Configuring EtherChannel and Link State Tracking

This chapter describes how to use the command-line interface (CLI) to configure EtherChannel on the Layer 2 or Layer 3 interfaces of the switch. It also provides guidelines, procedures, and configuration examples.

EtherChannel provides automatic recovery for the loss of a link by redistributing the load across the remaining links. If a link fails, EtherChannel redirects traffic from the failed link to the remaining links in the channel without intervention. This chapter also describes how to configure link-state tracking.

This chapter includes the following major sections:

> **Note** The commands in the following sections can be used on all Ethernet interfaces of the switch, including the uplink ports on the supervisor engine.

> **Note** For complete syntax and usage information for the switch commands used in this chapter, see the Cisco IOS Command Reference Guides for the Catalyst 4500 Series Switch.
>
> If a command is not in the *Cisco Catalyst 4500 Series Switch Command Reference* , you can locate it in the Cisco IOS Master Command List, All Releases.

About EtherChannel

EtherChannel bundles up to eight individual Ethernet links into a single logical ink that provides an aggregate bandwidth of up to 800 Mbps (Fast EtherChannel), 8 Gbps (Gigabit EtherChannel), or 80 Gbps (10 Gigabit EtherChannel) between a Catalyst 4500 or 4500X Series Switch and another switch or host.

Note   Because some linecards have a maximum bandwidth capacity toward the backplane, they can limit the aggregate bandwidth of an EtherChannel when all the EtherChannel members belong to the same linecard.

The switch supports a maximum of 254 EtherChannels on standalone switches, and 256 EtherChannels in virtual switching system (VSS) mode. You can form an EtherChannel with up to eight compatibly configured Ethernet interfaces across modules on a Catalyst 4500 series switch. All interfaces in each EtherChannel must be the same speed and must be configured as either Layer 2 or Layer 3 interfaces.

Note   The network device to which a Catalyst 4500 series switch is connected may impose its own limits on the number of interfaces in an EtherChannel.

If a segment within an EtherChannel fails, traffic previously carried over the failed link switches to the remaining segments within the EtherChannel. When the segment fails, an SNMP trap is sent, identifying the switch, the EtherChannel, and the failed link. Inbound broadcast and multicast packets on one segment in an EtherChannel are blocked from returning on any other segment of the EtherChannel.

Note   The port channel link failure switchover for the Catalyst 4500 series switch was measured at 50 milliseconds, which provides SONET-like link failure switchover time.

These subsections describe how EtherChannel works:

- Port Channel Interfaces, page 26-2
- Configuring EtherChannels, page 26-2
- Load Balancing, page 26-6

Port Channel Interfaces

Each EtherChannel has a numbered port channel interface. A configuration applied to the port channel interface affects all physical interfaces assigned to that interface.

Note   QoS does not propagate to members. The defaults, QoS cos = 0 and QoS dscp = 0, apply on the port channel. Input or output policies applied on individual interfaces are ignored.

After you configure an EtherChannel, the configuration that you apply to the port channel interface affects the EtherChannel; the configuration that you apply to the physical interfaces affects only the interface where you apply the configuration. To change the parameters of all ports in an EtherChannel, apply configuration commands to the port channel interface (such commands can be STP commands or commands to configure a Layer 2 EtherChannel as a trunk).

Configuring EtherChannels

These subsections describe how EtherChannels are configured:

- EtherChannel Configuration Overview, page 26-3
- Manual EtherChannel Configuration, page 26-3

- PAgP EtherChannel Configuration, page 26-4
- IEEE 802.3ad LACP EtherChannel Configuration, page 26-5

EtherChannel Configuration Overview

You can configure EtherChannels manually or use the Port Aggregation Control Protocol (PAgP) or the Link Aggregation Control Protocol (LACP) (Cisco IOS Release 12.2(31)SGA and later), to form EtherChannels. The EtherChannel protocols allow ports with similar characteristics to form an EtherChannel through dynamic negotiation with connected network devices. PAgP is a Cisco-proprietary protocol and LACP is defined in IEEE 802.3ad.

PAgP and LACP do not interoperate. Ports configured to use PAgP cannot form EtherChannels with ports configured to use LACP and vice versa.

Table 26-1 lists the user-configurable EtherChannel modes.

Table 26-1   EtherChannel Modes

Mode	Description
on	Mode that forces the LAN port to channel unconditionally. In the **on** mode, a usable EtherChannel exists only when a LAN port group in the **on** mode is connected to another LAN port group in the **on** mode. Because ports configured in the **on** mode do not negotiate, there is no negotiation traffic between the ports.
auto	PAgP mode that places a LAN port into a passive negotiating state in which the port responds to PAgP packets it receives but does not initiate PAgP negotiation.
desirable	PAgP mode that places a LAN port into an active negotiating state in which the port initiates negotiations with other LAN ports by sending PAgP packets.
passive	LACP mode that places a port into a passive negotiating state in which the port responds to LACP packets it receives but does not initiate LACP negotiation.
active	LACP mode that places a port into an active negotiating state in which the port initiates negotiations with other ports by sending LACP packets.

Manual EtherChannel Configuration

Manually configured EtherChannel ports do not exchange EtherChannel protocol packets. A manually configured EtherChannel forms only when you configure all ports compatibly in the EtherChannel.

Auto-LAG

You can automatically create EtherChannels on ports connected to a switch, using auto-LAG. By default, auto-LAG is disabled globally, but is enabled on all port interfaces. To configure auto-LAG on your switch, ensure that you enable auto-LAG globally.

When auto-LAG is enabled globally:

- All port interfaces participate in the creation of auto EtherChannels if the partner port interfaces have EtherChannel configured on them. For more information, see Table 26-1Supported auto-LAG configuration on actor and partner devices, page 26-4.
- Ports that are already part of manual EtherChannels cannot participate in creation of auto EtherChannels.

- When auto-LAG is disabled on a port interface that is already a part of an auto created EtherChannel, the port interface will unbundle from the auto EtherChannel.

The following table shows the supported auto-LAG configurations between the actor and partner devices:

Table 26-1   Supported auto-LAG configuration on actor and partner devices

Actor/Partner	Active	Passive	Auto
Active	Yes	Yes	Yes
Passive	Yes	No	Yes
Auto	Yes	Yes	Yes

On disabling auto-LAG globally, all auto created Etherchannels become manual EtherChannels.You cannot add any configurations in an existing auto created EtherChannel. To add an new configuration, convert the auto-created EtherChannel into a manual EtherChannel using the **port-channel** *channel-number* **persistent** command.

Note   Auto-LAG uses the LACP protocol to create an auto EtherChannel. Only one EtherChannel can be automatically created with the unique partner devices.

Auto-LAG Configuration Guidelines

Follow these guidelines when configuring auto-LAG:

- When auto-LAG is enabled globally and on the port interface, and you do not want the port interface to become a member of the auto EtherChannel, disable the auto-LAG on the port interface.

- A port interface does not bundle to an auto EtherChannel when it is already a member of a manual EtherChannel. To allow it to bundle with the auto EtherChannel, first unbundle the manual EtherChannel on the port interface.

- When auto-LAG is enabled and an auto EtherChannel is created, you can create multiple EtherChannels manually with the same partner device. But by default, the port tries to create auto EtherChannel with the partner device.

- Auto-LAG is supported only on Layer 2 EtherChannels. It is not supported on Layer 3 interfaces and Layer 3 EtherChannels.

PAgP EtherChannel Configuration

PAgP supports the automatic creation of EtherChannels by exchanging PAgP packets between LAN ports. PAgP packets are exchanged only between ports in **auto** and **desirable** modes.

The protocol learns the capabilities of LAN port groups dynamically and informs the other LAN ports. Once PAgP identifies correctly matched Ethernet links, it facilitates grouping the links into an EtherChannel. The EtherChannel is then added to the spanning tree as a single bridge port.

Both the **auto** and **desirable** modes allow PAgP to negotiate between LAN ports to determine if they can form an EtherChannel, based on criteria such as port speed and trunking state. Layer 2 EtherChannels also use VLAN numbers.

LAN ports can form an EtherChannel when they are in different PAgP modes if the modes are compatible. For example:

- A LAN port in **desirable** mode can form an EtherChannel successfully with another LAN port that is in **desirable** mode.

- A LAN port in **desirable** mode can form an EtherChannel with another LAN port in **auto** mode.

- A LAN port in **auto** mode cannot form an EtherChannel with another LAN port that is also in **auto** mode because neither port initiates negotiation.

IEEE 802.3ad LACP EtherChannel Configuration

Cisco IOS Release 12.2(31)SGA and later releases support IEEE 802.3ad LACP EtherChannels. LACP supports the automatic creation of EtherChannels by exchanging LACP packets between LAN ports. LACP packets are exchanged only between ports in **passive** and **active** modes.

The protocol learns the capabilities of LAN port groups dynamically and informs the other LAN ports. Once LACP identifies correctly matched Ethernet links, it facilitates grouping the links into an EtherChannel. The EtherChannel is then added to the spanning tree as a single bridge port.

Both the **passive** and **active** modes allow LACP to negotiate between LAN ports to determine if they can form an EtherChannel, based on criteria such as port speed and trunking state. Layer 2 EtherChannels also use VLAN numbers.

LAN ports can form an EtherChannel when they are in different LACP modes as long as the modes are compatible. For example:

- A LAN port in **active** mode can form an EtherChannel successfully with another LAN port that is in **active** mode.

- A LAN port in **active** mode can form an EtherChannel with another LAN port in **passive** mode.

- A LAN port in **passive** mode cannot form an EtherChannel with another LAN port that is also in **passive** mode, because neither port initiates negotiation.

LACP uses the following parameters:

- LACP system priority—You may configure an LACP system priority on each switch running LACP. The system priority can be configured automatically or through the CLI. See the "Configuring the LACP System Priority and System ID" section on page 26-16. LACP uses the system priority with the switch MAC address to form the system ID and also during negotiation with other systems.

  ✎
  Note   The LACP system ID is the combination of the LACP system priority value and the MAC address of the switch.

- LACP port priority—You must configure an LACP port priority on each port configured to use LACP. The port priority can be configured automatically or through the CLI. See the "Configuring Layer 2 EtherChannels" section on page 26-11. LACP uses the port priority with the port number to form the port identifier.

- LACP administrative key—LACP automatically configures an administrative key value equal to the channel group identification number on each port configured to use LACP. The administrative key defines the ability of a port to aggregate with other ports. A port's ability to aggregate with other ports is determined by these factors:

  – Port physical characteristics, such as data rate, duplex capability, and point-to-point or shared medium

  – Configuration restrictions that you establish

LACP tries to configure the maximum number of compatible ports in an EtherChannel up to the maximum allowed by the hardware (eight ports). If a port cannot be actively included in a channel, it is not included automatically if a channeled port fails.

You can change the LACP timer rate to modify the duration of the LACP timeout. Use the `lacp rate` command to set the rate at which LACP control packets are sent to an LACP-supported interface. You can change the timeout rate from the default rate (30 seconds) to the fast rate (1 second). This command is supported only on LACP-enabled interfaces. To configure the LACP fast time rate, see Configuring the LACP Fast Timer Rate.

ISSU and stateful switchover cannot be guaranteed with LACP fast timers.

Note   Standby and "sub-channeling" are not supported in LACP and PAgP.

Load Balancing

EtherChannel can balance the traffic load across the links in the channel by reducing part of the binary pattern formed from the addresses or ports in the frame to a numerical value that selects one of the links in the channel. To balance the load, EtherChannel uses MAC addresses, IP addresses, or Layer 4 port numbers, and either the message source or message destination, or both.

Use the option that provides the greatest variety in your configuration. For example, if the traffic on a channel is going only to a single MAC address, using the destination MAC address always chooses the same link in the channel; using source addresses or IP addresses might result in better load balancing.

Note   Load balancing can only be configured globally. As a result, all channels (manually configured, PagP, or LACP) use the same load-balancing method.

For additional information on load balancing, see the "Configuring EtherChannel Load Balancing" section on page 26-18.

EtherChannel Configuration Guidelines and Restrictions

If improperly configured, some EtherChannel interfaces are disabled automatically to avoid network loops and other problems. Follow these guidelines and restrictions to avoid configuration problems:

- All Ethernet interfaces on all modules support EtherChannel (maximum of eight interfaces) with no requirement that interfaces be physically contiguous or on the same module.

- Configure all interfaces in an EtherChannel to operate at the same speed and duplex mode.

- Enable all interfaces in an EtherChannel. Disabling an interface in an EtherChannel is treated as a link failure, and its traffic is transferred to one of the remaining interfaces in the EtherChannel.

- An EtherChannel does not form if one of the interfaces is a Switched Port Analyzer (SPAN) destination port.

- For Layer 3 EtherChannels:

  - Assign Layer 3 addresses to the port channel logical interface, not to the physical interfaces in the channel.

- For Layer 2 EtherChannels:

- Assign all interfaces in the EtherChannel to the same VLAN, or configure them as trunks.

- If you configure an EtherChannel from trunk interfaces, verify that the trunking mode and the native VLAN is the same on all the trunks. Interfaces in an EtherChannel with different trunk modes or different native VLANs can have unexpected results.

- An EtherChannel supports the same allowed range of VLANs on all the interfaces in a trunking Layer 2 EtherChannel. If the allowed ranges differ for selected interface differ, they do not form an EtherChannel.

- Interfaces with different Spanning Tree Protocol (STP) port path costs can form an EtherChannel as long they are otherwise compatibly configured. Setting different STP port path costs does not make interfaces incompatible for the formation of an EtherChannel.

- After you configure an EtherChannel, any configuration that you apply to the port channel interface affects the EtherChannel; any configuration that you apply to the physical interfaces affects only the interface you configure.

  Storm Control is an exception to this rule. For example, you cannot configure Storm Control on some of the members of an EtherChannel; Storm Control must be configured on all or none of the ports. If you configure Storm Control on only some of the ports, those ports are dropped from the EtherChannel interface (put in suspended state). You should configure Storm Control at the port channel interface level, and not at the physical interface level.

- A physical interface with port security enabled can join a Layer 2 EtherChannel only if port security is also enabled on the EtherChannel; otherwise the command is rejected by the CLI.

- You cannot configure a 802.1X port in an EtherChannel.

Configuring EtherChannel

These sections describe how to configure EtherChannel:

Note   Ensure that the interfaces are configured correctly. See the "EtherChannel Configuration Guidelines and Restrictions" section on page 26-6.

Configuring Layer 3 EtherChannels

To configure Layer 3 EtherChannels, create the port channel logical interface and then put the Ethernet interfaces into the port channel.

These sections describe Layer 3 EtherChannel configuration:

- Creating Port Channel Logical Interfaces, page 26-8
- Configuring Physical Interfaces as Layer 3 EtherChannels, page 26-8

Creating Port Channel Logical Interfaces

> **Note** To move an IP address from a physical interface to an EtherChannel, you must delete the IP address from the physical interface before configuring it on the port channel interface.

To create a port channel interface for a Layer 3 EtherChannel, perform this task:

	Command	Purpose
Step 1	Switch(config)# **interface port-channel** *port_channel_number*	Creates the port channel interface. The value for *port_channel_number* can range from 1 to 254 for a standalone switch, and from 1 to 256 for switches in VSS mode.
Step 2	Switch(config-if)# **ip address** *ip_address mask*	Assigns an IP address and subnet mask to the EtherChannel.
Step 3	Switch(config-if)# **end**	Exits configuration mode.
Step 4	Switch# **show running-config interface port-channel** *port_channel_number*	Verifies the configuration.

This example shows how to create port channel interface 1:

```
Switch# configure terminal
Switch(config)# interface port-channel 1
Switch(config-if)# ip address 172.32.52.10 255.255.255.0
Switch(config-if)# end
```

This example shows how to verify the configuration of port channel interface 1:

```
Switch# show running-config interface port-channel 1
Building configuration...

Current configuration:
!
interface Port-channel1
 ip address 172.32.52.10 255.255.255.0
 no ip directed-broadcast
end

Switch#
```

Configuring Physical Interfaces as Layer 3 EtherChannels

To configure physical interfaces as Layer 3 EtherChannels, perform this task for each interface:

	Command	Purpose
Step 1	`Switch(config)#` **`interface`** `{`**`fastethernet`** `|` **`gigabitethernet`** `|` **`tengigabitethernet`**`}` *`slot/port`*	Selects a physical interface to configure.
Step 2	`Switch(config-if)#` **`no switchport`**	Makes this a Layer 3 routed port.
Step 3	`Switch(config-if)#` **`no ip address`**	Ensures that no IP address is assigned to the physical interface.
Step 4	`Switch(config-if)#` **`channel-group`** *`port_channel_number`* **`mode`** `{`**`active`** `|` **`on`** `|` **`auto`** `|` **`passive`** `|` **`desirable`**`}`	Configures the interface in a port channel and specifies the PAgP or LACP mode. If you use PAgP, enter the keywords **auto** or **desirable**. If you use LACP, enter the keywords **active** or **passive**.
Step 5	`Switch(config-if)#` **`end`**	Exits configuration mode.
Step 6	`Switch#` **`show running-config interface port-channel`** *`port_channel_number`* `Switch#` **`show running-config interface`** `{`**`fastethernet`** `|` **`gigabitethernet`** `|` **`tengigabitethernet`**`}` *`slot/port`* `Switch#` **`show interfaces`** `{`**`fastethernet`** `|` **`gigabitethernet`** `|` **`tengigabitethernet`**`}` *`slot/port`* **`etherchannel`** `Switch#` **`show etherchannel 1 port-channel`**	Verifies the configuration.

This example shows how to configure Fast Ethernet interfaces 5/4 and 5/5 into port channel 1 with PAgP mode **desirable**:

```
Switch# configure terminal
Switch(config)# interface range fastethernet 5/4 - 5 (Note: Space is mandatory.)
Switch(config-if)# no switchport
Switch(config-if)# no ip address
Switch(config-if)# channel-group 1 mode desirable
Switch(config-if)# end
```

Note   See the "Configuring a Range of Interfaces" section on page 8-4 for information about the **range** keyword.

The following two examples show how to verify the configuration of Fast Ethernet interface 5/4:

```
Switch# show running-config interface fastethernet 5/4
Building configuration...

Current configuration:
!
interface FastEthernet5/4
 no ip address
 no switchport
 no ip directed-broadcast
 channel-group 1 mode desirable
end

Switch# show interfaces fastethernet 5/4 etherchannel
Port state    = EC-Enbld Up In-Bndl Usr-Config
```

```
Channel group = 1           Mode = Desirable     Gcchange = 0
Port-channel  = Po1         GC   = 0x00010001    Pseudo-port-channel = Po1
Port indx     = 0           Load = 0x55

Flags:  S - Device is sending Slow hello.  C - Device is in Consistent state.
        A - Device is in Auto mode.        P - Device learns on physical port.
Timers: H - Hello timer is running.        Q - Quit timer is running.
        S - Switching timer is running.    I - Interface timer is running.

Local information:
                               Hello    Partner  PAgP     Learning  Group
Port       Flags State  Timers  Interval Count   Priority  Method   Ifindex
Fa5/4      SC    U6/S7           30s     1        128       Any      55

Partner's information:

          Partner              Partner           Partner            Partner Group
Port      Name                 Device ID         Port     Age Flags Cap.
Fa5/4     JAB031301            0050.0f10.230c    2/45      1s  SAC   2D

Age of the port in the current state: 00h:54m:52s

Switch#
```

This example shows how to verify the configuration of port channel interface 1 after the interfaces have been configured:

```
Switch# show etherchannel 1 port-channel

             Channel-group listing:
             ---------------------
Group: 1
------------

             Port-channels in the group:
             ----------------------
Port-channel: Po1
------------

Age of the Port-channel   = 01h:56m:20s
Logical slot/port   = 10/1            Number of ports = 2
GC                  = 0x00010001      HotStandBy port = null
Port state          = Port-channel L3-Ag Ag-Inuse

Ports in the Port-channel:

Index   Load    Port
------------------
  1     00      Fa5/6
  0     00      Fa5/7

Time since last port bundled:    00h:23m:33s    Fa5/6

Switch#
```

This example shows how to display a one-line summary per channel group:

```
Switch# show etherchannel summary
Flags: D - down          P - bundled in  port-channel
       I -  stand-alone  s - suspended
       H - Hot-standby (LACP only)
       R  - Layer3       S -  Layer2
       U - in use        f - failed to allocate  aggregator
```

```
            M - not in  use, minimum links not  met
            u - unsuitable for  bundling
            w - waiting to be  aggregated
            d - default  port

Number of channel-groups in use: 2
Number of aggregators:          2

Group   Port-channel  Protocol    Ports
------+-------------+-----------+-----------------------------------------------
1      Po1(SD)         LACP       Gi1/23(H)   Gi1/24(H)
Switch#
```

Configuring Layer 2 EtherChannels

To configure Layer 2 EtherChannels, configure the Ethernet interfaces with the **channel-group** command. This operation creates the port channel logical interface.

Note   Cisco IOS software creates port channel interfaces for Layer 2 EtherChannels when you configure Layer 2 Ethernet interfaces with the **channel-group** command.

To configure Layer 2 Ethernet interfaces as Layer 2 EtherChannels, perform this task for each interface:

	Command	**Purpose**
Step 1	Switch(config)# **interface** {**fastethernet** | **gigabitethernet** | **tengigabitethernet**} *slot/port*	Selects a physical interface to configure.
Step 2	Switch(config-if)# **channel-group** *port_channel_number* **mode** {**active** | **on** | **auto** | **passive** | **desirable**}	Configures the interface in a port channel and specifies the PAgP or LACP mode. If you use PAgP, enter the keywords **auto** or **desirable**. If you use LACP, enter the keywords **active** or **passive**.
Step 3	Switch(config-if)# **end**	Exits configuration mode.
Step 4	Switch# **show running-config interface** {**fastethernet** | **gigabitethernet**} *slot/port* Switch# **show interface** {**fastethernet** | **gigabitethernet** | **tengigabitethernet**} *slot/port* **etherchannel**	Verifies the configuration.

This example shows how to configure Fast Ethernet interfaces 5/6 and 5/7 into port channel 2 with PAgP mode **desirable**:

```
Switch# configure terminal
Switch(config)# interface range fastethernet 5/6 - 7 (Note: Space is mandatory.)
Switch(config-if-range)# channel-group 2 mode desirable
Switch(config-if-range)# end
Switch# end
```

Note   See the "Configuring a Range of Interfaces" section on page 8-4 for information about the **range** keyword.

This example shows how to verify the configuration of port channel interface 2:

```
Switch# show running-config interface port-channel 2
Building configuration...

Current configuration:
!
interface Port-channel2
 switchport access vlan 10
 switchport mode access
end

Switch#
```

The following two examples show how to verify the configuration of Fast Ethernet interface 5/6:

```
Switch# show running-config interface fastethernet 5/6
Building configuration...

Current configuration:
!
interface FastEthernet5/6
 switchport access vlan 10
 switchport mode access
 channel-group 2 mode desirable
end

Switch# show interfaces fastethernet 5/6 etherchannel
Port state     = EC-Enbld Up In-Bndl Usr-Config
Channel group = 1           Mode = Desirable     Gcchange = 0
Port-channel  = Po1         GC   = 0x00010001
Port indx     = 0           Load = 0x55

Flags:  S - Device is sending Slow hello.  C - Device is in Consistent state.
        A - Device is in Auto mode.        P - Device learns on physical port.
        d - PAgP is down.
Timers: H - Hello timer is running.        Q - Quit timer is running.
        S - Switching timer is running.    I - Interface timer is running.
Local information:
                               Hello   Partner  PAgP     Learning  Group
Port      Flags State  Timers  Interval Count   Priority  Method   Ifindex
Fa5/6     SC    U6/S7           30s      1       128       Any      56

Partner's information:

          Partner              Partner         Partner          Partner Group
Port      Name                 Device ID       Port     Age Flags Cap.
Fa5/6     JAB031301            0050.0f10.230c  2/47      18s SAC   2F

Age of the port in the current state: 00h:10m:57s
```

This example shows how to verify the configuration of port channel interface 2 after the interfaces have been configured:

```
Switch# show etherchannel 2 port-channel
               Port-channels in the group:
               ----------------------

Port-channel: Po2
------------

Age of the Port-channel   = 00h:23m:33s
Logical slot/port   = 10/2          Number of ports in agport = 2
GC                  = 0x00020001    HotStandBy port = null
```

```
Port state          = Port-channel Ag-Inuse

Ports in the Port-channel:

Index   Load   Port
-------------------
  1     00     Fa5/6
  0     00     Fa5/7

Time since last port bundled:    00h:23m:33s    Fa5/6

Switch#
```

Configuring LACP Standalone or Independent Mode

This feature is particularly relevant when a port (A) in a Layer 2 LACP EtherChannel is connected to an unresponsive port (B) on the peer. When LACP standalone is disabled on the EtherChannel, all traffic arriving on A is blocked (the default behavior on a switch). In some scenarios, you might want to allow management traffic on such ports. You can do this by enabling LACP standalone (or independent) mode.

Note   This **port-channel standalone-disable** command only applies to Layer 2 EtherChannels

Note   LACP Standalone Disable is enabled by default.

To configure the LACP Standalone or Independent mode, perform this task:

	Command	Purpose
Step 1	Switch(config)# **no port-channel standalone-disable**	Enables the LACP standalone or independent mode.
	Switch(config)# **port-channel standalone-disable**	Reverts to the default.
Step 2	Switch(config-if)# **end**	Exits configuration mode.
Step 3	Switch# **show running configuration** {**fastethernet** | **gigabitethernet**} *slot*/*port* **port-channel** *port_channel_number*	Verifies the configuration.

This example shows how to configure the LACP Standalone mode:

```
Switch# configure terminal
Switch(config)# interface port-channel 1
Switch(config-if)# switchport
Switch(config-if)# exit
Switch(config)# int gi3/1
Switch(config-if)# channel-group 1 mode active
Switch(config-if)# exit
Switch(config)# interface port-channel 1
Switch(config-if)# no port-channel standalone-disable
Ports of Po12 already in suspend (S) mode require a shut/no shut.
Switch(config-if)# end
```

This example shows how to verify the configuration of port channel interface 1:

```
Switch# show running-config interface port-channel 1
```

```
Building configuration...

Current configuration:
!
interface Port-channel1
  switchport
  no port-channel standalone-disable
end

Switch#
```

This example shows how to verify the state of port channel interface 1:

```
Switch# show etherchannel 1 port-channel
                Port-channels in the group:
                --------------------------
Port-channel: Po13    (Primary Aggregator)
------------
Age of the Port-channel  = 0d:00h:07m:57s
Logical slot/port   = 11/13           Number of ports = 0
Port state           = Port-channel Ag-Not-Inuse
Protocol             =   LACP
Port security        = Disabled
Standalone           = Enabled (independent mode)
Switch#
```

Configuring LACP Port Channel Min-links

Beginning in Cisco IOS Release 15.2(4)E, and Cisco IOS XE Release 3.8.0E, you can specify the minimum number of active ports that must be in the link-up state and bundled in an EtherChannel for the port channel interface to transition to the link-up state. Using EtherChannel min-links, you can prevent low-bandwidth LACP EtherChannels from becoming active. Port channel min-links also cause LACP EtherChannels to become inactive if they have too few active member ports to supply the required minimum bandwidth.

Note   For a Multichassis EtherChannel (MEC), the LACP *min-links* command argument defines the minimum number of physical links in each chassis for the MEC to be operational.

To configure the minimum number of links that are required for a port channel, perform the following task:

	Command	Purpose
Step 1	Switch(config)# **interface port-channel** *channel-number*	Enters interface configuration mode for a port-channel. The valid range for *channel-number* is 1 to 256.

	Command	Purpose
Step 2	Switch(config-if)# **port-channel min-links** *min-links-number*	Specifies the minimum number of member ports that must be in the link-up state and bundled in the EtherChannel for the port channel interface to transition to the link-up state.
		The valid range for *min-links-number* is 2 to 8.
		For a switch in VSS mode, when setting up min-links, ensure that the port-channel consists of the same number of min-links on the active switch and on the standby switch. For example, if you set min-links to 2 for a port-channel, ensure that there are 2 links, each, available on the active and the standby switch.
Step 3	Switch# **end**	Returns to privileged EXEC mode

This example shows how to configure LACP port channel min-links:

```
Switch# configure terminal
Switch(config)# interface port-channel 25
switch(config-if)# port-channel min-links 3
Switch# show etherchannel 25 summary
switch# end
```

For a switch in VSS mode, when the minimum links requirement on a port channel is not met, the local ports in the EtherChannel are assigned m state, and all traffic is redirected out through the virtual switch link, via the member ports of the port channel on the peer switch, for example:

```
Switch# show etherchannel 25 summary
Flags: D - down P - bundled in port-channel
I - stand-alone s - suspended
H - Hot-standby (LACP only)
R - Layer3 S - Layer2
U - in use N - not in use, no aggregation
f - failed to allocate aggregator
M - not in use, no aggregation due to minimum links not met
m- not in use, port not aggregated due to minimum links not met
u - unsuitable for bundling
w - waiting to be aggregated
d - default port
Number of channel-groups in use: 125
Number of aggregators: 125
Group Port-channel Protocol Ports
------+-------------+-----------+----------------------------------------------
25 Po25(RU) LACP Gi1/3/1(D) Gi1/3/2(m) Gi2/2/25(P) Gi2/2/26(P)
```

When the minimum links requirement is not met in standalone switches, the port channel is flagged and assigned the SM/SN or RM/RN state, for example:

```
Switch# show etherchannel 25 summary
Flags: D - down P - bundled in port-channel
I - stand-alone s - suspended
H - Hot-standby (LACP only)
R - Layer3 S - Layer2
U - in use N- not in use, no aggregration
f - failed to allocate aggregator
M - not in use, no aggregation due to minimum links not met
m- not in use, port not aggregated due to minimum links not met
u - unsuitable for bundling
```

```
w - waiting to be aggregated
d - default port
Number of channel-groups in use: 125
Number of aggregators: 125
Group Port-channel Protocol Ports
------+-------------+----------+-----------------------------------------------
25 Po25(RM) LACP Gi1/3/1(D) Gi1/3/2(D) Gi2/2/25(D) Gi2/2/26(w)
```

Configuring the LACP System Priority and System ID

The LACP system ID is the LACP system priority value combined with the MAC address of the switch.

To configure the LACP system priority and system ID, perform this task:

	Command	Purpose
Step 1	Switch(config)# **lacp system-priority** *priority_value*	(Optional for LACP) Sets the LACP system priority and system ID. Valid values are 1 through 65535. Higher numbers have lower priority. The default is 32768.
	Switch(config)# **no system port-priority**	Reverts to the default.
Step 2	Switch(config)# **end**	Exits configuration mode.
Step 3	Switch# **show lacp sys-id**	Verifies the configuration.

This example shows how to configure the LACP system priority:

```
Switch# configure terminal
Switch(config)# lacp system-priority 23456
Switch(config)# end
Switch# show module

Mod  Ports Card Type                               Model            Serial No.
----+-----+--------------------------------------+----------------+-----------
 1      2 1000BaseX (GBIC) Supervisor(active)     WS-X4014         JAB063808YZ
 2     48 10/100BaseTX (RJ45)                      WS-X4148-RJ      JAB0447072W
 3     48 10/100BaseTX (RJ45)V                     WS-X4148-RJ45V   JAE061704J6
 4     48 10/100BaseTX (RJ45)V                     WS-X4148-RJ45V   JAE061704ML

M MAC addresses                   Hw  Fw           Sw              Status
--+--------------------------------+---+-----------+----------------+---------
 1 0005.9a39.7a80 to 0005.9a39.7a81 2.1 12.1(12r)EW  12.1(13)EW(0.26) Ok
 2 0002.fd80.f530 to 0002.fd80.f55f 0.1                               Ok
 3 0009.7c45.67c0 to 0009.7c45.67ef 1.6                               Ok
 4 0009.7c45.4a80 to 0009.7c45.4aaf 1.6                               Ok
```

This example shows how to verify the configuration:

```
Switch# show lacp sys-id
23456,0050.3e8d.6400
Switch#
```

The system priority is displayed first, followed by the MAC address of the switch.

Configuring LACP Fast Rate Timer

You can change the LACP timer rate to modify the duration of the LACP timeout. Use the **lacp rate command** to set the rate at which LACP control packets are received by an LACP-supported interface. You can change the timeout rate from the default rate (30 seconds) to the fast rate (1 second). This command is supported only on LACP-enabled interfaces.

	Command	Purpose
Step 4	Switch# **configure terminal**	Enters configuration mode.
Step 5	Switch(config)# **interface** {**fastethernet** | **gigabitethernet** | **tengigabitethernet**} *slot/port*	Selects the interface to configure.
Step 6	Switch(config-if)# **lacp rate** {**normal** | **fast**}	Configures the rate at which LACP control packets are received by an LACP-supported interface. To reset the timeout rate to its default, use the **no lacp rate** command.

This example shows how to configure the LACP rate:

```
Switch# configure terminal
Switch(config)# interface gigabitEthernet 2/1
Switch(config-if)# lacp rate fast
Switch(config-if)# exit
```

The **show lacp internal** command displays similar output:

```
Flags: S - Device is requesting Slow LACPDUs
F - Device is requesting Fast LACPDUs
A - Device is in Active mode P - Device is in Passive mode
Channel group 25
LACP port Admin Oper Port Port
Port Flags State Priority Key Key Number State
Te1/49 FA bndl 32768 0x19 0x19 0x32 0x3F
Te1/50 FA bndl 32768 0x19 0x19 0x33 0x3F
Te1/51 FA bndl 32768 0x19 0x19 0x34 0x3F
Te1/52 FA bndl 32768 0x19 0x19 0x35 0x3F
```

The **show lacp counters** command displays similar output

```
Switch# show lacp counters
LACPDUs Marker Marker Response LACPDUs
Port Sent Recv Sent Recv Sent Recv Pkts Err
--------------------------------------------------------------------
Channel group: 24
Te1/1/27 2 2 0 0 0 0 0
Te2/1/25 2 2 0 0 0 0 0
```

Configuring Auto-LAG Globally

✎

Note   By default, auto-LAG is disabled globally, on your device.

To configure auto-LAG globally, perform this task:

	Command	Purpose
Step 1	Switch(config)# port-channel auto	Configures auto-LAG globally on the device.
Step 2	Switch(config)# **end**	Exits configuration mode.
Step 3	Switch# **show etherchannel auto**	Displays the EtherChannel created automatically.

Configuring Auto-LAG on a Port Interface

To configure auto-LAG globally, perform this task:

	Command	Purpose
Step 1	Switch(config)# interface *interface-id*	Specifies the port interface to be enabled for auto-LAG, and enters interface configuration mode.
Step 2	Switch(config-if)# channel-group auto	(Optional) Enables auto-LAG feature on individual port interface. By default, the auto-LAG feature is enabled on the port.
Step 3	Switch(config)# **end**	Exits configuration mode.
Step 4	Switch# **show etherchannel auto**	Displays the EtherChannel created automatically.

Configuring Persistence with Auto-LAG

To convert the automatically created EtherChannel back to a manual one, perform this task:

	Command	Purpose
Step 1	Switch# port-channel *channel-number* persistent	Converts the auto created EtherChannel into a manual one and allows you to add configuration on the EtherChannel.
Step 2	Switch# **show etherchannel summary**	Displays EtherChannel information.

Configuring EtherChannel Load Balancing

Note   Load balancing can only be configured globally. As a result, all channels (manually configured, PagP, or LACP) use the same load-balancing method.

To configure EtherChannel load balancing, perform this task:

	Command	Purpose
Step 1	Switch(config)# [**no**] **port-channel load-balance** {**src-mac** | **dst-mac** | **src-dst-mac** | **src-ip** | **dst-ip** | **src-dst-ip** | **src-port** | **dst-port** | **src-dst-port**}	Configures EtherChannel load balancing. Use the **no** keyword to return EtherChannel load balancing to the default configuration.
Step 2	Switch(config)# **end**	Exits configuration mode.
Step 3	Switch# **show etherchannel load-balance**	Verifies the configuration.

The load-balancing keywords indicate these values:

- **src-mac**—Source MAC addresses
- **dst-mac**—Destination MAC addresses
- **src-dst-mac**—Source and destination MAC addresses
- **src-ip**—Source IP addresses
- **dst-ip**—Destination IP addresses
- **src-dst-ip**—Source and destination IP addresses (Default)
- **src-port**—Source Layer 4 port
- **dst-port**—Destination Layer 4 port
- **src-dst-port**—Source and destination Layer 4 port

This example shows how to configure EtherChannel to use source and destination IP addresses:

```
Switch# configure terminal
Switch(config)# port-channel load-balance src-dst-ip
Switch(config)# end
Switch#
```

This example shows how to verify the configuration:

```
Switch# show etherchannel load-balance
EtherChannel Load-Balancing Configuration:
        src-dst-ip

EtherChannel Load-Balancing Addresses Used Per-Protocol:
Non-IP: Source XOR Destination MAC address
  IPv4: Source XOR Destination IP address
  IPv6: Source XOR Destination IP address
Switch#
```

Removing an Interface from an EtherChannel

To remove an Ethernet interface from an EtherChannel, perform this task:

	Command	Purpose
Step 1	Switch(config)# **interface** {**fastethernet** | **gigabitethernet** | **tengigabitethernet**} *slot/port*	Selects a physical interface to configure.
Step 2	Switch(config-if)# **no channel-group**	Removes the interface from the port channel interface.

	Command	Purpose
Step 3	`Switch(config-if)# end`	Exits configuration mode.
Step 4	`Switch# show running-config interface {fastethernet | gigabitethernet | tengigabitethernet} slot/port` `Switch# show interface {fastethernet | gigabitethernet | tengigabitethernet} slot/port etherchannel`	Verifies the configuration.

This example shows how to remove Fast Ethernet interfaces 5/4 and 5/5 from port channel 1:

```
Switch# configure terminal
Switch(config)# interface range fastethernet 5/4 - 5 (Note: Space is mandatory.)
Switch(config-if)# no channel-group 1
Switch(config-if)# end
```

Removing an EtherChannel

If you remove an EtherChannel, the member ports are shut down and removed from the channel group.

Note   If you want to change an EtherChannel from Layer 2 to Layer 3, or Layer 3 to Layer 2, you must remove the EtherChannel and recreate it in the desired configuration.

To remove an EtherChannel, perform this task:

	Command	Purpose
Step 1	`Switch(config)# no interface port-channel port_channel_number`	Removes the port channel interface.
Step 2	`Switch(config)# end`	Exits configuration mode.
Step 3	`Switch# show etherchannel summary`	Verifies the configuration.

This example shows how to remove port channel 1:

```
Switch# configure terminal
Switch(config)# no interface port-channel 1
Switch(config)# end
```

Displaying EtherChannel to a Virtual Switch System

Catalyst 4500 series switches support enhanced PAgP. If a Catalyst 4500 series switch is connected to a Catalyst 6500 series Virtual Switch System (VSS) by using a PAgP EtherChannel, the Catalyst 4500 series switch automatically serve as a VSS client, using enhanced PAgP on this EtherChannel for dual-active detection. This VSS client feature has no impact on the performance of Catalyst 4500 series switch and does not require any user configuration.

This section includes these topics:

- Understanding VSS Client, page 26-21
- Displaying EtherChannel Links to VSS, page 26-23

Understanding VSS Client

This section describes these topics:

Virtual Switch System

The Cisco Catalyst 6500 Series VSS 1440 allows for the combination of two Cisco Catalyst 6500 Series switches into a single, logical network entity from the network control-plane and management perspectives. Within the Cisco VSS, one chassis is designated as the active virtual switch, acting as the single management point of the entire system, and the other is designated as the standby virtual switch. There two chassis are bound together by a special link, called Virtual Switch Link (VSL), which carries the internal signaling and control information between them.

Dual-Active Scenarios

One of the failure scenarios in a VSS is called *dual-active*, which occurs when the VSL fails completely. Neither virtual switch knows of the other's status. From the perspective of the active virtual switch, the standby chassis is lost. The standby virtual switch also views the active chassis as failed and transitions to active state by using an SSO switchover. Two active virtual switches exist in the network with identical configurations, causing duplicate IP addresses and bridge identifiers. This scenario has adverse effects on the network topology and traffic if it persists.

Dual-Active Detection Using Enhanced PAgP

One method for detecting a dual-active scenario is based on enhanced PAgP (PAgP+). Specifically, the VSS sends regularly scheduled PAgP messages with Type-Length-Values (TLVs) containing the ID of the current active virtual switch (Figure 26-1). When the VSL fails completely, the standby virtual switch immediately sends asynchronous PAgP messages with TLVs containing its own ID on all port channels enabled for enhanced PAgP dual-active detection (Figure 26-2). The remote switch (the VSS client) connected to both VSS components by using EtherChannel links, compares every received active ID with its stored active ID. If they match, the remote switch sends TLVs containing its stored active ID back to the VSS in its regularly scheduled PAgP messages. If they do not match, the remote switch stores the new active ID and immediately transmits asynchronous PAgP messages with TLVs containing the new active ID. Upon receiving the new active ID from the remote switch, the original active virtual switch detects the dual-active scenario and takes appropriate actions.

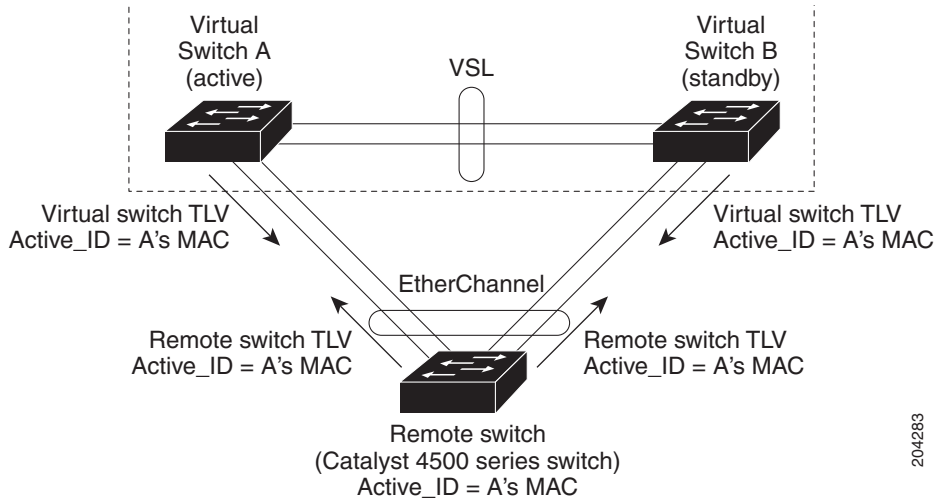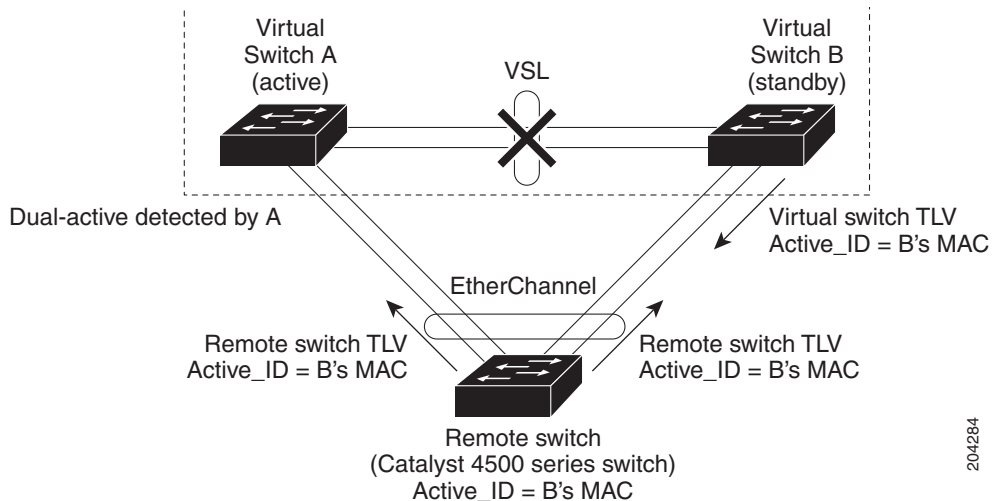*Figure 26-1   Enhanced PAgP in VSS Normal Operation*

Figure 26-2   Enhanced PAgP in VSS Dual-active Scenario

As a remote switch, the Catalyst 4500 series switch supports stateful VSS client. In particular, the ID of the current active virtual switch is synchronized from the active supervisor engine to the redundant supervisor engine of the Catalyst 4500 series switch. This ensures that dual-active detection is not disrupted even when the active supervisor engine switches over to the redundant supervisor engine.

Displaying EtherChannel Links to VSS

To display the dual-active detection capability of a configured PAgP port channel, enter the **show pagp** *port_channel_number* **dual-active** command.

The command provides the following information:

- A switch uses enhanced PAgP for dual-active detection.

  You should always see Yes after PAgP dual-active diction enabled on a Catalyst 4500 switch.

- The configured PAgP EtherChannel is connected to a Catalyst 6500 switch VSS.

  You see N/A below Partner Version if this EtherChannel is *not* connected to a VSS. Otherwise, you see the version of enhanced PAgP dual-active detection implemented in the VSS.

- This switch is capable of detecting dual-active scenarios in the connected VSS.

  You see Yes below Dual-Active Detect Capable if and only if the configured EtherChannel is connected to a Catalyst 6500 series VSS that uses the same version of enhanced PAgP dual-active detection.

Note   You can also see the name of the neighboring switch (Partner Name) and the ports to which this EtherChannel is connected (Partner Port).

If a Catalyst 4500 switch is connected to a Catalyst 6500 series VSS with the same version of enhanced PAgP dual-active detection, the switch can detect a dual-active scenario:

```
Switch# show pagp 1 dual-active
PAgP dual-active detection enabled: Yes
PAgP dual-active version: 1.1

Channel group 1
        Dual-Active      Partner         Partner         Partner
Port    Detect Capable   Name            Port            Version
Gi6/5   Yes              VSS             Gi1/8/1         1.1
Gi6/6   Yes              VSS             Gi2/8/1         1.1
```

If a Catalyst 4500 switch is not connected to a Catalyst 6500 series VSS, the switch cannot detect a dual-active scenario:

```
Switch# show pagp 1 dual-active
PAgP dual-active detection enabled: Yes
PAgP dual-active version: 1.1

Channel group 1
         Dual-Active     Partner              Partner   Partner
Port     Detect Capable  Name                 Port      Version
Gi6/5    No              Switch               Fa6/5     N/A
Gi6/6    No              Switch               Fa6/6     N/A
```

Understanding Link-State Tracking

Link-state tracking, also known as trunk failover, is a feature that binds the link state of multiple interfaces. For example, link-state tracking provides redundancy in the network when used with server NIC adapter teaming. When server network adapters are configured in a primary or secondary relationship known as teaming, if the link is lost on the primary interface, connectivity is transparently changed to the secondary interface.

Figure 26-3 on page 26-25 shows a network configured with link-state tracking. To enable link-state tracking, create a link-state group, and specify the interfaces that are assigned to the link-state group. An interface can be an aggregation of ports (an EtherChannel), a single physical port in access or trunk mode, or a routed port. In a link-state group, these interfaces are bundled together. The downstream interfaces are bound to the upstream interfaces. Interfaces connected to servers are referred to as downstream interfaces, and interfaces connected to distribution switches and network devices are referred to as upstream interfaces.

The configuration in Figure 26-3 ensures that the network traffic flow is balanced as follows:

- For links to switches and other network devices

  - Server 1 and server 2 use switch A for primary links and switch B for secondary links.

  - Server 3 and server 4 use switch B for primary links and switch A for secondary links.

- Link-state group 1 on switch A

  - Switch A provides primary links to server 1 and server 2 through link-state group 1. Port 1 is connected to server 1, and port 2 is connected to server 2. Port 1 and port 2 are the downstream interfaces in link-state group 1.

  - Port 5 and port 6 are connected to distribution switch 1 through link-state group 1. Port 5 and port 6 are the upstream interfaces in link-state group 1.

- Link-state group 2 on switch A

  - Switch A provides secondary links to server 3 and server 4 through link-state group 2. Port 3 is connected to server 3, and port 4 is connected to server 4. Port 3 and port 4 are the downstream interfaces in link-state group 2.

  - Port 7 and port 8 are connected to distribution switch 2 through link-state group 2. Port 7 and port 8 are the upstream interfaces in link-state group 2.

- Link-state group 2 on switch B

  - Switch B provides primary links to server 3 and server 4 through link-state group 2. Port 3 is connected to server 3, and port 4 is connected to server 4. Port 3 and port 4 are the downstream interfaces in link-state group 2.

  - Port 5 and port 6 are connected to distribution switch 2 through link-state group 2. Port 5 and port 6 are the upstream interfaces in link-state group 2.

- Link-state group 1 on switch B

  - Switch B provides secondary links to server 1 and server 2 through link-state group 1. Port 1 is connected to server 1, and port 2 is connected to server 2. Port 1 and port 2 are the downstream interfaces in link-state group 1.

  - Port 7 and port 8 are connected to distribution switch 1 through link-state group 1. Port 7 and port 8 are the upstream interfaces in link-state group 1.

In a link-state group, the upstream ports can become unavailable or lose connectivity because the distribution switch or router fails, the cables are disconnected, or the link is lost. These are the interactions between the downstream and upstream interfaces when link-state tracking is enabled:

- If any of the upstream interfaces are in the link-up state, the downstream interfaces can change to or remain in the link-up state.

- If all of the upstream interfaces become unavailable, link-state tracking automatically puts the downstream interfaces in the error-disabled state. Connectivity to and from the servers is automatically changed from the primary server interface to the secondary server interface.

As an example of a connectivity change from link-state group 1 to link-state group 2 on switch A, see Figure 26-3 on page 26-25. If the upstream link for port 6 is lost, the link states of downstream ports 1 and 2 do not change. However, if the link for upstream port 5 is also lost, the link state of the downstream ports changes to the link-down state. Connectivity to server 1 and server 2 is then changed from link-state group1 to link-state group 2. The downstream ports 3 and 4 do not change state because they are in link-group 2.

- If the link-state group is configured, link-state tracking is disabled, and the upstream interfaces lose connectivity, the link states of the downstream interfaces remain unchanged. The server does not recognize that upstream connectivity has been lost and does not failover to the secondary interface.

You can recover a downstream interface link-down condition by removing the failed downstream port from the link-state group. To recover multiple downstream interfaces, disable the link-state group.

Figure 26-3   Typical Link-State Tracking Configuration
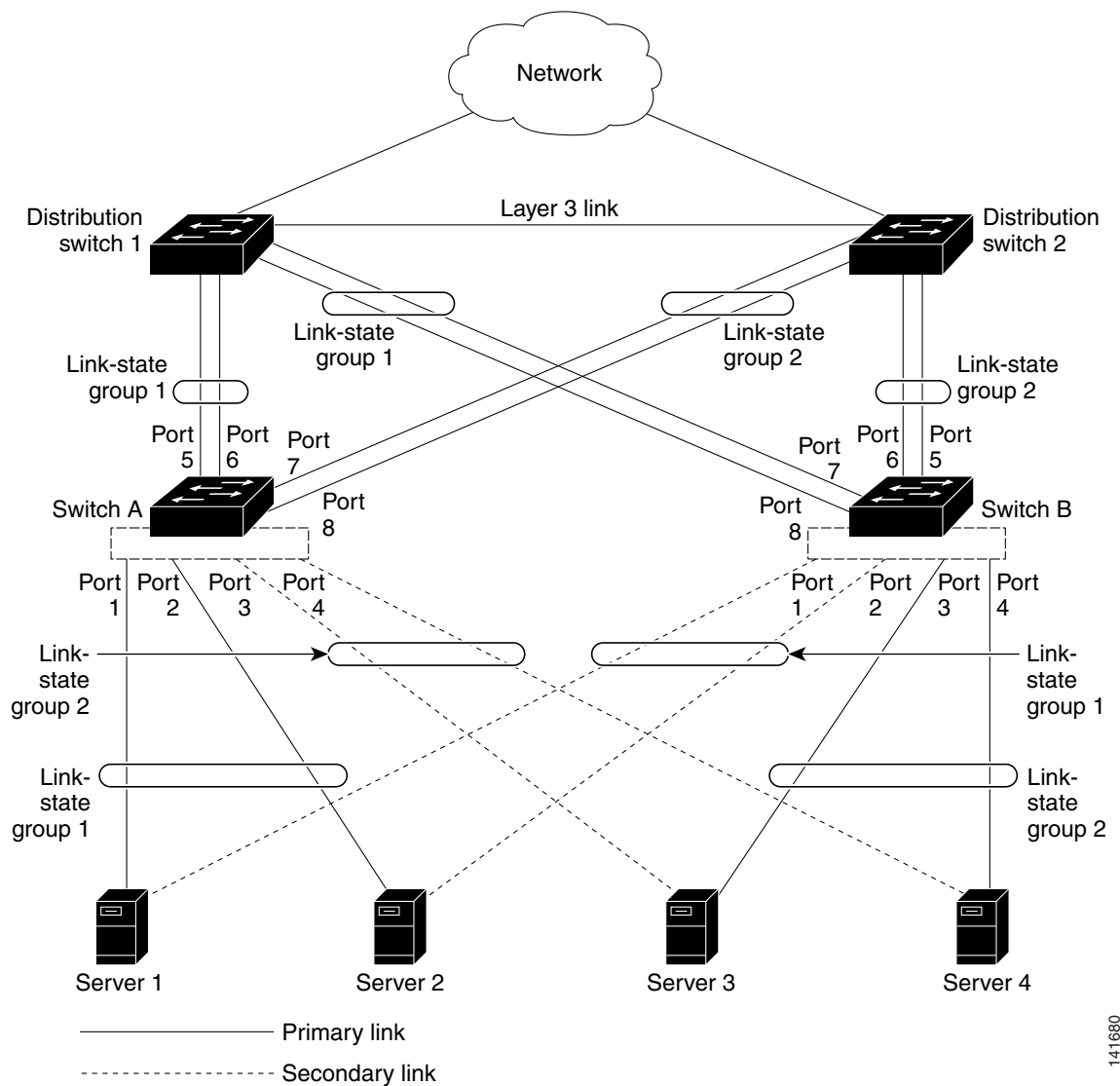
Configuring Link-State Tracking

These sections describe how to configure link-state tracking ports:

Default Link-State Tracking Configuration

No link-state groups are defined, and link-state tracking is not enabled for any group.

Link-State Tracking Configuration Guidelines

Follow these guidelines to avoid configuration problems:

- An interface that is defined as an upstream interface cannot also be defined as a downstream interface in the same or different link-state group. The reverse is also true.

- We recommend that you add the upstream interfaces to the link state group before adding the downstream interfaces. it is because when a downstream interface is added to a link state group without an upstream interface, the downstream interface is put in error-disabled state until an upstream interfaces is added to the group.

- An interface cannot be a member of more than one link-state group.

- You can configure up to 20 link-state groups per switch.

- If a SPAN destination port is configured as a downstream interface, it is error disabled when all upstream interfaces in its group are down. When an upstream interface is configured as a SPAN destination port, it is considered as a link down event on the interface.

Configuring Link-State Tracking

To configure a link-state group and to assign an interface to a group, perform this task beginning in privileged EXEC mode:

	Command	Purpose
Step 1	Switch# **configure terminal**	Enters global configuration mode.
Step 2	Switch(config)# **link state track** *number*	Creates a link-state group, and enables link-state tracking. The group number range is 1 to 20; the default is 1.
Step 3	Switch(config)# **interface** *interface-id*	Specifies a physical interface or range of interfaces to configure, and enters interface configuration mode.
		Valid interfaces include switch ports in access mode or trunk mode (IEEE 802.1q), routed ports, or multiple ports bundled into an EtherChannel interface (static or LACP), in trunk mode.

	Command	Purpose
Step 4	Switch(config-if)# **link state group** [*number*] {**upstream** | **downstream**}	Specifies a link-state group, and configures the interface as either an **upstream** or **downstream** interface in the group. The group number range is from 1 to 20; the default is 1.
Step 5	Switch(config-if)# **end**	Returns to privileged EXEC mode.
Step 6	Switch# **show running-config**	Verifies your entries.
Step 7	Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

This example shows how to create a link-state group and configure the interfaces:

```
Switch# configure terminal
Switch(config)# link state track 1
Switch(config)# interface gigabitethernet3/1
Switch(config-if)# link state group 1 upstream
Switch(config-if)# interface gigabitethernet3/3
Switch(config-if)# link state group 1 upstream
Switch(config-if)# interface gigabitethernet3/5
Switch(config-if)# link state group 1 downstream
Switch(config-if)# interface gigabitethernet3/7
Switch(config-if)# link state group 1 downstream
Switch(config-if)# end
```

To disable a link-state group, use the **no link state track** *number* global configuration command.

Displaying Link-State Tracking Status

To display the link-state group information, enter the **show link state group** command. Enter this command without keywords to display information about all link-state groups.

Enter the group number to display information specific to the group. Enter the **detail** keyword to display detailed information about the group.

it is an example of output from the **show link state group 1** command:

```
Switch> show link state group 1
Link State Group: 1 Status: Enabled, Down
```

it is an example of output from the **show link state group detail** command:

```
Switch> show link state group detail
(Up):Interface up (Dwn):Interface Down (Dis):Interface disabled
Link State Group: 1 Status: Enabled, Down
Upstream Interfaces : Gi3/5(Dwn) Gi3/6(Dwn)
Downstream Interfaces : Gi3/1(Dis) Gi3/2(Dis) Gi3/3(Dis) Gi3/4(Dis)
Link State Group: 2 Status: Enabled, Down
Upstream Interfaces : Gi3/15(Dwn) Gi3/16(Dwn) Gi3/17(Dwn)
Downstream Interfaces : Gi3/11(Dis) Gi3/12(Dis) Gi3/13(Dis) Gi3/14(Dis)
(Up):Interface up (Dwn):Interface Down (Dis):Interface disabled
```