CHAPTER 20

Configuring Optional Spanning-Tree Features

This chapter describes how to configure optional spanning-tree features on the 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 or switch stack is running the Multiple Spanning Tree Protocol (MSTP) or the rapid per-VLAN spanning-tree plus (rapid-PVST+) protocol. Unless otherwise noted, the term *switch* refers to a standalone switch and to a witch stack.

For information on configuring the PVST+ and rapid PVST+, see Chapter 18, "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 19, "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

These sections contain this conceptual information:

Understanding Port Fast

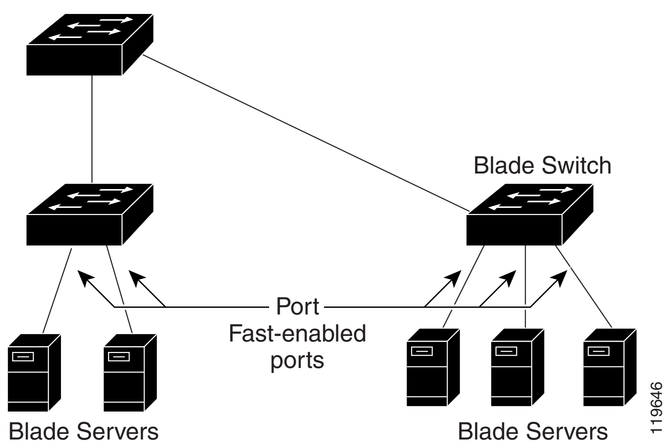
Port Fast immediately brings an interface configured as an access or trunk port to the forwarding state from a blocking state, bypassing the listening and learning states. You can use Port Fast on interfaces connected to a single workstation or server, as shown in Figure 20-1, to allow those devices to immediately connect to the network, rather than waiting for the spanning tree to converge.

Interfaces connected to a blade server should not receive bridge protocol data units (BPDUs). An interface 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 interfaces must wait for spanning-tree to converge, it is effective only when used on interfaces connected to end stations. If you enable Port Fast on an interface connecting 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 20-1 Port Fast-Enabled Interfaces



Understanding BPDU Guard

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

At the global level, you 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 them. In a valid configuration, Port Fast-enabled ports do not receive BPDUs. Receiving a BPDU on a Port Fast-enabled port means an invalid configuration, such as the connection of an unauthorized device, and the BPDU guard feature puts the port in the error-disabled state. When this happens, the switch shuts down the entire port on which the violation occurred.

To prevent the port from shutting down, you can use the **errdisable detect cause bpduguard shutdown vlan** global configuration command to shut down just the offending VLAN on the port where the violation occurred.

At the interface level, you 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 interface back in service. Use the BPDU guard feature in a service-provider network to prevent an access port from participating in the spanning tree.

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 interfaces by using the **spanning-tree portfast bpdufilter default** global configuration command. This command prevents interfaces that are in a Port Fast-operational state from sending or receiving BPDUs. The interfaces 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 interfaces do not receive BPDUs. If a BPDU is received on a Port Fast-enabled interface, the interface loses its Port Fast-operational status, and BPDU filtering is disabled.

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

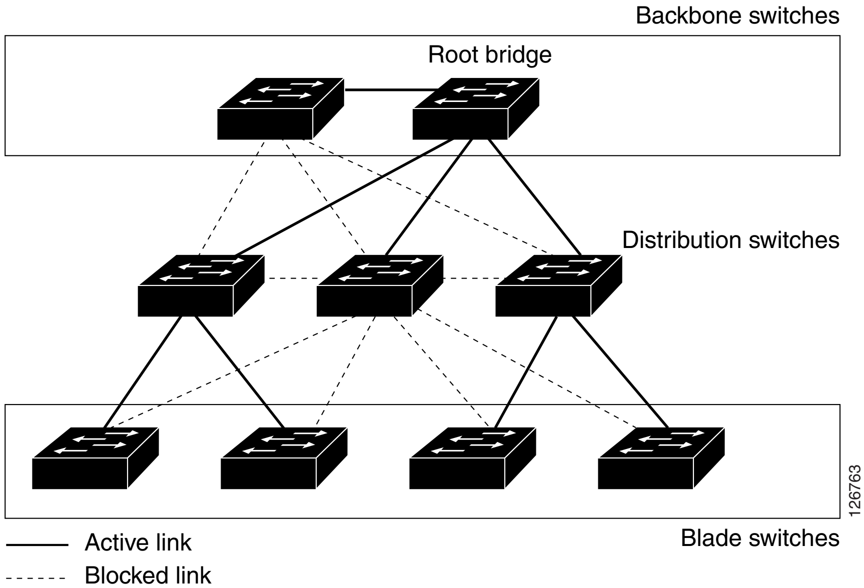
Caution Enabling BPDU filtering on an interface is the same as disabling spanning tree on it and can result in spanning-tree loops.

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 20-2 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 20-2 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.

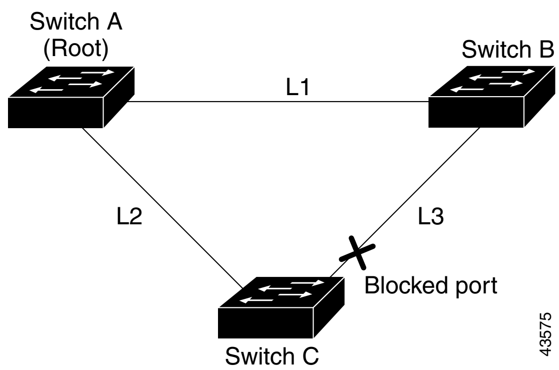
When the spanning tree reconfigures the new root port, other interfaces flood the network with multicast packets, one for each address that was learned on the interface. You can limit these bursts of multicast traffic by reducing the max-update-rate parameter (the default for this parameter is 150 packets 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.

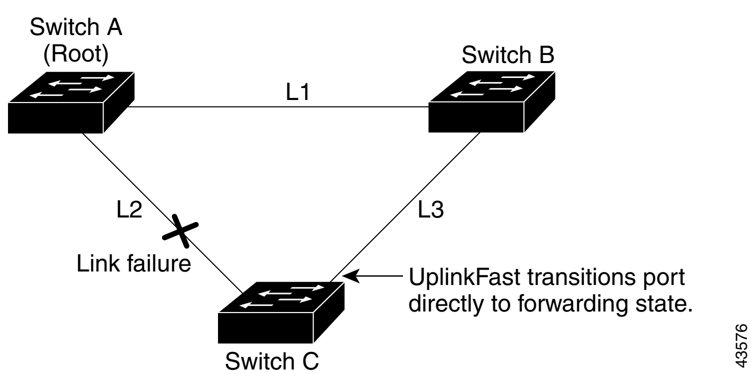
Figure 20-3 shows an example topology with no link failures. Switch A, the root switch, is connected directly to Switch B over link L1 and to Switch C over link L2. The Layer 2 interface on Switch C that is connected directly to Switch B is in a blocking state.

Figure 20-3 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 interface on Switch C and transitions it to the forwarding state without going through the listening and learning states, as shown in Figure 20-4. This change takes approximately 1 to 5 seconds.

Figure 20-4 UplinkFast Example After Direct Link Failure



Understanding Cross-Stack UplinkFast

The UplinkFast feature is the cross-stack UplinkFast feature. Cross-stack UplinkFast (CSUF) provides a fast spanning-tree transition (fast convergence in less than 1 second under normal network conditions) across a switch stack. During the fast transition, an alternate redundant link on the switch stack 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. CSUF is automatically enabled when you enable the UplinkFast feature by using the **spanning-tree uplinkfast** global configuration command.

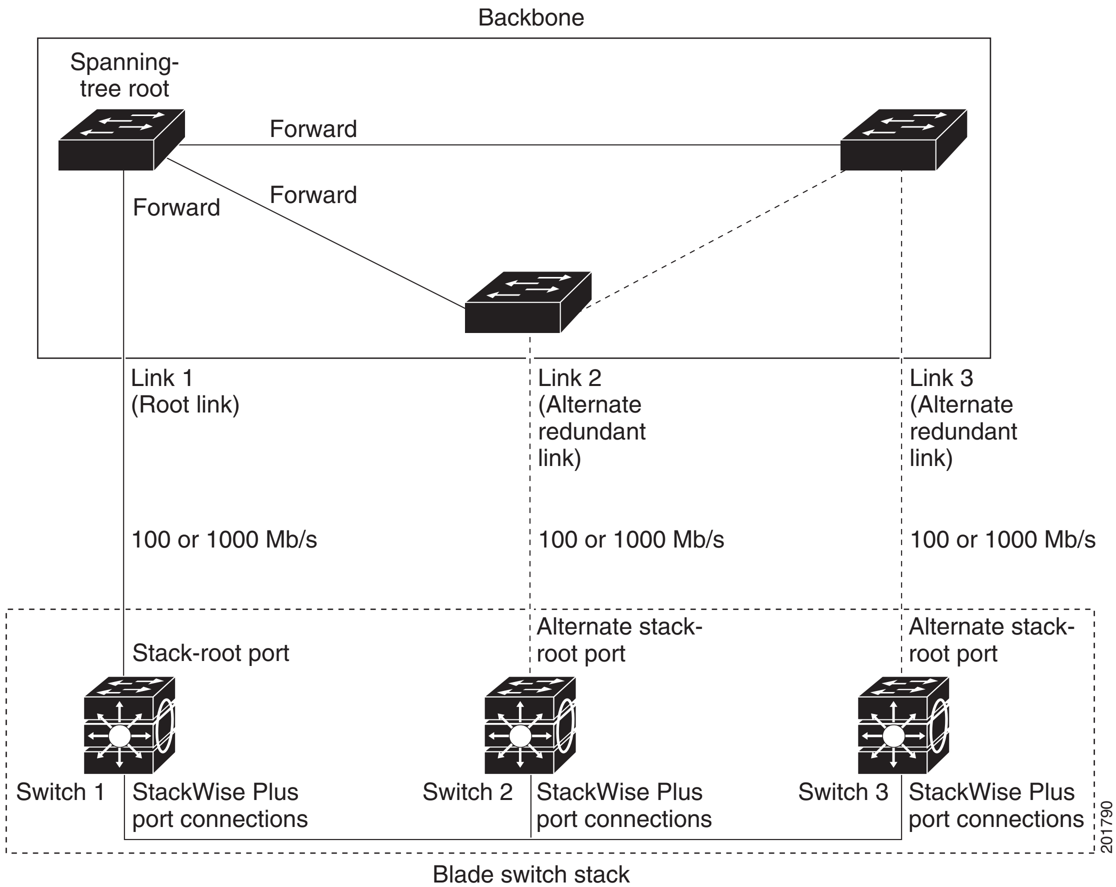
CSUF might not provide a fast transition all the time; in these cases, the normal spanning-tree transition occurs, completing in 30 to 40 seconds. For more information, see the "Events that Cause Fast Convergence" section on page 20-7.

How CSUF Works

CSUF ensures that one link in the stack is elected as the path to the root. As shown in Figure 20-5, the stack-root port on Switch 1 provides the path to the root of the spanning tree. The alternate stack-root ports on Switches 2 and 3 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 1, the root link, is in the spanning-tree forwarding state. Links 2 and 3 are alternate redundant links that are in the spanning-tree blocking state. If Switch 1 fails, if its stack-root port fails, or if Link 1 fails, CSUF selects either the alternate stack-root port on Switch 2 or Switch 3 and puts it into the forwarding state in less than 1 second.

Figure 20-5 Cross-Stack UplinkFast Topology



When certain link loss or spanning-tree events occur (described in the "Events that Cause Fast Convergence" section on page 20-7), the Fast Uplink Transition Protocol uses the neighbor list to send fast-transition requests 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 decides 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 sends a fast-transition request. 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 * forward-delay time + 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 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.

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 controls 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 interface 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 switch 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 find if it has an alternate path to the root switch. If the inferior BPDU arrives on a blocked interface, the root port and other blocked interfaces 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 interfaces become alternate paths to the root switch. If the inferior BPDU arrives on the root port and there are no blocked interfaces, 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 learn if any stack member has an alternate root to the root switch and waits for an RLQ reply from other switches in the network and in the stack

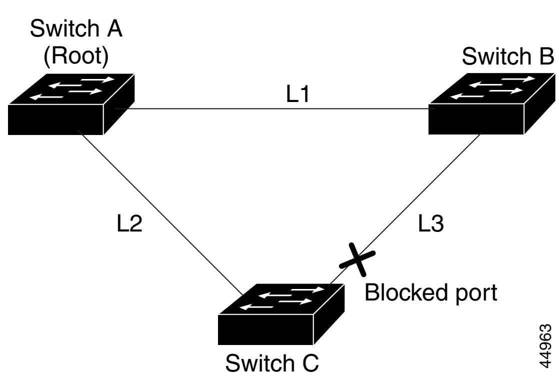
When a stack member receives an RLQ reply from a nonstack member on a blocked interface and the reply is destined for another nonstacked switch, it forwards the reply packet, regardless of the spanning-tree interface state.

When a stack member receives an RLQ reply from a nonstack member and the response is destined for the stack, the stack member forwards the reply so that all the other stack members receive it.

If the switch discovers that it still has an alternate path to the root, it expires the maximum aging time on the interface 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 interface that received the RLQ reply. If one or more alternate paths can still connect to the root switch, the switch makes all interfaces 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.

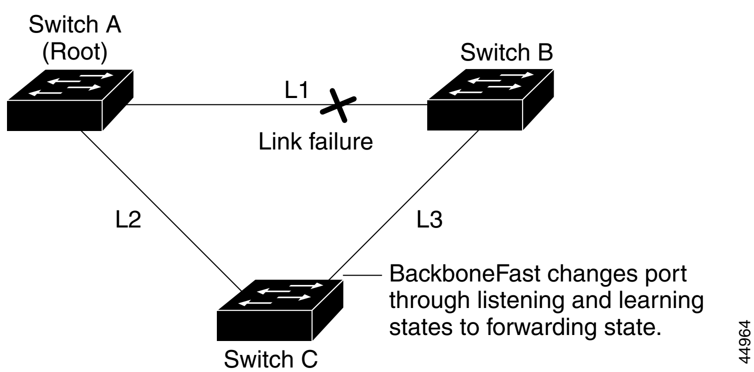
Figure 20-6 shows an example topology with no link failures. Switch A, the root switch, connects directly to Switch B over link L1 and to Switch C over link L2. The Layer 2 interface on Switch C that connects directly to Switch B is in the blocking state.

Figure 20-6 BackboneFast Example Before Indirect Link Failure



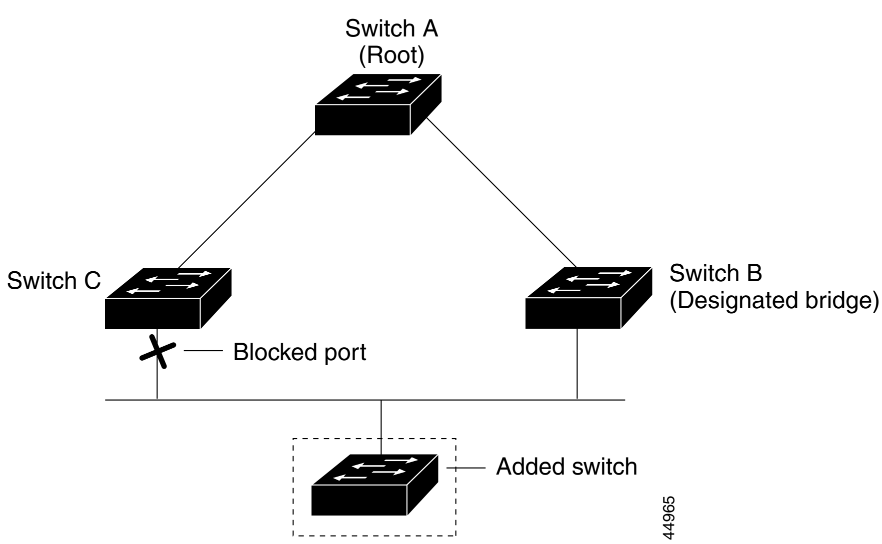
If link L1 fails as shown in Figure 20-7, Switch C cannot detect this failure because it is not connected directly to link L1. However, because Switch B is directly connected to the root switch over L1, it detects the failure, elects itself the root, and begins sending BPDUs to Switch C, identifying 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 interface on Switch C to move immediately to the listening state without waiting for the maximum aging time for the interface 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. The root-switch election takes approximately 30 seconds, twice the Forward Delay time if the default Forward Delay time of 15 seconds is set. Figure 20-7 shows how BackboneFast reconfigures the topology to account for the failure of link L1.

Figure 20-7 BackboneFast Example After Indirect Link Failure



If a new switch is introduced into a shared-medium topology as shown in Figure 20-8, BackboneFast is not activated because the inferior BPDUs did not come from the recognized designated switch (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 switch to Switch A, the root switch.

Figure 20-8 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 38-11.

If the switch detects a misconfiguration on the other device, EtherChannel guard places the switch interfaces in the error-disabled state, and displays an error message.

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 data center can include many connections to switches that are not owned by the data center. In such a topology, the spanning tree can reconfigure itself and select a *customer switch* as the root switch, as shown in Figure 20-9. You can avoid this situation by enabling root guard on data center 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 data center 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.

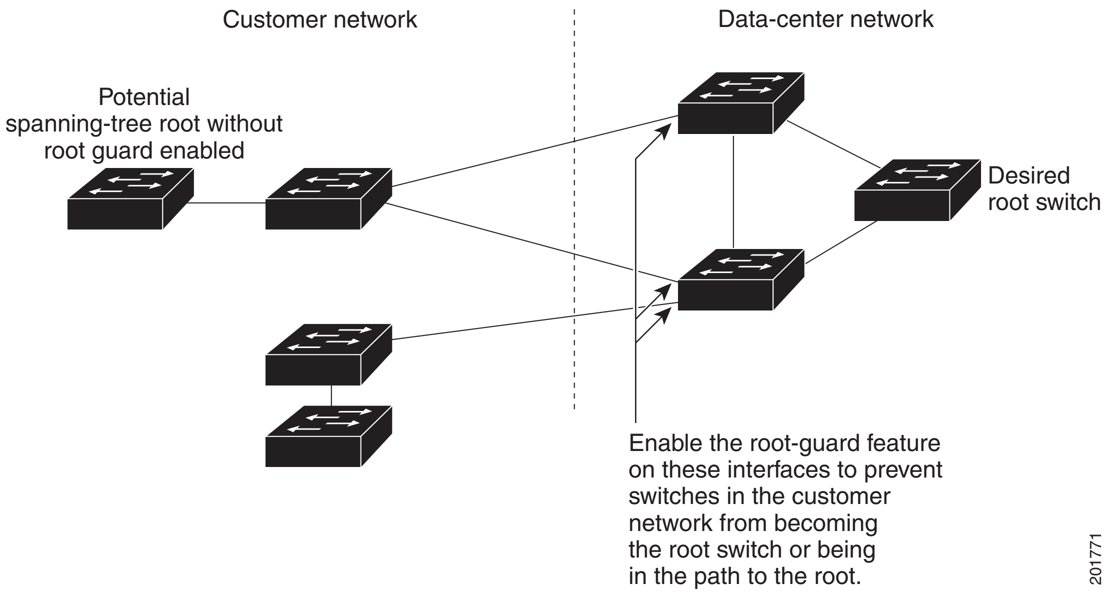
If the switch is operating in multiple spanning-tree (MST) mode, root guard forces the interface to be a designated port. If a boundary port is blocked in an internal spanning-tree (IST) instance because of root guard, the interface also is blocked in all MST instances. A boundary port is an interface 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 20-9 Root Guard in a Data-Center 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 enabled on the entire switched network. Loop guard prevents alternate and root ports from becoming designated ports, and spanning tree does not send BPDUs on root or alternate ports.

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 interface is blocked by loop guard in all MST instances. On a boundary port, loop guard blocks the interface in all MST instances.

Configuring Optional Spanning-Tree Features

These sections contain this configuration information:

Default Optional Spanning-Tree Configuration

Table 20-1 shows the default optional spanning-tree configuration.

Table 20-1 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. (The UplinkFast feature is the CSUF feature.)
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

An interface 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 an interface 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 15, "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** *interface-id*	Specify an interface to configure, and enter interface configuration mode.
Step 3	**spanning-tree portfast** [**trunk**]	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. **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. **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. By default, Port Fast is disabled on all interfaces.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show spanning-tree interface** *interface-id* **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 means an invalid configuration, such as the connection of an unauthorized device, and the BPDU guard feature puts the port in the error-disabled state. When this happens, the switch shuts down the entire port on which the violation occurred.

To prevent the port from shutting down, you can use the **errdisable detect cause bpduguard shutdown vlan** global configuration command to shut down just the offending VLAN on the port where the violation occurred.

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 also can 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 it 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** *interface-id*	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 interfaces, it prevents interfaces that are in a Port Fast-operational state from sending or receiving BPDUs. The interfaces 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 interfaces do not receive BPDUs. If a BPDU is received on a Port Fast-enabled interface, the interface loses its Port Fast-operational status, and BPDU filtering is disabled.

Caution Configure Port Fast only on interfaces 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 bpdufilter enable** interface configuration command to enable BPDU filtering on any interface without also enabling the Port Fast feature. This command prevents the interface 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 bpdufilter default**	Globally enable BPDU filtering. By default, BPDU filtering is disabled.
Step 3	**interface** *interface-id*	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 bpdufilter default** global configuration command.

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

Enabling UplinkFast for Use with Redundant Links

UplinkFast cannot be enabled on VLANs that have been configured with a 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 or switch stack. You cannot configure UplinkFast on an individual VLAN.

You can configure the UplinkFast or 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 UplinkFast and CSUF. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**spanning-tree uplinkfast** [**max-update-rate** *pkts-per-second*]	Enable UplinkFast. (Optional) For *pkts-per-second*, 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. When you enter this command, CSUF also is enabled on all nonstack port interfaces.
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 reduce the chance that a 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

When you enable or disable the UplinkFast feature by using the **spanning-tree uplinkfast** global configuration command, CSUF is automatically globally enabled or disabled on nonstack port interfaces.

For more information, see the "Enabling UplinkFast for Use with Redundant Links" section on page 20-16.

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 configure 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** *interface-id*	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.
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 interfaces 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**	Verify which interfaces 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 20-2:

Table 20-2 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 *interface-id*	Displays spanning-tree information for the specified interface.
show spanning-tree mst interface *interface-id*	Displays MST information for the specified interface.
show spanning-tree summary [**totals**]	Displays a summary of interface 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.