



Configuring Spanning-Tree PortFast, UplinkFast, and BackboneFast

This chapter describes how to configure the PortFast, UplinkFast, and BackboneFast spanning-tree enhancements.



Note For information on configuring spanning tree, see Chapter 8, “Configuring Spanning Tree.”



Note For complete syntax and usage information for the commands used in this chapter, refer to the *Command Reference* publication for your switch.

This chapter consists of these sections:

- Understanding How PortFast Works, page 9-1
- Configuring PortFast, page 9-2
- Understanding How PortFast BPDU Guard Works, page 9-3
- Configuring PortFast BPDU Guard, page 9-3
- Understanding How UplinkFast Works, page 9-6
- Configuring UplinkFast, page 9-7
- Understanding How BackboneFast Works, page 9-9
- Configuring BackboneFast, page 9-11

Understanding How PortFast Works

Spanning-tree PortFast causes a port to enter the spanning-tree forwarding state immediately, bypassing the listening and learning states. You can use PortFast on switch ports connected to a single workstation or server to allow those devices to connect to the network immediately, instead of waiting for the port to transition from the listening and learning states to the forwarding state.

**Caution**

PortFast should be used *only* when connecting a single end station to a switch port. If you enable PortFast on a port connected to another networking device, such as a switch, you can create network loops.

When the switch powers up, or when a device is connected to a port, the port normally enters the spanning-tree listening state. When the forward delay timer expires, the port enters the learning state. When the forward delay timer expires a second time, the port is transitioned to the forwarding or blocking state.

When you enable PortFast on a port, the port is immediately and permanently transitioned to the spanning-tree forwarding state.

Configuring PortFast

These sections describe how to configure spanning-tree PortFast on the switch:

- Enabling Spanning-Tree PortFast, page 9-2
- Disabling Spanning-Tree PortFast, page 9-3

Enabling Spanning-Tree PortFast

**Caution**

PortFast should be used *only* when connecting a single end station to a switch port. If you enable PortFast on a port connected to another networking device, such as a switch, you can create network loops.

To enable PortFast on a switch port, perform this task in privileged mode:

	Task	Command
Step 1	Enable PortFast on a switch port connected to a single workstation or server.	set spantree portfast mod_num/port_num enable
Step 2	Verify the PortFast setting.	show spantree mod_num/port_num

This example shows how to enable PortFast on a port and verify the configuration (the PortFast status is shown in the “Fast-Start” column):

```

Console> (enable) set spantree portfast 4/1 enable
Warning: Spantree port fast start should only be enabled on ports connected
to a single host. Connecting hubs, concentrators, switches, bridges, etc. to
a fast start port can cause temporary spanning tree loops. Use with caution.
Spantree port 4/1 fast start enabled.
Console> (enable) show spantree 4/1
Port      Vlan  Port-State  Cost  Priority  Fast-Start  Group-method
-----
4/1      1     blocking    19    20       enabled
4/1     100   forwarding   10    20       enabled
4/1     521   blocking    19    20       enabled
4/1     522   blocking    19    20       enabled
4/1     523   blocking    19    20       enabled

```

```

4/1      524   blocking      19      20   enabled
4/1      1003  not-connected  19      20   enabled
4/1      1005  not-connected  19      4    enabled
Console> (enable)

```

Disabling Spanning-Tree PortFast

To disable PortFast on a switch port, perform this task in privileged mode:

	Task	Command
Step 1	Disable PortFast on a switch port.	set spantree portfast <i>mod_num/port_num</i> disable
Step 2	Verify the PortFast setting.	show spantree <i>mod_num/port_num</i>

This example shows how to disable PortFast on a port:

```

Console> (enable) set spantree portfast 4/1 disable
Spantree port 4/1 fast start disabled.
Console> (enable)

```

Understanding How PortFast BPDU Guard Works

To prevent loops from occurring in a network, the spanning-tree PortFast mode is supported only on nontrunking access ports because these ports typically do not transmit or receive BPDUs. The most secure implementation of PortFast is to enable it only on ports that connect end stations to switches. Because PortFast can be enabled on nontrunking ports connecting two switches, spanning-tree loops can occur because BPDUs are still being transmitted and received on those ports.

The PortFast BPDU Guard feature provides a method for preventing loops by moving a nontrunking port into an ErrDisable state when a BPDU is received on that port. When the BPDU Guard feature is enabled on the switch, spanning tree shuts down PortFast-configured interfaces that receive BPDUs, instead of putting them into the spanning-tree blocking state. In a valid configuration, PortFast-configured interfaces do not receive BPDUs. If a PortFast-configured interface receives a BPDU, an invalid configuration exists, such as connection of an unauthorized device. The BPDU Guard feature provides a secure response to invalid configurations because the administrator must manually put the interface back in service.



Note

When enabled on the switch, spanning tree applies the BPDU Guard feature to all PortFast-configured interfaces.

Configuring PortFast BPDU Guard

These sections describe how to configure spanning-tree PortFast BPDU Guard on the switch:

- Enabling PortFast BPDU Guard, page 9-4
- Disabling PortFast BPDU Guard, page 9-5

Enabling PortFast BPDU Guard



Note

The PortFast feature is configured on an individual port and the PortFast BPDU Guard option is enabled globally. When PortFast is disabled on a port, PortFast BPDU Guard becomes inactive.

To enable PortFast BPDU Guard on a nontrunking switch port, perform this task in privileged mode:

	Task	Command
Step 1	Enable PortFast BPDU Guard on the switch.	set spantree portfast bpdu-guard enable
Step 2	Verify the PortFast BPDU Guard setting.	show spantree summary

This example shows how to enable PortFast BPDU Guard and verify the configuration:

```
console > (enable) set spantree portfast bpdu-guard enable
Spantree portfast bpdu-guard enabled on this switch.
Console> (enable) show spantree summary
Root switch for vlans: none.
→ Portfast bpdu-guard enabled for bridge.
Uplinkfast disabled for bridge.
Backbonefast disabled for bridge.
```

Vlan	Blocking	Listening	Learning	Forwarding	STP Active
1	0	0	0	4	4
2	0	0	0	4	4
3	0	0	0	4	4
4	0	0	0	4	4
5	0	0	0	4	4
6	0	0	0	4	4
10	0	0	0	4	4
20	0	0	0	4	4
50	0	0	0	4	4
100	0	0	0	4	4
152	0	0	0	4	4
200	0	0	0	5	5
300	0	0	0	4	4
400	0	0	0	4	4
500	0	0	0	4	4
521	0	0	0	4	4
524	0	0	0	4	4
570	0	0	0	4	4
801	0	0	0	0	0
802	0	0	0	0	0
850	0	0	0	4	4
917	0	0	0	4	4
999	0	0	0	4	4
1003	0	0	0	0	0
1005	0	0	0	0	0

	Blocking	Listening	Learning	Forwarding	STP Active

Total	0	0	0	85	85

```
Console> (enable)
```

Disabling PortFast BPDU Guard

To disable PortFast BPDU Guard, perform this task in privileged mode:

	Task	Command
Step 1	Disable PortFast BPDU Guard.	set spantree portfast bpdu-guard disable
Step 2	Verify the PortFast BPDU Guard setting.	show spantree summary

This example shows how to disable spanning-tree PortFast BPDU Guard on the switch and verify the configuration:

```
console > (enable) set spantree portfast bpdu-guard disable
Spantree portfast bpdu-guard disabled on this switch.
Console> (enable) show spantree summary
Summary of connected spanning tree ports by vlan
```

- Portfast bpdu-guard disabled for bridge.
- Uplinkfast disabled for bridge.
- Backbonefast disabled for bridge.

```
Vlan  Blocking  Listening  Learning  Forwarding  STP Active
-----
  1          0          0          0           4           4
  2          0          0          0           4           4
  3          0          0          0           4           4
  4          0          0          0           4           4
  5          0          0          0           4           4
  6          0          0          0           4           4
 10          0          0          0           4           4
 20          0          0          0           4           4
 50          0          0          0           4           4
100          0          0          0           4           4
152          0          0          0           4           4
200          0          0          0           5           5
300          0          0          0           4           4
400          0          0          0           4           4
500          0          0          0           4           4
521          0          0          0           4           4
524          0          0          0           4           4
570          0          0          0           4           4
801          0          0          0           0           0
802          0          0          0           0           0
850          0          0          0           4           4
917          0          0          0           4           4
999          0          0          0           4           4
1003         0          0          0           0           0
1005         0          0          0           0           0
```

```
          Blocking  Listening  Learning  Forwarding  STP Active
-----
Total          0          0          0           85          85
Console> (enable)
```

Understanding How UplinkFast Works

UplinkFast provides fast convergence in the network access layer after a spanning-tree topology change using uplink groups. An uplink group is a set of ports (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 (not including self-looped ports). The uplink group provides an alternate path in case the currently forwarding link fails.

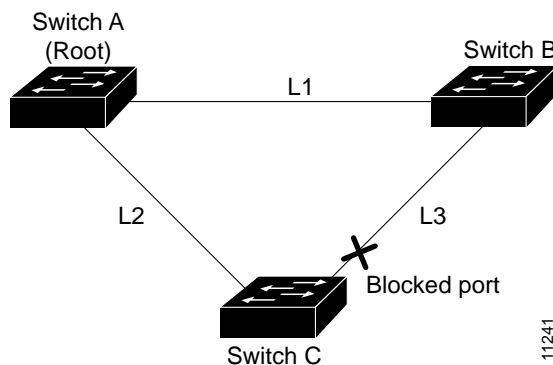


Note

UplinkFast is most useful in wiring-closet switches with a limited number of active VLANs. This enhancement might not be useful for other types of applications and should not be enabled on backbone or distribution layer switches.

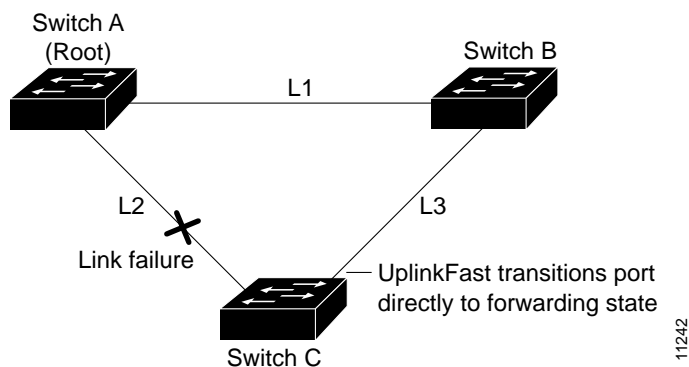
Figure 9-1 shows an example UplinkFast network topology. Switch A, the root switch, is connected directly to Switch B over link L1 and to Switch C over link L2. The port on Switch C that is connected to Switch B over link L3 is in blocking state.

Figure 9-1 UplinkFast Example Before Direct Link Failure



If Switch C detects a link failure on the currently active link L2 (a *direct* link failure), UplinkFast unblocks the blocked port on Switch C and transitions it to the forwarding state immediately, without transitioning the port through the listening and learning states (as shown in Figure 9-2). This switchover takes approximately one to five seconds.

Figure 9-2 UplinkFast Example After Direct Link Failure



As soon as the switch transitions the alternate port to the forwarding state, the switch begins transmitting dummy multicast frames on that port, one for each entry in the local EARL table (except those entries associated with the failed root port). By default, approximately 15 dummy multicast frames are transmitted per 100 milliseconds.

Each dummy multicast frame uses the station address in the EARL table entry as its source MAC address and a dummy multicast address (01-00-0C-CD-CD-CD) as the destination MAC address.

Switches receiving these dummy multicast frames immediately update their EARL table entries for each source MAC address to use the new port, allowing the switches to begin using the new path almost immediately.

If connectivity on the original root port is restored, the switch waits for a period equal to twice the forward delay time plus 5 seconds before transitioning the port to the forwarding state in order to allow the neighbor port time to transition through the listening and learning states to the forwarding state.

Configuring UplinkFast

These sections describe how to configure the UplinkFast feature on the switch:

- Enabling UplinkFast, page 9-7
- Disabling UplinkFast, page 9-8

Enabling UplinkFast

When you enable spanning-tree UplinkFast on the switch, UplinkFast processing is enabled and the spanning-tree bridge priority for all VLANs is set to 49152, making it unlikely that the switch will become the root switch. In addition, the spanning-tree port cost and port-VLAN cost of all ports on the switch is increased by 3000.

The *station_update_rate* value in the UplinkFast command represents the number of dummy multicast packets transmitted per 100 milliseconds (the default is 15 packets per 100 milliseconds) in the event of a direct link failure.

Use the **all-protocols on** keywords on switches that have UplinkFast enabled but do not have protocol filtering enabled, and that are connected to upstream switches in the network that have protocol filtering enabled. The **all-protocols on** keywords cause the switch to generate multicasts for each protocol-filtering group.

On switches with both UplinkFast and protocol filtering enabled, or if no other switches have protocol filtering enabled, you do not need to use the **all-protocols on** keywords.



Note

When you enable spanning-tree UplinkFast, it affects all VLANs on the switch. You cannot configure UplinkFast on a per-VLAN basis.

To enable UplinkFast, perform this task in privileged mode:

	Task	Command
Step 1	Enable UplinkFast on the switch.	set spantree uplinkfast enable [<i>rate station_update_rate</i>] [all-protocols { off on }]
Step 2	Verify that UplinkFast is enabled.	show spantree uplinkfast

This example shows how to enable UplinkFast with a station-update rate of 40 packets per 100 milliseconds and how to verify that UplinkFast is enabled:

```

Console> (enable) set spantree uplinkfast enable
VLANs 1-1005 bridge priority set to 49152.
The port cost and portvlancost of all ports set to above 3000.
Station update rate set to 15 packets/100ms.
uplinkfast all-protocols field set to off.
uplinkfast enabled for bridge.
Console> (enable) show spantree uplinkfast
Station update rate set to 15 packets/100ms.
uplinkfast all-protocols field set to off.
VLAN          port list
-----
1              1/1 (fwd) , 1/2
100            1/2 (fwd)
521            1/1 (fwd) , 1/2
522            1/1 (fwd) , 1/2
523            1/1 (fwd) , 1/2
524            1/1 (fwd) , 1/2
Console> (enable)

```

Disabling UplinkFast

To disable UplinkFast and restore the default spanning-tree bridge priority, port cost, and port-VLAN cost values to their default values, enter the **clear spantree uplinkfast** command.



Caution

Use caution when entering the **clear spantree uplinkfast** command. This command restores the default port-VLAN costs on all ports. If you have configured per-VLAN load sharing on redundant trunk links, the load-sharing configuration could be affected by this command.

You can disable only spanning-tree UplinkFast processing on the switch using the **set spantree uplinkfast disable** command. This command does not affect the bridge priority, port cost, and port-VLAN cost values on the switch.



Note

When you disable spanning-tree UplinkFast, it affects all VLANs on the switch. You cannot disable UplinkFast on a per-VLAN basis.

To disable UplinkFast on the switch, perform this task in privileged mode:

	Task	Command
Step 1	(Optional) Disable UplinkFast processing on the switch and restore the default bridge priority, port cost, and port-VLAN cost values.	clear spantree uplinkfast
Step 2	(Optional) Disable UplinkFast processing on the switch without affecting the bridge priority, port cost, and port-VLAN cost values.	set spantree uplinkfast disable
Step 3	Verify that UplinkFast is enabled.	show spantree uplinkfast

This example shows how to disable UplinkFast on the switch and restore the default bridge priority, port cost, and port-VLAN cost values:

```
Console> (enable) clear spantree uplinkfast
This command will cause all portcosts, portvlancosts, and the
bridge priority on all vlans to be set to default.
Do you want to continue (y/n) [n]? y
VLANs 1-1005 bridge priority set to 32768.
The port cost of all bridge ports set to default value.
The portvlancost of all bridge ports set to default value.
uplinkfast all-protocols field set to off.
uplinkfast disabled for bridge.
Console> (enable) show spantree uplinkfast
uplinkfast disabled for bridge.
Console> (enable)
```

Understanding How BackboneFast Works

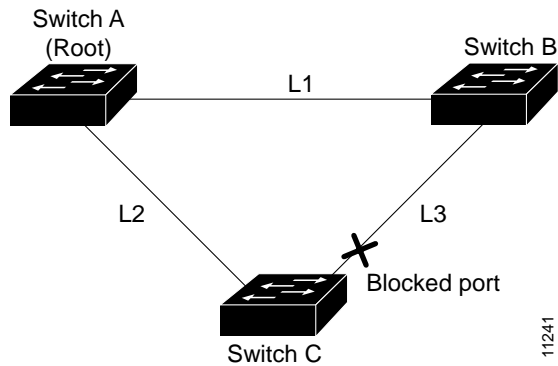
BackboneFast provides fast convergence in the network backbone after a spanning-tree topology change occurs. A switch detects an indirect link failure (the failure of a link to which the switch is not directly connected) when the switch receives inferior BPDUs from its designated bridge on its root port or blocked ports. These inferior BPDUs indicate that the designated bridge has lost its connection to the root bridge. An inferior BPDU identifies a single switch as both the root bridge and the designated bridge. Under normal spanning-tree rules, the switch ignores inferior BPDUs for the configured maximum aging time (specified by the **set spantree maxage** command).

The switch tries to determine if it has an alternate path to the root bridge. 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 bridge. If the inferior BPDU arrives on the root port, all blocked ports become alternate paths to the root bridge. 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 bridge, causes the maximum aging time on the root to expire, and becomes the root switch according to normal spanning-tree rules.

If the switch has alternate paths to the root bridge, it transmits Root Link Query PDUs out all alternate paths to the root bridge. If the switch determines that it still has an alternate path to the root, it causes the maximum aging time on the ports on which it received the inferior BPDU to expire. If all the alternate paths to the root bridge indicate that the switch has lost connectivity to the root bridge, the switch causes the maximum aging times on the ports on which it received an inferior BPDU to expire. If one or more alternate paths can still connect to the root bridge, the switch makes all ports on which it received an inferior BPDU its designated ports and moves them out of the blocking state (if they were in blocking state), through the listening and learning states, and into the forwarding state.

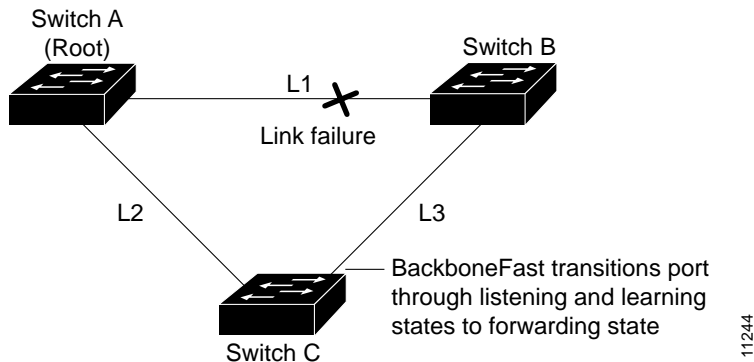
Figure 9-3 shows an example BackboneFast network topology. Switch A, the root switch, connects directly to Switch B over link L1 and to Switch C over link L2. The port on Switch C that connects directly to Switch B over link L3 is in the blocking state.

Figure 9-3 BackboneFast Example Before Indirect Link Failure



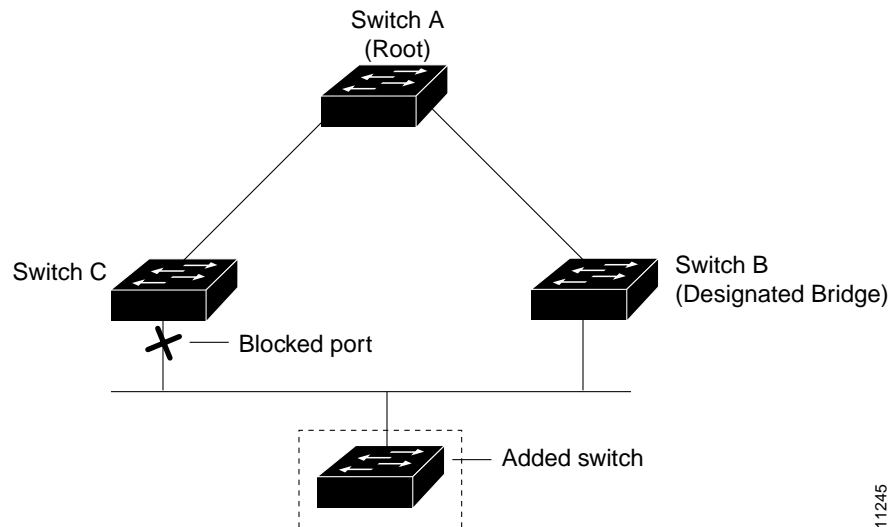
If link L1 fails, Switch C detects this failure as an indirect failure, since it is not connected directly to link L1. Switch B no longer has a path to the root switch. BackboneFast allows the blocked port on Switch C to move immediately to the listening state without waiting for the maximum aging time for the port to expire. BackboneFast then transitions the port on Switch C to the forwarding state, providing a path from Switch B to Switch A. This switchover takes approximately 30 seconds. Figure 9-4 shows how BackboneFast reconfigures the topology to account for the failure of link L1.

Figure 9-4 BackboneFast Example After Indirect Link Failure



If a new switch is introduced into a shared-medium topology, BackboneFast is not activated. Figure 9-5 shows a shared-medium topology in which a new switch is added. The new switch begins sending inferior BPDUs, which indicate that 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 9-5 Adding a Switch in a Shared-Medium Topology



11245

Configuring BackboneFast

These sections describe how to configure BackboneFast:

- Enabling BackboneFast, page 9-11
- Displaying BackboneFast Statistics, page 9-12
- Disabling BackboneFast, page 9-12

Enabling BackboneFast



Note

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

To enable BackboneFast on the switch, perform this task in privileged mode:

	Task	Command
Step 1	Enable BackboneFast on the switch.	set spantree backbonefast enable
Step 2	Verify that BackboneFast is enabled.	show spantree backbonefast

This example shows how to enable BackboneFast on the switch and how to verify the configuration:

```
Console> (enable) set spantree backbonefast enable
Backbonefast enabled for all VLANs
Console> (enable) show spantree backbonefast
Backbonefast is enabled.
Console> (enable)
```

Displaying BackboneFast Statistics

To display BackboneFast statistics, perform this task in privileged mode:

Task	Command
Display BackboneFast statistics.	show spantree summary

This example shows how to display BackboneFast statistics:

```

Console> (enable) show spantree summary
Summary of connected spanning tree ports by vlan

Uplinkfast disabled for bridge.
Backbonefast enabled for bridge.

Vlan  Blocking  Listening  Learning  Forwarding  STP Active
-----
   1      0         0         0           1           1

          Blocking  Listening  Learning  Forwarding  STP Active
          -----
Total      0         0         0           1           1

BackboneFast statistics
-----
Number of inferior BPDUs received (all VLANs) : 0
Number of RLQ req PDUs received (all VLANs)   : 0
Number of RLQ res PDUs received (all VLANs)   : 0
Number of RLQ req PDUs transmitted (all VLANs): 0
Number of RLQ res PDUs transmitted (all VLANs): 0
Console> (enable)

```

Disabling BackboneFast

To disable BackboneFast on the switch, perform this task in privileged mode:

	Task	Command
Step 1	Disable BackboneFast on the switch.	set spantree backbonefast disable
Step 2	Verify that BackboneFast is disabled.	show spantree backbonefast

This example shows how to disable BackboneFast on the switch and how to verify the configuration:

```

Console> (enable) set spantree backbonefast disable
Backbonefast enabled for all VLANs
Console> (enable) show spantree backbonefast
Backbonefast is disable.
Console> (enable)

```