



CHAPTER 17

Configuring Optional Spanning-Tree Features

This chapter describes how to configure optional spanning-tree features on your Catalyst 3550 switch. You can configure all of these features when your switch is running the per-VLAN spanning-tree plus (PVST+). You can configure only the noted features when your switch is running the Multiple Spanning Tree Protocol (MSTP) or the rapid per-VLAN spanning-tree plus (rapid-PVST+) protocol.

For information on configuring the PVST+ and rapid PVST+, see [Chapter 15, “Configuring STP.”](#) For information about the Multiple Spanning Tree Protocol (MSTP) and how to map multiple VLANs to the same spanning-tree instance, see [Chapter 16, “Configuring MSTP.”](#)



Note

For complete syntax and usage information for the commands used in this chapter, see the command reference for this release.

This chapter consists of these sections:

- [Understanding Optional Spanning-Tree Features, page 17-1](#)
- [Configuring Optional Spanning-Tree Features, page 17-13](#)
- [Displaying the Spanning-Tree Status, page 17-21](#)

Understanding Optional Spanning-Tree Features

These sections describe how the optional spanning-tree features work:

- [Understanding Port Fast, page 17-2](#)
- [Understanding BPDU Guard, page 17-2](#)
- [Understanding BPDU Filtering, page 17-3](#)
- [Understanding UplinkFast, page 17-3](#)
- [Understanding Cross-Stack UplinkFast, page 17-5](#)
- [Understanding BackboneFast, page 17-9](#)
- [Understanding EtherChannel Guard, page 17-11](#)
- [Understanding Root Guard, page 17-12](#)
- [Understanding Loop Guard, page 17-13](#)

Understanding Port Fast

Port Fast immediately brings an interface configured as an access or trunk port from a blocking state to the forwarding state, bypassing the listening and learning states. You can use Port Fast on ports connected to a single workstation or server, as shown in Figure 17-1, to allow those devices to immediately connect to the network, rather than waiting for the spanning tree to converge.

Ports connected to a single workstation or server should not receive bridge protocol data units (BPDUs). A port with Port Fast enabled goes through the normal cycle of spanning-tree status changes when the switch is restarted.

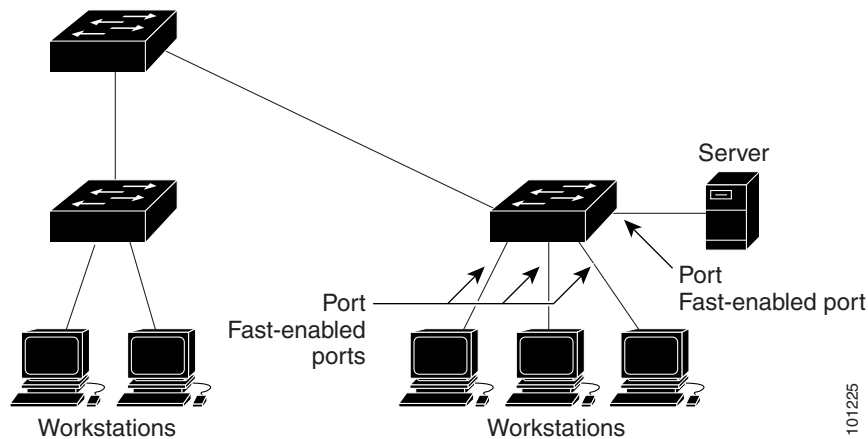


Note

Because the purpose of Port Fast is to minimize the time ports must wait for spanning-tree to converge, it is effective only when used on ports connected to end stations. If you enable Port Fast on a port connected to another switch, you risk creating a spanning-tree loop.

You can enable this feature by using the **spanning-tree portfast** interface configuration or the **spanning-tree portfast default** global configuration command.

Figure 17-1 Port Fast-Enabled Ports



Understanding BPDU Guard

The BPDU guard feature can be globally enabled on the switch or can be enabled per interface, but the feature operates with some differences.

At the global level, you can enable BPDU guard on Port Fast-enabled ports by using the **spanning-tree portfast bpduguard default** global configuration command. Spanning tree shuts down ports that are in a Port Fast-operational state if any BPDU is received on those interfaces. In a valid configuration, Port Fast-enabled ports do not receive BPDUs. Receiving a BPDU on a Port Fast-enabled port signals an invalid configuration, such as the connection of an unauthorized device, and the BPDU guard feature puts the port in the error-disabled state.

At the interface level, you can enable BPDU guard on any port by using the **spanning-tree bpduguard enable** interface configuration command without also enabling the Port Fast feature. When the port receives a BPDU, it is put in the error-disabled state.

The BPDU guard feature provides a secure response to invalid configurations because you must manually put the port back in service. Use the BPDU guard feature in a service-provider network to prevent an access port from participating in the spanning tree.

You can enable the BPDU guard feature for the entire switch or for an interface.

Understanding BPDU Filtering

The BPDU filtering feature can be globally enabled on the switch or can be enabled per interface, but the feature operates with some differences.

At the global level, you can enable BPDU filtering on Port Fast-enabled ports by using the **spanning-tree portfast bpdupfilter default** global configuration command. This command prevents ports that are in a Port Fast-operational state from sending or receiving BPDUs. The ports still send a few BPDUs at link-up before the switch begins to filter outbound BPDUs. You should globally enable BPDU filtering on a switch so that hosts connected to these ports do not receive BPDUs. If a BPDU is received on a Port Fast-enabled port, the port loses its Port Fast-operational status, and BPDU filtering is disabled.

At the interface level, you can enable BPDU filtering on any port without also enabling the Port Fast feature by using the **spanning-tree bpdupfilter enable** interface configuration command. This command prevents the port from sending or receiving BPDUs.



Caution

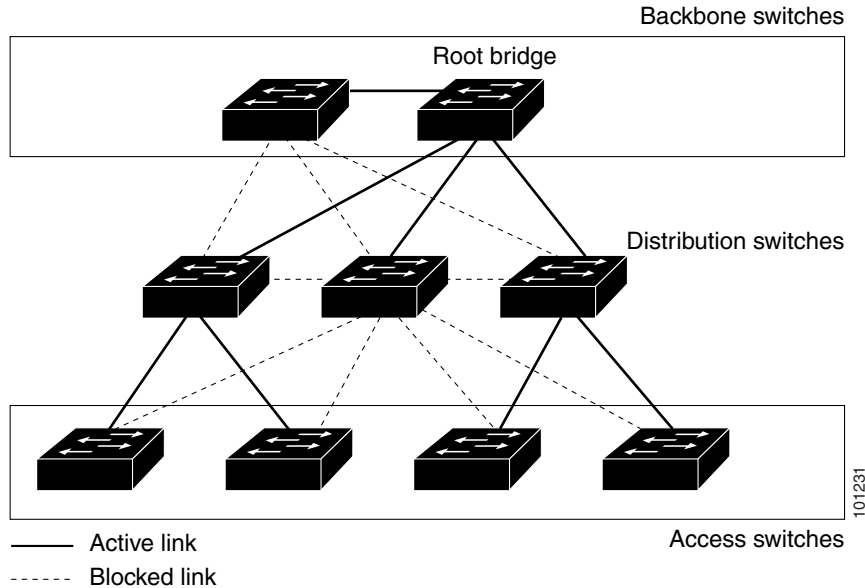
Enabling BPDU filtering on an interface is the same as disabling spanning tree on it and can result in spanning-tree loops.

You can enable the BPDU filtering feature for the entire switch or for an interface.

Understanding UplinkFast

Switches in hierarchical networks can be grouped into backbone switches, distribution switches, and access switches. [Figure 17-2](#) shows a complex network where distribution switches and access switches each have at least one redundant link that spanning tree blocks to prevent loops.

Figure 17-2 Switches in a Hierarchical Network



If a switch loses connectivity, it begins using the alternate paths as soon as the spanning tree selects a new root port. By enabling UplinkFast with the **spanning-tree uplinkfast** global configuration command, you can accelerate the choice of a new root port when a link or switch fails or when the spanning tree reconfigures itself. The root port transitions to the forwarding state immediately without going through the listening and learning states, as it would with the normal spanning-tree procedures.

When the spanning tree reconfigures the new root port, other interfaces flood the network with multicast packets, one for each address that was learned on the interface. You can limit these bursts of multicast traffic by reducing the max-update-rate parameter (the default for this parameter is 150 packets per second). However, if you enter zero, station-learning frames are not generated, so the spanning-tree topology converges more slowly after a loss of connectivity.

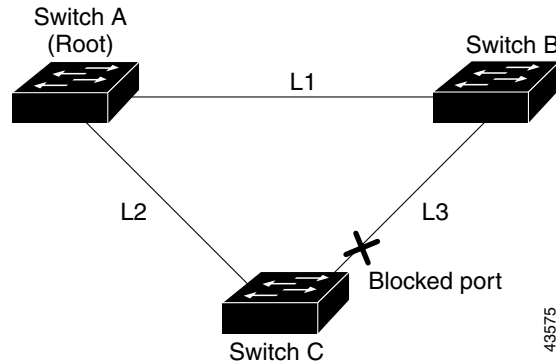
**Note**

UplinkFast is most useful in wiring-closet switches at the access or edge of the network. It is not appropriate for backbone devices. This feature might not be useful for other types of applications.

UplinkFast provides fast convergence after a direct link failure and achieves load balancing between redundant Layer 2 links using uplink groups. An uplink group is a set of Layer 2 interfaces (per VLAN), only one of which is forwarding at any given time. Specifically, an uplink group consists of the root port (which is forwarding) and a set of blocked ports, except for self-looping ports. The uplink group provides an alternate path in case the currently forwarding link fails.

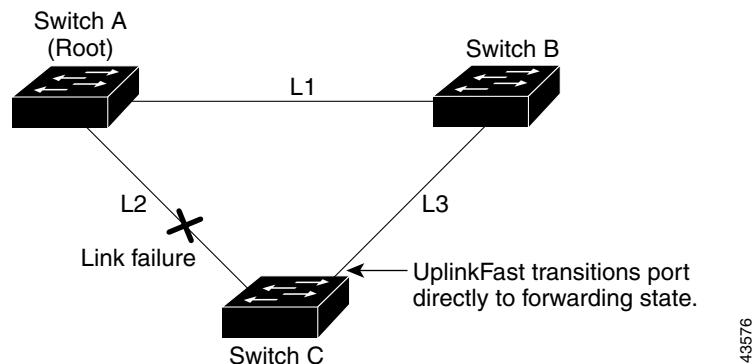
Figure 17-3 shows an example topology with no link failures. Switch A, the root switch, is connected directly to Switch B over link L1 and to Switch C over link L2. The Layer 2 interface on Switch C that is connected directly to Switch B is in a blocking state.

Figure 17-3 UplinkFast Example Before Direct Link Failure



If Switch C detects a link failure on the currently active link L2 on the root port (a *direct* link failure), UplinkFast unblocks the blocked port on Switch C and transitions it to the forwarding state without going through the listening and learning states, as shown in Figure 17-4. This change takes approximately 1 to 5 seconds.

Figure 17-4 UplinkFast Example After Direct Link Failure



Understanding Cross-Stack UplinkFast

Cross-stack UplinkFast (CSUF) provides a fast spanning-tree transition (fast convergence in less than 1 second under normal network conditions) across a stack of switches that use the GigaStack GBIC modules connected in a shared cascaded configuration (multidrop backbone). During the fast transition, an alternate redundant link on the stack of switches is placed in the forwarding state without causing temporary spanning-tree loops or loss of connectivity to the backbone. With this feature, you can have a redundant and resilient network in some configurations. You enable CSUF by using the **spanning-tree stack-port** interface configuration command.

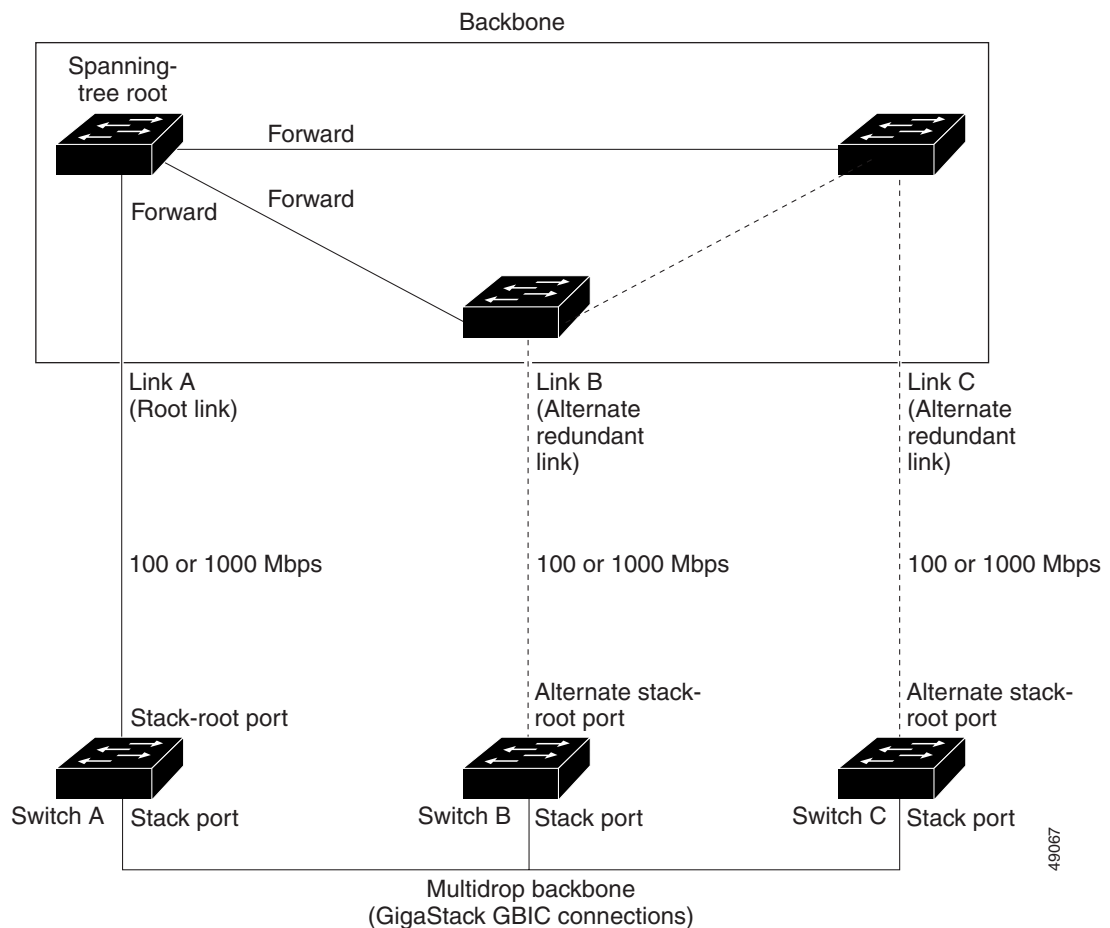
CSUF might not provide a fast transition all the time; in these cases, the normal spanning-tree transition occurs, completing in 30 to 40 seconds. For more information, see the “[Events that Cause Fast Convergence](#)” section on page 17-7.

How CSUF Works

CSUF ensures that one link in the stack is elected as the path to the root. As shown in [Figure 17-5](#), Switches A, B, and C are cascaded through the GigaStack GBIC module to form a multidrop backbone, which communicates control and data traffic across the switches at the access layer. The switches in the stack use their stack ports to communicate with each other and to connect to the stack backbone; stack ports are always in the spanning-tree forwarding state. The stack-root port on Switch A provides the path to the root of the spanning tree; the alternate stack-root ports on Switches B and C can provide an alternate path to the spanning-tree root if the current stack-root switch fails or if its link to the spanning-tree root fails.

Link A, the root link, is in the spanning-tree forwarding state; Links B and C are alternate redundant links that are in the spanning-tree blocking state. If Switch A fails, if its stack-root port fails, or if Link A fails, CSUF selects either the Switch B or Switch C alternate stack-root port and puts it into the forwarding state in less than 1 second.

Figure 17-5 Cross-Stack UplinkFast Topology



CSUF uses the Stack Membership Discovery Protocol to build a neighbor list of stack members through the receipt of discovery hello packets. When certain link loss or spanning-tree events occur (described in [“Events that Cause Fast Convergence”](#) section on page 17-7), the Fast Uplink Transition Protocol uses the neighbor list to send fast-transition requests on the stack port to stack members.

The switch sending the fast-transition request needs to do a fast transition to the forwarding state of a port that it has chosen as the root port, and it must obtain an acknowledgement from each stack switch before performing the fast transition.

Each switch in the stack determines if the sending switch is a better choice than itself to be the stack root of this spanning-tree instance by comparing the root, cost, and bridge ID. If the sending switch is the best choice as the stack root, each switch in the stack returns an acknowledgement; otherwise, it does not respond to the sending switch (drops the packet). The sending switch then has not received acknowledgements from all stack switches.

When acknowledgements are received from all stack switches, the Fast Uplink Transition Protocol on the sending switch immediately transitions its alternate stack-root port to the forwarding state. If acknowledgements from all stack switches are not obtained by the sending switch, the normal spanning-tree transitions (blocking, listening, learning, and forwarding) take place, and the spanning-tree topology converges at its normal rate ($2 * \text{forward-delay time} + \text{max-age time}$).

The Fast Uplink Transition Protocol is implemented on a per-VLAN basis and affects only one spanning-tree instance at a time.

Events that Cause Fast Convergence

Depending on the network event or failure, the CSUF fast convergence might or might not occur.

Fast convergence (less than 1 second under normal network conditions) occurs under these circumstances:

- The stack-root port link fails.
If two switches in the stack have alternate paths to the root, only one of the switches performs the fast transition.
- The failed link, which connects the stack root to the spanning-tree root, recovers.
- A network reconfiguration causes a new stack-root switch to be selected.
- A network reconfiguration causes a new port on the current stack-root switch to be chosen as the stack-root port.



Note

The fast transition might not occur if multiple events occur simultaneously. For example, if a stack member switch is powered off, and at the same time, the link connecting the stack root to the spanning-tree root comes back up, the normal spanning-tree convergence occurs.

Normal spanning-tree convergence (30 to 40 seconds) occurs under these conditions:

- The stack-root switch is powered off, or the software failed.
- The stack-root switch, which was powered off or failed, is powered on.
- A new switch, which might become the stack root, is added to the stack.
- A switch other than the stack root is powered off or failed.
- A link fails between stack ports on the multidrop backbone.

Limitations

These limitations apply to CSUF:

- CSUF uses the GigaStack GBIC module and runs on all Catalyst 3550 switches, all Catalyst 3500 XL switches, Catalyst 2950 switches with GBIC module slots, and only on modular Catalyst 2900 XL switches that have the 1000BASE-X module installed.
- Up to nine stack switches can be connected through their stack ports to the multidrop backbone. Only one stack port per switch is supported.
- Each stack switch can be connected to the spanning-tree backbone through one uplink.
- If the stack consists of a mixture of Catalyst 3550, Catalyst 3500 XL, Catalyst 2950, and Catalyst 2900 XL switches, up to 64 VLANs with spanning tree enabled are supported. If the stack consists of only Catalyst 3550 switches, up to 128 VLANs with spanning tree enabled are supported.

Connecting the Stack Ports

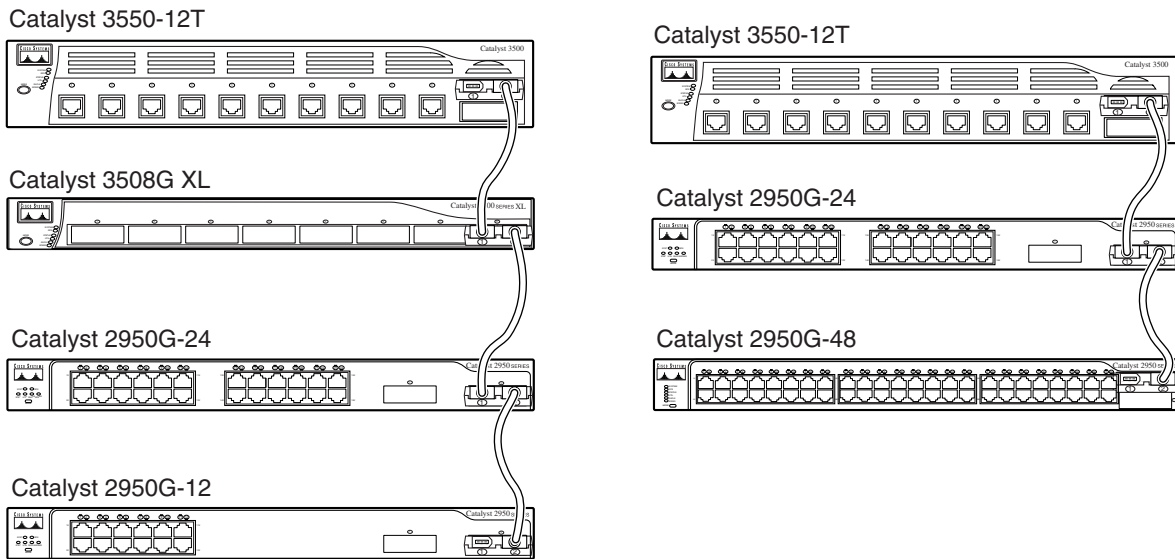
A fast transition occurs across the stack of switches if the multidrop backbone connections are a continuous link from one GigaStack GBIC module to another as shown in the top half of [Figure 17-6](#). The bottom half of [Figure 17-6](#) shows how to connect the GigaStack GBIC module to achieve a normal convergence time.

You should follow these guidelines:

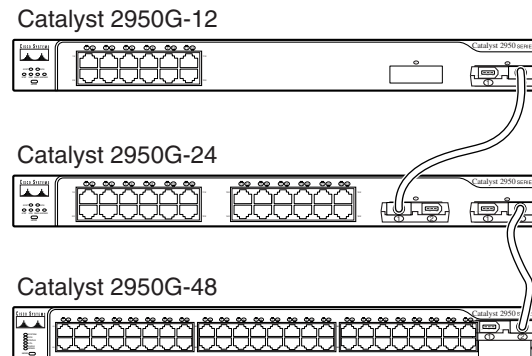
- A switch supports only one stack port.
- Do not connect alternate stack-root ports to stack ports.
- Connect all stack ports on the switch stack to the multidrop backbone.
- You can connect the open ports on the top and bottom GigaStack GBIC modules within the same stack to form a redundant link.

Figure 17-6 GigaStack GBIC Module Connections and Spanning-Tree Convergence

GigaStack GBIC connection for fast convergence



GigaStack GBIC connection for normal convergence



65276

Understanding BackboneFast

BackboneFast detects indirect failures in the core of the backbone. BackboneFast is a complementary technology to the UplinkFast feature, which responds to failures on links directly connected to access switches. BackboneFast optimizes the maximum-age timer, which determines the amount of time the switch stores protocol information received on an interface. When a switch receives an inferior BPDU from the designated port of another switch, the BPDU is a signal that the other switch might have lost its path to the root, and BackboneFast tries to find an alternate path to the root.

BackboneFast, which is enabled by using the **spanning-tree backbonefast** global configuration command, starts when a root port or blocked port on a switch receives inferior BPDUs from its designated switch. An inferior BPDU identifies a switch that declares itself as both the root bridge and the designated switch. When a switch receives an inferior BPDU, it means that a link to which the switch is not directly connected (an *indirect* link) has failed (that is, the designated bridge has lost its connection

to the root switch). Under spanning-tree rules, the switch ignores inferior BPDUs for the configured maximum aging time specified by the **spanning-tree vlan *vlan-id* max-age** global configuration command.

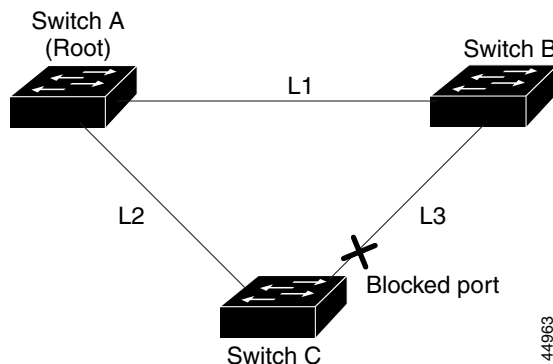
The switch tries to determine if it has an alternate path to the root switch. If the inferior BPDU arrives on a blocked port, the root port and other blocked ports on the switch become alternate paths to the root switch. (Self-looped ports are not considered alternate paths to the root switch.) If the inferior BPDU arrives on the root port, all blocked ports become alternate paths to the root switch. If the inferior BPDU arrives on the root port and there are no blocked ports, the switch assumes that it has lost connectivity to the root switch, causes the maximum aging time on the root port to expire, and becomes the root switch according to normal spanning-tree rules.

If the switch has alternate paths to the root switch, it uses these alternate paths to send a root link query (RLQ) request. The switch sends the RLQ request on all alternate paths to the root switch and waits for an RLQ reply from other switches in the network.

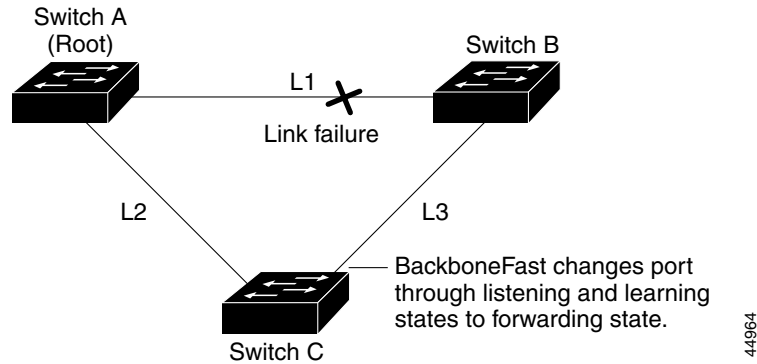
If the switch determines that it still has an alternate path to the root, it expires the maximum aging time on the port that received the inferior BPDU. If all the alternate paths to the root switch indicate that the switch has lost connectivity to the root switch, the switch expires the maximum aging time on the port that received the RLQ reply. If one or more alternate paths can still connect to the root switch, the switch makes all ports on which it received an inferior BPDU its designated ports and moves them from the blocking state (if they were in the blocking state), through the listening and learning states, and into the forwarding state.

Figure 17-7 shows an example topology with no link failures. Switch A, the root switch, connects directly to Switch B over link L1 and to Switch C over link L2. The Layer 2 interface on Switch C that connects directly to Switch B is in the blocking state.

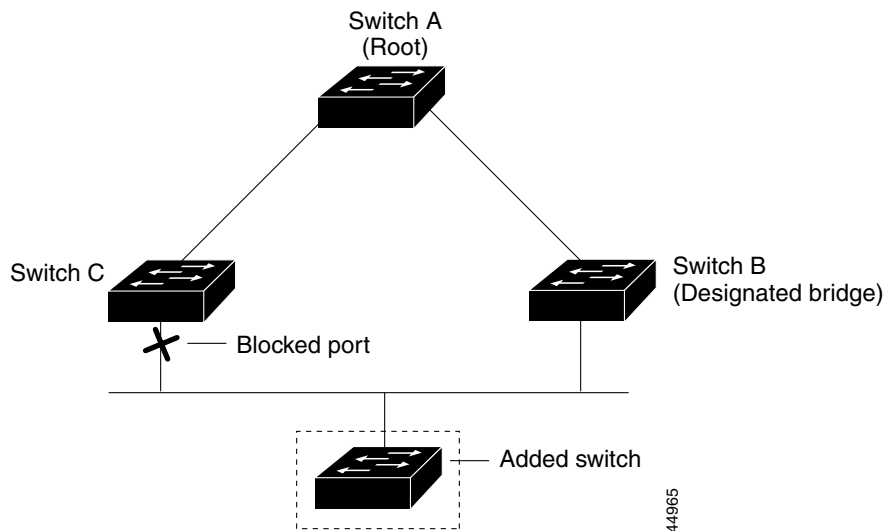
Figure 17-7 BackboneFast Example Before Indirect Link Failure



If link L1 fails as shown in Figure 17-8, Switch C cannot detect this failure because it is not connected directly to link L1. However, because Switch B is directly connected to the root switch over L1, it detects the failure, elects itself the root, and begins sending BPDUs to Switch C, identifying itself as the root. When Switch C receives the inferior BPDUs from Switch B, Switch C assumes that an indirect failure has occurred. At that point, BackboneFast allows the blocked port on Switch C to move immediately to the listening state without waiting for the maximum aging time for the port to expire. BackboneFast then transitions the Layer 2 interface on Switch C to the forwarding state, providing a path from Switch B to Switch A. This switchover takes approximately 30 seconds, twice the Forward Delay time if the default Forward Delay time of 15 seconds is set. Figure 17-8 shows how BackboneFast reconfigures the topology to account for the failure of link L1.

Figure 17-8 BackboneFast Example After Indirect Link Failure

If a new switch is introduced into a shared-medium topology as shown in [Figure 17-9](#), BackboneFast is not activated because the inferior BPDUs did not come from the recognized designated bridge (Switch B). The new switch begins sending inferior BPDUs that indicate it is the root switch. However, the other switches ignore these inferior BPDUs, and the new switch learns that Switch B is the designated bridge to Switch A, the root switch.

Figure 17-9 Adding a Switch in a Shared-Medium Topology

Understanding EtherChannel Guard

You can use EtherChannel guard to detect an EtherChannel misconfiguration between the switch and a connected device. A misconfiguration can occur if the switch interfaces are configured in an EtherChannel, but the interfaces on the other device are not. A misconfiguration can also occur if the channel parameters are not the same at both ends of the EtherChannel. For EtherChannel configuration guidelines, see the [“EtherChannel Configuration Guidelines”](#) section on page 30-8.

If the switch detects a misconfiguration on the other device, EtherChannel guard places the switch interfaces in the error-disabled state, and this error message appears:

```
PM-4-ERR_DISABLE: Channel-misconfig error detected on [chars], putting [chars] in
err-disable state.
```

You can enable this feature by using the **spanning-tree etherchannel guard misconfig** global configuration command.

Understanding Root Guard

The Layer 2 network of a service provider (SP) can include many connections to switches that are not owned by the SP. In such a topology, the spanning tree can reconfigure itself and select a *customer switch* as the root switch, as shown in [Figure 17-10](#). You can avoid this situation by enabling root guard on SP switch interfaces that connect to switches in your customer's network. If spanning-tree calculations cause an interface in the customer network to be selected as the root port, root guard then places the interface in the root-inconsistent (blocked) state to prevent the customer's switch from becoming the root switch or being in the path to the root.

If a switch outside the SP network becomes the root switch, the interface is blocked (root-inconsistent state), and spanning tree selects a new root switch. The customer's switch does not become the root switch and is not in the path to the root.

If the switch is operating in multiple spanning-tree (MST) mode, root guard forces the port to be a designated port. If a boundary port is blocked in an internal spanning-tree (IST) instance because of root guard, the port also is blocked in all MST instances. A boundary port is a port that connects to a LAN, the designated switch of which is either an IEEE 802.1D switch or a switch with a different MST region configuration.

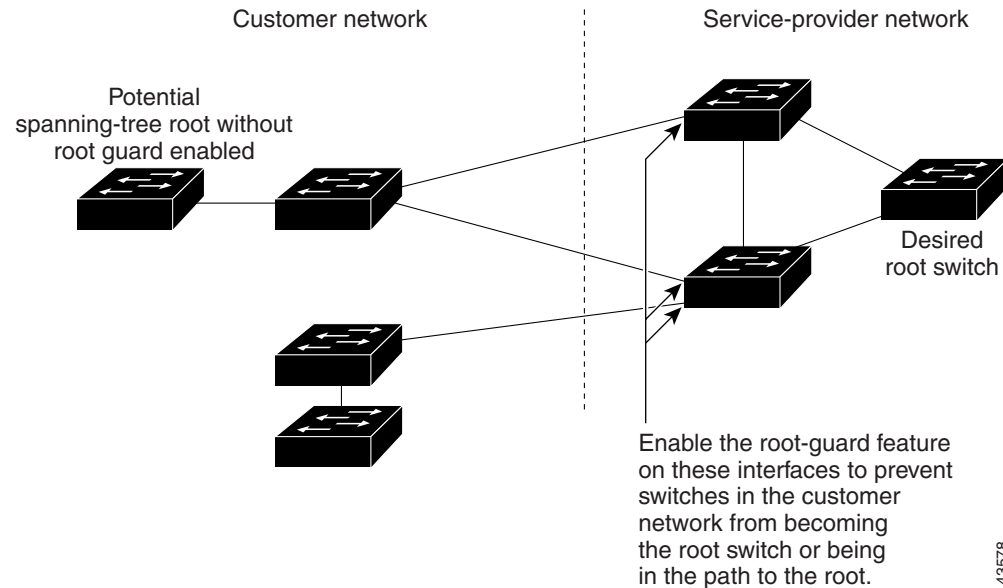
Root guard enabled on an interface applies to all the VLANs to which the interface belongs. VLANs can be grouped and mapped to an MST instance.

You can enable this feature by using the **spanning-tree guard root** interface configuration command.



Caution

Misuse of the root-guard feature can cause a loss of connectivity.

Figure 17-10 Root Guard in a Service-Provider Network

Understanding Loop Guard

You can use loop guard to prevent alternate or root ports from becoming designated ports because of a failure that leads to a unidirectional link. This feature is most effective when it is configured on the entire switched network.

You can enable this feature by using the **spanning-tree loopguard default** global configuration command.

When the switch is operating in PVST+ or rapid-PVST+ mode, loop guard prevents alternate and root ports from becoming designated ports, and spanning tree does not send BPDUs on root or alternate ports.

When the switch is operating in MST mode, BPDUs are not sent on nonboundary ports only if the port is blocked by loop guard in all MST instances. On a boundary port, loop guard blocks the port in all MST instances.

Configuring Optional Spanning-Tree Features

These sections describe how to configure optional spanning-tree features:

- [Default Optional Spanning-Tree Configuration, page 17-14](#)
- [Optional Spanning-Tree Configuration Guidelines, page 17-14](#)
- [Enabling Port Fast, page 17-14](#) (optional)
- [Enabling BPDU Guard, page 17-15](#) (optional)
- [Enabling BPDU Filtering, page 17-16](#) (optional)
- [Enabling UplinkFast for Use with Redundant Links, page 17-17](#) (optional)
- [Enabling Cross-Stack UplinkFast, page 17-18](#) (optional)

- [Enabling BackboneFast, page 17-19](#) (optional)
- [Enabling EtherChannel Guard, page 17-20](#) (optional)
- [Enabling Root Guard, page 17-20](#) (optional)
- [Enabling Loop Guard, page 17-21](#) (optional)

Default Optional Spanning-Tree Configuration

Table 17-1 shows the default optional spanning-tree configuration.

Table 17-1 Default Optional Spanning-Tree Configuration

Feature	Default Setting
Port Fast, BPDU filtering, BPDU guard	Globally disabled (unless they are individually configured per interface).
UplinkFast	Globally disabled.
CSUF	Disabled on all interfaces.
BackboneFast	Globally disabled.
EtherChannel guard	Globally enabled.
Root guard	Disabled on all interfaces.
Loop guard	Disabled on all interfaces.

Optional Spanning-Tree Configuration Guidelines

You can configure PortFast, BPDU guard, BPDU filtering, EtherChannel guard, root guard, or loop guard if your switch is running PVST+, rapid PVST+, or MSTP.

You can configure the UplinkFast, the BackboneFast, or the cross-stack UplinkFast feature for rapid PVST+ or for the MSTP, but the feature remains disabled (inactive) until you change the spanning-tree mode to PVST+.

Enabling Port Fast

A port with the Port Fast feature enabled is moved directly to the spanning-tree forwarding state without waiting for the standard forward-time delay.




Caution

Use Port Fast *only* when connecting a single end station to an access or trunk port. Enabling this feature on a port connected to a switch or hub could prevent spanning tree from detecting and disabling loops in your network, which could cause broadcast storms and address-learning problems.

If you enable the voice VLAN feature, the Port Fast feature is automatically enabled. When you disable voice VLAN, the Port Fast feature is not automatically disabled. For more information, see [Chapter 13, “Configuring Voice VLAN.”](#)

You can enable this feature if your switch is running PVST+, rapid PVST+, or MSTP.

Beginning in privileged EXEC mode, follow these steps to enable Port Fast. This procedure is optional.

	Command	Purpose
Step 1	configure terminal	Enter global configuration mode.
Step 2	interface <i>interface-id</i>	Specify an interface to configure, and enter interface configuration mode.
Step 3	spanning-tree portfast [trunk]	<p>Enable Port Fast on an access port connected to a single workstation or server. By specifying the trunk keyword, you can enable Port Fast on a trunk port.</p> <p>Note To enable Port Fast on trunk ports, you must use the spanning-tree portfast trunk interface configuration command. The spanning-tree portfast command will not work on trunk ports.</p> <p> Caution Make sure that there are no loops in the network between the trunk port and the workstation or server before you enable Port Fast on a trunk port.</p> <p>By default, Port Fast is disabled on all ports.</p>
Step 4	end	Return to privileged EXEC mode.
Step 5	show spanning-tree interface <i>interface-id</i> portfast	Verify your entries.
Step 6	copy running-config startup-config	(Optional) Save your entries in the configuration file.



Note You can use the **spanning-tree portfast default** global configuration command to globally enable the Port Fast feature on all nontrunking ports.

To disable the Port Fast feature, use the **spanning-tree portfast disable** interface configuration command.

Enabling BPDU Guard

When you globally enable BPDU guard on ports that are Port Fast-enabled (the ports are in a Port Fast-operational state), spanning tree continues to run on the ports. They remain up unless they receive a BPDU.

In a valid configuration, Port Fast-enabled ports do not receive BPDUs. Receiving a BPDU on a Port Fast-enabled port signals an invalid configuration, such as the connection of an unauthorized device, and the BPDU guard feature puts the port in the error-disabled state. The BPDU guard feature provides a secure response to invalid configurations because you must manually put the port back in service. Use the BPDU guard feature in a service-provider network to prevent an access port from participating in the spanning tree.

**Caution**

Configure Port Fast only on ports that connect to end stations; otherwise, an accidental topology loop could cause a data packet loop and disrupt switch and network operation.

You can also use the **spanning-tree bpduguard enable** interface configuration command to enable BPDU guard on any port without also enabling the Port Fast feature. When the port receives a BPDU, it is put in the error-disabled state.

You can enable the BPDU guard feature if your switch is running PVST+, rapid PVST+, or MSTP.

Beginning in privileged EXEC mode, follow these steps to globally enable the BPDU guard feature. This procedure is optional.

	Command	Purpose
Step 1	configure terminal	Enter global configuration mode.
Step 2	spanning-tree portfast bpduguard default	Globally enable BPDU guard. By default, BPDU guard is disabled.
Step 3	interface <i>interface-id</i>	Specify the interface connected to an end station, and enter interface configuration mode.
Step 4	spanning-tree portfast	Enable the Port Fast feature.
Step 5	end	Return to privileged EXEC mode.
Step 6	show running-config	Verify your entries.
Step 7	copy running-config startup-config	(Optional) Save your entries in the configuration file.

To disable BPDU guard, use the **no spanning-tree portfast bpduguard default** global configuration command.

You can override the setting of the **no spanning-tree portfast bpduguard default** global configuration command by using the **spanning-tree bpduguard enable** interface configuration command.

Enabling BPDU Filtering

When you globally enable BPDU filtering on Port Fast-enabled ports, it prevents ports that are in a Port Fast-operational state from sending or receiving BPDUs. The ports still send a few BPDUs at link-up before the switch begins to filter outbound BPDUs. You should globally enable BPDU filtering on a switch so that hosts connected to these ports do not receive BPDUs. If a BPDU is received on a Port Fast-enabled port, the port loses its Port Fast-operational status, and BPDU filtering is disabled.

**Caution**

Configure Port Fast only on ports that connect to end stations; otherwise, an accidental topology loop could cause a data packet loop and disrupt switch and network operation.

You can also use the **spanning-tree bpdudfilter enable** interface configuration command to enable BPDU filtering on any port without also enabling the Port Fast feature. This command prevents the port from sending or receiving BPDUs.

**Caution**

Enabling BPDU filtering on an interface is the same as disabling spanning tree on it and can result in spanning-tree loops.

You can enable the BPDU filtering feature if your switch is running PVST+, rapid PVST+, or MSTP.

Beginning in privileged EXEC mode, follow these steps to globally enable the BPDU filtering feature. This procedure is optional.

	Command	Purpose
Step 1	configure terminal	Enter global configuration mode.
Step 2	spanning-tree portfast bpdupfilter default	Globally enable BPDU filtering. By default, BPDU filtering is disabled.
Step 3	interface <i>interface-id</i>	Specify the interface connected to an end station, and enter interface configuration mode.
Step 4	spanning-tree portfast	Enable the Port Fast feature.
Step 5	end	Return to privileged EXEC mode.
Step 6	show running-config	Verify your entries.
Step 7	copy running-config startup-config	(Optional) Save your entries in the configuration file.

To disable BPDU filtering, use the **no spanning-tree portfast bpdupfilter default** global configuration command.

You can override the setting of the **no spanning-tree portfast bpdupfilter default** global configuration command by using the **spanning-tree bpdupfilter enable** interface configuration command.

Enabling UplinkFast for Use with Redundant Links

UplinkFast cannot be enabled on VLANs that have been configured for switch priority. To enable UplinkFast on a VLAN with switch priority configured, first restore the switch priority on the VLAN to the default value by using the **no spanning-tree vlan *vlan-id* priority** global configuration command.

**Note**

When you enable UplinkFast, it affects all VLANs on the switch. You cannot configure UplinkFast on an individual VLAN.

You can enable the UplinkFast feature for rapid PVST+ or for the MSTP, but the feature remains disabled (inactive) until you change the spanning-tree mode to PVST+.

Beginning in privileged EXEC mode, follow these steps to enable UplinkFast. This procedure is optional.

	Command	Purpose
Step 1	configure terminal	Enter global configuration mode.
Step 2	spanning-tree uplinkfast [max-update-rate <i>pkts-per-second</i>]	Enable UplinkFast. (Optional) For <i>pkts-per-second</i> , the range is 0 to 32000 packets per second; the default is 150. If you set the rate to 0, station-learning frames are not generated, and the spanning-tree topology converges more slowly after a loss of connectivity.
Step 3	end	Return to privileged EXEC mode.
Step 4	show spanning-tree summary	Verify your entries.
Step 5	copy running-config startup-config	(Optional) Save your entries in the configuration file.

When UplinkFast is enabled, the switch priority of all VLANs is set to 49152. If you change the path cost to a value less than 3000 and you enable UplinkFast or UplinkFast is already enabled, the path cost of all interfaces and VLAN trunks is increased by 3000 (if you change the path cost to 3000 or above, the path cost is not altered). The changes to the switch priority and the path cost reduces the chance that the switch will become the root switch.

When UplinkFast is disabled, the switch priorities of all VLANs and path costs of all interfaces are set to default values if you did not modify them from their defaults.

To return the update packet rate to the default setting, use the **no spanning-tree uplinkfast max-update-rate** global configuration command. To disable UplinkFast, use the **no spanning-tree uplinkfast** command.

Enabling Cross-Stack UplinkFast

Before enabling CSUF, make sure your stack switches are properly connected. For more information, see the [“Connecting the Stack Ports”](#) section on page 17-8.

You can enable the CSUF feature for rapid PVST+ or for the MSTP, but the feature remains disabled (inactive) until you change the spanning-tree mode to PVST+.

Beginning in privileged EXEC mode, follow these steps to enable CSUF. This procedure is optional.

	Command	Purpose
Step 1	configure terminal	Enter global configuration mode.
Step 2	spanning-tree uplinkfast [max-update-rate <i>pkts-per-second</i>]	Enable UplinkFast on the switch. (Optional) For max-update-rate <i>pkts-per-second</i> , specify the number of packets per second at which update packets are sent. The range is 0 to 65535; the default is 150 packets per second.
Step 3	interface <i>interface-id</i>	Specify the GBIC module interface on which to enable CSUF, and enter interface configuration mode.

	Command	Purpose
Step 4	spanning-tree stack-port	Enable CSUF on only one stack-port GBIC interface. The stack port connects to the GigaStack GBIC module multidrop backbone. If you try to enable CSUF on a Fast Ethernet or a Gigabit-capable Ethernet port, you receive an error message. If CSUF is already enabled on an interface and you try to enable it on another interface, you receive an error message. You must disable CSUF on the first interface before enabling it on a new interface. Use this command only on access switches.
Step 5	end	Return to privileged EXEC mode.
Step 6	show running-config	Verify your entries.
Step 7	copy running-config startup-config	(Optional) Save your entries in the configuration file.

To disable CSUF on an interface, use the **no spanning-tree stack-port** interface configuration command. To disable UplinkFast on the switch and all its VLANs, use the **no spanning-tree uplinkfast** global configuration command.

Enabling BackboneFast

You can enable BackboneFast to detect indirect link failures and to start the spanning-tree reconfiguration sooner.



Note

If you use BackboneFast, you must enable it on all switches in the network. BackboneFast is not supported on Token Ring VLANs. This feature is supported for use with third-party switches.

You can enable the BackboneFast feature for rapid PVST+ or for the MSTP, but the feature remains disabled (inactive) until you change the spanning-tree mode to PVST+.

Beginning in privileged EXEC mode, follow these steps to enable BackboneFast. This procedure is optional.

	Command	Purpose
Step 1	configure terminal	Enter global configuration mode.
Step 2	spanning-tree backbonefast	Enable BackboneFast.
Step 3	end	Return to privileged EXEC mode.
Step 4	show spanning-tree summary	Verify your entries.
Step 5	copy running-config startup-config	(Optional) Save your entries in the configuration file.

To disable the BackboneFast feature, use the **no spanning-tree backbonefast** global configuration command.

Enabling EtherChannel Guard

You can enable EtherChannel guard to detect an EtherChannel misconfiguration if your switch is running PVST+, rapid PVST+, or MSTP.

Beginning in privileged EXEC mode, follow these steps to enable EtherChannel guard. This procedure is optional.

	Command	Purpose
Step 1	configure terminal	Enter global configuration mode.
Step 2	spanning-tree etherchannel guard misconfig	Enable EtherChannel guard.
Step 3	end	Return to privileged EXEC mode.
Step 4	show spanning-tree summary	Verify your entries.
Step 5	copy running-config startup-config	(Optional) Save your entries in the configuration file.

To disable the EtherChannel guard feature, use the **no spanning-tree etherchannel guard misconfig** global configuration command.

You can use the **show interfaces status err-disabled** privileged EXEC command to show which switch ports are disabled because of an EtherChannel misconfiguration. On the remote device, you can enter the **show etherchannel summary** privileged EXEC command to verify the EtherChannel configuration.

After the configuration is corrected, enter the **shutdown** and **no shutdown** interface configuration commands on the port-channel interfaces that were misconfigured.

Enabling Root Guard

Root guard enabled on an interface applies to all the VLANs to which the interface belongs.

Do not enable the root guard on interfaces to be used by the UplinkFast feature. With UplinkFast, the backup interfaces (in the blocked state) replace the root port in the case of a failure. However, if root guard is also enabled, all the backup interfaces used by the UplinkFast feature are placed in the root-inconsistent state (blocked) and are prevented from reaching the forwarding state.



Note

You cannot enable both root guard and loop guard at the same time.

You can enable this feature if your switch is running PVST+, rapid PVST+, or MSTP.

Beginning in privileged EXEC mode, follow these steps to enable root guard on an interface. This procedure is optional.

	Command	Purpose
Step 1	configure terminal	Enter global configuration mode.
Step 2	interface <i>interface-id</i>	Specify an interface to configure, and enter interface configuration mode.
Step 3	spanning-tree guard root	Enable root guard on the interface. By default, root guard is disabled on all interfaces.

	Command	Purpose
Step 4	end	Return to privileged EXEC mode.
Step 5	show running-config	Verify your entries.
Step 6	copy running-config startup-config	(Optional) Save your entries in the configuration file.

To disable root guard, use the **no spanning-tree guard** interface configuration command.

Enabling Loop Guard

You can use loop guard to prevent alternate or root ports from becoming designated ports because of a failure that leads to a unidirectional link. This feature is most effective when it is configured on the entire switched network. Loop guard operates only on ports that are considered point-to-point by the spanning tree.



Note

You cannot enable both loop guard and root guard at the same time.

You can enable this feature if your switch is running PVST+, rapid PVST+, or MSTP.

Beginning in privileged EXEC mode, follow these steps to enable loop guard. This procedure is optional.

	Command	Purpose
Step 1	show spanning-tree active or show spanning-tree mst	Determine which ports are alternate or root ports.
Step 2	configure terminal	Enter global configuration mode.
Step 3	spanning-tree loopguard default	Enable loop guard. By default, loop guard is disabled.
Step 4	end	Return to privileged EXEC mode.
Step 5	show running-config	Verify your entries.
Step 6	copy running-config startup-config	(Optional) Save your entries in the configuration file.

To globally disable loop guard, use the **no spanning-tree loopguard default** global configuration command. You can override the setting of the **no spanning-tree loopguard default** global configuration command by using the **spanning-tree guard loop** interface configuration command.

Displaying the Spanning-Tree Status

To display the spanning-tree status, use one or more of the privileged EXEC commands in [Table 17-2](#):

Table 17-2 **Commands for Displaying the Spanning-Tree Status**

Command	Purpose
show spanning-tree active	Displays spanning-tree information on active interfaces only.
show spanning-tree detail	Displays a detailed summary of interface information.
show spanning-tree interface <i>interface-id</i>	Displays spanning-tree information for the specified interface.
show spanning-tree mst interface <i>interface-id</i>	Displays MST information for the specified interface.
show spanning-tree summary [totals]	Displays a summary of port states or displays the total lines of the spanning-tree state section.

You can clear spanning-tree counters by using the **clear spanning-tree [interface *interface-id*]** privileged EXEC command.

For information about other keywords for the **show spanning-tree** privileged EXEC command, see the command reference for this release.