



CHAPTER 5

Managing Switch Stacks

This chapter provides the concepts and procedures to manage Catalyst 3750 stacks. See the command reference for command syntax and usage information.

- [Understanding Stacks, page 5-1](#)
- [Configuring the Switch Stack, page 5-18](#)
- [Accessing the CLI of a Specific Member, page 5-23](#)
- [Displaying Stack Information, page 5-24](#)
- [Troubleshooting Stacks, page 5-24](#)

For other switch stack-related information, such as cabling the switches through their StackWise ports and using the LEDs for switch stack status, see the hardware installation guide.

Understanding Stacks

A *switch stack* is a set of up to nine Catalyst 3750 switches connected through their StackWise ports. One of the switches controls the operation of the stack and is called the *stack master*. The stack master and the other switches in the stack are *stack members*. Layer 2 and Layer 3 protocols present the entire switch stack as a single entity to the network.



Note

A switch stack is different from a *switch cluster*. A switch cluster is a set of switches connected through their LAN ports, such as the 10/100/1000 ports. For more information about how switch stacks differ from switch clusters, see the “Planning and Creating Clusters” chapter in the *Getting Started with Cisco Network Assistant* on Cisco.com.

The master is the single point of stack-wide management. From the master, you configure:

- System-level (global) features that apply to all members
- Interface-level features for each member

If the master is running the cryptographic version (that is, supports encryption) of the IP base or IP services image software, the encryption features are available.

Every member is uniquely identified by its own *stack member number*.

All members are eligible masters. If the master becomes unavailable, the remaining members elect a new master from among themselves. One of the factors is the *stack member priority value*. The switch with the highest stack-member priority-value becomes the master.

The system-level features supported on the master are supported on the entire stack.

If a switch in the stack is running the cryptographic version of the IP base or IP services image software, we recommend that this switch be the stack master.

The master contains the saved and running configuration files for the stack. The configuration files include the system-level settings for the stack and the interface-level settings for each member. Each member has a current copy of these files for back-up purposes.

You manage the stack through a single IP address. The IP address is a system-level setting and is not specific to the master or to any other member. You can manage the stack through the same IP address even if you remove the master or any other member from the stack.

You can use these methods to manage stacks:

- Network Assistant (available on Cisco.com)
- Command-line interface (CLI) over a serial connection to the console port of any member
- A network management application through the Simple Network Management Protocol (SNMP)



Note Use SNMP to manage network features across the stack that are defined by supported MIBs. The switch does not support MIBs to manage stacking-specific features such as stack membership and election.

- CiscoWorks network management software

To manage stacks, you should understand:

- These concepts on stack formations:
 - [Stack Membership, page 5-3](#)
 - [Master Election, page 5-4](#)
- These concepts on stack and member configurations:
 - [Stack MAC Address and Router MAC Address, page 5-6](#)
 - [Member Numbers, page 5-6](#)
 - [Member Priority Values, page 5-7](#)
 - [Stack Offline Configuration, page 5-7](#)
 - [Hardware Compatibility and SDM Mismatch Mode in Switch Stacks, page 5-9](#)
 - [Stack Software Compatibility Recommendations, page 5-10](#)
 - [Stack Protocol Version Compatibility, page 5-10](#)
 - [Major Version Number Incompatibility Among Switches, page 5-10](#)
 - [Minor Version Number Incompatibility Among Switches, page 5-10](#)
 - [Incompatible Software and Member Image Upgrades, page 5-14](#)
 - [Stack Configuration Files, page 5-14](#)
 - [Additional Considerations for System-Wide Configuration on Switch Stacks, page 5-15](#)

- [Stack Management Connectivity, page 5-15](#)
- [Stack Configuration Scenarios, page 5-16](#)

Stack Membership

A *standalone switch* is a stack with one member that is also the master. You can connect one standalone switch to another ([Figure 5-1 on page 5-4](#)) to create a stack containing two stack members, with one of them as the master. You can connect standalone switches to an existing stack ([Figure 5-2 on page 5-4](#)) to increase the stack membership.

If you replace a stack member with an identical model, the new switch functions with the same configuration as the replaced switch (assuming that the new switch is using the same member number as the replaced switch). For information about the benefits of provisioning a switch stack, see the “[Stack Offline Configuration](#)” section on [page 5-7](#). For information about replacing a failed switch, see the “[Troubleshooting](#)” chapter in the hardware installation guide.

The operation of the stack continues uninterrupted during membership changes unless you remove the master or you add powered-on standalone switches or stacks.

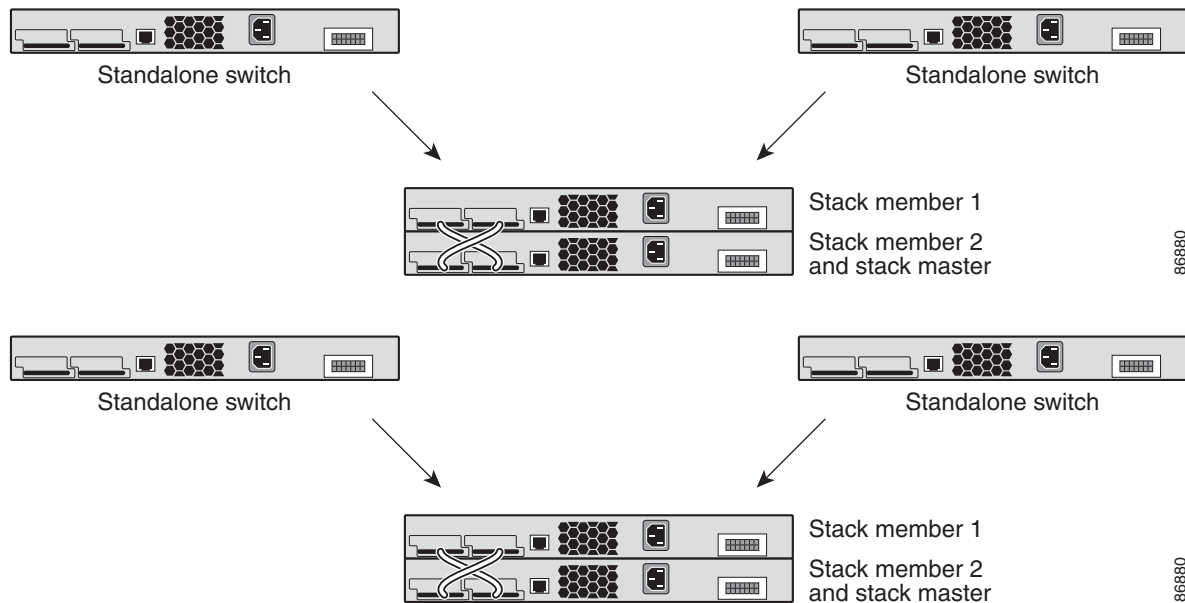


Note

To prevent interrupted stack operations, make sure the switches that you add to or remove from the stack are powered off.

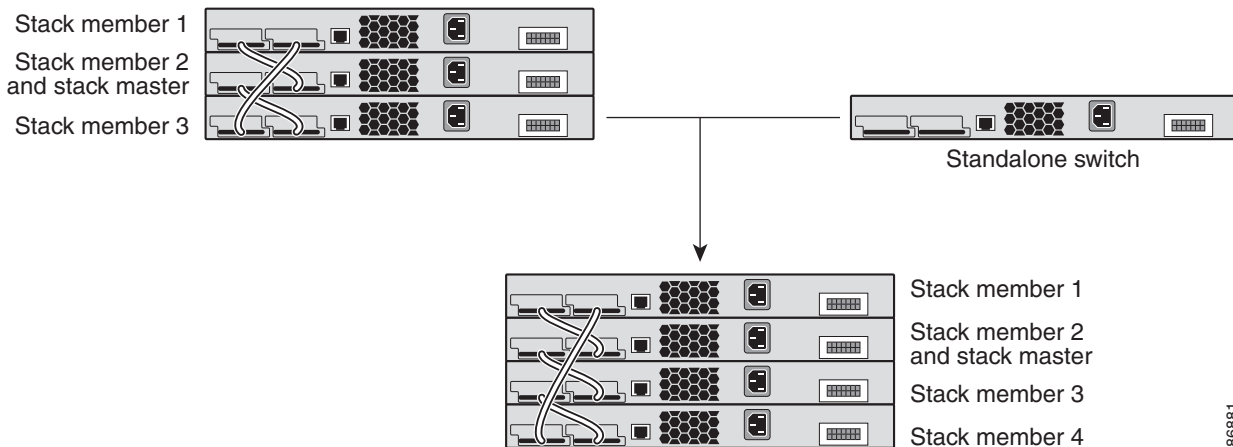
After adding or removing members, make sure that the stack ring is operating at full bandwidth (32 Gb/s). Press the Mode button on a member until the Stack mode LED is on. The last two port LEDs on all switches in the stack should be green. If any one or both of any the last two port LEDs are not green, the stack is not operating at full bandwidth.

- Adding powered-on switches (merging) causes the masters of the merging stacks to elect a master from among themselves. The new master keeps its role and configuration and so do its members. All remaining switches, including the former masters, reload and join the stack as members. They change their member numbers to the lowest available numbers and use the configuration of the new master.
- Removing powered-on members divides (partitions) the stack into two or more switch stacks, each with the same configuration. This can create an IP address configuration conflict in your network. If you want the stacks to remain separate, change the IP address or addresses of the newly created stacks.

Figure 5-1 Creating a Switch Stack from Two Standalone Switches

86880

86880

Figure 5-2 Adding a Standalone Switch to a Switch Stack

86881

For information about cabling and powering switch stacks, see the “Switch Installation” chapter in the hardware installation guide.

Master Election

The stack master is elected based on one of these factors in the order listed:

1. The switch that is currently the stack master.
2. The switch with the highest stack member priority value.



Note

We recommend you assign the highest priority value to the switch that you want to be the master. The switch is then re-elected as master if a re-election occurs.

3. The switch that is not using the default interface-level configuration.
4. The switch with the higher priority switch software version. These switch software versions are listed from highest to lowest priority:
 - Cryptographic IP services image software
 - Noncryptographic IP services image software
 - Cryptographic IP base image software
 - Noncryptographic IP base image software

If two or more switches in the stack use different software images, a switch running the noncryptographic IP base image might be selected as the master. A switch running the cryptographic IP services image takes 10 seconds longer to start than does the switch running the noncryptographic image. The switch running the cryptographic image is excluded from the master election process that lasts 10 seconds. To avoid this problem, upgrade the switch running the IP base image to a software release later than Cisco IOS Release 12.1(11)AX, or manually start the master and wait at least 8 seconds before starting the new member.

5. The switch with the lowest MAC address.

A stack master keeps its role unless one of these events occurs:

- The stack is reset.*
- The master is removed from the stack.
- The master is reset or powered off.
- The master fails.
- The stack membership is increased by adding powered-on standalone switches or switch stacks.*

In the events marked by an asterisk (*), the current stack master *might* be re-elected based on the listed factors.

When you power on or reset an entire stack, some stack members *might not* participate in the master election.

- All members participate in re-elections.
- Members that are powered on within the same 20-second time frame participate in the master election and have a chance to become the master.
- Members that are powered on after the 20-second time frame do not participate in this initial election and only become members.

**Note**

Stack master elections occur over a 10-second time frame on switches running releases earlier than Cisco IOS Release 12.2(20)SE3.

The new master is available after a few seconds. In the meantime, the switch stack uses the forwarding tables in memory to minimize network disruption. The physical interfaces on the other available stack members are not affected while a new stack master is elected and is resetting.

When a new master is elected and the previous stack master becomes available, the previous master *does not* resume its role as stack master.

For all powering considerations that affect stack-master elections, see the “Switch Installation” chapter in the hardware installation guide.

Stack MAC Address and Router MAC Address

The MAC address of the master determines the stack MAC address.

When the stack initializes, the MAC address of the master determines the bridge ID and router MAC address that identify the stack in the network.

If the master changes, the MAC address of the *new* master determines the new bridge ID and router MAC address. However, when the persistent MAC address feature is enabled, there is an approximate 4-minute delay before the stack MAC address changes. During this time period, if the previous master rejoins the stack, the stack continues to use its MAC address as the stack MAC address, even if the switch is now a member and not a master. If the previous master does not rejoin the stack during this period, the stack takes the MAC address of the new stack master as the stack MAC address. See the [“Enabling Persistent MAC Address” section on page 5-19](#) for more information.

Member Numbers

The member number (1 to 9) identifies each member in the stack. The member number also determines the interface-level configuration that a member uses.

A new, out-of-the-box switch (one that has not joined a stack or has not been manually assigned a member number) ships with a default member number of 1. When it joins a stack, its default stack member number changes to the lowest available member number in the stack.

Members in the same stack cannot have the same member number.

- If you manually change the member number by using the **switch** *current-stack-member-number renumber new-stack-member-number* global configuration command, the new number goes into effect after that member resets (or after you use the **reload slot stack-member-number** privileged EXEC command) and only if that number is not already changed.

You can also change the stack member number is by using the SWITCH_NUMBER environment variable.

If the number is being used by another member in the stack, the switch selects the lowest available number in the stack.

If you manually change the member number and no interface-level configuration is associated with that number, that member resets to its default configuration.

You cannot use the **switch** *current-stack-member-number renumber new-stack-member-number* global configuration command on a provisioned switch. If you do, the command is rejected.

- If you move a stack member to a different switch stack, the stack member keeps its number only if the number is not being used by another member in the stack. If it is being used by another member in the stack, the switch selects the lowest available number in the stack.

See the following sections for information about stack member configuration:

- The procedure to change a member number, see the [“Assigning a Member Number” section on page 5-21](#).
- The SWITCH_NUMBER environment variable, see the [“Controlling Environment Variables” section on page 3-22](#).
- Member numbers and configurations, see the [“Stack Configuration Files” section on page 5-14](#).
- Merging stacks, see the [“Stack Membership” section on page 5-3](#).

Member Priority Values

A high priority value for a member increases the chance that it will be elected master and keep its member number. The priority value can be 1 to 15. The default priority value is 1.

**Note**

We recommend that you assign the highest priority value to the switch that you want to be the stack master. The switch is then re-elected as master if a re-election occurs.

The new priority value takes effect immediately but does not affect the current master until the current master or the stack resets.

Stack Offline Configuration

You can use the offline configuration feature to *provision* (to configure) a new switch before it joins the stack. You can configure the member number, the switch type, and the interfaces associated with a switch that is not yet part of the stack. That configuration is the *provisioned configuration*. The switch to be added to the stack and to get this configuration is the *provisioned switch*.

The provisioned configuration is automatically created when a switch is added to a stack that is running Cisco IOS Release 12.2(20)SE or later and when no provisioned configuration exists. You can manually create the provisioned configuration by using the **switch stack-member-number provision type** global configuration command.

When you configure the interfaces for a provisioned switch (for example, as part of a VLAN), the information appears in the stack running configuration whether or not the provisioned switch is part of the stack. The interface for the provisioned switch is not active and does not appear in the display of a specific feature (for example, in the **show vlan** user EXEC command output). Entering the **no shutdown** interface configuration command has no effect.

The startup configuration file ensures that the stack can reload and can use the saved information whether or not the provisioned switch is part of the stack.

Effects of Adding a Provisioned Switch to a Stack

When you add a provisioned switch to the switch stack, the stack applies either the provisioned configuration or the default configuration to it. [Table 5-1](#) lists the events that occur when the switch stack compares the provisioned configuration with the provisioned switch.

Table 5-1 Results of Comparing the Provisioned Configuration with the Provisioned Switch

Scenario		Result
The stack member numbers and the switch types match.	<ol style="list-style-type: none"> 1. If the stack member number of the provisioned switch matches the stack member number in the provisioned configuration on the stack, and 2. If the switch type of the provisioned switch matches the switch type in the provisioned configuration on the stack. 	The switch stack applies the provisioned configuration to the provisioned switch and adds it to the stack.
The stack member numbers match but the switch types do not match.	<ol style="list-style-type: none"> 1. If the stack member number of the provisioned switch matches the stack member number in the provisioned configuration on the stack, but 2. The switch type of the provisioned switch does not match the switch type in the provisioned configuration on the stack. 	<p>The switch stack applies the default configuration to the provisioned switch and adds it to the stack.</p> <p>The provisioned configuration is changed to reflect the new information.</p>
The stack member number is not found in the provisioned configuration.		<p>The switch stack applies the default configuration to the provisioned switch and adds it to the stack.</p> <p>The provisioned configuration is changed to reflect the new information.</p>
The stack member number of the provisioned switch is in conflict with an existing stack member.	<p>The stack master assigns a new stack member number to the provisioned switch.</p> <p>The stack member numbers and the switch types match:</p> <ol style="list-style-type: none"> 1. If the new stack member number of the provisioned switch matches the stack member number in the provisioned configuration on the stack, and 2. If the switch type of the provisioned switch matches the switch type in the provisioned configuration on the stack. 	<p>The switch stack applies the provisioned configuration to the provisioned switch and adds it to the stack.</p> <p>The provisioned configuration is changed to reflect the new information.</p>
	<p>The stack member numbers match, but the switch types do not match:</p> <ol style="list-style-type: none"> 1. If the stack member number of the provisioned switch matches the stack member number in the provisioned configuration on the stack, but 2. The switch type of the provisioned switch does not match the switch type in the provisioned configuration on the stack. 	<p>The switch stack applies the default configuration to the provisioned switch and adds it to the stack.</p> <p>The provisioned configuration is changed to reflect the new information.</p>

Table 5-1 Results of Comparing the Provisioned Configuration with the Provisioned Switch (continued)

Scenario	Result
The stack member number of the provisioned switch is not found in the provisioned configuration.	The switch stack applies the default configuration to the provisioned switch and adds it to the stack.

If you add a provisioned switch that is a different type than specified in the provisioned configuration to a powered-down switch stack and then apply power, the switch stack rejects the (now incorrect) **switch stack-member-number provision type** global configuration command in the startup configuration file. However, during stack initialization, the nondefault interface configuration information in the startup configuration file for the provisioned interfaces (potentially of the wrong type) are executed. Depending on how different the actual switch type is from the previously provisioned switch type, some commands are rejected, and some commands are accepted.

For example, suppose the switch stack is provisioned for a 48-port switch with Power over Ethernet (PoE), the configuration is saved, and the stack is powered down. Then, a 24-port switch without PoE support is connected to the switch stack, and the stack is powered up. In this situation, the configuration for ports 25 through 48 is rejected, and error messages appear on the stack master switch console during initialization. In addition, any configured PoE-related commands that are valid only on PoE-capable interfaces are rejected, even for ports 1 through 24.

**Note**

If the switch stack does not contain a provisioned configuration for a new switch, the switch joins the stack with the default interface configuration. The switch stack then adds to its running configuration a **switch stack-member-number provision type** global configuration command that matches the new switch.

For configuration information, see the [“Provisioning a New Member for a Stack”](#) section on page 5-22.

Effects of Replacing a Provisioned Switch in a Stack

When a provisioned switch in a switch stack fails, is removed from the stack, and is replaced with another switch, the stack applies either the provisioned configuration or the default configuration to it. The events that occur when the switch stack compares the provisioned configuration with the provisioned switch are the same as those described in the [“Effects of Adding a Provisioned Switch to a Stack”](#) section on page 5-8.

Effects of Removing a Provisioned Switch from a Stack

If a switch stack is running Cisco IOS Release 12.2(20)SE or later and you remove a provisioned switch from the switch stack, the configuration associated with the removed stack member remains in the running configuration as provisioned information. To completely remove the configuration, use the **no switch stack-member-number provision** global configuration command.

Hardware Compatibility and SDM Mismatch Mode in Switch Stacks

The Catalyst 3750-12S switch supports desktop and aggregator Switch Database Management (SDM) templates. All other Catalyst 3750 switches support only the desktop SDM templates.

All stack members use the SDM template configured on the stack master. If the stack master is using an aggregator template, only Catalyst 3750-12S switches can be stack members. All other switches attempting to join this switch stack enter SDM-mismatch mode. These switches can join the stack only when the stack master is running a desktop SDM template.

We recommend that your stack master use an aggregator template only if you plan to create a switch stack of Catalyst 3750-12S switches. If you plan to have a switch stack with different Catalyst 3750 switch models, configure the stack master to use one of the desktop templates.

**Note**

Version-mismatch (VM) mode has priority over SDM-mismatch mode. If a VM-mode condition and an SDM-mismatch mode exist, the switch stack attempts to resolve the VM-mode condition first.

You can use the **show switch** privileged EXEC command to see if any stack members are in SDM-mismatch mode.

For more information about SDM templates and SDM-mismatch mode, see [Chapter 8, “Configuring SDM Templates.”](#)

Stack Software Compatibility Recommendations

All stack members must run the same Cisco IOS software version to ensure compatibility in the stack protocol version among the members.

Stack Protocol Version Compatibility

The stack protocol version has a *major* version number and a *minor* version number (for example 1.4, where 1 is the major version number and 4 is the minor version number).

Switches with the same Cisco IOS software version have the same stack protocol version. All features function properly across the stack. These switches with the same software version as the master immediately join the stack.

If an incompatibility exists, a system message describes the cause of the incompatibility on the specific stack members. The master sends the message to all members.

For more information, see the “[Major Version Number Incompatibility Among Switches](#)” procedure on page 5-10 and the “[Minor Version Number Incompatibility Among Switches](#)” procedure on page 5-10.

Major Version Number Incompatibility Among Switches

Switches with different Cisco IOS software versions likely have different stack protocol versions. Switches with different major version numbers are incompatible and cannot exist in the same stack.

Minor Version Number Incompatibility Among Switches

Switches with the same major version number but with a different minor version number as the master are considered partially compatible. When connected to a stack, a partially compatible switch enters version-mismatch mode and cannot join the stack as a fully functioning member. The software detects

the mismatched software and tries to upgrade (or downgrade) the switch in version-mismatch mode with the stack image or with a tar file image from the stack flash memory. The software uses the automatic upgrade (auto-upgrade) and the automatic advise (auto-advise) features.

The port LEDs on switches in version-mismatch mode will also stay off. Pressing the Mode button does not change the LED mode.

**Note**

Auto-advise and auto-copy identify which images are running by examining the info file and by searching the directory structure on the switch stack. If you download your image by using the **copy tftp:** command instead of by using the **archive download-sw** privileged EXEC command, the correct directory structure is not properly created. For more information about the info file, see the [“tar File Format of Images on a Server or Cisco.com” section on page B-25](#).

Understanding Auto-Upgrade and Auto-Advise

When the software detects mismatched software and tries to upgrade the switch in version-mismatch mode, two software processes are involved: automatic upgrade and automatic advise.

- The automatic upgrade (auto-upgrade) process includes an auto-copy process and an auto-extract process. By default, auto-upgrade is enabled (the **boot auto-copy-sw** global configuration command is enabled). You can disable auto-upgrade by using the **no boot auto-copy-sw** global configuration command on the master. You can check the status of auto-upgrade by using the **show boot** privileged EXEC command and by checking the *Auto upgrade* line in the display.
 - Auto-copy automatically copies the software image running on any member to the switch in version-mismatch mode to upgrade (auto-upgrade) it. Auto-copy occurs if auto-upgrade is enabled, if there is enough flash memory in the switch in version-mismatch mode, and if the software image running on the stack is suitable for the switch in version-mismatch mode.

**Note**

A switch in version-mismatch mode might not run all released software. For example, new switch hardware is not recognized in earlier versions of software.

- Automatic extraction (auto-extract) occurs when the auto-upgrade process cannot find the appropriate software in the stack to copy to the switch in version-mismatch mode. In that case, the auto-extract process searches all switches in the stack, whether they are in version-mismatch mode or not, for the tar file needed to upgrade the switch stack or the switch in version-mismatch mode. The tar file can be in any flash file system in the stack (including the switch in version-mismatch mode). If a tar file suitable for the switch in version-mismatch mode is found, the process extracts the file and automatically upgrades that switch.

The auto-upgrade (auto-copy and auto-extract) processes start a few minutes after the mismatched software is detected.

When the auto-upgrade process is complete, the switch that was in version-mismatch mode reloads and joins the stack as a fully functioning member. If you have both StackWise cables connected during the reload, network downtime does not occur because the stack operates on two rings.

**Note**

Auto-upgrade does not upgrade switches that are loaded with images of different packaging levels. For example, you cannot use auto-upgrade to upgrade a switch running an IP base image to an IP services image. However, auto-upgrade does support upgrades between cryptographic and non-cryptographic images of the same packaging level.

- Automatic advise (auto-advise)—when the auto-upgrade process cannot find appropriate version-mismatch member software to copy to the switch in version-mismatch mode, the auto-advise process tells you the command (**archive copy-sw** or **archive download-sw** privileged EXEC command) and the image name (tar filename) needed to manually upgrade the switch stack or the switch in version-mismatch mode. The recommended image can be the running stack image or a tar file in any flash file system in the stack (including the switch in version-mismatch mode). If an appropriate image is not found in the stack flash file systems, the auto-advise process tells you to install new software on the stack. Auto-advise cannot be disabled, and there is no command to check its status.

The auto-advise software does *not* give suggestions when the stack software and the software of the switch in version-mismatch mode do not contain the same feature sets. For example, if the switch stack is running the IP base image and you add a switch that is running the IP services image, the auto-advise software does not provide a recommendation. The same events occur when cryptographic and noncryptographic images are running.

You can use the **archive-download-sw /allow-feature-upgrade** privileged EXEC command to allow installing an image with a different feature set.

Auto-Upgrade and Auto-Advise Example Messages

When you add a switch that has a different minor version number to the stack, the software displays messages in sequence (assuming that there are no other system messages generated by the switch).

This example shows that the stack detected a new switch that is running a different minor version number than the stack. Auto-copy launches, finds suitable software to copy from a member to the switch in version-mismatch mode, upgrades the switch in version-mismatch mode, and then reloads it:

```
*Mar 11 20:31:19.247:%STACKMGR-6-STACK_LINK_CHANGE:Stack Port 2 Switch 2 has changed to
state UP
*Mar 11 20:31:23.232:%STACKMGR-6-SWITCH_ADDED_VM:Switch 1 has been ADDED to the stack
(VERSION_MISMATCH)
*Mar 11 20:31:23.291:%STACKMGR-6-SWITCH_ADDED_VM:Switch 1 has been ADDED to the stack
(VERSION_MISMATCH) (Stack_1-3)
*Mar 11 20:33:23.248:%IMAGEMGR-6-AUTO_COPY_SW_INITIATED:Auto-copy-software process
initiated for switch number(s) 1
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:Searching for stack member to act
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:as software donor...
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:Found donor (system #2) for
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:member(s) 1
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:System software to be uploaded:
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:System Type:          0x00000000
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:archiving c3750-ip-services-mz.122-25.SEB
(directory)
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:archiving
c3750-ip-services-mz.122-25.SEB/c3750-ip-services-mz.122-25.SEB.bin (4945851 bytes)
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:archiving
c3750-ip-services-mz.122-25.SEB/info (450 bytes)
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:archiving info (104 bytes)
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:examining image...
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:extracting info (104 bytes)
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:extracting
c3750-ip-services-mz.122-25.SEB/info (450 bytes)
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:extracting info (104 bytes)
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:Stacking Version Number:1.4
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:System Type:          0x00000000
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW: Ios Image File Size: 0x004BA200
```

```

*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW: Total Image File Size:0x00818A00
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW: Minimum Dram required:0x08000000
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW: Image Suffix:universalk9-122-53.SE
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW: Image
Directory:c3750-ipservices-mz.122-25.SEB
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW: Image Name:c3750-ipservices-mz.122-25.SEB
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW: Image
Feature:IP|LAYER_3|PLUS|MIN_DRAM_MEG=128
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:Old image for switch
1:flash1:c3750-ipservices-mz.122-25.SEB
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW: Old image will be deleted after download.
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:Extracting images from archive into flash on
switch 1...
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:c3750-i5-mz.122-0.0.313.SE (directory)
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:extracting
c3750-i5-mz.122-0.0.313.SE/c3750-ipservices-mz.122-25.SEB (4945851 bytes)
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:extracting
c3750-ipservices-mz.122-25.SEB/info (450 bytes)
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:extracting info (104 bytes)
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:Installing
(renaming):`flash1:update/c3750-i5-mz.122-0.0.313.SE' ->
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:
`flash1:c3750-ipservices-mz.122-25.SEB'
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:New software image installed in
flash1:c3750-i5-mz.122-0.0.313.SE
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:Removing old
image:flash1:c3750-i5-mz.121-19.EA1
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:All software images installed.
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:Requested system reload in progress...
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:Software successfully copied to
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:system(s) 1
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:Done copying software
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:Reloading system(s) 1

```

This example shows that the stack detected a new switch that is running a different minor version number than the stack. Auto-copy launches but cannot find software in the stack to copy to the switch in version-mismatch mode to make it compatible with the stack. The auto-advise process launches and recommends that you download a tar file from the network to the switch in version-mismatch mode:

```

*Mar 1 00:01:11.319:%STACKMGR-6-STACK_LINK_CHANGE:Stack Port 2 Switch 2 has changed to
state UP
*Mar 1 00:01:15.547:%STACKMGR-6-SWITCH_ADDED_VM:Switch 1 has been ADDED to the stack
(VERSION_MISMATCH)
stack_2#
*Mar 1 00:03:15.554:%IMAGEMGR-6-AUTO_COPY_SW_INITIATED:Auto-copy-software process
initiated for switch number(s) 1
*Mar 1 00:03:15.554:%IMAGEMGR-6-AUTO_COPY_SW:
*Mar 1 00:03:15.554:%IMAGEMGR-6-AUTO_COPY_SW:Searching for stack member to act
*Mar 1 00:03:15.554:%IMAGEMGR-6-AUTO_COPY_SW:as software donor...
*Mar 1 00:03:15.554:%IMAGEMGR-6-AUTO_COPY_SW:Software was not copied
*Mar 1 00:03:15.562:%IMAGEMGR-6-AUTO_ADVISE_SW_INITIATED:Auto-advise-software process
initiated for switch number(s) 1
*Mar 1 00:04:22.537:%IMAGEMGR-6-AUTO_ADVISE_SW:
*Mar 1 00:04:22.537:%IMAGEMGR-6-AUTO_ADVISE_SW:
*Mar 1 00:04:22.537:%IMAGEMGR-6-AUTO_ADVISE_SW:Systems with incompatible software
*Mar 1 00:04:22.537:%IMAGEMGR-6-AUTO_ADVISE_SW:have been added to the stack. The
*Mar 1 00:04:22.537:%IMAGEMGR-6-AUTO_ADVISE_SW:storage devices on all of the stack

```

```
*Mar 1 00:04:22.537:%IMAGEGR-6-AUTO_ADVISE_SW:members have been scanned, and it has
*Mar 1 00:04:22.537:%IMAGEGR-6-AUTO_ADVISE_SW:been determined that the stack can be
*Mar 1 00:04:22.537:%IMAGEGR-6-AUTO_ADVISE_SW:repaired by issuing the following
*Mar 1 00:04:22.537:%IMAGEGR-6-AUTO_ADVISE_SW:command(s):
*Mar 1 00:04:22.537:%IMAGEGR-6-AUTO_ADVISE_SW:
*Mar 1 00:04:22.537:%IMAGEGR-6-AUTO_ADVISE_SW: archive download-sw /force-reload
/overwrite /dest 1 flash1:c3750-ipservices-mz.122-25.SEB.tar
*Mar 1 00:04:22.537:%IMAGEGR-6-AUTO_ADVISE_SW:
```

For information about using the **archive download-sw** privileged EXEC command, see the [“Working with Software Images”](#) section on page B-24.

Incompatible Software and Member Image Upgrades

You can upgrade a switch that has an incompatible software image by using the **archive copy-sw** privileged EXEC command to copy the software image from an existing member. That switch automatically reloads with the new image and joins the stack as a fully functioning member.

For more information, see the [“Copying an Image File from One Stack Member to Another”](#) section on page B-38.

Stack Configuration Files

The master has the saved and running configuration files for the stack. All members periodically receive synchronized copies of the configuration files from the master. If the master becomes unavailable, any member assuming the role of master has the latest configuration files.

- System-level (global) configuration settings—such as IP, STP, VLAN, and SNMP settings—that apply to all members
- Member interface-specific configuration settings, which are specific for each member

A new, out-of-box switch joining a stack uses the system-level settings of that stack. If a switch is moved to a different stack, it loses its saved configuration file and uses the system-level configuration of the new stack.

The interface-specific configuration of each member is associated with its member number. A stack member keeps its number unless it is manually changed or it is already used by another member in the same stack.

- If an interface-specific configuration does not exist for that member number, the member uses its default interface-specific configuration.
- If an interface-specific configuration exists for that member number, the member uses the interface-specific configuration associated with that member number.

If you replace a failed member with an identical model, the replacement member automatically uses the same interface-specific configuration. You do not need to reconfigure the interface settings. The replacement switch must have the same member number as the failed switch.

You back up and restore the stack configuration in the same way as you do for a standalone switch configuration.

For information about

- The benefits of provisioning a switch stack, see the [“Stack Offline Configuration”](#) section on page 5-7.
- File systems and configuration files, see [Appendix B, “Working with the Cisco IOS File System, Configuration Files, and Software Images.”](#)

Additional Considerations for System-Wide Configuration on Switch Stacks

- “Planning and Creating Clusters” chapter in the *Getting Started with Cisco Network Assistant*, available on Cisco.com
- [“MAC Addresses and Switch Stacks”](#) section on page 7-14
- [“Setting the SDM Template”](#) section on page 8-6
- [“802.1x Authentication and Switch Stacks”](#) section on page 10-11
- [“VTP and Switch Stacks”](#) section on page 14-7
- [“Private VLANs and Switch Stacks”](#) section on page 16-6
- [“Spanning Tree and Switch Stacks”](#) section on page 18-12
- [“MSTP and Switch Stacks”](#) section on page 19-8
- [“DHCP Snooping and Switch Stacks”](#) section on page 22-7
- [“IGMP Snooping and Switch Stacks”](#) section on page 24-6
- [“Port Security and Switch Stacks”](#) section on page 25-18
- [“CDP and Switch Stacks”](#) section on page 26-2
- [“SPAN and RSPAN and Switch Stacks”](#) section on page 29-9
- [“Configuring QoS”](#) section on page 35-1
- [“ACLs and Switch Stacks”](#) section on page 34-6
- [“EtherChannel and Switch Stacks”](#) section on page 36-10
- [“IP Routing and Switch Stacks”](#) section on page 38-3
- [“IPv6 and Switch Stacks”](#) section on page 39-10
- [“HSRP and Switch Stacks”](#) section on page 42-5
- [“Multicast Routing and Switch Stacks”](#) section on page 46-10
- [“Fallback Bridging and Switch Stacks”](#) section on page 48-3

Stack Management Connectivity

You manage the stack and the member interfaces through the master. You can use the CLI, SNMP, Network Assistant, and CiscoWorks network management applications. You cannot manage members as individual switches.

- [Stack Through an IP Address](#), page 5-16
- [Stack Through an SSH Session](#), page 5-16
- [Stack Through Console Ports](#), page 5-16
- [Specific Members](#), page 5-16

Stack Through an IP Address

The stack is managed through a system-level IP address. You can still manage the stack through the same IP address even if you remove the master or any other stack member from the stack, provided there is IP connectivity.

**Note**

Members keep their IP addresses when you remove them from a stack. To avoid having two devices with the same IP address in your network, change the IP address of the switch that you removed from the stack.

For related information about switch stack configurations, see the [“Stack Configuration Files” section on page 5-14](#).

Stack Through an SSH Session

The Secure Shell (SSH) connectivity to the stack can be lost if a master running the cryptographic version of the IP base image or IP services image software fails and is replaced by a switch that is running a noncryptographic version. We recommend that a switch running the cryptographic version of the IP base image or IP services image software be the master.

Stack Through Console Ports

You can connect to the master through the console port of one or more members.

Be careful when using multiple CLI sessions to the master. Commands that you enter in one session are not displayed in the other sessions. Therefore, it is possible that you might not be able to identify the session from which you entered a command.

We recommend that you use only one CLI session when managing the stack.

Specific Members

If you want to configure a specific member port, you must include the stack member number in the CLI notation.

To access a specific member, see the [“Accessing the CLI of a Specific Member” section on page 5-23](#).

Stack Configuration Scenarios

Most of the scenarios in [Table 5-2](#) assume at least two switches are connected through their StackWise ports.

Table 5-2 Switch Stack Configuration Scenarios

Scenario		Result
Master election specifically determined by existing masters	Connect two powered-on stacks through the StackWise ports.	Only one of the two masters becomes the new stack master.
Master election specifically determined by the member priority value	<ol style="list-style-type: none"> 1. Connect two switches through their StackWise ports. 2. Use the switch stack-member-number priority new-priority-number global configuration command to set one member with a higher member priority value. 3. Restart both members at the same time. 	The member with the higher priority value is elected master.
Master election specifically determined by the configuration file	<p>Assuming that both members have the same priority value:</p> <ol style="list-style-type: none"> 1. Make sure that one member has a default configuration and that the other member has a saved (nondefault) configuration file. 2. Restart both members at the same time. 	The member with the saved configuration file is elected master.
Master election specifically determined by the cryptographic IP services image software	<p>Assuming that all members have the same priority value:</p> <ol style="list-style-type: none"> 1. Make sure that one member has the cryptographic IP services image software installed and that the other member has the noncryptographic IP services image software installed. 2. Restart both members at the same time. 	The member with the cryptographic IP services image software is elected master.
Master election specifically determined by the cryptographic IP base image software	<p>Assuming that all members have the same priority value:</p> <ol style="list-style-type: none"> 1. Make sure that one member has the cryptographic IP base image software installed and that the other member has the noncryptographic IP base image software installed. 2. Restart both members at the same time. 	The member with the cryptographic IP base image software is elected master.
Master election specifically determined by the MAC address	Assuming that both members have the same priority value, configuration file, and software image, restart both stack members at the same time.	The member with the lower MAC address is elected master.

Table 5-2 Switch Stack Configuration Scenarios (continued)

Scenario		Result
Member number conflict	Assuming that one member has a higher priority value than the other member: <ol style="list-style-type: none"> 1. Ensure that both members have the same member number. If necessary, use the switch current-stack-member-number renumber new-stack-member-number global configuration command. 2. Restart both members at the same time. 	The member with the higher priority value keeps its member number. The other member has a new stack member number.
Add a member	<ol style="list-style-type: none"> 1. Power off the new switch. 2. Through their StackWise ports, connect the new switch to a powered-on stack. 3. Power on the new switch. 	The master is kept. The new switch is added to the stack.
Master failure	Remove (or power off) the master.	One of the remaining stack members becomes the new master. All other members in the stack remain members and do not restart.
Add more than nine members	<ol style="list-style-type: none"> 1. Through their StackWise ports, connect ten switches. 2. Power on all switches. 	Two switches become masters. One master has nine stack members. The other master remains a standalone switch. Use the Mode button and port LEDs on the switches to identify which switches are masters and which switches belong to each master. For information about the Mode button and the LEDs, see the hardware installation guide.

Configuring the Switch Stack

- [Default Switch Stack Configuration, page 5-19](#)
- [Enabling Persistent MAC Address, page 5-19](#)
- [Assigning Stack Member Information, page 5-21](#)
- [Changing the Stack Membership, page 5-23](#)

Default Switch Stack Configuration

Table 5-3 shows the default switch stack configuration.

Table 5-3 Default Switch Stack Configuration

Feature	Default Setting
Stack MAC address timer	Disabled.
Member number	1
Member priority value	1
Offline configuration	The switch stack is not provisioned.
Persistent MAC address	Disabled.

Enabling Persistent MAC Address

The MAC address of the master determines the stack MAC address. When a master is removed from the stack and a new master takes over, the MAC address of the new master becomes the new stack MAC address. However, you can set the persistent MAC address feature with a time delay before the stack MAC address changes. During this time period, if the previous master rejoins the stack, the stack continues to use that MAC address as the stack MAC address, even if the switch is now a member and not a master. You can also configure stack MAC persistency so that the stack MAC address never changes to the new master MAC address.



Caution

When you configure this feature, a warning message displays the consequences of your configuration. You should use this feature cautiously. Using the old master MAC address elsewhere in the domain could result in lost traffic.

You can set the time period from 0 to 60 minutes.

- If you enter the command with no value, the default delay is 4 minutes. We recommend that you always enter a value. The time delay appears in the configuration file with an explicit timer value of 4 minutes.
- If you enter **0**, the stack MAC address of the previous master is used until you enter the **no stack-mac persistent timer** global configuration command, which changes the stack MAC address to that of the current master. If you do not enter this command, the stack MAC address does not change.
- If you enter a time delay of 1 to 60 minutes, the stack MAC address of the previous master is used until the configured time period expires or until you enter the **no stack-mac persistent timer** command.


If the previous master does not rejoin the stack during this period, the stack uses the MAC address of the new master as the stack MAC address.



Note

If the entire switch stack reloads, it acquires the MAC address of the master as the stack MAC address.

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

	Command	Purpose
Step 1	configure terminal	Enter global configuration mode.
Step 2	stack-mac persistent timer [0 <i>time-value</i>]	<p>Enable a time delay after a stack-master change before the stack MAC address changes to that of the new stack master. If the previous stack master rejoins the stack during this period, the stack uses that MAC address as the stack MAC address.</p> <ul style="list-style-type: none"> • Enter the command with no value to set the default delay of 4 minutes. We recommend that you always configure a value. • Enter 0 to use the MAC address of the current master indefinitely. • Enter a <i>time-value</i> from 1 to 60 to configure the time period (in minutes) before the stack MAC address changes to the new master. <p> Caution When you enter this command, a warning states that traffic might be lost if the old master MAC address appears elsewhere in the network domain.</p> <p>If you enter the no stack-mac persistent timer command after a new stack master takes over, before the time expires, the stack uses the current master MAC address.</p>
Step 3	end	Return to privileged EXEC mode.
Step 4	show running-config or show switch	<p>Verify that the stack MAC address timer is enabled.</p> <p>The output shows <code>stack-mac persistent timer</code> and the time in minutes.</p> <p>The output shows <code>Mac persistency wait time</code> with the number of minutes configured and the stack MAC address.</p>
Step 5	copy running-config startup-config	(Optional) Save your entries in the configuration file.

Use the **no stack-mac persistent timer** global configuration command to disable the persistent MAC address feature.

This example shows how to configure the persistent MAC address feature for a 7-minute time delay and to verify the configuration:

```
Switch(config)# stack-mac persistent timer 7
WARNING: The stack continues to use the base MAC of the old Master
WARNING: as the stack MAC after a master switchover until the MAC
WARNING: persistency timer expires. During this time the Network
WARNING: Administrators must make sure that the old stack-mac does
WARNING: not appear elsewhere in this network domain. If it does,
WARNING: user traffic may be blackholed.
Switch(config)# end
Switch# show switch
Switch/Stack Mac Address : 0016.4727.a900
Mac persistency wait time: 7 mins

                H/W   Current
Switch#  Role   Mac Address      Priority Version  State
-----
*1      Master 0016.4727.a900    1         0         Ready
```

Assigning Stack Member Information

- [Assigning a Member Number, page 5-21](#) (optional)
- [Setting the Member Priority Value, page 5-22](#) (optional)
- [Provisioning a New Member for a Stack, page 5-22](#) (optional)

Assigning a Member Number



Note

This task is available only from the master.

Beginning in privileged EXEC mode, follow these steps to assign a member number to a member. This procedure is optional.

	Command	Purpose
Step 1	configure terminal	Enter global configuration mode.
Step 2	switch <i>current-stack-member-number</i> renumber <i>new-stack-member-number</i>	Specify the current member number and the new member number for the member. The range is 1 to 9. You can display the current member number by using the show switch user EXEC command.
Step 3	end	Return to privileged EXEC mode.
Step 4	reload slot <i>stack-member-number</i>	Reset the stack member.
Step 5	show switch	Verify the stack member number.
Step 6	copy running-config startup-config	Save your entries in the configuration file.

Setting the Member Priority Value



Note This task is available only from the master.

Beginning in privileged EXEC mode, follow these steps to assign a priority value to a member: This procedure is optional.

	Command	Purpose
Step 1	configure terminal	Enter global configuration mode.
Step 2	switch <i>stack-member-number</i> priority <i>new-priority-number</i>	Specify the member number and the new priority for the member. The member number range is 1 to 9. The priority value range is 1 to 15. You can display the current priority value by using the show switch user EXEC command. The new priority value takes effect immediately but does not affect the current master until the current master or the stack resets.
Step 3	end	Return to privileged EXEC mode.
Step 4	reload slot <i>stack-member-number</i>	Reset the member, and apply this configuration.
Step 5	show switch <i>stack-member-number</i>	Verify the member priority value.
Step 6	copy running-config startup-config	(Optional) Save your entries in the configuration file.

You can also set the SWITCH_PRIORITY environment variable. For more information, see the [“Controlling Environment Variables” section on page 3-22](#).

Provisioning a New Member for a Stack



Note This task is available only from the master.

Beginning in privileged EXEC mode, follow these steps to provision a new member for a stack. This procedure is optional.

	Command	Purpose
Step 1	show switch	Display summary information about the stack.
Step 2	configure terminal	Enter global configuration mode.
Step 3	switch <i>stack-member-number</i> provision <i>type</i>	Specify the member number for the provisioned switch. By default, no switches are provisioned. For <i>stack-member-number</i> , the range is 91 to 4. Enter a member number that is not already used in the stack. See Step 1. For <i>type</i> , enter the model number of the member.
Step 4	end	Return to privileged EXEC mode.
Step 5	show running-config	Verify the correct numbering of interfaces in the configuration.

	Command	Purpose
Step 6	<code>show switch stack-member-number</code>	Verify the status of the provisioned switch. For <i>stack-member-number</i> , enter the same number as in Step 2.
Step 7	<code>copy running-config startup-config</code>	(Optional) Save your entries in the configuration file.

To remove provisioned information and to avoid receiving an error message, remove the specified switch from the stack before you use the **no** form of this command.

This example shows how to provision a Catalyst 3750G-12S switch with a stack member number of 2 for the stack. The **show running-config** command output shows the interfaces associated with the provisioned switch:

```
Switch(config)# switch 2 provision WS-C3750G-12S
Switch(config)# end
Switch# show running-config | include switch 2
!
interface GigabitEthernet2/0/1
!
interface GigabitEthernet2/0/2
!
interface GigabitEthernet2/0/3
<output truncated>
```

Changing the Stack Membership

If you remove powered-on members but do not want to partition the stack:

-
- Step 1** Power off the newly created stacks.
 - Step 2** Reconnect them to the original stack through their StackWise ports.
 - Step 3** Power on the switches.
-

Accessing the CLI of a Specific Member



Note

This task is only for debugging purposes, and is only available from the master.

You can access all or specific members by using the **remote command** `{all | stack-member-number}` privileged EXEC command. The stack member number range is 1 to 9.

You can access specific members by using the **session** `stack-member-number` privileged EXEC command. The member number is appended to the system prompt. For example, the prompt for member 2 is `Switch-2#`, and system prompt for the master is `Switch#`. Enter `exit` to return to the CLI session on the master. Only the **show** and **debug** commands are available on a specific member.

For more information, see the [“Using Interface Configuration Mode”](#) section on page 12-11.

Displaying Stack Information

To display saved configuration changes after resetting a specific member or the stack, use these privileged EXEC commands:

Table 5-4 *Commands for Displaying Stack Information*

Command	Description
<code>show platform stack passive-links all</code>	Display all stack information, such as the stack protocol version.
<code>show platform stack ports {buffer history}</code>	Display the StackWise port events and history.
<code>show switch</code>	Display summary information about the stack, including the status of provisioned switches and switches in version-mismatch mode.
<code>show switch stack-member-number</code>	Display information about a specific member.
<code>show switch detail</code>	Display detailed information about the stack ring.
<code>show switch neighbors</code>	Display the stack neighbors.
<code>show switch stack-ports [summary]</code>	Display port information for the stack. Use the summary keyword to display the StackWise cable length, the stack link status, and the loopback status.
<code>show switch stack-ring activity [detail]</code>	Display the number of frames per member that are sent to the stack ring. The detail keyword displays the number of frames per member that are sent to the stack ring, the receive queues, and the ASIC.

Troubleshooting Stacks

- [Manually Disabling a StackWise Port, page 5-24](#)
- [Re-Enabling a StackWise Port While Another Member Starts, page 5-25](#)
- [Understanding the show switch stack-ports summary Output, page 5-25](#)
- [Identifying Loopback Problems, page 5-26](#)
- [Finding a Disconnected Cable, page 5-31](#)

Manually Disabling a StackWise Port

If a StackWise port is flapping and causing instability in the stack ring, to disable the port, enter the `switch stack-member-number stack port port-number disable` privileged EXEC command. To re-enable the port, enter the `switch stack-member-number stack port port-number enable` command.



Note

Be careful when using the `switch stack-member-number stack port port-number disable` command. When you disable the StackWise port, the stack operates at half bandwidth.

- A stack is in the *full-ring* state when all members are connected through the StackWise ports and are in the ready state.
- The stack is in the *partial-ring* state when
 - All members are connected through the StackWise ports, but some all are not in the ready state.
 - Some members are not connected through the StackWise ports.

When you enter the **switch stack-member-number stack port port-number disable** privileged EXEC command and

- The stack is in the full-ring state, you can disable only one StackWise port. This message appears:
Enabling/disabling a stack port may cause undesired stack changes. Continue?[confirm]
- The stack is in the partial-ring state, you cannot disable the port. This message appears:
Disabling stack port not allowed with current stack configuration.

Re-Enabling a StackWise Port While Another Member Starts

StackWise Port 1 on Switch 1 is connected to Port 2 on Switch 4. If Port 1 is flapping, disable Port 1 with the **switch 1 stack port 1 disable** privileged EXEC command.

While Port 1 on Switch 1 is disabled and Switch 1 is still powered on:

1. Disconnect the stack cable between Port 1 on Switch 1 and Port 2 on Switch 4.
2. Remove Switch 4 from the stack.
3. Add a switch to replace Switch 4 and assign it switch-number 4.
4. Reconnect the cable between Port 1 on Switch 1 and Port 2 on Switch 4 (the replacement switch).
5. Re-enable the link between the switches. Enter the **switch 1 stack port 1 enable** privileged EXEC command to enable Port 1 on Switch 1.
6. Power on Switch 4.



Caution

Powering on Switch 4 before enabling the Port 1 on Switch 1 might cause one of the switches to reload.

If Switch 4 is powered on first, you might need to enter the **switch 1 stack port 1 enable** and the **switch 4 stack port 2 enable** privileged EXEC commands to bring up the link.

Understanding the show switch stack-ports summary Output

Only Port 1 on stack member 2 is disabled.

```
Switch# show switch stack-ports summary
Switch#/  Stack  Neighbor  Cable  Link  Link  Sync  #  In
Port#     Port   Status    Length OK   Active OK  Changes  Loopback
          Status
-----
1/1       OK     3         50 cm  Yes  Yes  Yes  1       No
1/2       Down   None      3 m    Yes  No   Yes  1       No
2/1       Down   None      3 m    Yes  No   Yes  1       No
2/2       OK     3         50 cm  Yes  Yes  Yes  1       No
3/1       OK     2         50 cm  Yes  Yes  Yes  1       No
3/2       OK     1         50 cm  Yes  Yes  Yes  1       No
```

Table 5-5 *show switch stack-ports summary Command Output*

Field	Description
Switch#/Port#	Member number and its StackWise port number.
Stack Port Status	<ul style="list-style-type: none"> Absent—No cable is detected on the StackWise port. Down—A cable is detected, but either no connected neighbor is up, or the StackWise port is disabled. OK—A cable is detected, and the connected neighbor is up.
Neighbor	Switch number of the active member at the other end of the StackWise cable.
Cable Length	Valid lengths are 50 cm, 1 m, or 3 m. If the switch cannot detect the cable length, the value is <i>no cable</i> . The cable might not be connected, or the link might be unreliable.
Link OK	This shows if the link is stable. The <i>link partner</i> is a StackWise port on a neighbor switch. <ul style="list-style-type: none"> No—The link partner receives invalid protocol messages from the port. Yes—The link partner receives valid protocol messages from the port.
Link Active	This shows if the StackWise port is in the same state as its link partner. <ul style="list-style-type: none"> No—The port cannot send traffic to the link partner. Yes—The port can send traffic to the link partner.
Sync OK	<ul style="list-style-type: none"> No—The link partner does not send valid protocol messages to the StackWise port. Yes—The link partner sends valid protocol messages to the port.
# Changes to LinkOK	This shows the relative stability of the link. If a large number of changes occur in a short period of time, link flapping can occur.
In Loopback	<ul style="list-style-type: none"> No—At least one StackWise port on the member has an attached StackWise cable. Yes—None of the StackWise ports on the member has an attached StackWise cable.

Identifying Loopback Problems

- [Software Loopback, page 5-27](#)
- [Software Loopback Example: No Connected StackWise Cable, page 5-28](#)
- [Software Loopback Examples: Connected StackWise Cables, page 5-28](#)
- [Hardware Loopback, page 5-29](#)

- [Hardware Loopback Example: LINK OK event, page 5-29](#)
- [Hardware Loop Example: LINK NOT OK Event, page 5-30](#)

Software Loopback

In a stack with three members, StackWise cables connect all the members.

```
Switch# show switch stack-ports summary
Switch#/ Stack Neighbor Cable Link Link Sync # In
Port# Port Length OK Active OK Changes Loopback
Status To LinkOK
-----
1/1 OK 3 50 cm Yes Yes Yes 1 No
1/2 OK 2 3 m Yes Yes Yes 1 No
2/1 OK 1 3 m Yes Yes Yes 1 No
2/2 OK 3 50 cm Yes Yes Yes 1 No
3/1 OK 2 50 cm Yes Yes Yes 1 No
3/2 OK 1 50 cm Yes Yes Yes 1 No
```

If you disconnect the stack cable from Port 1 on Switch 1, these messages appear:

```
01:09:55: %STACKMGR-4-STACK_LINK_CHANGE: Stack Port 2 Switch 3 has changed to state DOWN
01:09:56: %STACKMGR-4-STACK_LINK_CHANGE: Stack Port 1 Switch 1 has changed to state DOWN
```

```
Switch# show switch stack-ports summary
Switch#/ Stack Neighbor Cable Link Link Sync # In
Port# Port Length OK Active OK Changes Loopback
Status To LinkOK
-----
1/1 Absent None No cable No No No 1 No
1/2 OK 2 3 m Yes Yes Yes 1 No
2/1 OK 1 3 m Yes Yes Yes 1 No
2/2 OK 3 50 cm Yes Yes Yes 1 No
3/1 OK 2 50 cm Yes Yes Yes 1 No
3/2 Down None 50 cm No No No 1 No
```

If you disconnect the stack cable from Port 2 on Switch 1, the stack splits.

Switch 2 and Switch 3 are now in a two-member stack connected through stack cables.

```
Switch# show sw stack-ports summary
Switch#/ Stack Neighbor Cable Link Link Sync # In
Port# Port Length OK Active OK Changes Loopback
Status To LinkOK
-----
2/1 Down None 3 m No No No 1 No
2/2 OK 3 50 cm Yes Yes Yes 1 No
3/1 OK 2 50 cm Yes Yes Yes 1 No
3/2 Down None 50 cm No No No 1 No
```

Switch 1 is a standalone switch.

```
Switch# show switch stack-ports summary
Switch#/ Stack Neighbor Cable Link Link Sync # In
Port# Port Length OK Active OK Changes Loopback
Status To LinkOK
-----
1/1 Absent None No cable No No No 1 Yes
1/2 Absent None No cable No No No 1 Yes
```

Software Loopback Example: No Connected StackWise Cable

Catalyst 3750 switch port status:

```
Switch# show switch stack-ports summary
Switch#/ Stack Neighbor Cable Link Link Sync # In
Port# Port Length OK Active OK Changes In
Status To LinkOK Loopback
-----
1/1 Absent None No cable Yes No Yes 1 Yes
1/2 Absent None No cable Yes No Yes 1 Yes
```

Catalyst 3750-E switch port status:

```
Switch# show switch stack-ports summary
Switch#/ Stack Neighbor Cable Link Link Sync # In
Port# Port Length OK Active OK Changes In
Status To LinkOK Loopback
-----
1/1 Absent None No cable No No No 1 Yes
1/2 Absent None No cable No No No 1 Yes
```

Software Loopback Examples: Connected StackWise Cables

- On Port 1 on Switch 1, the port status is *Down*, and a cable is connected.
- On Port 2 on Switch 1, the port status is *Absent*, and no cable is connected.

```
Switch# show switch stack-ports summary
Switch#/ Stack Neighbor Cable Link Link Sync # In
Port# Port Length OK Active OK Changes In
Status To LinkOK Loopback
-----
1/1 Down None 50 Cm No No No 1 No
1/2 Absent None No cable No No No 1 No
```

- In a *physical loopback*, a cable connects both stack ports on a switch. You can use this configuration to test
 - Cables on a switch that is running properly
 - StackWise ports with a cable that works properly

```
Switch# show switch stack-ports summary
Switch#/ Stack Neighbor Cable Link Link Sync # In
Port# Port Length OK Active OK Changes In
Status To LinkOK Loopback
-----
2/1 OK 2 50 cm Yes Yes Yes 1 No
2/2 OK 2 50 cm Yes Yes Yes 1 No
```

The port status shows that

- Switch 2 is a standalone switch.
- The ports can send and receive traffic.

Hardware Loopback

The **show platform stack ports buffer** privileged EXEC command output shows the hardware loopback values.

```
Switch# show platform stack ports buffer
          Stack Debug Event Data Trace
=====
Event type LINK: Link status change
Event type RAC: RAC changes to Not OK
Event type SYNC: Sync changes to Not OK
=====

  Event   Stack           Stack PCS Info           Ctrl-Status  Loopback  Cable
  Count   Port                                     IOS / HW     length
=====  =====
Event type: LINK OK Stack Port 1
0000000011  1  FF08FF00 860302A5 AA55FFFF FFFFFFFF  1CE61CE6  Yes/Yes  No cable
0000000011  2  FF08FF00 86031805 55AFFFFF FFFFFFFF  1CE61CE6  Yes/Yes  No cable
Event type: LINK OK Stack Port 2
0000000012  1  FF08FF00 860302A5 AA55FFFF FFFFFFFF  1CE61CE6  Yes/Yes  No cable
0000000012  2  FF08FF00 86031805 55AFFFFF FFFFFFFF  1CE61CE6  Yes/Yes  No cable
Event type: RAC
0000000013  1  FF08FF00 860302A5 AA55FFFF FFFFFFFF  1CE61CE6  Yes/Yes  No cable
0000000013  2  FF08FF00 86031805 55AFFFFF FFFFFFFF  1CE61CE6  Yes/Yes  No cable
```

On a Catalyst 3750v2 member, the *Loopback HW* value is always *N/A*.

On a Catalyst 3750 member,

- If at least one stack port has an connected stack cable, the *Loopback HW* value for both stack ports is *No*.
- If neither stack port has an connected stack cable, the *Loopback HW* value for both stack ports is *Yes*.

On a Catalyst 3750-E member,

- If a stack port has an connected stack cable, the *Loopback HW* value for the stack port is *No*.
- If the stack port does not have an connected stack cable, the *Loopback HW* value for the stack port is *Yes*.

Hardware Loopback Example: LINK OK event

On a Catalyst 3750 switch:

```
Switch# show platform stack ports buffer
          Stack Debug Event Data Trace
=====
Event type LINK: Link status change
Event type RAC: RAC changes to Not OK
Event type SYNC: Sync changes to Not OK
=====

  Event   Stack           Stack PCS Info           Ctrl-Status  Loopback  Cable
  Count   Port                                     IOS / HW     length
=====  =====
Event type: LINK OK Stack Port 1
0000000008  1  FF08FF00 8603F083 55AFFFFF FFFFFFFF  0CE60C10  No /No   50 cm
0000000008  2  FF08FF00 0001DBDF 01000B00 FFFFFFFF  0CE60C10  No /No   No cable
Event type: RAC
0000000009  1  FF08FF00 8603F083 55AFFFFF FFFFFFFF  0CE60C10  No /No   50 cm
0000000009  2  FF08FF00 0001DC1F 02000100 FFFFFFFF  0CE60C10  No /No   No cable
```

On a Catalyst 3750-E switch:

```
Switch# show platform stack ports buffer
```

```

Stack Debug Event Data Trace
=====
Event type LINK: Link status change
Event type RAC: RAC changes to Not OK
Event type SYNC: Sync changes to Not OK
=====

Event Count Stack Port Stack PCS Info Ctrl-Status Loopback Cable
Count Port Info Status IOS / HW length
=====
Event type: LINK OK Stack Port 1
0000000153 1 FF01FF00 860351A5 55A5FFFF FFFFFFFF 0CE60C10 No /No 50 cm
0000000153 2 FF01FF00 00017C07 00000000 0000FFFF 0CE60C10 No /No 3 m
Event type: RAC
0000000154 1 FF01FF00 860351A5 55A5FFFF FFFFFFFF 0CE60C10 No /No 50 cm
0000000154 2 FF01FF00 00017C85 00000000 0000FFFF 0CE60C10 No /No 3 m

```

Hardware Loop Example: LINK NOT OK Event

On a Catalyst 3750 switch:

```
Switch# show platform stack ports buffer
```

```

Stack Debug Event Data Trace
=====
Event type LINK: Link status change
Event type RAC: RAC changes to Not OK
Event type SYNC: Sync changes to Not OK
=====

Event Count Stack Port Stack PCS Info Ctrl-Status Loopback Cable
Count Port Info Status IOS / HW length
=====
Event type: LINK OK Stack Port 2
0000000005 1 FF08FF00 0001FBD3 0801080B EFFFFFFF 0C100CE6 No /No No cable
0000000005 2 FF08FF00 8603E4A9 5555FFFF FFFFFFFF 0C100CE6 No /No 50 cm
Event type: RAC
0000000006 1 FF08FF00 0001FC14 08050204 EFFFFFFF 0C100CE6 No /No No cable
0000000006 2 FF08FF00 8603E4A9 5555FFFF FFFFFFFF 0C100CE6 No /No 50 cm
Event type: LINK NOT OK Stack Port 2
0000000939 1 FF08FF00 00016879 00010000 EFFFFFFF 0C100C14 No /No No cable
0000000939 2 FF08FF00 0001901F 00000000 FFFFFFFF 0C100C14 No /No No cable
Event type: RAC
0000000940 1 FF08FF00 000168BA 00010001 EFFFFFFF 0C100C14 No /No No cable
0000000940 2 FF08FF00 0001905F 00000000 FFFFFFFF 0C100C14 No /No No cable
Event type: LINK OK Stack Port 1
0000000956 1 FF08FF00 86034DAC 5555FFFF FFFFFFFF 1CE61CE6 Yes/Yes No cable
0000000956 2 FF08FF00 86033431 55AAFFFF FFFFFFFF 1CE61CE6 Yes/Yes No cable
Event type: LINK OK Stack Port 2
0000000957 1 FF08FF00 86034DAC 5555FFFF FFFFFFFF 1CE61CE6 Yes/Yes No cable
0000000957 2 FF08FF00 86033431 55AAFFFF FFFFFFFF 1CE61CE6 Yes/Yes No cable
Event type: RAC
0000000958 1 FF08FF00 86034DAC 5555FFFF FFFFFFFF 1CE61CE6 Yes/Yes No cable
0000000958 2 FF08FF00 86033431 55AAFFFF FFFFFFFF 1CE61CE6 Yes/Yes No cable

```

On a Catalyst 3750-E switch:

```
Switch# show platform stack ports buffer
```

```
Stack Debug Event Data Trace
```

```
Event type LINK: Link status change
Event type RAC: RAC changes to Not OK
Event type SYNC: Sync changes to Not OK
```

```
=====
Event type LINK: Link status change
Event type RAC: RAC changes to Not OK
Event type SYNC: Sync changes to Not OK
=====
```

Event Count	Stack Port	Stack PCS Info	Ctrl-Status	Loopback IOS / HW	Cable length
Event type: LINK OK Stack Port 1					
0000000014	1	FF01FF00 860204A7 5555FFFF 00000000	0CE60CA6	No /No	50 cm
0000000014	2	FF01FF00 85020823 AAAAFFFF 00000000	0CE60CA6	No /No	3 m
Event type: RAC					
0000000015	1	FF01FF00 860204A7 5555FFFF 00000000	0CE60CA6	No /No	50 cm
0000000015	2	FF01FF00 85020823 AAAAFFFF 00000000	0CE60CA6	No /No	3 m
Event type: LINK OK Stack Port 2					
0000000029	1	FF01FF00 860204A7 5555FFFF 00000000	1CE61CE6	No /No	50 cm
0000000029	2	FF01FF00 86020823 AAAAFFFF 00000000	1CE61CE6	No /No	3 m
Event type: RAC					
0000000030	1	FF01FF00 860204A7 5555FFFF 00000000	1CE61CE6	No /No	50 cm
0000000030	2	FF01FF00 86020823 AAAAFFFF 00000000	1CE61CE6	No /No	3 m
Event type: LINK NOT OK Stack Port 1					
0000009732	1	FF01FF00 00015B12 5555FFFF A49CFFFF	0C140CE4	No /No	50 cm
0000009732	2	FF01FF00 86020823 AAAAFFFF 00000000	0C140CE4	No /No	3 m
Event type: RAC					
0000009733	1	FF01FF00 00015B4A 5555FFFF A49CFFFF	0C140CE4	No /No	50 cm
0000009733	2	FF01FF00 86020823 AAAAFFFF 00000000	0C140CE4	No /No	3 m
Event type: LINK NOT OK Stack Port 2					
0000010119	1	FF01FF00 00010E69 25953FFF FFFFFFFF	0C140C14	No /Yes	No cable
0000010119	2	FF01FF00 0001D98C 81AAC7FF 0300FFFF	0C140C14	No /No	3 m
Event type: RAC					
0000010120	1	FF01FF00 00010EEA 25953FFF FFFFFFFF	0C140C14	No /Yes	No cable
0000010120	2	FF01FF00 0001DA0C 81AAC7FF 0300FFFF	0C140C14	No /No	3 m

Finding a Disconnected Cable

StackWise cables connect all stack members. Port 2 on Switch 1 connects to Port 1 on Switch 2.

This is the port status for the members:

```
Switch# show switch stack-ports summary
```

Switch#/Port#	Stack Port Status	Neighbor	Cable Length	Link OK	Link Active	Sync OK	# Changes To LinkOK	In Loopback
1/1	OK	2	50 cm	Yes	Yes	Yes	0	No
1/2	OK	2	50 cm	Yes	Yes	Yes	0	No
2/1	OK	1	50 cm	Yes	Yes	Yes	0	No
2/2	OK	1	50 cm	Yes	Yes	Yes	0	No

If you disconnect the cable from Port 2 on Switch 1, these messages appear:

```
%STACKMGR-4-STACK_LINK_CHANGE: Stack Port 1 Switch 2 has changed to state DOWN
```

```
%STACKMGR-4-STACK_LINK_CHANGE: Stack Port 2 Switch 1 has changed to state DOWN
```

This is now the port status:

```
Switch# show switch stack-ports summary
Switch#/ Stack Neighbor Cable Link Link Sync # In
Port# Port Length OK Active OK Changes Loopback
Status To LinkOK
-----
1/1 OK 2 50 cm Yes Yes Yes 1 No
1/2 Absent None No cable No No No 2 No
2/1 Down None 50 cm No No No 2 No
2/2 OK 1 50 cm Yes Yes Yes 1 No
```

Only one end of the cable connects to a StackWise port, Port 1 on Switch 2.

- The *Stack Port Status* value for Port 2 on Switch 1 is *Absent*, and the value for Port 1 on Switch 2 is *Down*.
- The *Cable Length* value is *No cable*.

Diagnosing the problem:

- Verify the cable connection for Port 2 on Switch 1.
- Port 2 on Switch 1 has a port or cable problem if
 - The *In Loopback* value is *Yes*.
 - or
 - The *Link OK*, *Link Active*, or *Sync OK* value is *No*.

Fixing a Bad Connection Between StackWise Ports

StackWiseStack cables connect all members. Port 2 on Switch 1 connects to Port 1 on Switch 2.

This is the port status:

```
Switch# show switch stack-ports summary
Switch#/ Stack Neighbor Cable Link Link Sync # In
Port# Port Length OK Active OK Changes Loopback
Status To LinkOK
-----
1/1 OK 2 50 cm Yes Yes Yes 1 No
1/2 Down None 50 cm No No No 2 No
2/1 Down None 50 cm No No No 2 No
2/2 OK 1 50 cm Yes Yes Yes 1 No
```

Diagnosing the problem:

- The *Stack Port Status* value is *Down*.
- *Link OK*, *Link Active*, and *Sync OK* values are *No*.
- The *Cable Length* value is *50 cm*. The switch detects and correctly identifies the cable.

The connection between Port 2 on Switch 1 and Port 1 on Switch 2 is unreliable on at least one of the connector pins.