Managing Switch Stacks

A switch stack can have up to eight stacking-capable switches connected through their StackWise ports. The stack members work together as a unified system. Layer 2 and Layer 3 protocols present the entire switch stack as a single entity to the network.

Prerequisites for Switch Stacks

- All the switches in the stack must be running the same license level as the active switch. For information about license levels, see the *System Management* section of this guide.
- All the switches in the switch stack must be running compatible software versions.

Restrictions for Switch Stacks

The following are the restrictions for switch stack configuration:

- This feature is not supported on C9200CX-12P-2X2G, C9200CX-8P-2X2G, and C9200CX-12T-2X2G models of the Cisco Catalyst 9200CX Series Switches
- A switch stack can have up to eight stacking-capable switches connected through their StackWise ports.
- Only homogenous stacking is supported, that is, a stack of Cisco Catalyst 9200 Series Switches with only Cisco Catalyst 9200 Series Switches as stack members.
- You cannot have a switch stack containing a mix of different license levels.
- Do not stack Cisco Catalyst 9200L Series Switches with Cisco Catalyst 9200 Series Switches.
- C9200-24PB and C9200-48PB switch models can only be stacked with each other and not with other models of the Cisco Catalyst 9200 Series Switches.

- During a switchover, when the standby device syncs with the active device, the following log message is displayed on the console:

```
%SM-4-BADEVENT: Event 'standby_phy_link_up' is invalid for
the current state 'NO_NEIGHBOR': rep_lsl_rx Gix/x/x -Traceback=
```

  Ignore this message. It does not have any functional or operational impact.

Information About Switch Stacks

Switch Stack Overview

A switch stack can have up to eight stacking-capable switches connected through their StackWise ports. The stack members work together as a unified system. Layer 2 and Layer 3 protocols present the entire switch stack as a single entity to the network.

The active switch controls the operation of the switch stack, and is the single point of stack-wide management.

From the active switch, you configure:

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

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

Switch Stack Bridge ID and MAC Address

A switch stack is identified in the network by its *bridge ID* and, if it is operating as a Layer 3 device, its router MAC address. The bridge ID and router MAC address are determined by the MAC address of the active switch.

If the active switch changes, the MAC address of the new active switch determines the new bridge ID and router MAC address.

If the entire switch stack reloads, the switch stack uses the MAC address of the active switch.

Persistent MAC Address on the Switch Stack

You can use the persistent MAC address feature to set a time delay before the stack MAC address changes. During this time period, if the previous active switch rejoins the stack, the stack continues to use its MAC address as the stack MAC address, even if the switch is now a stack member and not an active switch. If the previous active switch does not rejoin the stack during this period, the switch stack takes the MAC address of the new active switch as the stack MAC address. By default, the stack MAC address will be the MAC address of the first active switch, even if a new active switch takes over.

You can use the persistent MAC address feature to set a time delay before the stack MAC address changes to the MAC address of the new stack master. When this feature is enabled, the stack MAC address changes in approximately 4 minutes. During this time, if the previous stack master rejoins the stack, the stack continues to use its MAC address as the stack MAC address, even if the switch is now a stack member and not a stack

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

Note You can also configure stack MAC persistency so that the stack MAC address never changes to the new active switch MAC address, by using the **stack-mac persistent timer 0** command. This avoids Link Aggregation Control Protocol (LACP) and Port Aggregation Protocol (PAgP) flaps or inconsistencies.

Upgrading a Switch Running Incompatible Software

The auto-upgrade and auto-advise features enable a switch with software packages that are incompatible with the switch stack to be upgraded to a compatible software version so that it can join the switch stack.

Switch Stack Management Connectivity

You manage the switch stack and the stack member interfaces through the active switch. You can use the CLI, SNMP, and any of the supported network management applications. You cannot manage stack members on an individual basis.

How to Configure a Switch Stack

Monitoring the Device Stack

Table 1: Commands for Displaying Stack Information

Command	Description
show module	Displays summary informaton about the stack.
show switch detail	Displays detailed information about the stack.
show switch neighbors	Displays the stack neighbors.
show switch stack-ports [**summary**]	Displays port information for the stack. Use the **summary** keyword to display the stack cable length, the stack link status, and the loopback status.
show switch stack-ports [**detail**]	Displays the stack link status and information for each stack member. Use the **detail** keyword to display the stack interface status, errors, drops, packet transmission and bandwidth details.
show redundancy	Displays the redundant system and the current processor information. The redundant system information includes the system uptime, standby failures, switchover reason, hardware, configured and operating redundancy mode. The current processor information displayed includes the active location, the software state, the uptime in the current state and so on.

Command	Description
show redundancy state	Displays all the redundancy states of the active and standby devices.

Configuration Examples for Switch Stacks

Switch Stack Configuration Scenarios

Most of these switch stack configuration scenarios assume that at least two devices are connected through their StackWise ports.

Table 2: Configuration Scenarios

Scenario		Result
Active switch election specifically determined by existing active switches	Connect two powered-on switch stacks through the StackWise ports.	Only one of the two active switches becomes the new active switch.
Active switch election specifically determined by the stack member priority value	**1.** Connect two switches through their StackWise ports. **2.** Use the **switch** *stack-member-number* **priority** *new-priority-number* command to set one stack member with a higher member priority value. **3.** Restart both member switches at the same time.	The stack member with the higher priority value is elected active switch.
Active switch election specifically determined by the configuration file	Assuming that both member switches have the same priority value: **1.** Make sure that one stack member has a default configuration and that the other stack member has a saved (nondefault) configuration file. **2.** Restart both member switches at the same time.	The stack member with the saved configuration file is elected active switch.
Active switch election specifically determined by the MAC address	Assuming that both member switches have the same priority value, configuration file, and license level, restart both member switches at the same time.	The stack member with the lower MAC address is elected active switch .

Scenario		Result
Stack member number conflict	Assuming that one stack member has a higher priority value than the other stack member: **1.** Ensure that both member switches have the same stack member number. If necessary, use the **switch** *current-stack-member-number* **renumber** *new-stack-member-number* command. **2.** Restart both member switches at the same time.	The stack member with the higher priority value retains its stack member number. The other stack member has a new stack member number.
Add a stack member	**1.** Power off the new switch. **2.** Through their StackWise ports, connect the new switch to a powered-on switch stack. **3.** Power on the new switch.	The active switch is retained. The new switch is added to the switch stack.
Active switch failure	Remove (or power off) the active switch.	One of the remaining member switches becomes the new active switch. All other member switches in the stack remain as member switches and do not reboot.
Add member switches	**1.** Through their StackWise ports, connect devices. **2.** Power on all devices.	Two devices become active switches. One active switch has member switches. The other active switch remains as a standalone device. Use the Mode button and port LEDs on the device to identify which devices are active switches and which devices belong to each active switch.

Enabling the Persistent MAC Address Feature: Example

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

```
Device(config)# stack-mac persistent timer 7
  WARNING: The stack continues to use the base MAC of the old active
  WARNING: as the stack-MAC after a active 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.
  Device(config)# end
  Device# 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       Active 0016.4727.a900     1      P2B     Ready
```

show switch stack-ports summary Command Output: Example

Only Port 1 on stack member 2 is disabled.

```
# show switch stack-ports summary
  #/  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     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 3: show switch stack-ports summary Command Output

Field	Description
Switch#/Port#	Member number and its stack port number.
Stack Port Status	Status of the stack port. • Down—A cable is detected, but either no connected neighbor is up, or the stack 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 stack cable.
Cable Length	Valid lengths are 50 cm, 1 m, or 3 m. If the switch cannot detect the cable length, the value is *no cable*. The cable might not be connected, or the link might be unreliable. When there is no cable connected to the stack port, the value displayed is *no cable* along with the cable length value.
Link OK	Whether the stack cable is connected and functional. There may or may not be a neighbor connected on the other end. The *link partner* is a stack port on a neighbor switch. • No—There is no stack cable connected to this port or the stack cable is not functional. • Yes—There is a functional stack cable connected to this port.

Field	Description
Link Active	Whether a neighbor is connected on the other end of the stack cable. • No—No neighbor is detected on the other end. The port cannot send traffic over this link. • Yes—A neighbor is detected on the other end. The port can send traffic over this link.
Sync OK	Whether the link partner sends valid protocol messages to the stack port. • No—The link partner does not send valid protocol messages to the stack port. • Yes—The link partner sends valid protocol messages to the port.
# Changes to LinkOK	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	Whether a stack cable is attached to a stack port on the member. • No—At least one stack port on the member has an attached stack cable. • Yes—None of the stack ports on the member has an attached stack cable.

show switch stack-ports detail Command Output: Example

The following is a sample output of the command for a working stack:

```
Device# show switch stack-ports detail
1/1 is DOWN Loopback No
Cable Length 50cm     Neighbor NONE
Link Ok Yes Sync Ok Yes Link Active No
Changes to LinkOK 1
Five minute input rate  0 bytes/sec
Five minute output rate 0 bytes/sec
     752 bytes input
     240 bytes output
CRC Errors
            Data CRC 0
        Ringword CRC 0
        InvRingWord  0
         PcsCodeWord 667
1/2 is OK Loopback No
Cable Length 50cm     Neighbor 3
Link Ok Yes Sync Ok Yes Link Active Yes
Changes to LinkOK 1
Five minute input rate  7 bytes/sec
Five minute output rate 0 bytes/sec
     54332 bytes input
     1120 bytes output
CRC Errors
            Data CRC 0
        Ringword CRC 0
        InvRingWord  0
         PcsCodeWord 0
2/1 is OK Loopback No
Cable Length 50cm     Neighbor 3
```

```
Link Ok Yes Sync Ok Yes Link Active Yes
Changes to LinkOK 1
Five minute input rate  0 bytes/sec
Five minute output rate 30 bytes/sec
     146390 bytes input
     217587 bytes output
CRC Errors
            Data CRC 0
        Ringword CRC 0
        InvRingWord  0
         PcsCodeWord 0
2/2 is DOWN Loopback No
Cable Length 50cm     Neighbor NONE
Link Ok Yes Sync Ok Yes Link Active No
Changes to LinkOK 1
Five minute input rate  0 bytes/sec
Five minute output rate 0 bytes/sec
     1208 bytes input
     480 bytes output
CRC Errors
            Data CRC 0
        Ringword CRC 0
        InvRingWord  0
         PcsCodeWord 0
3/1 is OK Loopback No
Cable Length 50cm     Neighbor 1
Link Ok Yes Sync Ok Yes Link Active Yes
Changes to LinkOK 1
Five minute input rate  0 bytes/sec
Five minute output rate 0 bytes/sec
     41245 bytes input
     240 bytes output
CRC Errors
            Data CRC 0
        Ringword CRC 0
        InvRingWord  0
         PcsCodeWord 0
3/2 is OK Loopback No
Cable Length 50cm     Neighbor 2
Link Ok Yes Sync Ok Yes Link Active Yes
Changes to LinkOK 1
Five minute input rate  10 bytes/sec
Five minute output rate 0 bytes/sec
     60412 bytes input
     480 bytes output
CRC Errors
            Data CRC 0
        Ringword CRC 0
        InvRingWord  0
         PcsCodeWord 0
```

Table 4: show switch stack-ports detail Command Output

Field	Description
Neighbor	Switch number of the active member at the other end of the stack cable.
Cable Length	Valid lengths are 50 cm, 1 m, or 3 m. If the switch cannot detect the cable length, the value is *Unknown*. The cable might not be connected, or the link might be unreliable.

Field	Description
Link OK	Whether the stack cable is connected and functional. There may or may not be a neighbor connected on the other end. The *link partner* is a stack port on a neighbor switch. • No: There is no stack cable connected to this port or the stack cable is not functional. • Yes: There is a functional stack cable connected to this port.
Link Active	Whether a neighbor is connected on the other end of the stack cable. • No: No neighbor is detected on the other end. The port cannot send traffic over this link. • Yes: A neighbor is detected on the other end. The port can send traffic over this link.
Sync OK	Whether the link partner sends valid protocol messages to the stack port. • No: The link partner does not send valid protocol messages to the stack port. • Yes: The link partner sends valid protocol messages to the port.
# Changes to LinkOK	The relative stability of the link. If a large number of changes occur in a short period of time, link flapping can occur.
Five minute input rate	The average rate (calculated over a five minute period) at which packets are received, measured in packets/sec.
Five minute output rate	The average rate (calculated over a five minute period) at which packets are transmitted, measured in packets/sec.
CRC Errors	Different types of Cyclic Redundancy Check (CRC) errors that are seen on a stack interface: • **Data CRC**: Stack interface data CRC error • **Ringword CRC**: Stack interface ring word CRC error • **InvRingWord**: Stack interface invalid ring word error • **PcsCodeWord**: Stack interface Physical Coding Sublayer (PCS) error These errors normally occur when a stack interface state changes due to a switchover or a switch reload. You can ignore such errors. But when these error counters increase significantly or when they increase continuously over a period of time, check the stack cable for issues.

he following is a sample output when the stack port flaps:

```
Device# show switch stack-ports detail
1/1 is OK Loopback No
Cable Length 50cm     Neighbor 2
```

```
Link Ok Yes Sync Ok Yes Link Active Yes
Changes to LinkOK 4
Five minute input rate  0 bytes/sec
Five minute output rate 0 bytes/sec
     320 bytes input
     80 bytes output
CRC Errors
            Data CRC 0
        Ringword CRC 0
        InvRingWord  0
         PcsCodeWord 770
1/2 is OK Loopback No
Cable Length 50cm     Neighbor 3
Link Ok Yes Sync Ok Yes Link Active Yes
Changes to LinkOK 1
Five minute input rate  5 bytes/sec
Five minute output rate 1 bytes/sec
     2949 bytes input
     320 bytes output
CRC Errors
            Data CRC 0
        Ringword CRC 0
        InvRingWord  0
         PcsCodeWord 0
2/1 is OK Loopback No
Cable Length 50cm     Neighbor 3
Link Ok Yes Sync Ok Yes Link Active Yes
Changes to LinkOK 1
Five minute input rate  0 bytes/sec
Five minute output rate 0 bytes/sec
     49375 bytes input
     160 bytes output
CRC Errors
            Data CRC 0
        Ringword CRC 0
        InvRingWord  0
         PcsCodeWord 0
2/2 is OK Loopback No
Cable Length 50cm     Neighbor 1
Link Ok Yes Sync Ok Yes Link Active Yes
Changes to LinkOK 2
Five minute input rate  0 bytes/sec
Five minute output rate 0 bytes/sec
     1824 bytes input
     160 bytes output
CRC Errors
            Data CRC 0
        Ringword CRC 0
        InvRingWord  0
         PcsCodeWord 0
3/1 is OK Loopback No
Cable Length 50cm     Neighbor 1
Link Ok Yes Sync Ok Yes Link Active Yes
Changes to LinkOK 1
Five minute input rate  372 bytes/sec
Five minute output rate 7 bytes/sec
     111876 bytes input
     4613 bytes output
CRC Errors
            Data CRC 0
        Ringword CRC 0
        InvRingWord  0
         PcsCodeWord 0
3/2 is OK Loopback No
```

```
Cable Length 50cm     Neighbor 2
Link Ok Yes Sync Ok Yes Link Active Yes
Changes to LinkOK 2
Five minute input rate  0 bytes/sec
Five minute output rate 0 bytes/sec
     80 bytes input
     0 bytes output
CRC Errors
            Data CRC 0
         Ringword CRC 0
         InvRingWord  0
          PcsCodeWord 0
```

The following is a sample output when a switch reloads:

```
Device#show switch stack-ports detail
1/1 is OK Loopback No
Cable Length 50cm     Neighbor 2
Link Ok Yes Sync Ok Yes Link Active Yes
Changes to LinkOK 5
Five minute input rate  0 bytes/sec
Five minute output rate 0 bytes/sec
     2032 bytes input
     320 bytes output
CRC Errors
            Data CRC 184
         Ringword CRC 187
         InvRingWord  120
          PcsCodeWord 112
1/2 is OK Loopback No
Cable Length 50cm     Neighbor 3
Link Ok Yes Sync Ok Yes Link Active Yes
Changes to LinkOK 1
Five minute input rate  2 bytes/sec
Five minute output rate 0 bytes/sec
     24164 bytes input
     800 bytes output
CRC Errors
            Data CRC 0
         Ringword CRC 0
         InvRingWord  0
          PcsCodeWord 0
2/1 is OK Loopback No
Cable Length 50cm     Neighbor 3
Link Ok Yes Sync Ok Yes Link Active Yes
Changes to LinkOK 1
Five minute input rate  0 bytes/sec
Five minute output rate 0 bytes/sec
     3024 bytes input
     240 bytes output
CRC Errors
            Data CRC 0
         Ringword CRC 0
         InvRingWord  0
          PcsCodeWord 0
2/2 is OK Loopback No
Cable Length 50cm     Neighbor 1
Link Ok Yes Sync Ok Yes Link Active Yes
Changes to LinkOK 1
Five minute input rate  7 bytes/sec
Five minute output rate 0 bytes/sec
     9148 bytes input
     480 bytes output
CRC Errors
            Data CRC 0
```

```
        Ringword CRC 0
        InvRingWord  0
         PcsCodeWord 0
3/1 is OK Loopback No
Cable Length 50cm     Neighbor 1
Link Ok Yes Sync Ok Yes Link Active Yes
Changes to LinkOK 1
Five minute input rate  0 bytes/sec
Five minute output rate 15 bytes/sec
     1509354 bytes input
     27853 bytes output
CRC Errors
            Data CRC 0
        Ringword CRC 0
        InvRingWord  0
         PcsCodeWord 0
3/2 is OK Loopback No
Cable Length 50cm     Neighbor 2
Link Ok Yes Sync Ok Yes Link Active Yes
Changes to LinkOK 3
Five minute input rate  0 bytes/sec
Five minute output rate 0 bytes/sec
     240 bytes input
     160 bytes output
CRC Errors
            Data CRC 118
        Ringword CRC 74
        InvRingWord  125
         PcsCodeWord 373
```

Software Loopback: Examples

In a stack with three members, stack cables connect all the members:

```
# show switch stack-ports summary
  #
 Sw#/Port#  Port    Neighbor  Cable     Link  Link    Sync   #Changes   In
            Status            Length    OK    Active  OK     To LinkOK  Loopback
  --------  ------  --------  --------  ----  ------  ----  ---------   --------
    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

  # show switch stack-ports summary
  #
 Sw#/Port#  Port    Neighbor  Cable     Link  Link    Sync  #Changes   In
            Status            Length    OK    Active  OK    To LinkOK  Loopback

  --------  ------  --------  --------  ----  ------  ----  ---------  --------
    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:

```
# show sw stack-ports summary
  #
 Sw#/Port#  Port     Neighbor  Cable     Link  Link    Sync  #Changes   In
            Status             Length    OK    Active  OK    To LinkOK  Loopback
  --------  ------  --------  --------  ----  ------  ----  ---------  --------
    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:

```
# show switch stack-ports summary
  #
 Sw#/Port#  Port     Neighbor  Cable     Link  Link    Sync  #Changes   In
            Status             Length    OK    Active  OK    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 with Connected Stack Cables: Examples

- 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.

  ```
  # show switch stack-ports summary
    #
   Sw#/Port#  Port     Neighbor  Cable     Link  Link    Sync  #Changes   In
              Status             Length    OK    Active  OK    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
  - Stack ports with a cable that works properly

  ```
  # show switch stack-ports summary
    #
   Sw#/Port#  Port     Neighbor  Cable     Link  Link    Sync  #Changes   In
              Status             Length    OK    Active  OK    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.

Software Loopback with no Connected Stack Cable: Example

```
# show switch stack-ports summary
  #
 Sw#/Port#  Port    Neighbor  Cable     Link  Link    Sync  #Changes   In
            Status            Length    OK    Active  OK    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
```

Finding a Disconnected Stack Cable: Example

Stack 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:

```
# show switch stack-ports summary
  #
 Sw#/Port#  Port    Neighbor  Cable     Link  Link    Sync  #Changes   In
            Status            Length    OK    Active  OK    To LinkOK  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:

```
  # show switch stack-ports summary
  #
 Sw#/Port#  Port    Neighbor  Cable     Link  Link    Sync  #Changes   In
            Status            Length    OK    Active  OK    To LinkOK  Loopback
  --------  ------  --------  --------  ----  ------  ----  ---------  --------
    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 stack 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 Stack Ports: Example

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

This is the port status:

```
# show switch stack-ports summary
  #
 Sw#/Port#  Port     Neighbor  Cable     Link  Link    Sync  #Changes   In
            Status             Length    OK    Active  OK    To LinkOK  Loopback
  --------  ------   --------  --------  ----  ------  ----  ---------  --------
    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.

Additional References for Switch Stacks

Related Documents

Related Topic	Document Title
Cabling and powering on a switch stack.	*Cisco Catalyst 9200 Series Switches Hardware Installation Guide*
SGACL High Availability	"Cisco TrustSec SGACL High Availability" module of the *Cisco TrustSec Switch Configuration Guide*

Error Message Decoder

Description	Link
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

Standards and RFCs

Standard/RFC	Title
None	—

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and , use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature History for Switch Stacks

This table provides release and related information for features explained in this module.

These features are available on all releases subsequent to the one they were introduced in, unless noted otherwise.

Release	Feature	Feature Information
Cisco IOS XE Fuji 16.9.1	Switch Stack	A switch stack can have up to eight stacking-capable switches connected through their StackWise ports. The stack members work together as a unified system. Layer 2 and Layer 3 protocols present the entire switch stack as a single entity to the network.
Cisco IOS XE Amsterdam 17.2.1	Switch Stack	C9200-24PB and C9200-48PB switch models can only be stacked with each other and not with other models of the Cisco Catalyst 9200 Series Switches.
Cisco IOS XE Amsterdam 17.3.1	Switch Stack	A new command **show switch stack-ports detail** was introduced to display detailed information on the stack link of each stack member.

Use the Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to https://cfnng.cisco.com/. An account on Cisco.com is not required