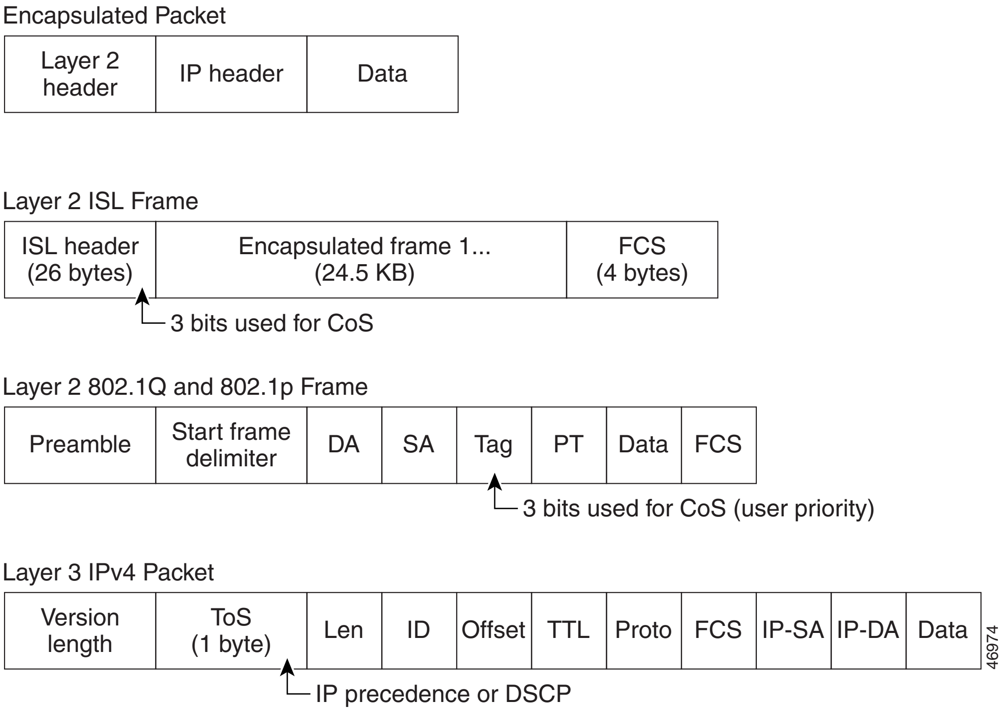
- Prioritization bits in Layer 3 packets:

  Layer 3 IP packets can carry either an IP precedence value or a Differentiated Services Code Point (DSCP) value. QoS supports the use of either value, because DSCP values are backward-compatible with IP precedence values.

  IP precedence values range from 0 to 7.

  DSCP values range from 0 to 63.

Figure 7 QoS Classification Layers in Frames and Packets



Note Layer 2 ISL Frame is not supported in this release.

All switches and routers across the Internet rely on the class information to provide the same forwarding treatment to packets with the same class information and different treatment to packets with different class information. The class information in the packet can be assigned by end hosts or by switches or routers along the way, based on a configured policy, detailed examination of the packet, or both. Detailed examination of the packet is expected to happen closer to the edge of the network so that the core switches and routers are not overloaded.

Switches and routers along the path can use the class information to limit the amount of resources allocated per traffic class. The behavior of an individual device when handling traffic in the DiffServ architecture is called per-hop behavior. If all devices along a path provide a consistent per-hop behavior, you can construct an end-to-end QoS solution.

Implementing QoS in your network can be a simple or complex task and depends on the QoS features offered by your internetworking devices, the traffic types and patterns in your network, and the granularity of control you need over incoming and outgoing traffic.

The Ethernet switch network module can function as a Layer 2 switch connected to a Layer 3 router. When a packet enters the Layer 2 engine directly from a switch port, it is placed into one of four queues in the dynamic, 32-MB shared memory buffer. The queue assignment is based on the dot1p value in the packet. Any voice bearer packets that come in from the Cisco IP phones on the voice VLAN are automatically placed in the highest priority (Queue 3) based on the 802.1p value generated by the IP phone. The queues are then serviced on a WRR basis. The control traffic, which uses a CoS or ToS of 3, is placed in Queue 2.

Table 2 summarizes the queues, CoS values, and weights for Layer 2 QoS on the Ethernet switch network module network module.

Table 2 Queues, CoS values, and Weights for Layer 2 QoS

Queue Number	CoS Value	Weight
3	5,6,7	255
2	3,4	64
1	2	16
0	0,1	1

The weights specify the number of packets that are serviced in the queue before moving on to the next queue. Voice Realtime Transport Protocol (RTP) bearer traffic marked with a CoS or ToS of 5 and Voice Control plane traffic marked with a CoS/ToS of 3 are placed into the highest priority Queues. If the queue has no packets to be serviced, it is skipped. Weighted Random Early Detection (WRED) is not supported on the Fast Ethernet ports.

The WRR default values cannot be changed. There are currently no CLI commands to determine QoS information for WRR weights and queue mappings. You cannot configure port based QoS on the Layer 2 switch ports.

Maximum Number of VLAN and Multicast Groups

The maximum number is less than or equal to 242. The number of VLANs is determined by multiplying the number of VLANs by the number of multicast groups. For example, the maximum number for 10 VLAN's and 20 groups would be 200, under the 242 limit.

IP Multicast Support

The maximum number of multicast groups is related to the maximum number of VLANs. The product of the number of multicast groups and the number of VLANs cannot exceed 242.

- Support for pim sparse mode/dense mode sparse-dense mode

IGMP snooping Versions 1 and 2

Understanding IGMP Snooping

In VLANs or subnets where you have configured IGMP support by enabling multicast routing on the Router and enabling PIM on the VLAN interfaces IGMP snooping manages multicast traffic at Layer2 dynamically forwarding multicast traffic only to those interfaces that want to receive it.

IGMP snooping constrains traffic in MAC multicast groups 01-00-5e-00-00-01 to 01-00-5e-ff-ff-ff. IGMP snooping does not constrain Layer 2 multicasts generated by routing protocols.

Note For more information on IP multicast and IGMP, refer to RFC 1112 and RFC 2236.

IGMP (on a router) sends out periodic general IGMP queries. When you enable IGMP snooping, the switch responds at Layer 2 to the IGMP queries with only one IGMP join request per Layer 2 multicast group. The switch creates one entry per subnet in the Layer 2 forwarding table for each Layer 2 multicast group from which it receives an IGMP join request. All hosts interested in this multicast traffic send IGMP join requests and are added to the forwarding table entry.

Layer 2 multicast groups learned through IGMP snooping are dynamic.

When a host connected to a Layer 2 interface wants to join an IP multicast group, it sends an IGMP join request specifying the IP multicast group it wants to join. When hosts want to leave a multicast group, they can either ignore the periodic general IGMP queries, or they can send an IGMP leave message. When the switch receives an IGMP leave message from a host, it sends out a group-specific IGMP query to determine if any devices connected to that interface are interested in traffic for the specific multicast group. The switch then updates the table entry for that Layer 2 multicast group so that only those hosts interested in receiving multicast traffic for the group are listed. IGMP Snooping is enabled on a switchport only when the SVI is configured for PIM. IGMP snooping is disabled by default.

Fast-Leave Processing

IGMP snooping fast-leave processing allows the switch to remove an interface from the forwarding-table entry without first sending out group specific queries to the interface. The VLAN interface is pruned from the multicast tree for the multicast group specified in the original leave message. Fast-leave processing ensures optimal bandwidth management for all hosts on a switched network, even when multiple multicast groups are in use simultaneously.

When a switch with IGMP snooping enabled receives an IP group-specific IGMPv2 leave message, it sends a group-specific query out the interface where the leave message was received to determine if there are any other hosts attached to that interface that are interested in the MAC multicast group. If the switch does not receive an IGMP join message within the query-response-interval and none of the other 31 IP groups corresponding to the MAC group are interested in the multicast traffic for that MAC group and no multicast routers have been learned on the interface, then the interface is removed from the portmask of the (mac-group, vlan) entry in the L2 forwarding table. With fast-leave enabled on the VLAN, an interface can be removed immediately from the portmask of the L2 entry when the IGMP leave message is received, unless a multicast router was learned on the port

Note Use fast-leave processing only on VLANs where only one host is connected to each interface. If fast-leave is enabled in VLANs where more than one host is connected to an interface, some hosts might be dropped inadvertently. Fast leave processing is supported only with IGMP version 2 hosts.

Storm-Control

Understanding Storm-Control

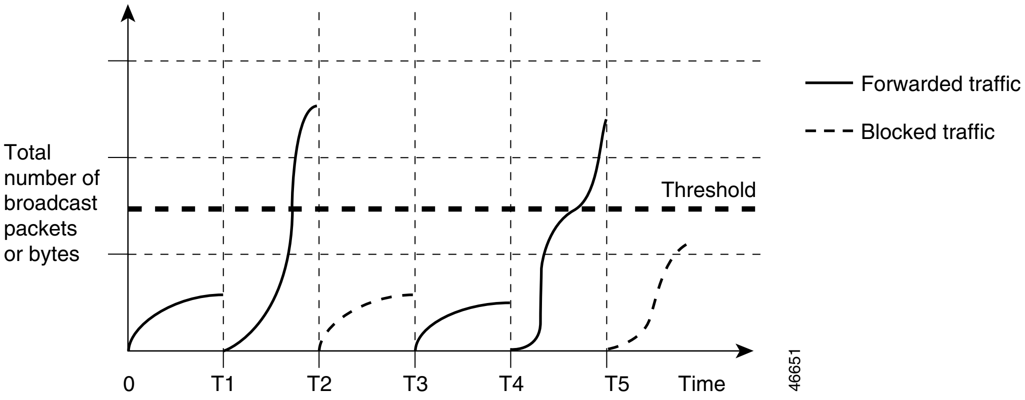
Storm-control prevents switchports on a LAN from being disrupted by a broadcast, multicast, or unicast storm on one of the interfaces. A LAN storm occurs when packets flood the LAN, creating excessive traffic and degrading network performance. Errors in the protocol-stack implementation or in the network configuration can cause a storm.

Storm-control monitors incoming traffic statistics over a time period and compares the measurement with a predefined suppression level threshold. The threshold represents the percentage of the total available bandwidth of the port. If the threshold of a traffic type is reached, further traffic of that type is suppressed until the incoming traffic falls below the threshold level. Storm-control is disabled by default.

The switch supports storm-control for broadcast, multicast, and unicast traffic. This example of broadcast suppression can also be applied to multicast and unicast traffic.

The graph in Figure 8 shows broadcast traffic patterns on an interface over a given period of time. In this example, the broadcast traffic exceeded the configured threshold between time intervals T1 and T2 and between T4 and T5. When the amount of specified traffic exceeds the threshold, all traffic of that kind is dropped. Therefore, broadcast traffic is blocked during those intervals. At the next time interval, if broadcast traffic does not exceed the threshold, it is again forwarded.

Figure 8 Broadcast Suppression Example



When storm-control is enabled, the switch monitors packets passing from an interface to the switching bus and determines if the packet is unicast, multicast, or broadcast. The switch monitors the number of broadcast, multicast, or unicast packets received within the 1-second time interval, and when a threshold for one type of traffic is reached, that type of traffic is dropped. This threshold is specified as a percentage of total available bandwidth that can be used by broadcast (multicast or unicast) traffic.

The combination of broadcast suppression threshold numbers and the 1-second time interval control the way the suppression algorithm works. A higher threshold allows more packets to pass through. A threshold value of 100 percent means that no limit is placed on the traffic.

Note Because packets do not arrive at uniform intervals, the 1-second time interval during which traffic activity is measured can affect the behavior of storm-control.

The switch continues to monitor traffic on the port, and when the utilization level is below the threshold level, the type of traffic that was dropped is forwarded again.

You use the **storm-control broadcast**, **storm-control multicast**, and **storm-control unicast** interface configuration commands to set up the storm-control threshold value.

Port Security

You can use port security to block input to an Ethernet, Fast Ethernet, or Gigabit Ethernet port when the MAC address of the station attempting to access the port is different from any of the MAC addresses specified for that port. Alternatively, you can use port security to filter traffic destined to or received from a specific host based on the host MAC address.

Ethernet Switching in Cisco AVVID Architecture

This section describes the Ethernet switching capabilities of the Ethernet switch network module, which is designed to work as part of the Cisco Architecture for Voice, Video, and Integrated Data (AVVID) solution.

The section outlines how to configure ethernet ports on the Ethernet switch network module to support Cisco IP phones in a branch office on your network. Also included is a section on performing basic tasks on the Ethernet switch network module.

The following topics are included:

- Configuring the Ethernet switch network module for Cisco AVVID/IP Telephony, page 25
- Default Switch Configuration, page 25

Configuring the Ethernet switch network module for Cisco AVVID/IP Telephony

The Ethernet switch network module has sixteen 10/100 switched Ethernet ports with integrated inline power and QoS features that make it an ideal choice for extending Cisco AVVID (Architecture for Voice, Video and Integrated Data) based voice-over-IP (VoIP) networks to small branch offices.

As an access gateway switch, the Ethernet switch network module can be deployed as a component of a centralized call-processing network using a centrally deployed Cisco CallManager (CCM). Instead of deploying and managing key systems or PBXs in small branch offices, applications are centrally located at the corporate headquarters or data center and are accessed via the IP WAN.

Default Switch Configuration

By default, the Ethernet switch network module provides the following settings with respect to Cisco AVVID:

- All switch ports are in access VLAN 1.
- All switch ports are static access ports, not 802.1Q trunk ports.
- Default voice VLAN is not configured on the switch.
- Inline power is automatically supplied on the 10/100 ports.

Stacking

Multiple switch modules may be installed simultaneously by connecting the Gigabit Ethernet (GE) ports of the Cisco EtherSwitch network module. This connection sustains a line-rate traffic similar to the switch fabric found in Cisco Catalyst switches and forms a single VLAN consisting of all ports in multiple Cisco EtherSwitch network modules. The stacking port must be configured for multiple switch modules to operate correctly in the same chassis.

- MAC address entries learned via intrachassis stacking are not displayed.
- Link status of intrachassis stacked ports are filtered.

For more details about the requirements for installing and connecting Cisco EtherSwitch network modules in a single chassis, go to the following URL:

http://www.cisco.com/univercd/cc/td/doc/product/access/acs_mod/cis2600/hw_inst/nm_inst/nm-doc/connswh.htm

Flow Control

Flow-control is a feature that Gigabit Ethernet ports use to inhibit the transmission of incoming packets. If a buffer on a Gigabit Ethernet port runs out of space, the port transmits a special packet that requests remote ports to delay sending packets for a period of time. This special packet is called a pause frame.

Using Flow-Control Keywords

Table 3 describes guidelines for using different configurations of the **send** and **receive** keywords with the **set port flowcontrol** command.

Table 3 Gigabit Ethernet Flow-Control Keyword Functions

Configuration	Description
send on	Enables a local port to send pause frames to a remote port. Use **send on** when a remote port is set to **receive on** or **receive desired**.
send off	Prevents a local port from sending pause frames to a remote port. Use **send off** when a remote port is set to **receive off** or **receive desired**.
send desired	Indicates preference to send pause frames, but autonegotiates flow control. You can use **send desired** when a remote port is set to **receive on**, **receive off**, or **receive desired**.
receive on	Enables a local port to process pause frames that a remote port sends. Use **receive on** when a remote port is set to **send on** or **send desired**.
receive off	Prevents a local port from processing pause frames. Use **receive off** when a remote port is set to **send off** or **send desired**.
receive desired	Indicates preference to process pause frames, but autonegotiates flow control. You can use **receive desired** when a remote port is set to **send on**, **send off**, or **send** desired.

Benefits

- Statistical gains by combining multiple traffic types over a common IP infrastructure.
- Long distance savings
- Support for Intra-chassis stacking
- Voice connectivity over data applications
- IPSEC, ACL, VPN and Firewall options
- New broadband WAN options

Interface Range Specification feature makes configuration easier because:

- Identical commands can be entered once for a range of interfaces, rather than being entered separately for each interface.
- Interface ranges can be saved as macros.

Restrictions

The following features are not supported in this release:

- Enable or disable per port based on unknown unicast or multicast flooding
- CGMP client, CGMP fast-leave
- Dynamic access ports
- Dynamic trunk protocol
- Dynamic VLANs
- GARP, GMRP and GVRP
- Inter-chassis stacking
- ISL tagging, the chip does not support ISL.
- Layer 3 switching onboard
- Monitoring of VLANs
- Multi-VLAN ports Network Port
- Shared STP instances
- STP backbone fast
- STP uplink fast for clusters
- VLAN-based SPAN
- VLAN Query Protocol
- VTP pruning protocol
- Web-based management interface

Related Features and Technologies

- IP Phone Telephony
- Voice over IP (VoIP)
- Wireless LAN

Related Documents

For information about installing voice network modules and voice interface cards in Cisco 2600 series, Cisco 3600 series, and Cisco 3700 series routers see these publications:

- *Cisco 2600 Series Modular Routers Quick Start Guide*
- Cisco 2600 Series Hardware Installation Guide
- *Quick Start Guides for Cisco 3600 series routers*
- *Cisco 3600 Series Hardware Installation Guide*
- Quick start guides for Cisco 3700 series routers
- Hardware installation documents for Cisco 3700 series
- *WAN Interface Card Hardware Installation Guide*

For information about configuring Voice over IP features, see these publications:

- Cisco 2600 Series Software Configuration Guide
- *Cisco IOS Voice, Video, and Fax Configuration Guide*, Release 12.2
- *Cisco IOS Voice, Video, and Fax Command Reference*, Release 12.2

For more information on Flow control, see the following publication:

- *Configuring Gigabit Ethernet Switching*

Supported Platforms

- Cisco 2600 series
- Cisco 3600 series
- Cisco 3700 series

Determining Platform Support Through Cisco Feature Navigator

Cisco IOS software is packaged in feature sets that support specific platforms. To get updated information regarding platform support for this feature, access Cisco Feature Navigator. Cisco Feature Navigator dynamically updates the list of supported platforms as new platform support is added for the feature.

Cisco Feature Navigator is a web-based tool that enables you to determine which Cisco IOS software images support a specific set of features and which features are supported in a specific Cisco IOS image. You can search by feature or release. Under the release section, you can compare releases side by side to display both the features unique to each software release and the features in common.

To access Cisco Feature Navigator, you must have an account on Cisco.com. If you have forgotten or lost your account information, send a blank e-mail to cco-locksmith@cisco.com. An automatic check will verify that your e-mail address is registered with Cisco.com. If the check is successful, account details with a new random password will be e-mailed to you. Qualified users can establish an account on Cisco.com by following the directions at http://www.cisco.com/register.

Cisco Feature Navigator is updated regularly when major Cisco IOS software releases and technology releases occur. For the most current information, go to the Cisco Feature Navigator home page at the following URL:

http://www.cisco.com/go/fn

Availability of Cisco IOS Software Images

Platform support for particular Cisco IOS software releases is dependent on the availability of the software images for those platforms. Software images for some platforms may be deferred, delayed, or changed without prior notice. For updated information about platform support and availability of software images for each Cisco IOS software release, refer to the online release notes or, if supported, Cisco Feature Navigator.

Supported Standards, MIBs, and RFCs

Standards

- 802.1d

- 802.1p
- 802.1q

MIBs

- RFC 1213
- IF MIB
- RFC 2037 ENTITY MIB
- CISCO-CDP-MIB
- CISCO-IMAGE-MIB
- CISCO-FLASH-MIB
- OLD-CISCO-CHASSIS-MIB
- CISCO-VTP-MIB
- CISCO-HSRP-MIB
- OLD-CISCO-TS-MIB
- CISCO-ENTITY-ASSET-MIB
- CISCO-ENTITY-FRU-CONTROL-MIB
- BRIDGE MIB (RFC 1493)
- CISCO-VLAN-MEMBERSHIP-MIB
- CISCO-VLAN-IFINDEX-RELATIONSHIP-MIB
- RMON1-MIB
- PIM-MIB
- CISCO-STP-EXTENSIONS-MIB
- OSPF MIB (RFC 1253)
- CISCO-VLAN-BRIDGE-MIB
- IPMROUTE-MIB
- CISCO-MEMORY-POOL-MIB
- ETHER-LIKE-MIB (RFC 1643)
- CISCO-ENTITY-FRU-CONTROL-MIB.my
- CISCO-RTTMON-MIB
- CISCO-PROCESS-MIB
- CISCO-COPS-CLIENT-MIB

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

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

RFCs

No new or modified RFCs are supported by this feature.s

Prerequisites

- Cisco IOS Release 12.2 or later release
- Basic configuration of the Cisco 2600 series, Cisco 3600 series, or Cisco 3700 series router

In addition, complete the following tasks before configuring this feature:

- Configure IP routing

  For more information on IP routing, refer to the *Cisco IOS IP Configuration Guide,* Release 12.2.
- Set up the call agents

  For more information on setting up call agents, refer to the documentation that accompanies the call agents used in your network configuration.

Configuration Tasks

See the following sections for configuration tasks for the Ethernet switch network module.

Configuring Layer 2 Interfaces

- Configuring a Range of Interfaces (required)
- Defining a Range Macro (optional)
- Configuring Layer 2 Optional Interface Features (optional)
- Configuring an Ethernet Interface as a Layer 2 Trunk (optional)
- Configuring an Ethernet Interface as a Layer 2 Access (optional)

Configuring a Range of Interfaces

To configure a range of interfaces, use the **interface range** command in global configuration mode:

	Command	Purpose
Step 1	`Router(config)#` **`interface range`** `{`**`vlan`** *`vlan_ID - vlan_ID`*`}	{{`**`ethernet`**`	`**`fastethernet`**`	`**`macro`** *`macro_name`*`}` *`slot/interface - interface`*`}[, {{`**`ethernet`**`	`**`fastethernet`**`	`**`macro`** *`macro_name`*`}` *`slot/interface - interface`*`}]`	Select the range of interfaces to be configured. • The space before the dash is required. For example, the command **interface range fastethernet 1 - 5** is valid; the command **interface range fastethernet 1-5** is not valid. • You can enter one macro or up to five comma-separated ranges. • Comma-separated ranges can include both VLANs and physical interfaces. • You are not required to enter spaces before or after the comma. • The **interface range** command only supports VLAN interfaces that are configured with the **interface vlan** command.

Defining a Range Macro

To define an interface range macro, use the **define interface-range** command in global configuration mode:

	Command	Purpose
Step 1	Router(config)# **define interface-range** *macro_name* {**vlan** *vlan_ID - vlan_ID*} | {{**ethernet** | **fastethernet**} *slot/interface - interface*} [, {{**ethernet** | **fastethernet**} *slot/interface - interface*}]	Define the interface-range macro and save it in NVRAM.

Verifying Configuration of a Range of Interfaces

Step 1 Use the **show running-configuration** command to show the defined interface-range macro configuration, as illustrated below:

```
Router# show running-configuration | include define define interface-range enet_list
FastEthernet5/1 - 4
```

Configuring Layer 2 Optional Interface Features

- Interface Speed and Duplex Configuration Guidelines, page 32
- Configuring the Interface Speed, page 32
- Configuring the Interface Duplex Mode, page 33
- Configuring a Description for an Interface, page 34
- Configuring an Ethernet Interface as a Layer 2 Trunk, page 34
- Configuring an Ethernet Interface as a Layer 2 Access, page 35

Interface Speed and Duplex Configuration Guidelines

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

- If both ends of the line support autonegotiation, Cisco highly recommends the default autonegotiation settings.
- 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.
- Both ends of the line need to be configured to the same setting. For example, both hard-set or both auto-negotiate. Mismatched settings are not supported.

Caution Changing the interface speed and duplex mode configuration might shut down and reenable the interface during the reconfiguration.

Configuring the Interface Speed

To set the interface speed, use the **interface fastethernet** command in global configuration mode:

	Command	Purpose
Step 1	`Router(config)#` **`interface fastethernet`** *`slot/interface`*	Select the interface to be configured.
Step 2	`Router(config-if)#` **`speed`** `[`**`10`** `|` **`100`** `|` **`auto`**`]`	Set the interface speed of the interface.

Note If you set the interface speed to auto on a 10/100-Mbps Ethernet interface, both speed and duplex are autonegotiated.

Configuring the Interface Duplex Mode

To set the duplex mode of an Ethernet or Fast Ethernet interface, use the following commands beginning in global configuration mode:

	Command	Purpose
Step 1	`Router(config)# interface fastethernet slot/interface`	Selects the interface to be configured.
Step 2	`Router(config-if)# duplex [auto | full | half]`	Sets the duplex mode of the interface.

Note If you set the port speed to auto on a 10/100-Mbps Ethernet interface, both speed and duplex are autonegotiated. You cannot change the duplex mode of autonegotiation interfaces.

The following example shows how to set the interface duplex mode to full on Fast Ethernet interface 5/4:

```
Router(config)# interface fastethernet 5/4
Router(config-if)# duplex full
```

Verifying Interface Speed and Duplex Mode Configuration

Step 1 Use the **show interfaces** command to verify the interface speed and duplex mode configuration for an interface, as illustrated below:

```
Router# show interfaces fastethernet 1/4

FastEthernet1/4 is up, line protocol is down
  Hardware is Fast Ethernet, address is 0000.0000.0c89 (bia 0000.0000.0c89)
  MTU 1500 bytes, BW 100000 Kbit, DLY 100 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation ARPA, loopback not set
  Keepalive set (10 sec)
  Auto-duplex, Auto-speed
  ARP type: ARPA, ARP Timeout 04:00:00
  Last input never, output never, output hang never
  Last clearing of "show interface" counters never
  Queueing strategy: fifo
  Output queue 0/40, 0 drops; input queue 0/75, 0 drops
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
     0 packets input, 0 bytes, 0 no buffer
   Received 0 broadcasts, 0 runts, 0 giants, 0 throttles
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored
     0 input packets with dribble condition detected
     3 packets output, 1074 bytes, 0 underruns(0/0/0)
     0 output errors, 0 collisions, 5 interface resets
     0 babbles, 0 late collision, 0 deferred
     0 lost carrier, 0 no carrier
     0 output buffer failures, 0 output buffers swapped out
Router#
```

Configuring 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 the following commands: **show configuration**, **show running-config**, and **show interfaces**.

To add a description for an interface, use the **description** command in interface configuration mode:

	Command	Purpose
Step 1	`Router(config-if)#` **`description`** *`string`*	Adds a description for an interface.

Configuring an Ethernet Interface as a Layer 2 Trunk

To configure an Ethernet interface as a Layer 2 trunk, use the following commands, beginning in global configuration mode:

	Command	Purpose
Step 1	`Router(config)#` **`interface`** `{`**`ethernet`** `|` **`fastethernet`**`}` *`slot/port`*	Selects the interface to configure.
Step 2	`Router(config-if)#` **`shutdown`**	(Optional) Shuts down the interface to prevent traffic flow until configuration is complete. **Note** Encapsulation is always dot1q.
Step 3	`Router(config-if)#` **`switch port mode trunk`**	Configures the interface as a Layer 2 trunk.
Step 4	`Router(config-if)#` **`switch port trunk native vlan`** *`vlan_num`*	For 802.1Q trunks, specifies the native VLAN.
Step 5	`Router(config-if)#` **`switch port trunk allowed vlan`** `{`**`add`** `|` **`except`** `|` **`none`** `|` **`remove`**`}` *`vlan1`*`[,`*`vlan`*`[,`*`vlan`*`[,...]]`	(Optional) Configures the list of VLANs allowed on the trunk. All VLANs are allowed by default. You cannot remove any of the default VLANs from a trunk.
Step 6	`Router(config-if)#` **`no shutdown`**	Activates the interface. (Required only if you shut down the interface.)
Step 7	`Router(config-if)#` **`end`**	Exits configuration mode.

Note Ports do not support Dynamic Trunk Protocol (DTP). Ensure that the neighboring switch is set to a mode that will not send DTP.

Verifying an Ethernet Interface as a Layer 2 Trunk

Step 1 Use the show commands to verify the configuration of an Ethernet interface as a Layer 2 trunk, as illustrated below:

```
Router# show running-config interface fastethernet 5/8

Building configuration...
Current configuration:
!
interface FastEthernet5/8
 no ip address
 switchport
 switchport trunk encapsulation dot1q
end
```

Step 2

```
Router# show interfaces fastethernet 5/8 switchport

Name: Fa5/8
Switchport: Enabled
Administrative Mode: static access
Operational Mode: static access
Administrative Trunking Encapsulation: dot1q
Operational Trunking Encapsulation: native
Negotiation of Trunking: Disabled
Access Mode VLAN: 1 (default)
Trunking Native Mode VLAN: 1 (default)
Trunking VLANs Enabled: ALL
Pruning VLANs Enabled: 2-1001
Protected: false
Unknown unicast blocked: false
Unknown multicast blocked: false
Broadcast Suppression Level: 100
Multicast Suppression Level: 100
Unicast Suppression Level: 100
Voice VLAN: none
Appliance trust: none
```

Step 3

```
Router# show interfaces fastethernet 5/8 trunk

Port      Mode         Encapsulation  Status        Native vlan
Fa1/15    off          802.1q         not-trunking  1
Port      Vlans allowed on trunk
Fa1/15    1
Port      Vlans allowed and active in management domain
Fa1/15    1
Port      Vlans in spanning tree forwarding state and not pruned
Fa1/15    1
```

Configuring an Ethernet Interface as a Layer 2 Access

To configure an Ethernet Interface as a Layer 2 access use the following commands, beginning in global configuration mode:

	Command	Purpose
Step 1	`Router(config)#` **`interface {ethernet	fastethernet}`** *`slot/port`*	Selects the interface to configure.
Step 2	`Router(config-if)#` **`shutdown`**	(Optional) Shuts down the interface to prevent traffic flow until configuration is complete. Encapsulation is always dot1q.
Step 3	`Router(config-if)#` **`switchport mode access`**	Configures the interface as a Layer 2 access.
Step 4	`Router(config-if)#` **`switchport access vlan`** *`vlan_num`*	For access ports, specifies the access vlan.
Step 5	`Router(config-if)#` **`no shutdown`**	Activates the interface. (Required only if you shut down the interface.)
Step 6	`Router(config-if)#` **`end`**	Exits configuration mode.

Verifying an Ethernet Interface as a Layer 2 Access

Step 1 Use the **show running-config interface** command to verify the running configuration of the interface, as illustrated below:

```
Router# show running-config interface {ethernet | fastethernet} slot/port
```

Step 1 Use the **show interfaces** command to verify the switch port configuration of the interface, as illustrated below:

```
Router# show interfaces [ethernet | fastethernet] slot/port switchport
```

Configuring VLANs

This section describes how to configure the VLANs on the Ethernet switch network modules, and contains the following sections:

- Configuring VLANs (optional)
- Deleting a VLAN from the Database (optional)

Configuring VLANs

To configure an Ethernet Interface as a Layer 2 access, use the following commands beginning in EXEC mode:

	Command	Purpose
Step 1	`Router# vlan database`	Enters VLAN configuration mode.
Step 2	`Router(vlan)# vlan vlan_id`	Adds an Ethernet VLAN.
Step 3	`Router(vlan)# exit`	Updates the VLAN database, propagates it throughout the administrative domain, and returns to privileged EXEC mode.

Verifying the VLAN Configuration.

Step 1 Use the **show vlan name** command to verify the VLAN configuration, as illustrated below:

```
Router# show vlan name VLAN0003

VLAN Name                             Status    Ports
---- -------------------------------- --------- -------------------------------
1    default                          active    Fa1/0, Fa1/1, Fa1/2, Fa1/3
                                                Fa1/4, Fa1/5, Fa1/6, Fa1/7
                                                Fa1/8, Fa1/9, Fa1/10, Fa1/11
                                                Fa1/12, Fa1/13, Fa1/14, Fa1/15
1002 fddi-default                     active
1003 token-ring-default               active
1004 fddinet-default                  active
1005 trnet-default                    active
```

```
VLAN Type  SAID       MTU   Parent RingNo BridgeNo Stp  BrdgMode Trans1 Trans2
---- ----- ---------- ----- ------ ------ -------- ---- -------- ------ ------
1    enet  100001     1500  -      -      -        -    -        1002   1003
1002 fddi  101002     1500  -      -      -        -    -        1      1003
1003 tr    101003     1500  1005   0      -        -    srb      1      1002
1004 fdnet 101004     1500  -      -      1        ibm  -        0      0
1005 trnet 101005     1500  -      -      1        ibm  -        0      0
Router#
```

Deleting a VLAN from the Database

When you delete a VLAN from a switch that is in VTP server mode, the VLAN is removed from all switches in the VTP domain. When you delete a VLAN from a switch that is in VTP transparent mode, the VLAN is deleted only on that specific switch.

You cannot delete the default VLANs for the different media types: Ethernet VLAN 1 and FDDI or Token Ring VLANs 1002 to 1005.

To delete a VLAN from the database, use the following commands beginning in privileged EXEC mode:

	Command	Purpose
Step 1	`Router# `**`vlan database`**	Enters VLAN configuration mode.
Step 2	`Router(vlan)# `**`no vlan`** *`vlan_id`*	Deletes the VLAN.
Step 3	`Router(vlan)# `**`exit`**	Updates the VLAN database, propagate it throughout the administrative domain, and return to privileged EXEC mode.

Verifying VLAN Deletion

Step 1 Use the **show vlan-switch brief** command to verify that a VLAN has been deleted from a switch, as illustrated below:

```
Router# show vlan-switch brief

VLAN Name                             Status    Ports
---- -------------------------------- --------- -------------------------------
1    default                          active    Fa0/2, Fa0/9, Fa0/14, Gi0/0
2    VLAN0002                         active
3    VLAN0003                         active    Fa0/4, Fa0/5, Fa0/10, Fa0/11
4    VLAN0004                         active    Fa0/6, Fa0/7, Fa0/12, Fa0/13
5    VLAN0005                         active
40   VLAN0040                         active    Fa0/15
50   VLAN0050                         active
1000 VLAN1000                         active
1002 fddi-default                     active
1003 token-ring-default               active
1004 fddinet-default                  active
1005 trnet-default                    active
Router#
```

Configuring VLAN Trunking Protocol (VTP)

This section describes how to configure the VLAN Trunking Protocol (VTP) on the Ethernet switch network module, and contains the following sections:

- Configuring VTP Server, page 38
- Configuring a VTP Client, page 38
- Disabling VTP (VTP Transparent Mode), page 39
- Configuring VTP version 2, page 39

Configuring VTP Server

When a switch is in VTP server mode, you can change the VLAN configuration and have it propagate throughout the network.

To configure the switch as a VTP server, use the following commands beginning in privileged EXEC mode:

	Command	Purpose
Step 1	`Router#` **`vlan database`**	Enters VLAN configuration mode.
Step 2	`Router(vlan)#` **`vtp server`**	Configures the switch as a VTP server.
Step 3	`Router(vlan)#` **`vtp domain`** *`domain_name`*	Defines the VTP domain name, which can be up to 32 characters long.
Step 4	`Router(vlan)#` **`vtp password`** *`password_value`*	(Optional) Sets a password, which can be from 8 to 64 characters long, for the VTP domain.
Step 5	`Router(vlan)#` **`exit`**	Exits VLAN configuration mode.

Configuring a VTP Client

When a switch is in VTP client mode, you cannot change the VLAN configuration on the switch. The client switch receives VTP updates from a VTP server in the management domain and modifies its configuration accordingly.

To configure the switch as a VTP client, use the following commands beginning in privileged EXEC mode:

	Command	Purpose
Step 1	`Router#` **`vlan database`**	Enters VLAN configuration mode.
Step 2	`Router(vlan)#` **`vtp client`**	Configures the switch as a VTP client.
Step 3	`Router(vlan)#` **`exit`**	Exits VLAN configuration mode.

Disabling VTP (VTP Transparent Mode)

When you configure the switch as VTP transparent, you disable VTP on the switch. A VTP transparent switch does not send VTP updates and does not act on VTP updates received from other switches. However, a VTP transparent switch running VTP version 2 does forward received VTP advertisements out all of its trunk links.

To disable VTP on the switch, use the following commands beginning in privileged EXEC mode:

	Command	Purpose
Step 1	`Router# vlan database`	Enters VLAN configuration mode.
Step 2	`Router(vlan)# vtp transparent`	Configures VTP transparent mode.
Step 3	`Router(vlan)# exit`	Exits VLAN configuration mode.

Configuring VTP version 2

To enable VTP version 2, use the following commands beginning in privileged EXEC mode:

	Command	Purpose
Step 1	`Router# vlan database`	Enters VLAN configuration mode.
Step 2	`Router(vlan)# [no] vtp v2-mode`	Enables VTP version 2. Use the **no** keyword to disable VTP version 2.
Step 3	`Router(vlan)# exit`	Exits VLAN configuration mode.

Verifying VTP

Step 1 Use the **show vtp status** to verify VTP status, as illustrated below:

```
Router# show vtp status

VTP Version                     : 2
Configuration Revision          : 247
Maximum VLANs supported locally : 1005
Number of existing VLANs        : 33
VTP Operating Mode              : Client
VTP Domain Name                 : Lab_Network
VTP Pruning Mode                : Enabled
VTP V2 Mode                     : Disabled
VTP Traps Generation            : Disabled
MD5 digest                      : 0x45 0x52 0xB6 0xFD 0x63 0xC8 0x49 0x80
Configuration last modified by 0.0.0.0 at 8-12-99 15:04:49
Router#
```

Configuring Layer 2 EtherChannels (Port-Channel Logical Interfaces)

Configuring Layer 2 EtherChannels (Port-Channel Logical Interfaces)

To configure Layer 2 EtherChannels, configure the Ethernet interfaces with the **channel-group** command, which creates the port-channel logical interface.

Note Cisco IOS software creates port-channel interfaces for Layer 2 EtherChannels when you configure Layer 2 Ethernet interfaces with the **channel-group** command. You cannot put Layer 2 Ethernet interfaces into a manually created port-channel interface.

Note Layer 2 interfaces must be connected and functioning for Cisco IOS software to create port-channel interfaces for Layer 2 EtherChannels.

To configure Layer 2 Ethernet interfaces as a Layer 2 EtherChannel, use the following commands beginning in global configuration mode for each interface:

	Command	Purpose
Step 1	`Router(config)#` **`interface fastethernet`** *`slot/port`*	Selects a physical interface to configure.
Step 2	`Router(config-if)#` **`channel-group`** *`port_channel_number`* **`mode`** `{`**`on`**`}`	Configures the interface in a port-channel.
Step 3	`Router(config-if)#` **`end`**	Exits configuration mode.

Verifying Layer 2 EtherChannels

Use the following **show** commands to verify Layer 2 EtherChannels, as illustrated below:

Step 1

```
Router# show running-config interface fastethernet 5/6
Building configuration...

Current configuration:
!
interface FastEthernet5/6
 no ip address
 switchport
 switchport access vlan 10
 switchport mode access
 channel-group 2 mode on
end
```

Step 2

```
Router# show interfaces fastethernet 5/6 etherchannel

Port state    = EC-Enbld Up In-Bndl Usr-Config
Channel group = 2           Mode = Desirable     Gcchange = 0
Port-channel  = Po2         GC   = 0x00020001
Port indx     = 1           Load = 0x55

Flags:  S - Device is sending Slow hello.  C - Device is in Consistent state.
        A - Device is in Auto mode.        P - Device learns on physical port.
Timers: H - Hello timer is running.        Q - Quit timer is running.
        S - Switching timer is running.    I - Interface timer is running.
Local information:
                                Hello    Partner  PAgP     Learning  Group
Port      Flags State   Timers  Interval Count   Priority   Method  Ifindex
Fa5/6     SC    U6/S7           30s      1        128         Any      56

Partner’s information:

          Partner             Partner          Partner         Partner Group
Port      Name                Device ID        Port       Age  Flags   Cap.
Fa5/6     JAB031301           0050.0f10.230c   2/47        18s SAC     2F

Age of the port in the current state: 00h:10m:57s
```

Step 3

```
Router# show running-config interface port-channel 2

Building configuration...

Current configuration:
!
interface Port-channel2
 no ip address
 switchport
 switchport access vlan 10
 switchport mode access
end

Router#
```

Step 4

```
Router# show etherchannel 2 port-channel

                Port-channels in the group:
                ----------------------

Port-channel: Po2
------------

Age of the Port-channel   = 00h:23m:33s
Logical slot/port   = 10/2           Number of ports in agport = 2
GC                  = 0x00020001      HotStandBy port = null
Port state          = Port-channel Ag-Inuse

Ports in the Port-channel:

Index   Load   Port
-------------------
  1     55      Fa5/6
  0     AA      Fa5/7
```

```
Time since last port bundled:    00h:23m:33s    Fa5/6
```

Configuring EtherChannel Load Balancing

To configure EtherChannel load balancing, use the following command in global configuration mode:

	Command	Purpose
Step 1	`Router(config)#` **`port-channel load-balance`** `{`**`src-mac`** `|` **`dst-mac`** `|` **`src-dst-mac`** `|` **`src-ip`** `|` **`dst-ip`** `|` **`src-dst-ip`**`}`	Configures EtherChannel load balancing, use the **no** form of this command to return EtherChannel load balancing to the default configuration.
Step 2	`Router(config)#` **`end`**	Exits configuration mode.

Note For new load balancing to take affect, the EtherChannel must be first configured to the default configuration.

Verifying EtherChannel Load Balancing

Step 1 Use the **show etherchannel load-balance** to verify Layer 2 EtherChannel load balancing, as illustrated below:

```
Router# show etherchannel load-balance

Source XOR Destination IP address
Router#
```

Removing an Interface from an EtherChannel

To remove an Ethernet interface from an EtherChannel, use the following commands in global configuration mode:

	Command	Purpose
Step 1	`Router(config)# [`**`no`**`]` **`port-channel load-balance`** `{`**`src-mac`** `|` **`dst-mac`** `|` **`src-dst-mac`** `|` **`src-ip`** `|` **`dst-ip`** `|` **`src-dst-ip`**`}`	Configures EtherChannel load balancing. Use the **no** keyword to return EtherChannel load balancing to the default configuration.
Step 2	`Router(config)#` **`end`**	Exits configuration mode.

Configuring Removing an EtherChannel

To remove an EtherChannel, use the following command in global configuration mode:

	Command	Purpose
Step 1	`Router(config)# no interface port-channel port_channel_number`	Removes the port-channel interface.
Step 2	`Router(config)# end`	Exits configuration mode.

Verify Removing an EtherChannel

Step 1 Use the **show etherchannel summary** command to verify that the Etherchannel is removed, as illustrated below:

```
Router# show etherchannel summary

Flags:  D - down        P - in port-channel
        I - stand-alone s - suspended
        R - Layer3      S - Layer2
        U - in use
Group Port-channel  Ports
-----+------------+-----------------------------------------------------------

Router#
```

Configuring Spanning Tree

- Enabling Spanning Tree, page 43
- Configuring Spanning Tree Port Priority, page 44
- Configuring Spanning Tree Port Cost, page 45
- Configuring the Bridge Priority of a VLAN, page 46
- Configuring the Hello Time, page 46
- Configuring the Forward-Delay Time for a VLAN, page 47
- Configuring the Maximum Aging Time for a VLAN, page 47
- Disabling Spanning Tree, page 48

Enabling Spanning Tree

You can enable spanning tree on a per-VLAN basis. The switch maintains a separate instance of spanning tree for each VLAN (except on VLANs on which you disable spanning tree).

To enable spanning tree on a per-VLAN basis, use the following command in global configuration mode:

	Command	Purpose
Step 1	`Router(config)# spanning-tree vlan vlan_ID`	Enables spanning tree on a per-VLAN basis.
Step 2	`Router(config)# end`	Exits configuration mode.

Verify Spanning Tree

Step 1 Use the **show spanning-tree vlan** to verify spanning tree configuration, as illustrated below:

```
Router# show spanning-tree vlan 200
 VLAN200 is executing the ieee compatible Spanning Tree protocol
  Bridge Identifier has priority 32768, address 0050.3e8d.6401
  Configured hello time 2, max age 20, forward delay 15
  Current root has priority 16384, address 0060.704c.7000
  Root port is 264 (FastEthernet5/8), cost of root path is 38
  Topology change flag not set, detected flag not set
  Number of topology changes 0 last change occurred 01:53:48 ago
  Times:  hold 1, topology change 24, notification 2
          hello 2, max age 14, forward delay 10
  Timers: hello 0, topology change 0, notification 0


 Port 264 (FastEthernet5/8) of VLAN200 is forwarding
   Port path cost 19, Port priority 128, Port Identifier 129.9.
   Designated root has priority 16384, address 0060.704c.7000
   Designated bridge has priority 32768, address 00e0.4fac.b000
   Designated port id is 128.2, designated path cost 19
   Timers: message age 3, forward delay 0, hold 0
   Number of transitions to forwarding state: 1
   BPDU: sent 3, received 3417

Router#
```

Configuring Spanning Tree Port Priority

To configure the spanning tree port priority of an interface, use the following commands beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# **interface** {{**ethernet** | **fastethernet**} *slot/port}* | *{port-channel port_channel*_number}	Selects an interface to configure.
Step 2	Router(config-if)# [**no**] **spanning-tree port-priority** *port_priority*	Configures the port priority for an interface. The of port_priority value can be from 1 to 255 in increments of 4. Use the **no** form of this command to restore the defaults.
Step 3	Router(config-if)# [**no**] **spanning-tree vlan** *vlan_ID* **port-priority** *port_priority*	Configures the VLAN port priority for an interface. The port_priority value can be from 1 to 255 in increments of 4. Use the **no** form of this command to restore the defaults.
Step 4	Router(config-if)# **end**	Exits configuration mode.

Verify Spanning Tree Port Priority

Step 1 Use the **show spanning-tree interface** to verify spanning-tree interface and the spanning-tree port priority configuration, as illustrated below:

```
Router# show spanning-tree interface fastethernet 5/8

 Port 264 (FastEthernet5/8) of VLAN200 is forwarding
   Port path cost 19, Port priority 100, Port Identifier 129.8.
   Designated root has priority 32768, address 0010.0d40.34c7
   Designated bridge has priority 32768, address 0010.0d40.34c7
   Designated port id is 128.1, designated path cost 0
   Timers: message age 2, forward delay 0, hold 0
   Number of transitions to forwarding state: 1
   BPDU: sent 0, received 13513
Router#
```

Configuring Spanning Tree Port Cost

To configure the spanning tree port cost of an interface, use the following commands beginning in global configuration mode:

	Command	Purpose
Step 1	`Router(config)# interface {{ethernet | fastethernet} slot/port} | {port-channel port_channel_number}`	Selects an interface to configure.
Step 2	`Router(config-if)# [no] spanning-tree cost port_cost`	Configures the port cost for an interface. The value of port_cost can be from 1 to 200,000,000 (1 to 65,535 in Cisco IOS Releases 12.1(2)E and earlier). Use the **no** form of this command to restore the defaults.
Step 3	`Router(config-if)# [no] spanning-tree vlan vlan_ID cost port_cost`	Configures the VLAN port cost for an interface. The value port_cost can be from 1 to 65,535. Use the **no** form of this command to restore the defaults.
Step 4	`Router(config-if)# end`	Exits configuration mode.

Verifying Spanning Tree Port Cost

Step 1 Use the **show spanning-tree vlan** to verify the spanning-tree port cost configuration, as illustrated below:

```
Router# show spanning-tree vlan 200

!
!
!
Port 264 (FastEthernet5/8) of VLAN200 is forwarding
Port path cost 17, Port priority 64, Port Identifier 129.8.
   Designated root has priority 32768, address 0010.0d40.34c7
   Designated bridge has priority 32768, address 0010.0d40.34c7
   Designated port id is 128.1, designated path cost 0
   Timers: message age 2, forward delay 0, hold 0
   Number of transitions to forwarding state: 1
   BPDU: sent 0, received 13513
!
!
!
Router#
```

Configuring the Bridge Priority of a VLAN

Caution Exercise care when using this command. For most situations **spanning-tree vlan vlan_ID root primary** and the **spanning-tree vlan vlan_ID root secondary** are the preferred commands to modify the bridge priority.

To configure the spanning tree bridge priority of a VLAN, use the following command in global configuration mode:

	Command	Purpose
Step 1	`Router(config)# [`**`no`**`]` **`spanning-tree vlan`** *`vlan_ID`* **`priority`** *`bridge_priority`*	Configures the bridge priority of a VLAN. The bridge_priority value can be from 1 to 65535. Use the **no** keyword to restore the defaults.
Step 2	`Router(config)#` **`end`**	Exits configuration mode.

Verifying the Bridge Priority of a VLAN

Step 1 Use the **show spanning-tree vlan bridge** command to verify the bridge priority, as illustrated below:

```
Router# show spanning-tree vlan 200 bridge brief

    Hello Max  Fwd
Vlan                   Bridge ID      Time  Age Delay  Protocol
---------------- -------------------- ---- ---- -----  --------
VLAN200          33792 0050.3e8d.64c8    2   20    15  ieee
Router#
```

Configuring the Hello Time

To configure the hello interval for the spanning tree, use the following command in global configuration mode:

	Command	Purpose
Step 1	`Router(config)# [`**`no`**`]` **`spanning-tree vlan`** *`vlan_ID`* **`hello-time`** *`hello_time`*	Configures the hello time of a VLAN. The **hello_time** value can be from 1 to 10 seconds. Use the **no** form of this command to restore the defaults.
Step 2	`Router(config)#` **`end`**	Exits configuration mode.

Configuring the Forward-Delay Time for a VLAN

To configure the forward delay for the spanning tree, use the following command in global configuration mode:

	Command	Purpose
Step 1	`Router(config)# [no] spanning-tree vlan vlan_ID forward-time forward_time`	Configures the forward time of a VLAN. The value of **forward_time** can be from 4 to 30 seconds. Use the **no** form of this command to restore the defaults.
Step 2	`Router(config)# end`	Exits configuration mode.

Configuring the Maximum Aging Time for a VLAN

To configure the maximum age interval for the spanning tree, use the following command in global configuration mode:

	Command	Purpose
Step 1	`Router(config)# [no] spanning-tree vlan vlan_ID max-age max_age`	Configures the maximum aging time of a VLAN. The value of **max_age** can be from 6 to 40 seconds. Use the **no** form of this command to restore the defaults.
Step 2	`Router(config)# end`	Exits configuration mode.

Configuring the Root Bridge

The Ethernet switch network module maintains a separate instance of spanning tree for each active VLAN configured on the switch. A bridge ID, consisting of the bridge priority and the bridge MAC address, is associated with each instance. For each VLAN, the switch with the lowest bridge ID will become the root bridge for that VLAN.

To configure a VLAN instance to become the root bridge, the bridge priority can be modified from the default value (32768) to a significantly lower value so that the bridge becomes the root bridge for the specified VLAN. Use the spanning-tree vlan vlan-ID root command to alter the bridge priority.

The switch checks the bridge priority of the current root bridges for each VLAN. The bridge priority for the specified VLANs is set to 8192 if this value will cause the switch to become the root for the specified VLANs.

If any root switch for the specified VLANs has a bridge priority lower than 8192, the switch sets the bridge priority for the specified VLANs to 1 less than the lowest bridge priority.

For example, if all switches in the network have the bridge priority for VLAN 100 set to the default value of 32768, entering the spanning-tree vlan 100 root primary command on a switch will set the bridge priority for VLAN 100 to 8192, causing the switch to become the root bridge for VLAN 100.

Note Note The root switch for each instance of spanning tree should be a backbone or distribution switch. Do not configure an access switch as the spanning tree primary root.

Use the diameter keyword to specify the Layer 2 network diameter (that is, the maximum number of bridge hops between any two end stations in the Layer 2 network). When you specify the network diameter, the switch automatically picks an optimal hello time, forward delay time, and maximum age time for a network of that diameter, which can significantly reduce the spanning tree convergence time. You can use the hello keyword to override the automatically calculated hello time.

Note

Note We recommend that you avoid configuring the hello time, forward delay time, and maximum age time manually after configuring the switch as the root bridge.

To configure the switch as the root, use the following command in global configuration mode:

	Command	Purpose
Step 1	Router(config)# [**no**] **spanning-tree vlan** *vlan_ID* **root primary** [*diameter hops* [**hello-time** *seconds*]]	Configures a switch as the root switch. Use the **no** form of this command to restore the defaults.
Step 2	Router(config)# **end**	Exits configuration mode.

Disabling Spanning Tree

To disable spanning tree on a per-VLAN basis, use the following command in global configuration mode:

	Command	Purpose
Step 1	Router(config)# **no spanning-tree vlan** *vlan_ID*	Disables spanning tree on a per-VLAN basis.
Step 2	Router(config)# **end**	Exits configuration mode.

Verifying that Spanning Tree is Disabled.

Step 1 Use the **show spanning-tree vlan** to verify the that the spanning tree is disabled, as illustrated below:

```
Router# show spanning-tree vlan 200
<...output truncated...>
Spanning tree instance for VLAN 200 does not exist.
Router#
```

Configuring Mac Table Manipulation — Port Security

Port security is implemented by providing the user with the option to make a port secure by allowing only well-known MAC addresses to send in data traffic.

- Enabling Known MAC Address Traffic, page 49
- Creating a Static or Dynamic Entry in the MAC Address Table, page 49
- Configuring aging-timer, page 50

Enabling Known MAC Address Traffic

To enable the MAC address secure option, use the following commands beginning in privileged EXEC mode:

	Command	Purpose
Step 1	Router# **configure terminal**	Enters global configuration mode.
Step 2	Router# [**no**] **mac-address-table secure** *<mac-address>* **fastethernet** *slot/port* [**vlan** *<vlan id>*]	Secures the MAC address traffic on the port.
Step 3	Router(config)# **end**	Exits configuration mode.

Verifying the mac-address-table secure

Step 1 Use the **show mac-address-table secure** to verify the configuration, as illustrated below:

```
Router# show mac-address-table secure

Secure Address Table:
Destination Address  Address Type  VLAN  Destination Port
-------------------  ------------  ----  --------------------
0003.0003.0003          Secure 1  FastEthernet    2/8
```

Creating a Static or Dynamic Entry in the MAC Address Table

To create a static or dynamic entry in the mac address table, use the following commands beginning in privileged EXEC mode:

	Command	Purpose
Step 1	Router# **configure terminal**	Enters global configuration mode.
Step 2	Router# **mac-address-table** [**dynamic** / **static**] *mac-address* **fastethernet** *slot/port* [**vlan** *<vlan id>*]	Creates static or dynamic entry in the MAC address table.
Step 3	Router(config)# **end**	Exits configuration mode.

Note Only port where the link is up will see the dynamic entry validated in the Ethernet switch network module.

Verifying the mac-address-table

Step 1 Use the **show mac** command to verify the mac-address-table, as illustrated below:

```
Router# show mac

Destination Address  Address Type  VLAN  Destination Port
-------------------  ------------  ----  --------------------
0001.6443.6440          Static       1      Vlan1
0004.c16d.9be1          Dynamic      1      FastEthernet2/13
0004.ddf0.0282          Dynamic      1      FastEthernet2/13
0006.0006.0006          Dynamic      1      FastEthernet2/13
001b.001b.ad45          Dynamic      1      FastEthernet2/13
```

Configuring aging-timer

To configure the aging-timer, use the following commands beginning in privileged EXEC mode:

	Command	Purpose
Step 1	`Router# configure terminal`	Enters global configuration mode.
Step 2	`Router(config)# mac-address-table aging-time <10-1000000>`	Configures the MAC address aging-timer age in seconds
Step 3	`Router(config)# end`	Exits configuration mode.

Caution Cisco advises that you not change the aging-timer, because the Ethernet switch network module could go out of synchronization.

Verifying the aging-timer

Step 1 Use the **show mac-address-table aging-time** command to verify the mac-address-table, as illustrated below:

```
Router # show mac-address-table aging-time

Mac address aging time 23
```

Configuring Cisco Discovery Protocol (CDP)

Configuring Cisco Discovery Protocol (CDP)

To enable CDP globally, use the following command in global configuration mode:

	Command	Purpose
Step 1	`Router(config)# [no] cdp run`	Enables CDP globally. Use the no keyword to disable CDP.

Verifying the CDP Global Configuration

Step 1 Use the **show cdp** command to verify the CDP configuration, as illustrated below:

```
Router# show cdp

Global CDP information:
        Sending CDP packets every 120 seconds
        Sending a holdtime value of 180 seconds
        Sending CDPv2 advertisements is enabled
Router#
```

Enabling CDP on an Interface

To enable CDP on an interface, use the following command in interface configuration mode:

	Command	Purpose
Step 1	`Router(config-if)# cdp enable`	Enables CDP on an interface.

The following example shows how to enable CDP on Fast Ethernet interface 5/1:

```
Router(config)# interface fastethernet 5/1
Router(config-if)# cdp enable
```

Verifying the CDP Interface Configuration

Step 1 Use the **show cdp interface** command to verify the CDP configuration for an interface, as illustrated below:

```
Router# show cdp interface fastethernet 5/1

FastEthernet5/1 is up, line protocol is up
  Encapsulation ARPA
  Sending CDP packets every 120 seconds
  Holdtime is 180 seconds
Router#
```

Verifying CDP Neighbors

Step 1 Use the **show cdp neighbors** command to verify information about the neighboring equipment, as illustrated below:

```
Router# show cdp neighbors

Capability Codes: R - Router, T - Trans Bridge, B - Source Route Bridge
                  S - Switch, H - Host, I - IGMP, r - Repeater
Device ID        Local Intrfce     Holdtme    Capability  Platform  Port ID
JAB023807H1      Fas 5/3            127          T S       WS-C2948  2/46
JAB023807H1      Fas 5/2            127          T S       WS-C2948  2/45
JAB023807H1      Fas 5/1            127          T S       WS-C2948  2/44
JAB023807H1      Gig 1/2            122          T S       WS-C2948  2/50
JAB023807H1      Gig 1/1            122          T S       WS-C2948  2/49
JAB03130104      Fas 5/8            167          T S       WS-C4003  2/47
JAB03130104      Fas 5/9            152          T S       WS-C4003  2/48
```

Monitoring and Maintaining CDP

To monitor and maintain CDP on your device, use one or more of the following commands beginning in privileged EXEC mode:

Command	Purpose
`Router# clear cdp counters`	Resets the traffic counters to zero.
`Router# clear cdp table`	Delete the CDP table of information about neighbors.
`Router# show cdp`	Verifies global information such as frequency of transmissions and the holdtime for packets being transmitted.
`Router# show cdp entry entry_name [protocol | version]`	Verifies information about a specific neighbor. The display can be limited to protocol version information.
`Router# show cdp interface [slot/port]`	Verifies information about interfaces on which CDP is enabled.
`Router# show cdp neighbors [slot/port] [detail]`	Verifies information about neighbors. The display can be limited to neighbors on a specific interface and can be expanded to provide more detailed information.
`Router# show cdp traffic`	Verifies CDP counters, including the number of packets sent and received and checksum errors.

Switched Port Analyzer (SPAN)

- Configuring Switched Port Analyzer (SPAN), page 53
- Configuring SPAN Destinations, page 53
- Removing Sources or Destinations from a SPAN Session, page 53

Configuring Switched Port Analyzer (SPAN)

To configure the source for a SPAN session, use the following command in global configuration mode:

	Command	Purpose
Step 1	Router(config)# **monitor session** {*session_number*} {**source** {**interface** slot/port} | {**vlan** *vlan_ID*}} [, | - | **rx** | **tx** | **both**]	Specifies the SPAN session number (1 or 2), the source interfaces or VLANs, and the traffic direction to be monitored.

Note Multiple SPAN sessions can be configured. But only one SPAN session is supported at a time.

The following example shows how to configure SPAN session 1 to monitor bidirectional traffic from source interface Fast Ethernet 5/1:

```
Router(config)# monitor session 1 source interface fastethernet 5/1
```

Configuring SPAN Destinations

To configure the destination for a SPAN session, use the following command in global configuration mode:

	Command	Purpose
Step 1	Router(config)# **monitor session** {*session_number*} {**destination** {**interface** *type*/*num*} [, | -] | {**vlan** *vlan*_ID}}	Specifies the SPAN session number (1 or 2) and the destination interfaces or VLANs.

Removing Sources or Destinations from a SPAN Session

To remove sources or destinations from a SPAN session, use the following command in global configuration mode:

	Command	Purpose
Step 1	Router(config)# **no monitor session** *session_number*	Clears existing SPAN configuration for a session.

Configuring Power Management on the Interface

To manage the powering of the Cisco IP phones, use the following commands beginning in privileged EXEC mode:

	Command	Purpose
Step 1	Router# **`configure terminal`**	Enters global configuration mode.
Step 2	Router(config)# **`int fastethernet`** *`port`*/*`slot`*	Selects a particular Fast Ethernet interface for configuration.
Step 3	Router(config)# **`power inline auto`**/**`never`**	Configures the port to supply inline power automatically to a Cisco IP phone. Use never to permanently disable inline power on the port.

Verifying Power Management on the Interface

Step 1 Use the **show power inline** command to verify the power configuration on the ports, as illustrated below:

```
Router# show power inline

PowerSupply   SlotNum.   Maximum   Allocated       Status
-----------   --------   -------   ---------       ------
 EXT-PS          1       165.000    20.000         PS1 GOOD PS2 ABSENT

Interface           Config   Phone    Powered    PowerAllocated
---------           ------   -----    -------    --------------
FastEthernet1/0      auto    no          off      0.000 Watts
FastEthernet1/1      auto    no          off      0.000 Watts
FastEthernet1/2      auto    no          off      0.000 Watts
FastEthernet1/3      auto    no          off      0.000 Watts
FastEthernet1/4      auto    unknown     off      0.000 Watts
FastEthernet1/5      auto    unknown     off      0.000 Watts
FastEthernet1/6      auto    unknown     off      0.000 Watts
FastEthernet1/7      auto    unknown     off      0.000 Watts
FastEthernet1/8      auto    unknown     off      0.000 Watts
FastEthernet1/9      auto    unknown     off      0.000 Watts
FastEthernet1/10     auto    unknown     off      0.000 Watts
FastEthernet1/11     auto    yes         on       6.400 Watts
FastEthernet1/12     auto    yes         on       6.400 Watts
FastEthernet1/13     auto    no          off      0.000 Watts
FastEthernet1/14     auto    unknown     off      0.000 Watts
FastEthernet1/15     auto    unknown     off      0.000 Watts
```

Verifying Other Power Management CLI

Step 1 Use the **show power inline** command to verify the power configuration on the ports, as illustrated below:

```
Router# show power inline [actual | interface fastethernet port/slot | configured]
```

Configuring IP Multicast Layer 3 Switching

These sections describe how to configure IP multicast Layer 3 switching:

- Enabling IP Multicast Routing Globally, page 55
- Enabling IP PIM on Layer 3 Interfaces, page 55
- Verifying IP Multicast Layer 3 Hardware Switching Summary, page 56
- Verifying the IP Multicast Routing Table, page 57
- Configuring IGMP Snooping, page 58

Enabling IP Multicast Routing Globally

You must enable IP multicast routing globally before you can enable IP multicast Layer 3 switching on Layer 3 interfaces.

For complete information and procedures, refer to these publications:

- *Cisco IOS IP Configuration Guide*, Release 12.2.
- Cisco IOS IP Command Reference, Volume 1 of 3: Addressing and Services, Release 12.2.
- Cisco IOS IP Command Reference, Volume 2 of 3: Routing Protocols, Release 12.2.
- Cisco IOS IP Command Reference, Volume 3 of 3: Multicast, Release 12.2.

To enable IP multicast routing globally, Use this command in global configuration mode:

Command	Purpose
`Router(config)#` **`ip multicast-routing`**	Enables IP multicast routing globally.

Enabling IP PIM on Layer 3 Interfaces

You must enable PIM on the Layer 3 interfaces before IP multicast Layer 3 switching functions on those interfaces.

To enable IP PIM on a Layer 3 interface, use the following commands beginning in global configuration mode:

	Command	Purpose
Step 1	`Router(config)#` **`interface vlan`** *`vlan_id`* `{slot/port}`	Selects the interface to be configured.
Step 2	`Router(config-if)#` **`ip pim`** `{`**`dense-mode`** `|` **`sparse-mode`** `|` **`sparse-dense-mode`**`}`	Enables IP PIM on a Layer 3 interface.

This example shows how to enable PIM on an interface using the default mode (**sparse-dense-mode**):

```
Router(config-if)# ip pim
Router(config-if)#
```

This example shows how to enable PIM sparse mode on an interface:

```
Router(config-if)# ip pim sparse-mode
Router(config-if)#
```

Verifying IP Multicast Layer 3 Hardware Switching Summary

Note The **show interface statistics** command does not verify hardware-switched packets, only packets switched by software.

The **show ip pim interface count** command verifies the IP multicast Layer 3 switching enable state on IP PIM interfaces and the number of packets received and sent on the interface.

Use the following show commands to verify IP multicast Layer 3 switching information for an IP PIM Layer 3 interface, as illustrated below:

Step 1

```
Router# show ip pim interface count

State:* - Fast Switched, D - Distributed Fast Switched
      H - Hardware Switching Enabled
Address          Interface          FS  Mpackets In/Out
10.15.1.20       GigabitEthernet4/8 * H 952/4237130770
10.20.1.7        GigabitEthernet4/9 * H 1385673757/34
10.25.1.7        GigabitEthernet4/10* H 0/34
10.11.1.30       FastEthernet6/26   * H 0/0
10.37.1.1        FastEthernet6/37   * H 0/0
1.22.33.44       FastEthernet6/47   * H 514/68
```

Step 2

```
Router# show ip mroute count

IP Multicast Statistics
56 routes using 28552 bytes of memory
13 groups, 3.30 average sources per group
Forwarding Counts:Pkt Count/Pkts per second/Avg Pkt Size/Kilobits per second
Other counts:Total/RPF failed/Other drops(OIF-null, rate-limit etc)

Group:224.2.136.89, Source count:1, Group pkt count:29051
  Source:132.206.72.28/32, Forwarding:29051/-278/1186/0, Other:85724/8/56665
Router#
```

Note The -tive counter means that the outgoing interface list of the corresponding entry is NULL, and this indicates that this flow is still active.

Step 3

```
Router# show ip interface vlan 10

Vlan10 is up, line protocol is up
  Internet address is 10.0.0.6/8
  Broadcast address is 255.255.255.255
  Address determined by non-volatile memory
  MTU is 1500 bytes
  Helper address is not set
  Directed broadcast forwarding is disabled
  Multicast reserved groups joined: 224.0.0.1 224.0.0.2 224.0.0.13 224.0.0.10
  Outgoing access list is not set
  Inbound  access list is not set
  Proxy ARP is enabled
  Security level is default
  Split horizon is enabled
  ICMP redirects are always sent
  ICMP unreachables are never sent
  ICMP mask replies are never sent
  IP fast switching is enabled
  IP fast switching on the same interface is disabled
  IP Flow switching is disabled
  IP CEF switching is enabled
  IP Fast switching turbo vector
  IP Normal CEF switching turbo vector
  IP multicast fast switching is enabled
  IP multicast distributed fast switching is disabled
  IP route-cache flags are Fast, CEF
  Router Discovery is disabled
  IP output packet accounting is disabled
  IP access violation accounting is disabled
  TCP/IP header compression is disabled
  RTP/IP header compression is disabled
  Probe proxy name replies are disabled
  Policy routing is disabled
  Network address translation is disabled
  WCCP Redirect outbound is disabled
  WCCP Redirect exclude is disabled
  BGP Policy Mapping is disabled
IP multicast multilayer switching is enabled
IP mls switching is enabled
Router#
```

Verifying the IP Multicast Routing Table

Step 1 Use the **show ip mroute** command to verify the IP multicast routing table, as illustrated below:

```
Router# show ip mroute 230.13.13.1

IP Multicast Routing Table
Flags:D - Dense, S - Sparse, s - SSM Group, C - Connected, L - Local,
       P - Pruned, R - RP-bit set, F - Register flag, T - SPT-bit set,
       J - Join SPT, M - MSDP created entry, X - Proxy Join Timer Running
       A - Advertised via MSDP, U - URD, I - Received Source Specific Host
           Report
Outgoing interface flags:H - Hardware switched
Timers:Uptime/Expires
Interface state:Interface, Next-Hop or VCD, State/Mode

(*, 230.13.13.1), 00:16:41/00:00:00, RP 10.15.1.20, flags:SJC
  Incoming interface:GigabitEthernet4/8, RPF nbr 10.15.1.20
  Outgoing interface list:
 GigabitEthernet4/9, Forward/Sparse-Dense, 00:16:41/00:00:00, H

(*, 230.13.13.2), 00:16:41/00:00:00, RP 10.15.1.20, flags:SJC
 Incoming interface:GigabitEthernet4/8, RPF nbr 10.15.1.20, RPF-MFD
  Outgoing interface list:
    GigabitEthernet4/9, Forward/Sparse-Dense, 00:16:41/00:00:00, H

(10.20.1.15, 230.13.13.1), 00:14:31/00:01:40, flags:CJT
 Incoming interface:GigabitEthernet4/8, RPF nbr 10.15.1.20, RPF-MFD
  Outgoing interface list:
    GigabitEthernet4/9, Forward/Sparse-Dense, 00:14:31/00:00:00, H
(132.206.72.28, 224.2.136.89), 00:14:31/00:01:40, flags:CJT
  Incoming interface:GigabitEthernet4/8, RPF nbr 10.15.1.20, RPF-MFD
 Outgoing interface list:Null
Router#
```

Note The RPF-MFD flag indicates that the flow is completely hardware switched. The H flag indicates that the flow is hardware-switched on the outgoing interface.

Configuring IGMP Snooping

Default IGMP Snooping Configuration

IGMP Snooping is enabled by default on a VLAN or subnet basis.   Multicast routing has to be enabled on the router first and then PIM (Multicast routing protocol) has to be enabled on the VLAN interface so that the switch acknowledges the IGMP join and leave messages which are sent from the hosts connected to the switch.

```
Router(config)# ip multicast-routing
Router(config-if)# interface VLAN1
    ip-address 192.168.10.1 255.255.255.0
    ip pim sparse-mode
```

To verify multicasting support:

```
Router# show ip igmp group
```

To verify IGMP snooping:

```
Router# show mac-address-table multicast igmp-snooping
```

To verify the multicast routing table:

```
Router# show ip mroute
```

Configuring Storm-Control

This section describes how to configure storm-control and characteristics on your router and consists of the following configuration information and procedures:

- Default Storm-Control Configuration, page 58
- Enabling Storm-Control, page 58
- Verifying Storm-Control, page 59

Default Storm-Control Configuration

By default, unicast, broadcast, and multicast suppression is disabled on the switch.

Enabling Storm-Control

Enable **storm-control** globally and enter the percentage of total available bandwidth that you want to be used by a all traffic (multicast, unicast,); entering 100 percent would allow all traffic.

To enable a particular type of storm-control, use the following commands beginning in privileged EXEC mode:

	Command	Purpose
Step 1	Router# `configure terminal`	Enters global configuration mode.
Step 2	Router(config)# `[`**`no`**`]` **`storm-control broadcast threshold`** *`<0-100>`*	Specifies the broadcast suppression level for an interface as a percentage of total bandwidth. A threshold value of 100 percent means that no limit is placed on broadcast traffic. Use the **no** keyword to restore the defaults.
Step 3	Router(config)# `[`**`no`**`]` **`storm-control multicast threshold`** *`<0-100>`*	Specifies the multicast suppression level for an interface as a percentage of total bandwidth. Use the **no** keyword to restore the defaults.
Step 4	Router(config)# `[`**`no`**`]` **`storm-control unicast threshold`** *`<0-100>`*	Specifies the unicast suppression level for an interface as a percentage of total bandwidth. Use the **no** keyword to restore the defaults.
Step 5	Router(config)# `end`	Returns to privileged EXEC mode.

Verifying Storm-Control

Step 1 Use the **show storm-control** command to view switchport characteristics, including storm-control levels set on the interface, as illustrated below:

```
Router# show storm-control
```

Step 2 Use the show interface counters privileged EXEC commands display the count of discarded packets.

To verify storm-control statistics on an interface, use the following commands beginning in privileged EXEC mode:

Command	Purpose
`show interface [`*`interface-id`*`] counters broadcast`	Verifies the broadcast suppression discard counter for all interfaces or a specific interface. Verify the number of packets discarded.
`show interface [`*`interface-id`*`] counters multicast`	Verifies the multicast suppression discard counter for all interfaces or a specific interface. Verify the number of packets discarded.
`show interface [`*`interface-id`*`] counters unicast`	Verifies the unicast suppression discard counter for all interfaces or a specific interface. Verify the number of packets discarded.

The following is sample output from the show interface counters broadcast privileged EXEC command:

```
Router# show interface counters broadcast

Port      BcastSuppDiscards
Fa0/1                     0
Fa0/2                     0
```

Configuring Separate Voice and Data VLANs

For ease of network administration and increased scalability, network managers can configure the Ethernet switch network module to support Cisco IP phones such that the voice and data traffic reside on separate VLANs. We recommend configuring separate VLANs when you are able to segment the existing IP address space of your branch office.

User priority bits in the 802.1p portion of the 802.1Q standard header are used to provide prioritization in Ethernet switches. This is a vital component in designing Cisco AVVID networks.

The Ethernet switch network module provides the performance and intelligent services of Cisco IOS Software for branch office applications. The Ethernet switch network module can identify user applications—such as voice or multicast video—and classify traffic with the appropriate priority levels. QoS policies are enforced using Layer 2 and 3 information such as 802.1p, IP precedence, and DSCP.

Note Refer to the *Cisco AVVID QoS Design Guide* for more information on how to implement end-to-end QoS as you deploy Cisco AVVID solutions.

To automatically configure Cisco IP phones to send voice traffic on the voice VLAN ID (VVID) on a per-port basis (see the "Voice Traffic and VVID" section on page 60), use the following commands beginning in global configuration mode:

	Command	Purpose
Step 1	`Router(config)# enable`	Enters the privileged EXEC mode. A preset password may be required to enter this mode.
Step 2	`Router(config)# configure terminal`	Enters global configuration mode.
Step 3	`Router(config)#` **`interface`** *`<interface>`*	Enters the interface configuration mode and the port to be configured (for example, interface fa5/1).
Step 4	`Router(config)# switchport mode trunk`	Configures the interface type as trunk mode. **Note** Voice VLANs require the port to be configured as a trunk port.
Step 5	`Router(config)# switchport voice vlan` *`<vlan-id>`*	Configures the voice port with a VVID that will be used exclusively for voice traffic.

Voice Traffic and VVID

The Ethernet switch network module can automatically configure voice VLAN. This capability overcomes the management complexity of overlaying a voice topology onto a data network while maintaining the quality of voice traffic. With the automatically configured voice VLAN feature, network administrators can segment phones into separate logical networks, even though the data and voice infrastructure is physically the same. The voice VLAN feature places the phones into their own VLANs without the need for end-user intervention. A user can plug the phone into the switch, and the switch provides the phone with the necessary VLAN information

Configuring a Single Subnet for Voice and Data

For network designs with incremental IP telephony deployment, network managers can configure the Ethernet switch network module so that the voice and data traffic coexist on the same subnet. This might be necessary when it is impractical either to allocate an additional IP subnet for IP phones or to divide the existing IP address space into an additional subnet at the remote branch, it might be necessary to use a single IP address space for branch offices. (This is one of the simpler ways to deploy IP telephony.) When this is the case, you must still prioritize voice above data at both Layer 2 and Layer 3.

Layer 3 classification is already handled because the phone sets the Type of Service (ToS) bits in all media streams to an IP Precedence value of 5. (With Cisco CallManager Release 3.0(5), this marking changed to a Differentiated Services Code Point ([DSCP]) value of EF.) However, to ensure that there is Layer 2 classification for admission to the multiple queues in the branch office switches, the phone must also use the User Priority bits in the Layer 2 802.1p header to provide Class of Service (CoS) marking. Setting the bits to provide marking can be done by having the switch look for 802.1p headers on the native VLAN.

This configuration approach must address two key considerations:

- Network managers should ensure that existing subnets have enough available IP addresses for the new Cisco IP phones, each of which requires a unique IP address.
- Administering a network with a mix of IP phones and workstations on the same subnet might pose a challenge.

To automatically configure Cisco IP phones to send voice and data traffic on the same VLAN, use the following commands beginning in privileged EXEC mode:

	Command	Purpose
Step 1	`Router# configure terminal`	Enters global configuration mode.
Step 2	`Router(config)#` **`interface`** *`<interface>`*	Enters the interface configuration mode and the port to be configured (e.g., interface fa5/1).
Step 3	`Router(config)#` **`switchport access vlan`** *`<vlan-id>`*	Sets the native VLAN for untagged traffic. The value of *vlan-id* represents the ID of the VLAN that is sending and receiving untagged traffic on the port. Valid IDs are from 1 to 1001. Leading zeroes are not accepted.
Step 4	`Router(config)# switchport voice vlan dot1p`	Configures the Cisco IP phone to send voice traffic with higher priority (CoS=5 on 802.1Q tag) on the access VLAN. Data traffic (from an attached PC) is sent untagged for lower priority (port default=0).
Step 5	`Router# end`	Returns to the privileged EXEC mode.

Verifying Switchport Configuration

Step 1 Use the **show run interface** command to verify the switch port configuration and the **write memory** command to save the current configuration in flash memory, as illustrated below:

```
Router# show run interface <interface>
```

Step 2

```
Router# write memory
```

Configuring Ethernet Ports to Support Cisco IP Phones with Multiple Ports

You might want to use multiple ports to connect the Cisco IP phones if any of the following conditions apply to your Cisco IP telephony network:

- You are connecting Cisco IP phones that do not have a second Ethernet port for attaching a PC.
- You want to create a physical separation between the voice and data networks.
- You want to provide in-line power easily to the IP phones without having to upgrade the data infrastructure.
- You want to limit the number of switches that need Uninterruptible Power Supply (UPS) power.

IP Addressing

The recommended configuration for using multiple cables to connect IP phones to the Cisco AVVID network is to use a separate IP subnet and separate VLANs for IP telephony.

Managing the Ethernet switch network module

This section illustrates how to perform basic management tasks on the Ethernet switch network module with the Cisco IOS CLI. You might find this information useful when you configure the switch for the previous scenarios.

The following topics are included:

Adding Trap Managers

A trap manager is a management station that receives and processes traps. When you configure a trap manager, community strings for each member switch must be unique. If a member switch has an IP address assigned to it, the management station accesses the switch by using its assigned IP address.

By default, no trap manager is defined, and no traps are issued.

To add a trap manager and community string, use the following commands beginning in privileged EXEC mode:

	Command	Purpose
Step 1	`Router# config terminal`	Enters global configuration mode.
Step 2	`Router(config)# snmp-server host 172.2.128.263 traps1 snmp vlan-membership`	Enters the trap manager IP address, community string, and the traps to generate.
Step 3	`Router(config)# end`	Returns to privileged EXEC mode.

Verifying Trap Managers

Step 1 Use the **show running-config** command to verify that the information was entered correctly by displaying the running configuration, as illustrated below:

```
Router# show running-config
```

Configuring IP Information

This section describes how to assign IP information on the Ethernet switch network module. The following topics are included:

- Assigning IP Information to the Switch, page 63
- Specifying a Domain Name and Configuring the DNS, page 64

Assigning IP Information to the Switch

You can use a BOOTP server to automatically assign IP information to the switch; however, the BOOTP server must be set up in advance with a database of physical MAC addresses and corresponding IP addresses, subnet masks, and default gateway addresses. In addition, the switch must be able to access the BOOTP server through one of its ports. At startup, a switch without an IP address requests the information from the BOOTP server; the requested information is saved in the switch running the configuration file. To ensure that the IP information is saved when the switch is restarted, save the configuration by entering the **write memory** command in privileged EXEC mode.

You can change the information in these fields. The mask identifies the bits that denote the network number in the IP address. When you use the mask to subnet a network, the mask is then referred to as a subnet mask. The broadcast address is reserved for sending messages to all hosts. The CPU sends traffic to an unknown IP address through the default gateway.

To enter the IP information, use the following commands beginning in privileged EXEC mode:

	Command	Purpose
Step 1	Router# `configure terminal`	Enters global configuration mode.
Step 2	Router(config)# `interface vlan 1`	Enter interface configuration mode, and enter the VLAN to which the IP information is assigned. VLAN 1 is the management VLAN, but you can configure any VLAN from IDs 1 to 1001.

	Command	Purpose
Step 3	Router(config)# **`ip address`** *`ip_address subnet_mask`*	Enters the IP address and subnet mask.
Step 4	Router(config)# `exit`	Returns to global configuration mode.
Step 5	Router# **`ip default-gateway`** *`ip_address`*	Enters the IP address of the default router.
Step 6	Router# `end`	Returns to privileged EXEC mode.

Use the following procedure to remove the IP information from a switch.

Note Using the **no ip address** command in configuration mode disables the IP protocol stack and removes the IP information. Cluster members without IP addresses rely on the IP protocol stack being enabled.

To remove an IP address, use the following commands beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# `interface vlan 1`	Enters interface configuration mode, and enters the VLAN to which the IP information is assigned. VLAN 1 is the management VLAN, but you can configure any VLAN from IDs 1 to 1001.
Step 2	Router(config-subif)# `no ip address`	Removes the IP address and subnet mask.
Step 3	Router(config-subif)# `end`	Returns to privileged EXEC mode.

Caution **If you are removing the IP address through a telnet session, your connection to the switch will be lost.**

Specifying a Domain Name and Configuring the DNS

Each unique IP address can have a host name associated with it. The Cisco IOS software maintains a EC mode, and related Telnet support operations. This cache speeds the process of converting names to addresses.

IP defines a hierarchical naming scheme that allows a device to be identified by its location or domain. Domain names are pieced together with periods (.) as the delimiting characters. For example, Cisco Systems is a commercial organization that IP identifies by a *com* domain name, so its domain name is *cisco.com*. A specific device in this domain, the FTP system, for example, is identified as *ftp.cisco.com*.

To track domain names, IP has defined the concept of a domain name server (DNS), the purpose of which is to hold a cache (or database) of names mapped to IP addresses. To map domain names to IP addresses, you must first identify the host names and then specify a name server and enable the DNS, the Internet's global naming scheme that uniquely identifies network devices.

Specifying the Domain Name

You can specify a default domain name that the software uses to complete domain name requests. You can specify either a single domain name or a list of domain names. When you specify a domain name, any IP host name without a domain name has that domain name appended to it before being added to the host table.

Specifying a Name Server

You can specify up to six hosts that can function as a name server to supply name information for the DNS.

Enabling the DNS

If your network devices require connectivity with devices in networks for which you do not control name assignment, you can assign device names that uniquely identify your devices within the entire internetwork. The Internet's global naming scheme, the DNS, accomplishes this task. This service is enabled by default.

Configuring Voice Ports

This section describes how to configure voice ports on the Ethernet switch network module. The following topics are included:

- Configuring a Port to Connect to a Cisco 7960 IP phone, page 65
- Disabling Inline Power on a Ethernet switch network module, page 66

The Ethernet switch network module can connect to a Cisco 7960 IP phone and carry IP voice traffic. If necessary, the Ethernet switch network module can supply electrical power to the circuit connecting it to the Cisco 7960 IP phone.

Because the sound quality of an IP telephone call can deteriorate if the data is unevenly transmitted, the current release of the IOS software supports QoS based on IEEE 802.1p CoS. QoS uses classification and scheduling to transmit network traffic from the switch in a predictable manner.

The Cisco 7960 IP phone contains an integrated three-port 10/100 switch. The ports are dedicated to connect to the following devices:

- Port 1 connects to the Ethernet switch network module switch or other voice-over-IP device
- Port 2 is an internal 10/100 interface that carries the phone traffic
- Port 3 connects to a PC or other device

Configuring a Port to Connect to a Cisco 7960 IP phone

Because a Cisco 7960 IP phone also supports connection to a PC or other device, a port connecting a Ethernet switch network module to a Cisco 7960 IP phone can carry a mix of traffic. There are three ways to configure a port connected to a Cisco 7960 IP phone:

- All traffic is transmitted according to the default COS priority (0) of the port. This is the default.
- Voice traffic is given a higher priority by the phone, and all traffic is in the same VLAN.
- Voice and data traffic are carried on separate VLANs, and voice traffic always has a CoS priority of 5.

To instruct the phone to give voice traffic a higher priority and to forward all traffic through the 802.1Q native VLAN, use the following commands beginning in privileged EXEC mode:

	Command	Purpose
Step 1	Router# `configure terminal`	Enters global configuration mode.
Step 2	Router(config)# **`interface`** *`interface_id`*	Enters interface configuration mode, and enter the port to be configured.
Step 3	Router(`config-if`)# **`switchport voice vlan dot1p`**	Instruct the switch to use 802.1p priority tagging for voice traffic and to use VLAN 0 (default native VLAN) to carry all traffic.
Step 4	Router(`config-if`)# `end`	Returns to privileged EXEC mode.

Verifying Voice Traffic Configuration

Step 1 Use the **show interface interface switchport** command to verify the voice traffic configuration on the 802.1Q native VLAN, as illustrated below:

```
Router# show interface interface switchport
```

Disabling Inline Power on a Ethernet switch network module

The Ethernet switch network module can supply inline power to the, if necessary. The Cisco 7960 IP phone can also be connected to an AC power source and supply its own power to the voice circuit. When the Cisco 7960 IP phone is supplying its own power, a Ethernet switch network module can forward IP voice traffic to and from the phone.

A detection mechanism on the Ethernet switch network module determines whether it is connected to a Cisco 7960 IP phone. If the switch senses that there is *no* power on the circuit, the switch supplies the power. If there is power on the circuit, the switch does not supply it.

You can configure the switch to never supply power to the Cisco 7960 IP phone and to disable the detection mechanism.

To configure a port to never supply power to Cisco 7960 IP phones, use the following commands beginning in privileged EXEC mode:

	Command	Purpose
Step 1	Router# `configure terminal`	Enters global configuration mode.
Step 2	Router(config)# **`interface`** *`interface_id`*	Enters interface configuration mode, and enter the port to be configured.
Step 3	Router(`config-if`)# `power inline never`	Permanently disables inline power on the port.
Step 4	Router(`config-if`)# `end`	Returns to privileged EXEC mode.

Verifying Inline Power Configuration

Step 1 Use the show **power inline** *interface* **configured** command to verifies the change by displaying the setting as configured, as illustrated below:

```
Router# show power inline interface configured
```

Enabling Switch Port Analyzer (SPAN)

You can monitor traffic on a given port by forwarding incoming and outgoing traffic on the port to another port in the same VLAN. A Switch Port Analyzer (SPAN) port cannot monitor ports in a different VLAN, and a SPAN port must be a static-access port. Any number of ports can be defined as SPAN ports, and any combination of ports can be monitored. SPAN is supported for up to 2 sessions.

To enable SPAN, use the following commands beginning in privileged EXEC mode:

	Command	Purpose
Step 1	Router# `configure terminal`	Enters global configuration mode.
Step 2	`Router(config)#` **`monitor session`** `<session_id>` `{`**destination** | **source**`}{`**`interface`** | **`vlan`** *`interface_id`* | *`vlan_id`*`}}` [, | - | both | tx | rx]	Enables port monitoring for a specific session (“*number”).* Optionally, supply a SPAN *destination* interface, and a *source* interface.
Step 3	Router(config)# `end`	Returns to privileged EXEC mode.

To disable SPAN, use the following commands beginning in privileged EXEC mode:

	Command	Purpose
Step 1	Router# `configure terminal`	Enters global configuration mode.
Step 2	Router(config)# **`no monitor session`** *`session_id`*	Disables port monitoring for a specific session.
Step 3	Router(config)# `end`	Returns to privileged EXEC mode.

Managing the ARP Table

To communicate with a device (on Ethernet, for example), the software first must determine the 48-bit MAC or local data link address of that device. The process of determining the local data link address from an IP address is called *address resolution*.

The Address Resolution Protocol (ARP) associates a host IP address with the corresponding media or MAC addresses and VLAN ID. Taking an IP address as input, ARP determines the associated MAC address. Once a MAC address is determined, the IP-MAC address association is stored in an ARP cache for rapid retrieval. Then the IP datagram is encapsulated in a link-layer frame and sent over the network. Encapsulation of IP datagrams and ARP requests and replies on IEEE 802 networks other than Ethernet is specified by the Subnetwork Access Protocol (SNAP). By default, standard Ethernet-style ARP encapsulation (represented by the **arpa** keyword) is enabled on the IP interface.

When you manually add entries to the ARP Table by using the CLI, you must be aware that these entries do not age and must be manually removed.

Managing the MAC Address Tables

This section describes how to manage the MAC address tables on the Ethernet switch network module. The following topics are included:

- MAC Addresses and VLANs, page 68
- Changing the Address Aging Time, page 69
- Configuring the Aging Time, page 69

The switch uses the MAC address tables to forward traffic between ports. All MAC addresses in the address tables are associated with one or more ports. These MAC tables include the following types of addresses:

- Dynamic address—a source MAC address that the switch learns and then drops when it is not in use.
- Secure address—a manually entered unicast address that is usually associated with a secured port. Secure addresses do not age.
- Static address—a manually entered unicast or multicast address that does not age and that is not lost when the switch resets.

The address tables list the destination MAC address and the associated VLAN ID, module, and port number associated with the address. The following shows an example of a list of addresses as they would appear in the dynamic, secure, or static address table.

```
Router# show mac

4d01h:%SYS-5-CONFIG_I:Configured from console by consolec
Slot # :0
--------------
Destination Address  Address Type  VLAN  Destination Port
-------------------  ------------  ----  --------------------
0004.272f.49de          Dynamic      1      FastEthernet0/8
0004.2762.3235          Dynamic      1      FastEthernet0/3
0004.4d07.6960          Dynamic      1      FastEthernet0/0
0004.ddbb.6700          Self         1      Vlan1
0020.18d7.4304          Dynamic      1      FastEthernet0/2
beef.beef.beef          Static       1      FastEthernet0/11
0004.2762.3235          Dynamic      2      FastEthernet0/3
0004.ddbb.6700          Self         2      Vlan2
0002.7e48.cc38          Dynamic      3      FastEthernet0/4
    0002.7e48.cc39          Dynamic      3      FastEthernet0/5
```

MAC Addresses and VLANs

All addresses are associated with a VLAN. An address can exist in more than one VLAN and have different destinations in each. Multicast addresses, for example, could be forwarded to port 1 in VLAN 1 and ports 9, 10, and 11 in VLAN 5.

Each VLAN maintains its own logical address table. A known address in one VLAN is unknown in another until it is learned or statically associated with a port in the other VLAN. An address can be secure in one VLAN and dynamic in another. Addresses that are statically entered in one VLAN must be static addresses in all other VLANs.

Changing the Address Aging Time

Dynamic addresses are source MAC addresses that the switch learns and then drops when they are not in use. Use the Aging Time field to define how long the switch retains unseen addresses in the table. This parameter applies to all VLANs.

Configuring the Aging Time

Setting too short an aging time can cause addresses to be prematurely removed from the table. Then when the switch receives a packet for an unknown destination, it floods the packet to all ports in the same VLAN as the receiving port. This unnecessary flooding can impact performance. Setting too long an aging time can cause the address table to be filled with unused addresses; it can cause delays in establishing connectivity when a workstation is moved to a new port.

To configure the dynamic address table aging time, use the following commands beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# `configure terminal`	Enters global configuration mode.
Step 2	Router(config)# **`mac-address-table aging-time`** *`seconds`*	Enters the number of seconds that dynamic addresses are to be retained in the address table. Valid entries are from 10 to 1000000.
Step 3	Router(config)# `end`	Returns to privileged EXEC mode.

Verifying Aging-Time Configuration

Step 1 Use the **show mac-address-table aging-time** command to verify configuration, as illustrated below:

```
Router# show mac-address-table aging-time
```

Verifying Dynamic Addresses

To remove a dynamic address entry follow these steps, beginning in privileged EXEC mode:

	Command	Purpose
Step 1	Router# `configure terminal`	Enters global configuration mode.
Step 2	Router(config)# **`no mac-address-table dynamic`** *`hw-addr`*	Enters the MAC address to be removed from dynamic MAC address table.
Step 3	Router(config)# `end`	Returns to privileged EXEC mode.

You can remove all dynamic entries by using the **clear mac-address-table dynamic** command in privileged EXEC mode.

Step 1 Use the **show mac-address-table dynamic** command to verify configuration, as illustrated below:

```
Router# show mac-address-table dynamic
```

Adding Secure Addresses

The secure address table contains secure MAC addresses and their associated ports and VLANs. A secure address is a manually entered unicast address that is forwarded to only one port per VLAN. If you enter an address that is already assigned to another port, the switch reassigns the secure address to the new port.

You can enter a secure port address even when the port does not yet belong to a VLAN. When the port is later assigned to a VLAN, packets destined for that address are forwarded to the port.

To add a secure address, use the following commands beginning in privileged EXEC mode:

	Command	Purpose
Step 1	`Router# configure terminal`	Enters global configuration mode.
Step 2	`Router(config)#` **`mac-address-table secure address`** *`hw-addr`* **`interface`** *`interface_id`* **`vlan`** *`vlan-id`*	Enters the MAC address, its associated port, and the VLAN ID.
Step 3	`Router(config)# end`	Returns to privileged EXEC mode.

To remove a secure address, use the following commands beginning in privileged EXEC mode:

	Command	Purpose
Step 1	`Router# configure terminal`	Enters global configuration mode.
Step 2	`Router(config)#` **`no mac-address-table secure`** *`hw-addr`* **`vlan`** *`vlan-id`*	Enters the secure MAC address, its associated port, and the VLAN ID to be removed.
Step 3	`Router(config)# end`	Returns to privileged EXEC mode.

You can remove all secure addresses by using the **clear mac-address-table secure** command in privileged EXEC mode.

Verifying Secure Addresses

Step 1 Use the **show mac-address-table secure** command to verify configuration, as illustrated below:

```
Router# show mac-address-table secure
```

Configuring Static Addresses

A static address has the following characteristics:

- It is manually entered in the address table and must be manually removed.
- It can be a unicast or multicast address.
- It does not age and is retained when the switch restarts.

Because all ports are associated with at least one VLAN, the switch acquires the VLAN ID for the address from the ports that you select on the forwarding map. A static address in one VLAN must be a static address in other VLANs. A packet with a static address that arrives on a VLAN where it has not been statically entered is flooded to all ports and not learned.

To add a static address, use the following commands beginning in privileged EXEC mode:

	Command	Purpose
Step 1	`Router#` **`configure terminal`**	Enters global configuration mode.
Step 2	`Router(config)#` **`mac-address-table static`** *`hw-addr`* `[`**`interface`**`]` *`interface-id`* `[`**`vlan`**`]` *`vlan-id`*	Enters the static MAC address, the interface, and the VLAN ID of those ports.
Step 3	`Router(config)# end`	Returns to privileged EXEC mode.

To remove a static address, use the following commands beginning in privileged EXEC mode

:

	Command	Purpose
Step 1	`Router# configure terminal`	Enters global configuration mode.
Step 2	`Router(config)#` **`no mac-address-table static`** *`hw-addr`* `[`**`interface`**`]` *`interface-id`* `[`**`vlan`**`]` *`vlan-id`*	Enters the static MAC address, the interface, and the VLAN ID of the port to be removed.
Step 3	`Router(config)# end`	Returns to privileged EXEC mode.

You can remove all secure addresses by using the **clear mac-address-table static** command in privileged EXEC mode.

Verifying Static Addresses

Step 1 Use the **show mac**-address-**table static** command to verify configuration, as illustrated below:

```
Router # show mac-address-table static

4d01h:%SYS-5-CONFIG_I:Configured from console by consolec
Slot # :0
--------------
Destination Address  Address Type  VLAN  Destination Port
-------------------  ------------  ----  --------------------
0004.272f.49de          Dynamic      1      FastEthernet0/8
0004.2762.3235          Dynamic      1      FastEthernet0/3
0004.4d07.6960          Dynamic      1      FastEthernet0/0
0004.ddbb.6700          Self         1      Vlan1
0020.18d7.4304          Dynamic      1      FastEthernet0/2
beef.beef.beef          Static       1      FastEthernet0/11
0004.2762.3235          Dynamic      2      FastEthernet0/3
0004.ddbb.6700          Self         2      Vlan2
0002.7e48.cc38          Dynamic      3      FastEthernet0/4
0002.7e48.cc39          Dynamic      3      FastEthernet0/5
```

Clearing all MAC Address Tables

To remove all addresses, Use the **clear mac-address** command in privileged EXEC mode:

	Command	Purpose
Step 1	`Router# clear mac-address-table`	Enters to clear all MAC address tables.
Step 2	`Router# end`	Returns to privileged EXEC mode.

Configuring Intrachassis Stacking

Perform this task to extend Layer 2 switching in the router by connecting the Gigabit Ethernet ports of the Cisco EtherSwitch network module.

For more details about the requirements for installing and connecting Cisco EtherSwitch network modules in a single chassis, go to the following URL:

http://www.cisco.com/univercd/cc/td/doc/product/access/acs_mod/cis2600/hw_inst/nm_inst/nm-doc/connswh.htm

To extend Layer 2 switching in the router by connecting the Gigabit Ethernet ports of the Cisco EtherSwitch network module, use the following commands beginning in global configuration mode:

	Command	Purpose
Step 1	`Router(config)#` **`interface Gigabit`** *`slot/port`*	Enters the current Gigabit Ethernet interface being used for intrachassis stacking.
Step 1	`Router(config-if)# [`**`no`**`]` **`switchport stacking-link interface Gigabit`** *`slot/port`*	Creates the intrachassis stacking between the current GE interface and the stacking link partner GE interface. To restore the defaults, use the **no** form of this command .
Step 2	`Router(config)#` **`end`**	Exits configuration mode.

Verifying Intra-chassis Stacking

Use the **show interface** command to verify configuration, as illustrated below:

```
Router# show interface gigabit slot/port
```

If intra-chassis stacking is used, use the **show interface g1/0** command and **show interface g2/0** to display the status of each interface. The display will indicate whether the interface is up or down. Notice the status in the line of the display labelled "Internal Stacking Link Active."

```
Router# show interface g1/0
Current configuration : 94 bytes
!
interface GigabitEthernet1/0
 switchport stacking-partner interface GigabitEthernet2/0 and interface GigabitEthernet2/0
switchport stacking-partner interface GigabitEthernet1/0


GigabitEthernet1/0 is up, line protocol is down
  Internal Stacking Link Active : Gi1/0 is stacked with Gi2/0
```

```
GigabitEthernet2/0 is up, line protocol is down
  Internal Stacking Link Active : Gi2/0 is stacked with Gi1/0
```

Configuring Flow Control on Gigabit Ethernet Ports

To configure flow control on a Gigabit Ethernet port, use the following commands in privileged mode:

	Command	Purpose
Step 1	`Router#` **`set port flowcontrol`** `{`**`receive`** `|` **`send`**`}` *`mod_num/port_num`* `{`**`off`** `|` **`on`** `|` **`desired`**`}`	Sets the flow control parameters on a Gigabit Ethernet port.
Step 2	`Router# show port flowcontrol`	Verifies the flow control configuration.

Configuration Examples

This section provides the following configuration examples:

- Range of Interface Examples, page 74
- Optional Interface Feature Examples, page 75
- VLAN Configuration Example, page 76
- VTP Examples, page 76
- EtherChannel Load Balancing Example, page 78
- Spanning Tree Examples, page 78
- Mac Table Manipulation Examples, page 81
- Cisco Discovery Protocol (CDP) Example, page 81
- Switched Port Analyzer (SPAN) Source Examples, page 81
- IGMP Snooping Example, page 82
- Storm-Control Example, page 83
- Ethernet Switching Examples, page 84
- Intrachassis Stacking Example, page 86
- Flow Control on Gigabit Ethernet Ports Example, page 87

Range of Interface Examples

- Single Range Configuration Example, page 74
- Multiple Range Configuration Example, page 75
- Range Macro Definition Example, page 75

Single Range Configuration Example

The following example shows all Fast Ethernet interfaces 5/1 to 5/5 being reenabled:

```
Router(config)# interface range fastethernet 5/1 - 5
Router(config-if)# no shutdown
Router(config-if)#
*Oct  6 08:24:35: %LINK-3-UPDOWN: Interface FastEthernet5/1, changed state to up
*Oct  6 08:24:35: %LINK-3-UPDOWN: Interface FastEthernet5/2, changed state to up
*Oct  6 08:24:35: %LINK-3-UPDOWN: Interface FastEthernet5/3, changed state to up
*Oct  6 08:24:35: %LINK-3-UPDOWN: Interface FastEthernet5/4, changed state to up
*Oct  6 08:24:35: %LINK-3-UPDOWN: Interface FastEthernet5/5, changed state to up
*Oct  6 08:24:36: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet5/5, changed
state to up
*Oct  6 08:24:36: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet5/3, changed
state to up
*Oct  6 08:24:36: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet5/4, changed
state to up
Router(config-if)#
```

Multiple Range Configuration Example

The following example shows how to use a comma to add different interface type strings to the range to reenable all Fast Ethernet interfaces in the range 5/1 to 5/5 and both Gigabit Ethernet interfaces 1/1 and 1/2:

```
Router(config-if)# interface range fastethernet 5/1 - 5, gigabitethernet 1/1 - 2
Router(config-if)# no shutdown
Router(config-if)#
*Oct  6 08:29:28: %LINK-3-UPDOWN: Interface FastEthernet5/1, changed state to up
*Oct  6 08:29:28: %LINK-3-UPDOWN: Interface FastEthernet5/2, changed state to up
*Oct  6 08:29:28: %LINK-3-UPDOWN: Interface FastEthernet5/3, changed state to up
*Oct  6 08:29:28: %LINK-3-UPDOWN: Interface FastEthernet5/4, changed state to up
*Oct  6 08:29:28: %LINK-3-UPDOWN: Interface FastEthernet5/5, changed state to up
*Oct  6 08:29:28: %LINK-3-UPDOWN: Interface GigabitEthernet1/1, changed state to up
*Oct  6 08:29:28: %LINK-3-UPDOWN: Interface GigabitEthernet1/2, changed state to up
*Oct  6 08:29:29: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet5/5, changed
state to up
*Oct  6 08:29:29: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet5/3, changed
state to up
*Oct  6 08:29:29: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet5/4, changed
state to up
Router(config-if)#
```

Range Macro Definition Example

The following example shows an interface-range macro named enet_list being defined to select Fast Ethernet interfaces 5/1 through 5/4:

```
Router(config)# define interface-range enet_list fastethernet 5/1 - 4

Router(config)#
```

The following example shows how to change to the interface-range configuration mode using the interface-range macro enet_list:

```
Router(config)# interface range macro enet_list

Router(config-if)#
```

Optional Interface Feature Examples

- Interface Speed Example, page 75
- Setting the Interface Duplex Mode Example, page 76
- Adding a Description for an Interface Example, page 76
- Configuring an Ethernet Interface as a Layer 2 Trunk Example, page 76

Interface Speed Example

The following example shows the interface speed being set to 100 Mbps on the Fast Ethernet interface 5/4:

```
Router(config)# interface fastethernet 5/4

Router(config-if)# speed 100
```

Setting the Interface Duplex Mode Example

The following example shows the interface duplex mode being set to full on Fast Ethernet interface 5/4:

```
Router(config)# interface fastethernet 5/4

Router(config-if)# duplex full
```

Adding a Description for an Interface Example

The following example shows how to add a description on Fast Ethernet interface 5/5:

```
Router(config)# interface fastethernet 5/5
Router(config-if)# description Channel-group to "Marketing"
```

Configuring an Ethernet Interface as a Layer 2 Trunk Example

The following example shows how to configure the Fast Ethernet interface 5/8 as an 802.1Q trunk. This example assumes that the neighbor interface is configured to support 802.1Q trunking:

```
Router# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)# interface fastethernet 5/8
Router(config-if)# shutdown
Router(config-if)# switchport trunk encapsulation dot1q
Router(config-if)# switchport mode trunk
Router(config-if)# no shutdown
Router(config-if)# end
Router# exit
```

VLAN Configuration Example

The following example shows how to configure the VLAN

```
Router# vlan database
Router(vlan)# vlan 3
VLAN 3 added:
Name: VLAN0003
Router(vlan)# exit
APPLY completed.
Exiting....
```

VTP Examples

- VTP Server Example, page 77
- VTP Client Example, page 77
- Disabling VTP (VTP Transparent Mode) Example, page 77
- VTP version 2 Example, page 77

VTP Server Example

The following example shows how to configure the switch as a VTP server:

```
Router# vlan database
Router(vlan)# vtp server
Setting device to VTP SERVER mode.
Router(vlan)# vtp domain Lab_Network
Setting VTP domain name to Lab_Network
Router(vlan)# vtp password WATER
Setting device VLAN database password to WATER.
Router(vlan)# exit
APPLY completed.
Exiting....
Router#
```

VTP Client Example

The following example shows how to configure the switch as a VTP client:

```
Router# vlan database
Router(vlan)# vtp client
Setting device to VTP CLIENT mode.
Router(vlan)# exit

In CLIENT state, no apply attempted.
Exiting....
Router#
```

Disabling VTP (VTP Transparent Mode) Example

The following example shows how to configure the switch as VTP transparent:

```
Router# vlan database
Router(vlan)# vtp transparent
Setting device to VTP TRANSPARENT mode.
Router(vlan)# exit
APPLY completed.
Exiting....
Router#
```

VTP version 2 Example

The following example shows VTP version 2 being enabled:

```
Router# vlan database

Router(vlan)# vtp v2-mode

V2 mode enabled.
Router(vlan)# exit

APPLY completed.
Exiting....
Router#
```

EtherChannel Load Balancing Example

Layer 2 EtherChannels Example

The following example shows Fast Ethernet interfaces 5/6 and 5/7 being configured into port-channel 2 with PAgP mode desirable:

```
Router# configure terminal
Router(config)# interface range fastethernet 5/6 -7
Router(config-if)# channel-group 2 mode desirable
Router(config-if)# end
```

EtherChannel Load Balancing Example

This example shows EtherChannel being configured to use source and destination IP addresses:

```
Router# configure terminal
Router(config)# port-channel load-balance src-dst-ip
Router(config)# end
Router(config)#
```

Removing an EtherChannel Example

This example shows port-channel 1 being removed:

```
Router# configure terminal
Router(config)# no interface port-channel 1
Router(config)# end
```

Note Removing the port-channel also removes the channel-group command from the interfaces belonging to it.

Spanning Tree Examples

Spanning-Tree Interface and Spanning-Tree Port Priority Example

The following example shows the VLAN port priority of an interface being configured:

```
Router# configure terminal
Router(config)# interface fastethernet 5/8
Router(config-if)# spanning-tree vlan 200 port-priority 64
Router(config-if)# end
Router#
```

The following example shows how to verify the configuration of VLAN 200 on the interface when it is configured as a trunk port:

```
Router# show spanning-tree vlan 200
!
!
!
Port 264 (FastEthernet5/8) of VLAN200 is forwarding
Port path cost 19, Port priority 64, Port Identifier 129.8.
   Designated root has priority 32768, address 0010.0d40.34c7
   Designated bridge has priority 32768, address 0010.0d40.34c7
   Designated port id is 128.1, designated path cost 0
   Timers: message age 2, forward delay 0, hold 0
   Number of transitions to forwarding state: 1
   BPDU: sent 0, received 13513
Router#
!
!
!
```

Spanning-Tree Port Cost Example

The following example shows how to change the spanning-tree port cost of a Fast Ethernet interface:

```
Router# configure terminal
Router(config)# interface fastethernet 5/8
Router(config-if)# spanning-tree cost 18
Router(config-if)# end
Router#
```

The following example shows how to verify the configuration of the interface when it is configured as an access port:

```
Router# show spanning-tree interface fastethernet 5/8
 Port 264 (FastEthernet5/8) of VLAN200 is forwarding
   Port path cost 18, Port priority 100, Port Identifier 129.8.
   Designated root has priority 32768, address 0010.0d40.34c7
   Designated bridge has priority 32768, address 0010.0d40.34c7
   Designated port id is 128.1, designated path cost 0
   Timers: message age 2, forward delay 0, hold 0
   Number of transitions to forwarding state: 1
   BPDU: sent 0, received 13513
Router#
```

The following example shows how to configure the spanning-tree VLAN port cost of a Fast Ethernet interface:

```
Router# configure terminal
Router(config)# interface fastethernet 5/8
Router(config-if)# spanning-tree vlan 200 cost 17
Router(config-if)# exit
Router(config)# exit
Router#
```

Bridge Priority of a VLAN

The following example shows the bridge priority of VLAN 200 being configured to 33792:

```
Router# configure terminal
Router(config)# spanning-tree vlan 200 priority 33792
Router(config)# end
Router#
```

Hello Time Example

The following example shows the hello time for VLAN 200 being configured to 7 seconds:

```
Router# configure terminal
Router(config)# spanning-tree vlan 200 hello-time 7
Router(config)# end
Router#
```

Forward-Delay Time for a VLAN Example

The following example shows the forward delay time for VLAN 200 being configured to 21 seconds:

```
Router# configure terminal
Router(config)# spanning-tree vlan 200 forward-time 21
Router(config)# end
Router#
```

Maximum Aging Time for a VLAN Example

The following example configures the maximum aging time for VLAN 200 to 36 seconds:

```
Router# configure terminal
Router(config)# spanning-tree vlan 200 max-age 36
Router(config)# end
Router#
```

Spanning Tree Examples

The following example shows spanning tree being enabled on VLAN 200:

```
Router# configure terminal
Router(config)# spanning-tree vlan 200
Router(config)# end
Router#
```

Note Because spanning tree is enabled by default, issuing a show running command to view the resulting configuration will not display the command you entered to enable spanning tree.

The following example shows spanning tree being disabled on VLAN 200:

```
Router# configure terminal
Router(config)# no spanning-tree vlan 200
Router(config)# end
Router#
```

Spanning Tree Root Example

The following example shows the switch being configured as the root bridge for VLAN 10, with a network diameter of 4:

```
Router# configure terminal
Router(config)# spanning-tree vlan 10 root primary diameter 4
Router(config)# exit
Router#
```

Mac Table Manipulation Examples

The following example shows a dynamic entry being configured in the MAC address table:

```
Router# configure terminal
Router (config)# mac-address-table dynamic 6.6.6 fastEthernet 2/13 vlan 1
Router (config)# end
```

The following example shows a static entry being configured in the MAC address table:

```
Router(config)# mac-address-table static beef.beef.beef int fa0/11 vlan 1
Router(config)# end
```

Cisco Discovery Protocol (CDP) Example

The following example shows CDP counter configuration being configured on the NM-16ESW:

```
Router# clear cdp counters
```

Switched Port Analyzer (SPAN) Source Examples

- SPAN Source Configuration Example, page 81
- SPAN Destinations Example, page 81
- Removing Sources or Destinations from a SPAN Session Example, page 82

SPAN Source Configuration Example

The following example shows SPAN session 1 being configured to monitor bidirectional traffic from source interface Fast Ethernet 5/1:

```
Router(config)# monitor session 1 source interface fastethernet 5/1
```

SPAN Destinations Example

The following example shows interface Fast Ethernet 5/48 being configured as the destination for SPAN session 1:

```
Router(config)# monitor session 1 destination interface fastethernet 5/48
```

Removing Sources or Destinations from a SPAN Session Example

This following example shows interface Fast Ethernet 5/2 being removed as a SPAN source for SPAN session 1:

```
Router(config)# no monitor session 1 source interface fastethernet 5/2
```

IGMP Snooping Example

The following example shows the output from configuring IGMP snooping:

```
Router# show mac-address-table multicast igmp-snooping

Slot # :3
--------------
    MACADDR     VLANID     INTERFACES

0100.5e00.0001    1
0100.5e00.0002    1
0100.5e00.000d    1
0100.5e00.0016    1
0100.5e05.0505    1        Fa3/12
0100.5e06.0606    1        Fa3/13
0100.5e7f.ffff    1        Fa3/13
0100.5e00.0001    2
0100.5e00.0002    2
0100.5e00.000d    2
0100.5e00.0016    2
0100.5e00.0128    2
0100.5e05.0505    2        Fa3/10
0100.5e06.0606    2        Fa3/11
Router#
```

The following is an example of output from the **show run int** privileged EXEC command for VLAN 1:

```
Router#show run int vlan 1

Building configuration...

Current configuration :82 bytes
!
interface Vlan1
 ip address 192.168.4.90 255.255.255.0
 ip pim sparse-mode
end

Router# show run int vlan 2

Building configuration...

Current configuration :82 bytes
!
interface Vlan2
 ip address 192.168.5.90 255.255.255.0
 ip pim sparse-mode
end

Router#
Router# sh ip igmp group

IGMP Connected Group Membership
Group Address    Interface                Uptime    Expires   Last Reporter
```

```
239.255.255.255  Vlan1                    01:06:40  00:02:20  192.168.41.101
224.0.1.40       Vlan2                    01:07:50  00:02:17  192.168.5.90
224.5.5.5        Vlan1                    01:06:37  00:02:25  192.168.41.100
224.5.5.5        Vlan2                    01:07:40  00:02:21  192.168.31.100
224.6.6.6        Vlan1                    01:06:36  00:02:22  192.168.41.101
224.6.6.6        Vlan2                    01:06:39  00:02:20  192.168.31.101
Router#

Router# show ip mroute

IP Multicast Routing Table
Flags:D - Dense, S - Sparse, B - Bidir Group, s - SSM Group, C -
Connected,
       L - Local, P - Pruned, R - RP-bit set, F - Register flag,
       T - SPT-bit set, J - Join SPT, M - MSDP created entry,
       X - Proxy Join Timer Running, A - Candidate for MSDP Advertisement,
       U - URD, I - Received Source Specific Host Report
Outgoing interface flags:H - Hardware switched
Timers:Uptime/Expires
Interface state:Interface, Next-Hop or VCD, State/Mode

(*, 239.255.255.255), 01:06:43/00:02:17, RP 0.0.0.0, flags:DC
  Incoming interface:Null, RPF nbr 0.0.0.0
  Outgoing interface list:
    Vlan1, Forward/Sparse, 01:06:43/00:02:17

(*, 224.0.1.40), 01:12:42/00:00:00, RP 0.0.0.0, flags:DCL
  Incoming interface:Null, RPF nbr 0.0.0.0
  Outgoing interface list:
    Vlan2, Forward/Sparse, 01:07:53/00:02:14

(*, 224.5.5.5), 01:07:43/00:02:22, RP 0.0.0.0, flags:DC
  Incoming interface:Null, RPF nbr 0.0.0.0
  Outgoing interface list:
    Vlan1, Forward/Sparse, 01:06:40/00:02:22
    Vlan2, Forward/Sparse, 01:07:44/00:02:17

(*, 224.6.6.6), 01:06:43/00:02:18, RP 0.0.0.0, flags:DC
  Incoming interface:Null, RPF nbr 0.0.0.0
  Outgoing interface list:
    Vlan1, Forward/Sparse, 01:06:40/00:02:18
    Vlan2, Forward/Sparse, 01:06:43/00:02:16

Router#
```

Storm-Control Example

The following example shows bandwidth-based multicast suppression being enabled at 70 percent on Gigabit Ethernet interface 1 and the configuration being verified:

```
Router# configure terminal
Router(config)# interface gigabitethernet0/2
Router(config-if)# storm-control threshold 70
Router(config-if)# end
Router# show storm-control

Name: Gi0/2
Switchport: Enabled
Administrative Mode: dynamic desirable
Operational Mode: down
Administrative Trunking Encapsulation: dot1q
Negotiation of Trunking: On
```

```
Access Mode VLAN: 1 (default)
Trunking Native Mode VLAN: 1 (default)
Trunking VLANs Enabled: ALL
Pruning VLANs Enabled: 2-1001

Port Protected: Off
Unknown Unicast Traffic: Allowed
Unknown Multicast Traffic: Not Allowed

Broadcast Suppression Level: 100
Multicast Suppression Level: 70
Unicast Suppression Level: 100
```

Ethernet Switching Examples

- Subnets for Voice and Data Example, page 84
- Inter-VLAN Routing Example, page 85
- Single Subnet Configuration Example, page 85
- Ethernet Ports on IP Phones with Multiple Ports Example, page 85

Subnets for Voice and Data Example

The following example shows separate subnets being configured for voice and data on the Ethernet switch network module:

```
interface FastEthernet5/1
     description DOT1Q port to IP Phone
     switchport native vlan 50
     switchport mode trunk
     switchport voice vlan 150

interface Vlan 150
    description voice vlan
ip address 10.150.1.1 255.255.255.0
ip helper-address 172.20.73.14 (See Note below)

interface Vlan 50
    description data vlan
ip address 10.50.1.1 255.255.255.0
```

This configuration instructs the IP phone to generate a packet with an 802.1Q VLAN ID of 150 with an 802.1p value of 5 (default for voice bearer traffic).

Note In a centralized CallManager deployment model, the DHCP server might be located across the WAN link. If so, an **ip helper-address** command pointing to the DHCP server should be included on the voice VLAN interface for the IP phone. This is done to obtain its IP address as well as the address of the TFTP server required for its configuration.

Be aware that IOS supports a DHCP server function. If this function is used, the Ethernet switch network module serves as a local DHCP server and a helper address would not be required.

Inter-VLAN Routing Example

Configuring inter-vlan routing is identical to the configuration on a Ethernet switch network module with an MSFC. Configuring an interface for WAN routing is consistent with other IOS platforms.

The following example provides a sample configuration:

```
interface Vlan 160
     description voice vlan
     ip address 10.6.1.1 255.255.255.0

interface Vlan 60
    description data vlan
ip address 10.60.1.1 255.255.255.0

interface Serial1/0
ip address 160.3.1.2 255.255.255.0
```

Note Standard IGP routing protocols such as RIP, IGRP, EIGRP, and OSPF are supported on the Ethernet switch network module. Multicast routing is also supported for PIM dense mode, sparse mode and sparse-dense mode.

Single Subnet Configuration Example

The Ethernet switch network module supports the use of an 802.1p-only option when configuring the voice VLAN. Using this option allows the IP phone to tag VoIP packets with a CoS of 5 on the native VLAN, while all PC data traffic is sent untagged.

The following example shows a single subnet configuration for the Ethernet switch network module switch:

```
Router# FastEthernet 5/2
description Port to IP Phone in single subnet
     switchport access vlan 40
     switchport voice vlan dot1p
     spanning-tree portfast
```

The Ethernet switch network module instructs the IP phone to generate an 802.1Q frame with a null VLAN ID value but with an 802.1p value (default is COS of 5 for bearer traffic). The voice and data vlans are both 40 in this example.

Ethernet Ports on IP Phones with Multiple Ports Example

The following example illustrates the configuration on the IP phone:

```
interface FastEthernetx/x
     switchport voice vlan x
     switchport mode trunk
```

The following example illustrates the configuration on the PC:

```
interface FastEthernetx/y
     switchport access vlan y
```

Note Using a separate subnet, and possibly a separate IP address space, may not be an option for some small branch offices due to the IP routing configuration. If the IP routing can handle an additional subnet at the remote branch, you can use Cisco Network Registrar and secondary addressing.

Intrachassis Stacking Example

The following example shows how to stack GE port 2/0 to GE port 3/0 to form an extended VLAN within one chassis:

```
Router #config terminal
Router(config)# interface Gigabit 2/0
Router(config-if)# switchport stacking-link interface Gigabit3/0
```

The following example shows interchassis stacking being verified between GE port 2/0 and GE port 3/0:

```
Router# show interface gigabit 2/0

 GigabitEthernet2/0 is up, line protocol is down
   Internal Stacking Link Active : Gi2/0 is stacked with Gi3/0
   Hardware is Gigabit Ethernet, address is 001b.3f2b.2c24 (bia 001b.3f2b.2c24)
   MTU 1500 bytes, BW 1000000 Kbit, DLY 10 usec,
      reliability 255/255, txload 1/255, rxload 1/255
   Encapsulation ARPA, loopback not set
   Keepalive set (10 sec)
   Full-duplex mode, link type is force-up, media type is unknown 0
   output flow-control is off, input flow-control is off
   Full-duplex, 1000Mb/s
   ARP type: ARPA, ARP Timeout 04:00:00
   Last input 1d22h, output never, output hang never
   Last clearing of "show interface" counters 1d22h
   Queueing strategy: fifo
   Output queue 0/40, 0 drops; input queue 0/75, 0 drops
   5 minute input rate 0 bits/sec, 0 packets/sec
   5 minute output rate 0 bits/sec, 0 packets/sec
      250707 packets input, 19562597 bytes, 0 no buffer
      Received 7 broadcasts, 0 runts, 0 giants, 0 throttles
      0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored
      0 watchdog, 0 multicast, 0 pause input
      0 input packets with dribble condition detected
      7469804 packets output, 582910831 bytes, 0 underruns(0/0/0)
      0 output errors, 0 collisions, 0 interface resets
      0 babbles, 0 late collision, 0 deferred
      0 lost carrier, 0 no carrier, 0 pause output
      0 output buffer failures, 0 output buffers swapped out
```

Flow Control on Gigabit Ethernet Ports Example

The following examples show how to turn transmit and receive flow control on and how to verify the flow-control configuration:

Port 2/1 flow control send administration status set to on (port will send flowcontrol to far end):

```
Router> set port flowcontrol send 2/1 on
```

Port 2/1 flow control receive administration status set to on (port will require far end to send flowcontrol):

```
Router> set port flowcontrol receive 2/1 on
```

```
The following example shows flow control configuration being verified:
```

```
Router> show port flowcontrol 2/1

Port   Send FlowControl    Receive FlowControl   RxPause TxPause Unsupported
       admin    oper       admin    oper                      opcodes
-----  -------- --------   -------- --------     ------- ------- -----------
 2/1   on       on         on       on           0       0       0
Console> (enable)
```

Glossary

802.1p—IEEE standard for queuing and multicast support.

802.1q—IEEE standard for VLAN frame tagging.

ATM—Asynchronous Transfer Mode.

AVVID—Architecture for Voice, Video, and Integrated Data.

BRI—Basic Rate Interface.

CAC—connection admission control.

CBWFQ—class-based weighted fair queuing.

CCN—Cisco Communications Network (Cisco IP phones and IP PBX).

CoS—class of service.

DSL—digital subscriber line.

E&M—ear and mouth.

FXO—Foreign Exchange Office.

FXS—Foreign Exchange Station.

IP—Internet Protocol.

MIB—Management Information Base.

PRI—Primary Rate Interface.

PVC—permanent virtual circuit.

PSTN—public switched telephone network.

QoS—quality of service.

RSVP—Resource Reservation Protocol.

SIP—session initiation protocol.

SNMP—Simple Network Management Protocol.

VBR—variable bit rate.

VPN—virtual private network.

VoIP—Voice over IP.

VoIPoFR—Voice-over-IP over Frame-Relay.

WAN—wide area network.

WFQ—weighted fair queuing.

WRR—weighted round-robin.