Cisco Catalyst 6500 Series
Virtual Switching System (VSS) 1440

Table of Contents

Introduction

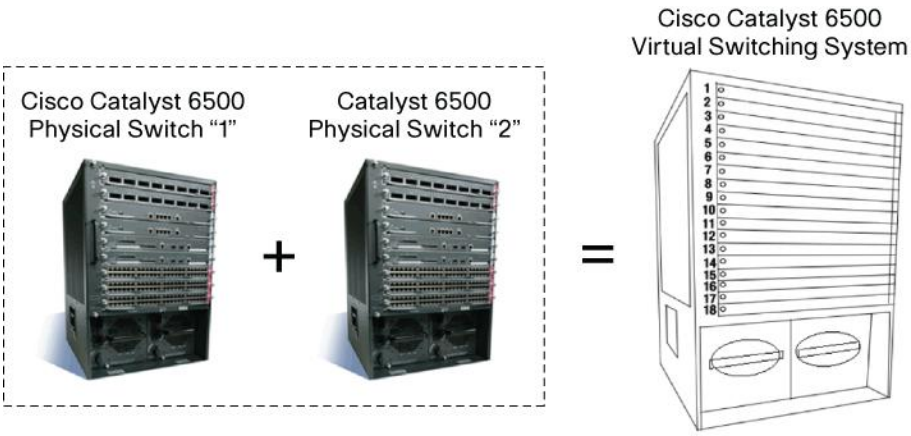
The Cisco® Catalyst® 6500 Series Virtual Switching System (VSS) 1440 is an exciting innovation on the Cisco Catalyst 6500 Series Switches that effectively allows the clustering of two or more physical chassis together into a single, logical entity. This technology allows for new enhancements in all areas of network design, including high availability, scalability, management, and maintenance.

This paper analyzes the Cisco VSS technology, including its benefits and requirements, and highlights potential deployment caveats you should consider before deploying Cisco Virtual Switching System.

Cisco Catalyst 6500 Series Virtual Switching System 1440: An Overview

The Cisco Catalyst 6500 Series Virtual Switching System (VSS) 1440 allows for the merging of two physical Cisco Catalyst 6500 Series Switches together into a single, logically managed entity. Figure 1 graphically represents this concept where you can manage two Cisco Catalyst 6509 chassis as a single, 18-slot chassis after enabling Cisco Virtual Switching System.

Figure 1. Cisco Virtual Switching System

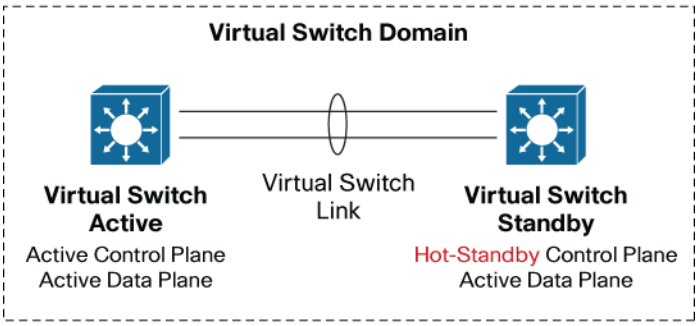


Cisco Catalyst 6500 Series Virtual Switching System 1440 Architecture

The Cisco Catalyst 6500 Series Virtual Switching System 1440 allows for the combination of two switches into a single, logical network entity from the network control-plane and management perspectives. To the neighboring devices, the Cisco Virtual Switching System appears as a single, logical switch or router.

Within the Cisco Virtual Switching System, one chassis is designated as the active virtual switch and the other is designated as the standby virtual switch. All control-plane functions, including management (Simple Network Management Protocol [SNMP], Telnet, Secure Shell [SSH] Protocol), Layer 2 protocols (bridge protocol data units [BPDUs], protocol data units [PDUs], Link Aggregation Control Protocol [LACP], Layer 3 protocols (routing protocols and so on), and software data path are centrally managed by the active supervisor engine of the active virtual switch chassis. The supervisor engine on the active virtual switch is also responsible for programming the hardware forwarding information onto all the distributed forwarding cards (DFCs) across the entire Cisco Virtual Switching System as well as the policy feature card (PFC) on the standby virtual switch supervisor engine. (See Figure 2.)

Figure 2. Components of Cisco Virtual Switching System



From data-plane and traffic-forwarding perspectives, both switches in the Cisco Catalyst 6500 Series Virtual Switching System 1440 actively forward traffic. The PFC on the active virtual switch supervisor engine performs central forwarding lookups for all traffic that ingresses the active virtual switch, whereas the PFC on the standby virtual switch supervisor engine performs central forwarding lookups for all traffic that ingresses the standby virtual switch.

Additionally, all DFCs across the entire Cisco Virtual Switching System can also simultaneously perform packet lookups. As a result, the Cisco Virtual Switching System in aggregate offers greater than 800 Mpps of IPv4 lookup performance. Because the switch fabrics of both switches are also in an active state, the Cisco Virtual Switching System in aggregate has the switch fabric capacity of 1440 Gbps, or 1.44 Tbps.

Centralized Management

The fundamental design of a Cisco Catalyst 6500 Series Virtual Switching System 1440 allows for the centralized management of all network and device resources, including Layer 3 protocols (Open Shortest Path First [OSPF], Enhanced Interior Gateway Routing Protocol [EIGRP], Border Gateway Protocol [BGP], and so on) and Layer 2 protocols (Spanning Tree Protocol, Unidirectional Link Detection Protocol [UDLD], Flow Control, LACP, and so on). A single supervisor engine in the Cisco Virtual Switching System is elected as the central management point for the entire system.

The chassis containing the supervisor engine acting as the single management point is referred to as the active virtual switch. The peer chassis is referred to as the standby virtual switch. The single supervisor engine acting as the single management point is referred to as the active supervisor engine, and the peer supervisor engine in the standby virtual switch chassis is referred to as the hot-standby supervisor engine. You can verify this setup with the following commands:

```
vss#show switch virtual
Switch mode : Virtual Switch
Virtual switch domain number : 200
Local switch number : 1
Local switch operational role: Virtual Switch Active
Peer switch number : 2
Peer switch operational role : Virtual Switch Standby
vss#show switch virtual redundancy
My Switch Id = 1
Peer Switch Id = 2
Last switchover reason = none
```

```
Configured Redundancy Mode = sso
Operating Redundancy Mode = sso
Switch 1 Slot 5 Processor Information :
-----------------------------------------------
Current Software state = ACTIVE
Uptime in current state = 3 weeks, 4 days, 9 minutes
Image Version = Cisco IOS Software, s72033_rp Software (s72033_rp-
ADVENTERPRISEK9_WAN_DBG-M), Version 12.2(SIERRA_INTEG_070502) INTERIM SOFTWARE
Synced to V122_32_8_11, 12.2(32.8.11)SR on rainier, Weekly 12.2(32.8.11)SX76
Technical Support: http://www.cisco.com/techsupport
Copyright (c) 1986-2007 by Cisco Systems, Inc.
Compiled Thu 03-May-07 09:46 by kchristi
BOOT = sup-bootdisk:s72033-adventerprisek9_wan_dbg-mz.SIERRA_INTEG_070502,1;
CONFIG_FILE =
BOOTLDR =
Configuration register = 0x2102
Fabric State = ACTIVE
Control Plane State = ACTIVE
Switch 2 Slot 5 Processor Information :
-----------------------------------------------
Current Software state = STANDBY HOT (switchover target)
Uptime in current state = 3 weeks, 4 days, 8 minutes
Image Version = Cisco IOS Software, s72033_rp Software (s72033_rp-
ADVENTERPRISEK9_WAN_DBG-M), Version 12.2(SIERRA_INTEG_070502) INTERIM SOFTWARE
Synced to V122_32_8_11, 12.2(32.8.11)SR on rainier, Weekly 12.2(32.8.11)SX76
Technical Support: http://www.cisco.com/techsupport
Copyright (c) 1986-2007 by Cisco Systems, Inc.
Compiled Thu 03-May-07 09:46 by kchristi
BOOT = sup-bootdisk:s72033-adventerprisek9_wan_dbg-mz.SIERRA_INTEG_070502,1;
CONFIG_FILE =
BOOTLDR =
Configuration register = 0x2102
Fabric State = ACTIVE
Control Plane State = STANDBY
```

Beginning in Cisco IOS® Software Release 12.2(33)SXI, the Virtual Switching System will fully support a redundant supervisor installed within a single chassis. This is an added level of redundancy compared to previous implementations where only a single supervisor per chassis is supported. This enhancement feature is called Quad-Sup Uplink Forwarding.

With Quad-Sup Uplink Forwarding a second supervisor installed in a Virtual Switch chassis is fully operational from a traffic forwarding perspective. The supervisor's 10 Gigabit Ethernet interfaces can be used for building the Virtual Switch Link as well as providing connectivity to external devices and hosts.

There is a new redundancy mode created specifically for the Virtual Switching System. The new redundancy mode is called "RPR-WARM." The RPR-WARM redundancy mode allows the redundant supervisor to operate primarily as DFC-enabled line card but can also assume the role of the active supervisor for the given chassis if needed. The chassis must undergo a reload for the redundant supervisor to assume the role of the active supervisor. However, this provides a deterministic recovery method of the individual chassis in the rare event of a supervisor failure. For more information on the Quad-Sup Uplink Forwarding feature please refer to the High Availability section of this paper.

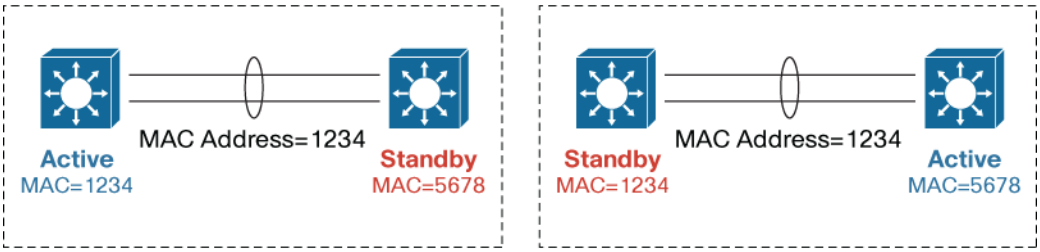
Router MAC Addresses

Router MAC addresses are assigned to Layer 3 interfaces (physical interfaces or VLANs). They are used primarily to address the Layer 2 fields of the interface for communications, but are also fundamental for the device to perform a Layer 3 lookup; Layer 3 lookups are initiated only if the destination MAC address of the frame is equal to the router MAC address of the interface.

In a standalone Cisco Catalyst 6500, the router MAC address is derived from the MAC erasable electronic programmable read only memory (EEPROM) that is embedded in each Cisco Catalyst 6500 chassis. In a Cisco Virtual Switching System environment, because two physical chassis form the single, logical device, the router MAC addresses must be consistent across both physical chassis. Therefore, the assignment of the router MAC address varies in a Cisco Virtual Switching System environment.

In a Cisco Virtual Switching System environment, the router MAC address assigned to the entire Cisco Virtual Switching System is the router MAC address derived from the MAC EEPROM of the active virtual switch chassis upon the initial system activation. When the virtual switch transitions to active state, it assigns all its Layer 3 interfaces with its own router MAC address local to its MAC EEPROM. When the standby virtual switch is brought online after VSL activation, it also derives its router MAC addresses from the MAC EEPROM of the active virtual switch. From this point onward, even if a switchover occurs between the virtual switches (causing a role change), the MAC address remains consistent (Figure 3).

Figure 3. MAC Address Synchronization Across Cisco Virtual Switching System



If the entire Cisco Virtual Switching System is restarted and brought online again but the peer switch assumes the active virtual switch role on activation, the router MAC address changes. In most environments, this change does not represent a problem because gratuitous Address Resolution Protocol (ARP) frames advertising the new router MAC addresses are transmitted upon interface initialization. If you have devices that do not interpret gratuitous ARP frames in your network, you should configure a static router MAC address on the interface:

```
vss#conf t
Enter configuration commands, one per line. End with CNTL/Z.
vss(config)#int te2/1/2
```

```
vss(config-if)#mac-address 000f.f8aa.9c00
vss(config-if)#^Z
vss-demo-1#
*Jul 12 06:49:56.441: %SYS-5-CONFIG_I: Configured from console by console
vss#sh int ten 2/1/2 | include address
Hardware is C6k 10000Mb 802.3, address is 000f.f8aa.9c00 (bia 000f.f8aa.9c00)
```

If you issue this command and save the configuration, the Cisco Virtual Switching System will always use the manually configured address, regardless of the role it assumes.

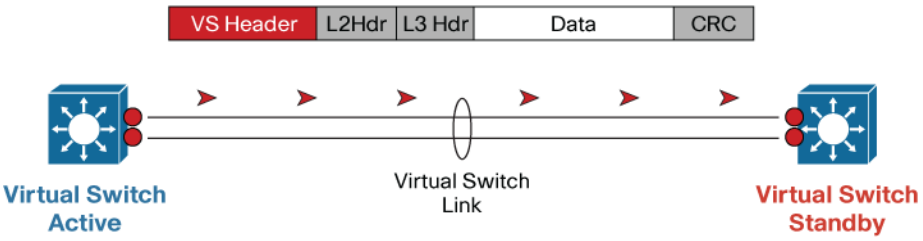
In addition to the first two methods of specifying the router MAC address, beginning in 12.2(33)SXH2, a method to select a router MAC address from a VSS reserved pool is available. The router MAC address will be derived from a formula that uses the domain-id of the VSS pairs. This will make the router MAC unique for a virtual switch domain and will remain the same irrespective of which becomes active.

```
VSS(config-vs-domain)# switch virtual domain 100
VSS(config-vs-domain)# mac-address use-virtual
VSS (config-vs-domain)#mac-address use-virtual
Configured Router mac address (0008.e3ff.fd34) is different from operational value
(0013.5f48.fe40). Change will take effect after the configuration is saved and the
entire Virtual Switching System (Active and Standby) is reloaded.
VSS(config-vs-domain)#
```

Virtual Switch Link

The Cisco Catalyst 6500 Series Virtual Switching System 1440 consists initially of two Cisco Catalyst 6500 chassis. In order to bond the two chassis together into a single, logical node, special signaling and control information must be exchanged between the two chassis in a timely manner. To facilitate this information exchange, you need a special link to transfer both data and control traffic between the peer chassis. This link is referred to as the virtual switch link (VSL). (Refer to Figure 4.)

Figure 4. Virtual Switch Header



Because a VSH is appended onto every frame that is sent across the VSL, there are also new port ASIC requirements to be able to form the VSL. The interfaces that can currently form the VSL are the 10 Gigabit Ethernet uplink interfaces on the Cisco Catalyst 6500 Series Supervisor Engine 720-10G VSS as well as those on the 8-port 10 Gigabit Ethernet line card (WS-X6708-10G-3C/XL) and the those on the 16-port 10 Gigabit Ethernet line card (WS-X6716-10G-3C/XL). Note that if the WS-6716-10G-3C/XL line card ports are used to form the VSL, the specific port must be configured in the “performance mode.”

The VSL is both the enabling technology of the Cisco Virtual Switching System and a critical link of the system. Internal Cisco Catalyst 6500 control information that is usually retained within the chassis must now be exchanged across the VSL to the peer switch, extending the backplane between the two switches. You should, therefore, always provision the VSL to contain at least two member links, which should be deployed along different physical paths or conduits. You should minimize data traffic use of the VSL wherever possible. Refer to the section "Cisco EtherChannel Concepts" for more details.

Control-Plane Communication

The VSL is crucial for both CPUs in each supervisor engine to communicate with each other. It is also used to determine which virtual switch becomes the active virtual switch and which becomes the standby virtual switch. Because this determination affects the behavior of each switch, the roles must be negotiated very early during the chassis bootup cycle. As a result, the system must bring the VSL and its associated ports online before initializing the rest of the system.

Communication between the two chassis is facilitated with internal messaging that is sent across the VSL. Because the VSL is implemented as a Cisco EtherChannel interface, it is resilient to single-link failures. However, realistically only a single link of the VSL is chosen as the control link at any given time because the hash algorithm of the Cisco EtherChannel interface is based on the source and destination MAC addresses, which are always the same for each CPU.

Virtual Switch Link Initialization

The system must bring the VSL online before activating the Cisco Virtual Switching System. The initialization sequence consists of the following steps:

1. In-chassis Role Resolution

   Beginning in software release 12.2(33)SXI4 the Virtual Switching System may be configured with two supervisor modules in a single chassis. If the chassis is configured with two supervisors then the two supervisors start their boot process by performing role negotiation. This initial role negotiation is to determine which supervisor will become the active supervisor for the chassis. One Supervisor will become the In-chassis Active and the other will become the In-chassis Standby. The In-chassis Active will continue to boot as a supervisor module and proceed with the VSL initialization. Alternatively, the In-chassis Standby supervisor will boot to a VSS unique redundancy mode which allows it to perform primarily as a line card but with some data synchronized for redundancy. More details on the In-chassis redundant supervisor are provided in the High Availability section of this paper.

2. VSL initialization

   The VSL link initialization occurs very early in the system boot cycle—it actually occurs even before the configuration file is parsed and the system is initialized. To determine which ports form members of the VSL, the configuration file is preparsed to extract the appropriate VSL commands and their associated interfaces, so that the modules containing these interfaces can be powered up, diagnostics run, and VSL interfaces brought online.

   The Link Management Protocol (LMP) operates on each link of the VSL and is part of the Virtual Switch Link Protocol (VSLP). It performs the following functions:

   - Identifies and rejects unidirectional links
   - Exchanges switch IDs between the two chassis
   - Exchanges other information required to establish communication between the two chassis

3. VSS Role resolution

   The role of each physical chassis is resolved by another protocol that forms part of VSLP—the Role Resolution Protocol (RRP), which performs the following functions:

   - Determines whether the hardware and software versions allow a Cisco Virtual Switching System to be formed

- Determines which chassis will become the active virtual switch and which will become the standby virtual switch chassis from a control-plane perspective.

4. High-availability role determination

After the role resolution, the active and standby image versions and configurations are checked for compatibility helping make sure that the hardware and software versions are the same on both chassis supervisor engines. The configuration check helps make sure that the VSL-related configurations on the two switches are compatible. If either of the two checks fails, then the standby chassis comes up in route-processor redundancy (RPR), mode where all modules are powered down, as opposed to Nonstop Forwarding/Stateful Switchover (NSF/SSO) mode, where the standby chassis is fully initialized and can forward traffic.

An example of how configuration checking may force the system into RPR mode is provided in the following output:

```
*Jun 29 14:05:44.731: %VSLP-SW2_SP-5-RRP_ROLE_RESOLVED: Role resolved as ACTIVE by VSLP
*Jun 29 14:05:44.735: %VSL-SW2_SP-5-VSL_CNTRL_LINK: vsl_new_control_link NEW VSL Control Link 5/4
*Jun 29 14:05:44.735: %VSL-SW2_SP-2-VSL_STATUS: === VSL is UP
*Jun 29 14:08:22.294: %VS_PARSE-3-CONFIG_MISMATCH: The system:/running-config VSL config comparison failed
Switch 2 has the following configs that mismatch with Switch 1:
Interface TenGigabitEthernet1/5/4 shutdown
*Jun 29 14:08:22.210: SW2_SP: VS_PARSE_DBG_ERR: vs_redun_send_check_vs_config: icc_req_resp_timeout_and_success: Failed
*Jun 29 14:08:22.210: SW2_SP: VS_PARSE_DBG: vs_redun_check_vs_config: running config check on rp not ok
*Jun 29 14:08:22.218: %PFREDUN-SW2_SP-6-ACTIVE: Standby initializing for RPR mode
```

This output shows that the configuration consistency check failed because of a mismatch in the VSL configuration between switch 1 and switch 2. In this case, switch 2 has an extra “shutdown” statement under one of its VSL members, whereas switch 1 does not, forcing the standby virtual switch (in this case switch 1) into RPR mode.

In order to recover from this situation, make any necessary changes to the configuration, save the configuration, and reload the standby chassis:

```
vss#conf t
Enter configuration commands, one per line. End with CNTL/Z.
vss(config)#int te1/5/4
vss(config-if)#no shut
vss(config-if)#^Z
vss#wr
Building configuration...
*Jun 29 14:28:53.906: %SYS-5-CONFIG_I: Configured from console by console
*Jun 29 14:29:04.834: %PFINIT-SW2_SP-5-CONFIG_SYNC: Sync’ing the startup configuration to the standby Router. [OK]
vss#redundancy reload shelf 1
```

```
Reload the entire remote shelf[confirm]
Preparing to reload remote shelf
```

Upon reload of switch 1, you should be able to observe that configurations are now synchronized and both switches can enter into NSF/SSO mode:

```
*Jun 29 14:40:46.101: VS_PARSE_DBG: vsl_mgr_parse_config_file:
vsl_mgr_parse_config_file:Open Succeeded for running config system:/running-config
*Jun 29 14:40:46.029: SW2_SP: VS_PARSE_DBG: vs_redun_check_vs_config: running config
check on rp ok
*Jun 29 14:40:46.037: %PFREDUN-SW2_SP-6-ACTIVE: Standby initializing for SSO mode
*Jun 29 14:40:49.874: %PFINIT-SW2_SP-5-CONFIG_SYNC: Sync'ing the startup configuration
to the standby Router.
```

Cisco Catalyst 6500 Series Virtual Switching System 1440 Hardware Requirements

Specific hardware is required to enable the Cisco Virtual Switching System feature; it exists in the form of the supervisor engine and system forwarding engines, the VSL-capable modules, and modules that may exist in a Cisco Virtual Switching System-enabled system.

Forwarding Engine

The forwarding engine of the Cisco Catalyst 6500 may exist in the form of a PFC on the supervisor engine or a DFC that is installed on the individual line cards. These forwarding engines perform lookup functions for every frame that enters into the system and can determine the ultimate destination of the packet as well as providing value-added services such as security access control list (ACL) and quality of service (QoS) lookups.

A standalone Cisco Catalyst 6500 and a Cisco Virtual Switching System-enabled Cisco Catalyst 6500 have two notable differences:

- Both the active and the hot-standby supervisor engine PFCs are active and are used to perform packet lookups for centralized lookup on each chassis.
- All forwarding engines are required to cater for an increased amount of port index information to be able to address a fully populated Cisco Virtual Switching System-enabled chassis. In the initial release of software, this requirement generally indicates a requirement to address twice the number of physical ports.

As a result, a new system forwarding engine mode is required to enable the Cisco Virtual Switching System capabilities. This new mode is the Policy Feature Card 3C (PFC3C) or Policy Feature Card 3CXL (PFC3CXL) mode. You can verify the operating mode of the system with the following command:

```
vss#sh platform hardware pfc mode
PFC operating mode : PFC3C
Configured PFC operating mode : None
```

Because the system also operates in a lowest-common-denominator mode, it is important to make sure that all other forwarding engines in the chassis are either at PFC3C mode or above (PFC3CXL). If a lower-mode module was previously inserted into the chassis that forced the system mode of operation to PFC3A, PFC3B, or PFC3BXL mode, then the Cisco Virtual Switching System function will not be enabled on the system. Likewise, if a module with a lower-mode DFC is inserted into the chassis after conversion to Cisco Virtual Switching System mode, the system will not grant power to the module.

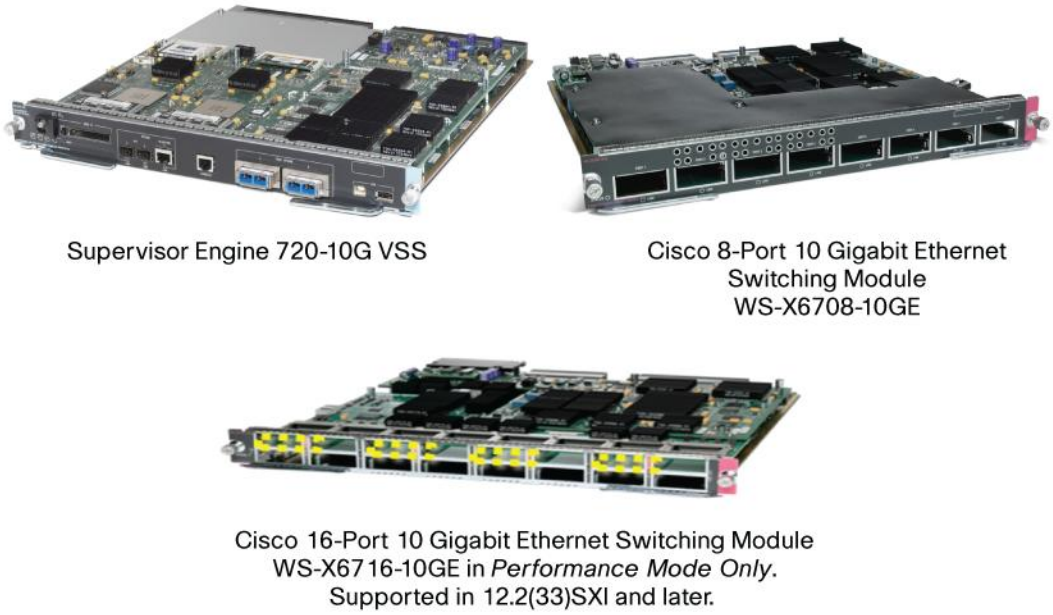
Additionally, the supervisor engines of both chassis may prenegotiate their modes to be in PFC3CXL mode during VSL initiation. If modules are installed in a system that contains a DFC3C, then the module may never be allowed to power up. This example is further highlighted by the case where the VSL module itself is a DFC3C, but the supervisor engine has a PFC3CXL. In this case, the VSL-enabled module may never power up. To avoid this situation, configure the following global command if installation of a mix of PFC3C and PFC3CXL modules is possible:

```
vss(config)#platform hardware vsl pfc mode pfc3c
System EARL mode will be forced to PFC3C on next bootup
vss#sh platform hardware pfc mode
PFC operating mode : PFC3C
Configured PFC operating mode : PFC3C
```

Virtual Switch Link-Capable Interfaces

VSL-capable interfaces are required to create a VSL port channel. The current modules that support the formation of a VSL port channel are the 10 Gigabit Ethernet uplink ports on the Supervisor Engine 720-10G (VS-S720-10G-3C/XL), Cisco 8-Port 10 Gigabit Ethernet Switching Module (WS-X6708-10G-3C/XL), and Cisco 16-Port 10 Gigabit Ethernet Switching Module (WS-X6716-10G-3C/XL). (Refer to Figure 5.)

Figure 5. VSL-Capable Modules



These interfaces contain new port ASICs that allow the VSH to be encapsulated on each frame forwarded out of the port and also support the ability to de-encapsulate VSH-tagged frames.

Supported Chassis

From a chassis perspective, both E-Series chassis and non E-Series chassis are supported within a Cisco Virtual Switching System environment, with the exception of the Cisco Catalyst 6503 (non E-Series) and Cisco Catalyst 6509-NEB (non E-Series). Table 1 gives a complete list of the chassis supported with the initial release of Cisco Virtual Switching System.

Table 1. Chassis Supported by Cisco Virtual Switching System

Model Number	Description
WS-C6503-E	E-Series 3-slot chassis
WS-C6504-E	E-Series 4-slot chassis
WS-C6506	6-slot chassis
WS-C6506-E	E-Series 6-slot chassis
WS-C6509	9-slot chassis
WS-C6509-E	E-Series 9-slot chassis
WS-C6509-NEB-A	9-slot vertical Network Equipment Building Standards (NEBS) chassis
WS-C6509-V-E	E-Series 9-slot vertical chassis
WS-C6513	13-slot chassis

It should be noted that there is no requirement that the two members of the Cisco Virtual Switching System use the same chassis type. The members consisting of the Cisco Virtual Switching System can be different chassis with varying slot counts.

Additionally, note that no Cisco 7600 Series chassis will be supported after the system is converted to Cisco Virtual Switching System mode.

Supervisor Engine

The supervisor engine that supports the Cisco Virtual Switching System feature is the Supervisor Engine 720-10G VSS (VS-S720-10G-3C/XL, Figure 6). This supervisor engine has both the PFC3C/XL- as well as the VSL-capable interfaces integrated on the module

Figure 6. Supervisor Engine 720-10G VSS

The initial release of the Cisco Virtual Switching System supports only one Supervisor Engine 720-10G VSS per chassis. If a redundant supervisor is installed in a Virtual Switching System chassis the redundant Supervisor will stop the boot process at the ROMMON stage. Beginning with the 12.2(33)SXI4 software release the Quad-Sup Uplink forwarding feature is available where the redundant supervisor will fully load a Cisco IOS Software image. More details on the Quad-Sup Uplink Forwarding feature is provided in the High Availability section of this paper.

Other Supported Modules

You can use all other modules in the chassis to connect to other network nodes or devices, including interface modules or integrated services modules.

Supported interface modules that can coexist within a Cisco Virtual Switching System-enabled chassis include all CEF720 modules (WS-X6700 series). These modules can also support either a centralized forwarding card (CFC) or a DFC. If a DFC is installed, it must either be DFC3C or DFC3CXL. A lower-mode DFC inserted in the module will be denied system power until the system-wide PFC mode has been configured and the system is reloaded.

Cisco Catalyst 6500 Series Virtual Switching System Hardware Deployment Recommendations

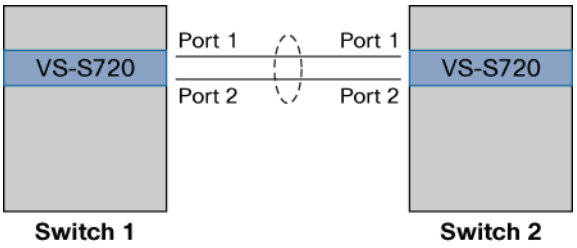
You can deploy Cisco Virtual Switching System in your network in numerous ways. In order to maximize system availability and capacity, note the following recommendations with their associated benefits and caveats.

Two-Port VSL Using Supervisor-Engine Uplinks

In this scenario, the two members of the Cisco Virtual Switching System are connected through a 2-port VSL bundle. Figure 7 shows the VSL being forme

d out of the two 10 Gigabit Ethernet uplink ports on the Supervisor Engine 720-10G VSS.

Figure 7. VSL Formed Out of Two 10 Gigabit Ethernet Uplink Ports

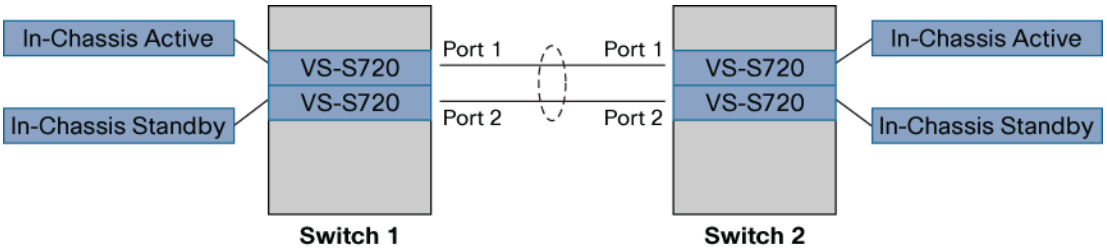


This deployment scenario is the minimum recommended configuration and allows for a redundant VSL interface connection without requiring additional hardware modules. However, if a VSL module fails (in this case, the VSL module is also the supervisor engine), the VSL as well as the chassis associated with the failed supervisor engine will stop operating. Note that this scenario does not allow for future scaling of VSL bandwidth because no extra VSL-capable 10 Gigabit Ethernet interfaces are available. See the Failure Scenarios section of this document for a description of the outage associated with a single chassis failure.

Two-Port VSL Using Quad-Sup Uplink Forwarding

This deployment scenario is supported beginning in the 12.2(33)SXI4 software release using the Quad-Sup Uplink Forwarding capability. (See Figure 8.)

Figure 8.

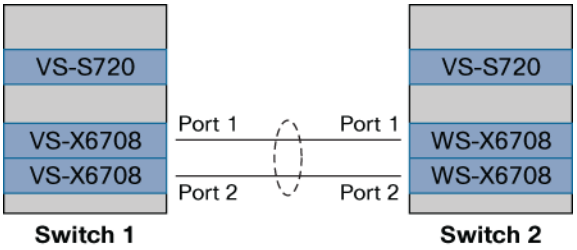


In this configuration the In-chassis Standby supervisors are fully operational and can used to form the VSL or as uplinks to other devices. Also the In-chassis Standby is capable of assuming the In-chassis Active role in the event the In-chassis Active should fail. Therefore it is important that VSL is built using interfaces from more than a single Supervisor engine. In this case the VSL is built using the ten gigabit interfaces from both supervisor engines. More details on the Quad-Sup Uplink Forwarding feature are provided in the High Availability section of this paper.

Two-Port VSL Using Line Cards Only

Figure 9 shows the two members of the Cisco Virtual Switching System connected through a 2-port VSL bundle, but takes full advantage of line cards instead of the supervisor-engine uplinks.

Figure 9. VSL Composed of Interfaces on the Cisco 8-Port 10 Gigabit Ethernet Switching Module

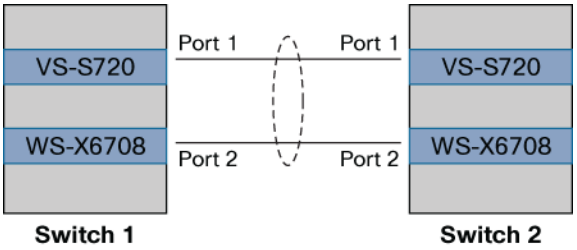


This deployment scenario allows for a redundant VSL connection as well, but uses the two Cisco 8-port 10 Gigabit Ethernet modules if the uplinks on the supervisor engine are already in use. As a result, the scenario depicted in Figure 8 involves more hardware, but it is highly redundant: the VSL can survive both a physical link failure as well as a complete VSL module failure. This solution also provides for future VSL bandwidth scalability.

Two-Port VSL Using Supervisor Engine and Line Card

Figure 10 shows the two members of the VSS connected through a 2-port VSL bundle; this scenario uses both supervisor-engine uplinks and a single 8-port 10 Gigabit Ethernet line card for bandwidth and redundancy.

Figure 10. VSL Members Across Supervisor Engine Ports and 8-Port 10 Gigabit Ethernet Switching Module



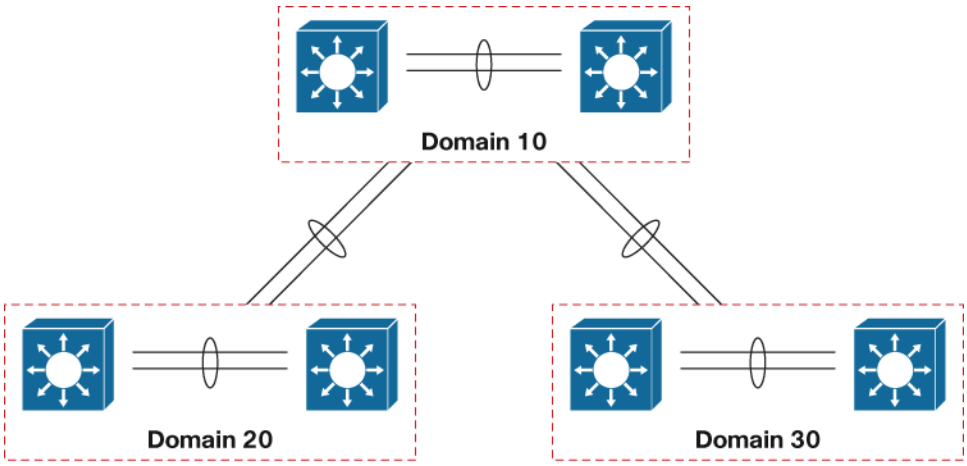
This deployment scenario, which allows for a redundant VSL connection, may require more hardware than using a single supervisor engine, but offers both link and line-card redundancy as well as the added benefit of future scaling requirements if you need extra VSL bandwidth in the future.

Multiple Cisco Virtual Switching System Domains

Multiple deployments of Cisco Virtual Switching System can exist in a given network design, adding to the availability and scalability of the network. As a result, Cisco requires you use unique virtual switch domain identifiers for each pair of VSS switches.

Figure 11 shows an example of multiple Cisco Virtual Switching System domains in a network design. The figure shows three unique VSS domains, each with a unique domain ID. You can also deploy multichassis Cisco EtherChannel links across other Cisco Virtual Switching Systems, removing the reliance on protocols such as Spanning Tree Protocol.

Figure 11. Multiple Cisco Virtual Switching System Domains



Another example of multiple Cisco Virtual Switching System domains is in the area of Layer 2 adjacent WAN deployments. It may be a requirement for the business or the applications that a routed Layer 3 WAN connection may not be possible requiring a Layer 2 connection between two disparate geographic sites, yet still providing link redundancy at the same time. This will ultimately require some form of Layer 2 redundancy protocol be implemented (such as Spanning Tree Protocol), resulting in complex topologies as well as inefficient bandwidth utilization across network links. By using Cisco Virtual Switching System, such inefficiencies will be mitigated through the formation of multichassis Cisco EtherChannel connections.

Cisco EtherChannel Concepts

Cisco EtherChannel interfaces on the Cisco Catalyst 6500 platform represent a grouping of one or more physical ports into a single, logical port from the perspective of either a Layer 2 switching or Layer 3 routing environment. Cisco EtherChannel interfaces allow for individual link resiliency as well as providing added bandwidth without the necessity of complex protocols.

There are generally no restrictions with regard to which ports or modules can form members of a Cisco EtherChannel link, except that the member interfaces need to be of the same speed and no more than 8 members can belong to a single Cisco EtherChannel grouping. You can, therefore, extend members of the Cisco EtherChannel interface across switching modules to allow for the maximum availability of the Cisco EtherChannel interface if either a single link or module fails.

Traffic Distribution and Hashing

The distribution of traffic across the various members of the Cisco EtherChannel link is accomplished through different hash schemes, each using a fixed set of fields within the frame to determine which Cisco EtherChannel member is used to forward a particular traffic flow. With the PFC3C running Cisco IOS Software Release 12.2(33)SXH, you can choose from 13 possible different hash schemes:

```
vss(config)#port-channel load-balance ?
dst-ip Dst IP Addr
dst-mac Dst Mac Addr
dst-mixed-ip-port Dst IP Addr and TCP/UDP Port
dst-port Dst TCP/UDP Port
mpls Load Balancing for MPLS packets
src-dst-ip Src XOR Dst IP Addr
```

```
src-dst-mac Src XOR Dst Mac Addr
src-dst-mixed-ip-port Src XOR Dst IP Addr and TCP/UDP Port
src-dst-port Src XOR Dst TCP/UDP Port
src-ip Src IP Addr
src-mac Src Mac Addr
src-mixed-ip-port Src IP Addr and TCP/UDP Port
src-port Src TCP/UDP Port
```

Selection of the hash scheme of choice largely depends on the traffic mix through the Cisco EtherChannel interface, noting that these hash schemes may be selected only on a global basis.

Determination of Hash Result

With the release of Cisco Virtual Switching System, a new mechanism has been implemented to allow you to determine which physical link a given flow of traffic uses within a port-channel group. You provide inputs to the command and the hashing algorithm computes the physical link that is selected for the traffic mix and algorithm.

```
vss#sh etherchannel load-balance hash-result ?
interface Port-channel interface
ip IP address
ipv6 IPv6
l4port Layer 4 port number
mac Mac address
mixed Mixed mode: IP address and Layer 4 port number
mpls MPLS
vss#sh etherchannel load-balance hash-result interface port-channel 120 ip
192.168.220.10 192.168.10.10
Computed RBH: 0x4
Would select Gi1/2/1 of Po120
```

Adaptive Load Balancing

The addition or removal of a member port from a Cisco EtherChannel interface has always led to a varied amount of traffic loss for customers. The current generation of port ASICs uses a 3-bit Result Bundle Hash (RBH) value from the PFC or DFC result to index into a load register to allow a packet to be transmitted if the corresponding bit is set.

When a new port is added or deleted, the load value is reset on all the ports. A new load is then distributed on all the ports in the Cisco EtherChannel interface, including the new member, and reprogrammed into the port ASIC for each port. This process causes packets to be dropped during the short outage window (approximately 200 to 300 ms), an undesirable result for higher-speed interfaces such as 10 Gigabit Ethernet connections where a large amount of traffic may be lost during this brief outage window.

This problem has led to the development of an enhanced load-distribution mechanism such that when ports are added or removed from a Cisco EtherChannel interface, the load result does not need to be reset on existing member ports, resulting in better traffic recovery times.

You can implement this new algorithm either globally or on a per-port channel basis, where **fixed** is the current default mode and **adaptive** uses the enhanced mode:

```
vss(config)#port-channel hash-distribution ?
adaptive selective distribution of the bndl_hash among port-channel members
fixed distribution of the bndl_hash among port-channel members
vss(config)#int port-channel 4
vss(config-if)#port-channel port hash-distribution ?
adaptive selective distribution of the bndl_hash among port-channel members
fixed fixed distribution of the bndl_hash among port-channel members
```

The algorithm selected with these commands is applied only at the next hash-distribution instance, which usually occurs on a port-channel member link transition event.

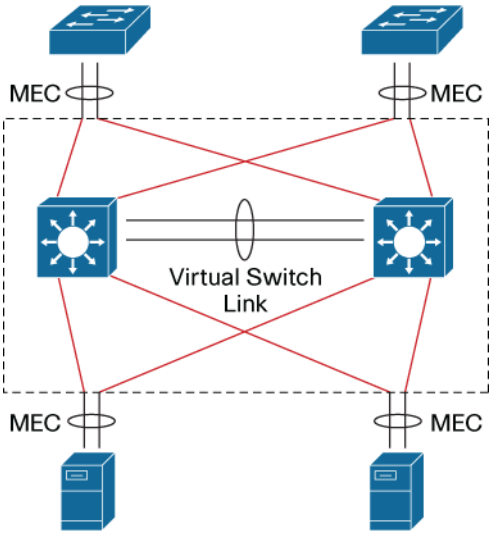
```
vss#sh etherchannel 4 summary
Flags: D - down P - bundled in port-channel
I - stand-alone s - suspended
H - Hot-standby (LACP only)
R - Layer3 S - Layer2
U - in use N - not in use, no aggregation
f - failed to allocate aggregator
M - not in use, no aggregation due to minimum links not met
m - not in use, port not aggregated due to minimum links not
met
u - unsuitable for bundling
d - default port
w - waiting to be aggregated
Number of channel-groups in use: 9
Number of aggregators: 9
Group Port-channel Protocol Ports
------+-------------+-----------+-----------------------------------
4 Po4(SU) PAgP Gi1/5/3(P) Gi2/5/3(P)
Last applied Hash Distribution Algorithm: Adaptive
```

Although this new load-distribution algorithm requires configuration for regular Cisco EtherChannel and multichassis Cisco EtherChannel interfaces, it is the default load-distribution algorithm used on the virtual switch links.

Multichassis Cisco EtherChannel Links

The multichassis Cisco EtherChannel interface spans more than a single physical switch (Figure 12). Cisco Virtual Switching System allows for the formation of this multichassis Cisco EtherChannel link and allows all the dual-homed connections to and from the upstream and downstream devices to be configured as Cisco EtherChannel links, as opposed to individual links. As a result, multichassis Cisco EtherChannel links allow for implementation of new network designs where true Layer 2 multipathing can be implemented without the reliance on Layer 2 redundancy protocols such as Spanning Tree Protocol. With 12.2(33)SXI and above, VSS supports 512 etherchannels.

Figure 12. Multichassis Cisco EtherChannel Links



Like regular Cisco EtherChannel interfaces, all ports within the multichassis Cisco EtherChannel link have the same source index regardless of the chassis in which they are physically present, making it possible to apply a single IP address for Layer 3 Cisco EtherChannel links or for Spanning Tree Protocol to view such a Cisco EtherChannel interface as a single, logical port.

One unique difference between multichassis Cisco EtherChannel and regular Cisco EtherChannel interfaces is the way traffic is load balanced across the channel group members. A regular Cisco EtherChannel link selects the appropriate channel group member to exit based purely on the hashing algorithm of choice. A multichassis Cisco EtherChannel link, however, has some extra intelligence to reduce the amount of traffic that requires transmission across the VSL. This optimization is accomplished by populating the index port only with the ports local to the physical chassis, allowing the chassis to favor the local ports of the multichassis Cisco EtherChannel link over those on the remote chassis.

For traffic that must be flooded on the VLAN (broadcasts, multicasts, and unknown unicasts), a copy is sent across the VSL to be sent out any single-homed ports belonging to the VLAN. Because the first chassis will have sent a copy out one of the multichassis Cisco EtherChannel ports, packets received from the VSL are not sent out of another multichassis Cisco EtherChannel port. If all of the multichassis Cisco EtherChannel ports on a given chassis are removed because of a failure, management control, and so on, the Cisco EtherChannel link is no longer a multichassis Cisco EtherChannel link, but a regular Cisco EtherChannel link, and hence flooded packets will be sent out of this EtherChannel link from the VSL.

Although the data traffic is spread across the two chassis, the active supervisor engine must terminate control traffic for the multichassis Cisco EtherChannel link on the active virtual switch, including most of the Layer 2 protocols such as Spanning Tree Protocol, Port Aggregation Protocol (PAgP), VLAN Trunking Protocol (VTP), and so on. All multichassis Cisco EtherChannel links have their control protocols terminated on the active supervisor engine. Any control protocols received by multichassis Cisco EtherChannel link ports on the standby virtual switch are redirected to the active supervisor engine through the VSL. Because the Cisco EtherChannel link is terminated in one chassis, PAgP and LACP have the same device identifier on all the member links, regardless of the chassis on which the link resides.

Multichassis Cisco EtherChannel Link Management Protocols

Multichassis Cisco EtherChannel links support both the Cisco proprietary Port Aggregation Protocol (PAgP) and the LACP, both of which run on the active supervisor engine on the active virtual switch. Protocol frames that the standby virtual switch receives are relayed to the active supervisor engine on the active virtual switch through the VSL.

Virtual Switch Mode

With the first release of software supporting the Cisco Virtual Switching System, you can run the switches in either standalone mode or virtual mode. The default configuration is for the individual chassis to operate in standalone mode. In order to migrate to virtual mode, you must perform a conversion procedure, outlined as follows.

After the chassis reloads and is operating in virtual mode, it begins the VSL initialization sequence. Additionally, the interface naming convention is changed to allow for the specification of a chassis identifier as part of the interface name. Please refer to the section “Operational Management” for more information.

Switch Identifier

Each chassis within the Cisco Virtual Switching System is allocated a unique chassis identifier upon conversion to virtual switch mode. This identifier is known as the switch identifier, or switch ID. This number is used as part of the interface naming to make sure that the interface name remains the same regardless of the active or standby virtual switch roles.

As mentioned previously, this variable is set during the conversion phase; if a replacement supervisor engine is required, it is set with an enable-mode command-line interface (CLI) command. The variable that has been set is stored as a variable in ROMmon, so it is locally significant to the individual supervisor engine. If you need to alter the switch ID, use the following CLI:

```
VSS#switch set switch_num 1

Set rommon’s switch_num to 1

VSS#switch read switch_num

Read switch_num from rommon is 1
```

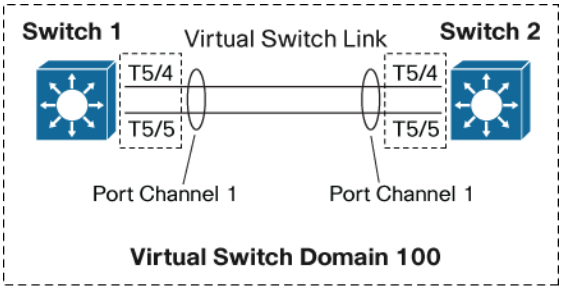
If there is a misconfiguration in switch IDs (when both switches have the same switch ID), the formation of the VSL will fail on initialization. When the two chassis are being brought up as a single Cisco Virtual Switching System, the VSL initialization handshake verifies that the switch IDs of the two chassis do not match. If the switch ID is found to be in conflict, then the VSL will not become active. If this situation occurs, both chassis assume the role of active virtual switch and you are informed of this conflict.

Conversion to Virtual Switch Mode

This section details the steps required to convert a standalone system into virtual switch mode. The interfaces forming the VSL should be connected prior to the conversion process to minimize the number of times the chassis is reloaded. Additionally, you should begin the conversion process using a default configuration because the conversion process removes any previous configuration that exists on the standalone chassis.

Refer to Figure 13 to reference the conversion process.

Figure 13. Cisco Virtual Switching System Conversion



Step 1. Configure virtual switch ID and domain.

On the two switches, configure the same virtual switch domain number (in this case it is 100), but unique switch IDs using the following configuration mode commands: VSS-sw1#conf t Enter configuration commands, one per line. End with CNTL/Z. VSS-sw1(config)#switch virtual domain 100 Domain ID 100 config will take effect only after the exec command 'switch convert mode virtual' is issued VSS-sw1(config-vs-domain)#switch 1 VSS-sw1(config-vs-domain)#	VSS-sw2#conf t Enter configuration commands, one per line. End with CNTL/Z. VSS-sw2(config)#switch virtual domain 100 Domain ID 100 config will take effect only Aft er the exec command 'switch convert mode virtual' is issued VSS-sw2(config-vs-domain)#switch 2 VSS-sw2(config-vs-domain)#

Note that as a result of this command, the domain ID is retained in the configuration, but the switch ID is not; this value is stored as a variable in ROMmon. It can be confirmed by issuing the following command on each switch:

VSS-sw1#switch read switch_num Read switch_num from rommon is 1	VSS-sw2#switch read switch_num Read switch_num from rommon is 2

Step 2. Configure the VSL port channel and member ports.

Choose unique port-channel IDs for each chassis to form the VSL and configure them with the corresponding switch ID using the following commands:

VSS-sw1#conf t Enter configuration commands, one per line. End with CNTL/Z. VSS-sw1(config)#interface port-channel 1 VSS-sw1(config-if)#switch virtual link 1 VSS-sw1(config-if)#no shut VSS-sw1(config-if)#	VSS-sw2#conf t Enter configuration commands, one per line. End with CNTL/Z. VSS-sw2(config)#interface port-channel 2 VSS-sw2(config-if)#switch virtual link 2 VSS-sw2(config-if)#no shut VSS-sw2(config-if)#

Now add the ports on each switch to the port channel that corresponds to the respective side of the VSL using the following commands:

VSS-sw1(config)#interface range tenGigabitEthernet 5/4 - 5	VSS-sw2(config)#interface range tenGigabitEthernet 5/4 - 5
VSS-sw1(config-if-range)#channel-group 1 mode on	VSS-sw2(config-if-range)#channel-group 2 mode on
VSS-sw1(config-if-range)#no shut	VSS-sw2(config-if-range)#no shut
VSS-sw1(config-if-range)#^Z	VSS-sw2(config-if-range)#^Z
VSS-sw1#	VSS-sw2

Note that only the local port channels and their associated members need to be configured on each switch. Because the switches are still in standalone mode, you do not need to configure the peer-switch ports.

Step 3. Convert to virtual switch mode.

Convert both switches to virtual switch mode using the following exec command:

VSS-sw1#switch convert mode virtual	VSS-sw2#switch convert mode virtual
This command will convert all interface names to naming convention "interface-type switch-number/slot/port," save the running config to startup-config and reload the switch.	This command will convert all interface names to naming convention "interface-type switch-number/slot/port," save the running config to startup-config and reload the switch.
Do you want to proceed? [yes/no]: yes	Do you want to proceed? [yes/no]: yes
Converting interface names	Converting interface names
Building configuration...	Building configuration...
[OK]	[OK]
Saving converted configurations to bootflash …	Saving converted configurations to bootflash…
[OK]	[OK]

The following actions occur when you issue this command:

- The running configuration of the individual switch is converted into a three-level virtual switch interface notation. Two-level interface configurations (such as TenGigabitEthernet 5/4) are converted into three-level interfaces (such as TenGigabitEthernet 1/5/4 in switch 1 and TenGigabitEthernet 2/5/4 in switch 2).
- The startup configuration is updated with the three-number notation.
- A copy of the original startup configuration converted to three-number notation is written to the multilayer switch feature card (MSFC) bootflash of the respective switch. Both switches reload.

Note that the command **switch convert mode** virtual is not stored in the startup configuration because it is not a configuration command. Instead, the following line is added to the startup configuration under the **switch virtual domain**:

When the two switches are brought online, they proceed with VSL initialization and initialize their respective VSL ports. The two switches communicate with each other and determine active and standby roles. This exchange of information is evident through the following console messages:

```
VSS#sh run | begin switch virtual domain
switch virtual domain 100
switch mode virtual
```

When the two switches are brought online, they proceed with VSL initialization and initialize their respective VSL ports. The two switches communicate with each other and determine active and standby roles. This exchange of information is evident through the following console messages:

System detected Virtual Switch configuration...	System detected Virtual Switch configuration...
Interface TenGigabitEthernet 1/5/4 is member of PortChannel 1	Interface TenGigabitEthernet 2/5/4 is member of PortChannel 2
Interface TenGigabitEthernet 1/5/5 is member of PortChannel 1	Interface TenGigabitEthernet 2/5/5 is member of PortChannel 2
<snip>	<snip>
00:00:26: %VSL_BRINGUP-6-MODULE_UP: VSL module in slot 5 switch 1 brought up	00:00:26: %VSL_BRINGUP-6-MODULE_UP: VSL module in slot 5 switch 2 brought up
Initializing as Virtual Switch active	Initializing as Virtual Switch standby

After the VSL is initialized and the Cisco Virtual Switching System becomes active, you may notice that the console is active only for the active virtual switch and has been disabled for the standby virtual switch:

00:08:01: SW1_SP: Card inserted in Switch_number = 2 ,	00:01:43: %CRYPTO-6-ISAKMP_ON_OFF: ISAKMP is OFF
physical slot 3, interfaces are now online	00:01:43: %CRYPTO-6-ISAKMP_ON_OFF: ISAKMP is OFF
VSS >	VSS-sdby>
VSS>en	Standby console disabled
VSS#	

Although not required, it is possible to verify that all modules have been automatically provisioned and their module types stored in the configuration by issuing the following command on the active virtual switch:

```
VSS#sh run | begin module provision
module provision switch 1
slot 1 slot-type 254 port-type 31 number 2 port-type 61 number 1 port-type 60 number 2 virtual-slot 17
slot 2 slot-type 148 port-type 60 number 4 virtual-slot 18
slot 3 slot-type 147 port-type 61 number 48 virtual-slot 19
!
module provision switch 2
```

```
slot 1 slot-type 254 port-type 31 number 2 port-type 61 number 1 port-type 60 number 2
virtual-slot 33
slot 2 slot-type 148 port-type 60 number 4 virtual-slot 34
slot 3 slot-type 147 port-type 61 number 48 virtual-slot 35
```

Modules are provisioned in the configuration to allow for parsing even if they are not present in the chassis. This situation occurs when one of the member switches is not yet online but the configuration needs to be parsed.

With the following command you can determine that the Cisco Virtual Switching System is now operating and that the two switches are acting as a single, logical network node.

```
VSS#show switch virtual
Switch mode : Virtual Switch
Virtual switch domain number : 100
Local switch number : 1
Local switch operational role: Virtual Switch Active
Peer switch number : 2
Peer switch operational role : Virtual Switch Standby
```

If the conversion process is performed using software release 12.2(33)SXI3 or newer then the conversion process is complete once the two supervisors reach the SSO Standby Hot redundancy mode. If the conversion is performed using a software release prior to 12.2(33)SXI3 then there is one more critical step to perform in order to finalize the conversion.

Again, this final, critical step is applicable only for a first-time conversion, and only applicable to systems converted using a software release prior to 12.2(33)SXI3.

During the conversion process, the configuration of the standby virtual switch (in this case, switch 2) is cleared, including the configuration of the two VSL interfaces on the switch. If the switch were to reload at this point it would not have the information available to determine which interfaces to use for VSL communication. Therefore the configuration for the VSL interfaces on the standby switch must be applied, or merged from the active switch configuration. In order to facilitate this information to be repopulated again, you must complete step 4.

Step 4. Finalize the Virtual Switch Conversion

When the standby virtual switch is in SSO hot mode, you must execute the following command to automatically configure the standby virtual switch configuration on the active virtual switch:

```
VSS#switch accept mode virtual
This command will bring in all VSL configurations from the standby switch and populate
it into the running configuration.
In addition the startup configurations will be updated with the new merged
configurations.
Do you want proceed? [yes/no]: yes
Merging the standby VSL configuration. . .
Building configuration...
[OK]
```

This command prompts you to accept all standby virtual switch VSL-related configurations and also updates the startup configuration with the new merged configurations. Note that only VSL-related configurations are merged with this step—all other configurations will be lost and require manual intervention.

If entering this command running a 12.2(33)SXI3 or newer software release the command returns a notification that this step is no longer necessary.

```
VSS#
Corel#switch accept mode virtual
This command is no longer required since standby VSL configuration merge is done
automatically.
VSS#
```

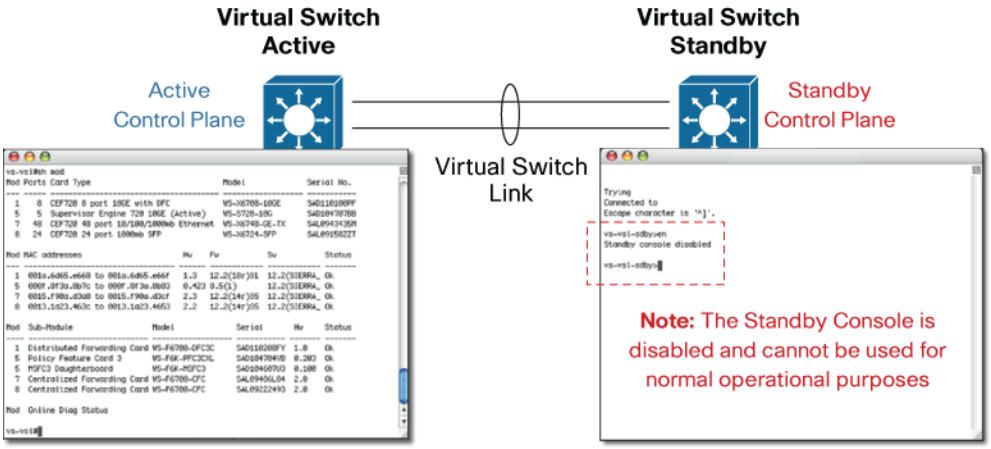
Operational Management

Management of the system as a whole changed with the advent of the Cisco Virtual Switching System. The fundamental concept of a switching system has evolved from a single physical entity managed separately to multiple physical entities that are managed as a single system. The following section examines areas such as console management, in-band (Telnet or SSH) management, SNMP and MIB changes, effects of NVRAM, NetFlow features, Switch Port Analyzer (SPAN), Embedded Event Manager (EEM), and CiscoWorks LAN Management System (LMS) management.

Console Management

After the two individual switches are converted into a Virtual Switching System, the console access is restricted to only the active virtual switch. In this case you can handle all configuration, monitoring, and troubleshooting under a single interface. The console output into the standby virtual switch indicates that the console is disabled for general administrative purposes. (See Figure 14.)

Figure 14. Active and Standby Consoles



If a switchover occurs and switch 2 becomes the active virtual switch, the console becomes active on that supervisor engine.

Interface Numbering

After conversion to a Virtual Switching System, the interface numbering changes from a traditional scheme:

```
<INTERFACE_TYPE> <MODULE>/<PORT>
```

To a new 3-number scheme:

```
<INTERFACE_TYPE> <SWITCH_ID>/<MODULE>/<PORT>
```

This new naming scheme allows a single configuration file to uniquely address the physical interfaces on both chassis that are part of the same virtual switch domain.

```
vss#sh ip interface brief
Interface IP-Address OK? Method Status Protocol
Vlan1 unassigned YES NVRAM administratively down down
Vlan10 192.168.10.1 YES NVRAM up up
Vlan20 192.168.20.1 YES NVRAM up up
Loopback0 3.3.3.3 YES NVRAM up up
Port-channel1 unassigned YES NVRAM up up
Port-channel2 unassigned YES NVRAM up up
Port-channel10 unassigned YES unset up up
Port-channel20 unassigned YES unset up up
Te1/1/1 unassigned YES unset up up
Te1/1/2 unassigned YES NVRAM administratively down down
Te1/1/3 unassigned YES NVRAM administratively down down
Te1/1/4 unassigned YES NVRAM administratively down down
Te1/1/5 unassigned YES NVRAM administratively down down
```

File-System Naming

To facilitate the ability to uniquely identify multiple file systems on each supervisor engine or module across the Cisco Virtual Switching System, a new role-independent file-system naming scheme has been implemented.

```
SW<SWITCH_ID>-SLOT<MODULE>-<FILESYSTEM>:
```

This naming scheme allows all unique file systems to be addressed and identified across the entire virtual switch domain, regardless of supervisor-engine redundancy state.

```
vss#dir sw1-slot?
sw1-slot1-dfc-bootflash: sw1-slot5-const_nvram: sw1-slot5-disk0:
sw1-slot5-nvram: sw1-slot5-sup-bootdisk: sw1-slot5-sup-
bootflash:
sw1-slot7-dfc-bootflash: sw1-slot8-dfc-bootflash:
vss#dir sw2-slot?
sw2-slot1-dfc-bootflash: sw2-slot5-bootflash: sw2-slot5-
const_nvram:
sw2-slot5-disk0: sw2-slot5-nvram: sw2-slot5-sup-
bootdisk:
sw2-slot5-sup-bootflash: sw2-slot7-dfc-bootflash: sw2-slot8-dfc-
bootflash:
```

You can still use the existing file-system naming scheme, but you need to determine the role of the switch (active or standby) before you access the file systems.

Reloading the Cisco Virtual Switching System and Its Members

It may sometimes be desirable to reload the entire system or to reset individual members of the Virtual Switching System; you can perform these tasks through the console of the active virtual switch.

Reloading the Cisco Virtual Switching System

If you need to reload the entire Cisco Virtual Switching System (both active virtual switch and standby virtual switch), the following command achieves this reload:

```
vss#reload
Proceed with reload? [confirm]
```

Reloading a Member of the Cisco Virtual Switching System

It may be more desirable to reload a member of the Cisco Virtual Switching System rather than the entire system. You can accomplish this reloading in multiple ways.

You can reset the active virtual switch in two ways. First, you can issue the command redundancy force-switchover, which essentially forces a SSO or RPR switchover from active to standby, reloading the previous active virtual switch in the process:

```
vss#redundancy force-switchover
This will reload the active unit and force switchover to standby[confirm]
Preparing for switchover..
```

You can also use the redundancy reload shelf command, where either switch 1 or switch 2 can be specified:

```
vss#redundancy reload shelf 1
Reload this shelf [confirm]
```

You also have two options to reset the standby virtual switch. First, you can use the same command as before, replacing the switch ID with the switch ID of the standby virtual switch:

```
vss#redundancy reload shelf 1
Reload the entire remote shelf[confirm]
Preparing to reload remote shelf
```

Alternatively, use the command redundancy reload peer to reload the standby virtual switch:

```
vss#redundancy reload peer
Reload peer [confirm]
Preparing to reload peer
```

Systemwide PFC Mode

Although only PFC3C or PFC3CXL modes are supported with Cisco Virtual Switching System, you can mix these modules to configure a lowest-common-denominator mode fallback.

In a standalone Cisco Catalyst 6500 system, the supervisor-engine module is the first module initiated in the system. As a result, a combination of the PFC of the supervisor engine as well as the inserted modules in the chassis during bootup determines the operational mode of the system. If you add a lower-revision module, the module will be disabled until the system is reloaded.

In a Cisco Virtual Switching System environment, the system PFC mode is negotiated between the supervisor engines much earlier in the system initiation process. Additionally, there may also be a discrepancy between PFCs used on the supervisor engines and those used on the DFCs for other VSL modules. As a result, you may encounter instances where a module configured to support VSL member links may not be brought up because of a PFC or DFC mismatch.

To remedy this situation, a new command allows you to manually specify the system-wide PFC mode to PFC3C (from PFC3CXL). This configuration follows:

```
vss#conf t
Enter configuration commands, one per line. End with CNTL/Z.
vss(config)#platform hardware vsl pfc mode pfc3c
vss(config)#^Z
```

You must restart the system to activate this command. When the system is reloaded, you can verify the operational mode of the system by issuing the following command:

```
vss#sh platform hardware pfc mode
PFC operating mode : PFC3C
Configured PFC operating mode : PFC3C
```

Using the Cisco Network Analysis Module in a Cisco Virtual Switching System Environment

The Cisco Network Analysis Modules (WS-SVC-NAM-1 and WS-SVC-NAM-2) are integrated and powerful traffic-monitoring solutions for the Cisco Catalyst 6500 (Figure 15). Cisco NAMs combine a rich set of embedded data collection and analysis capabilities with a remotely accessible, Web-based management console, all of which reside on a single blade. The embedded Web-based Cisco Traffic Analyzer console offers quick access to configuration menus and presents easy-to-read performance reports on data, voice, and video traffic. The Cisco NAM provides both the ability to analyze packets and the ability to analyze them from within the switch or router itself, giving greater insight into how the network is being used and how users experience the services on the network.

Figure 15. Cisco Network Analysis Module

In order to use the network analysis module in a Cisco Virtual Switching System environment, the minimum version must be Cisco Network Analysis Module Software 3.6(1a).

Administration of the Cisco NAM

The most direct change with Cisco NAM in a Cisco Virtual Switching System environment relates to the way you can access or manage it. In a Cisco Virtual Switching System environment, a switch ID is required for many of the commands used to administer the NAM. In the following example, the NAM is installed in switch 1, slot 3:

```
vss#sh module switch 1 slot 3
Switch Number: 1 Role: Virtual Switch Active
---------------------- ----------------------------
```

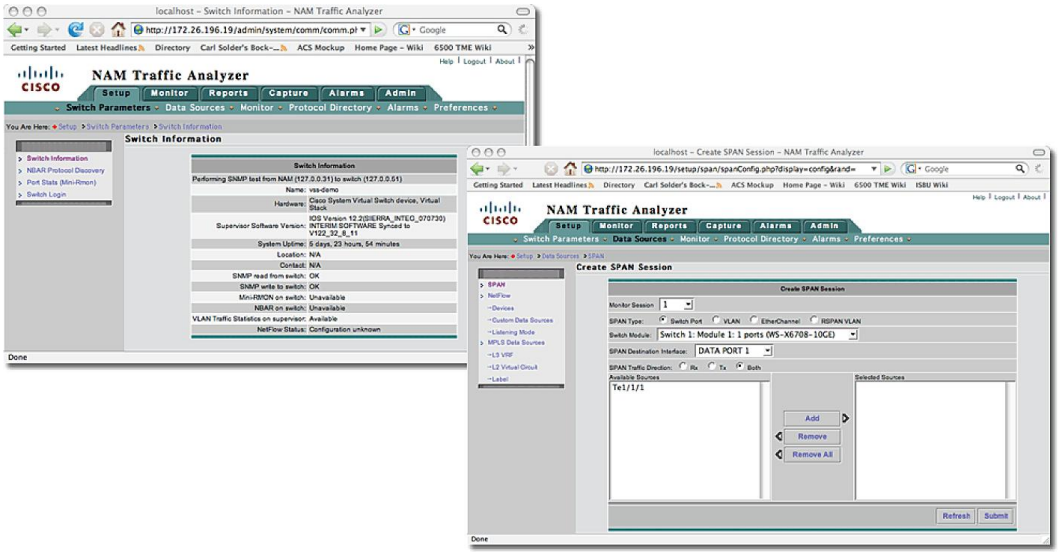
```
Mod Ports Card Type Model Serial No.
--- ----- ---------------------------- ------------------ -----------
3 8 Network Analysis Module WS-SVC-NAM-2 SAD101902NX
Mod MAC addresses Hw Fw Sw Status
--- --------------------------------- ----- ------- ------- -------
3 0017.0ee1.c4ac to 0017.0ee1.c4b3 4.0 7.2(1) 3.6(1a) Ok
Mod Online Diag Status
---- -------------------
3 Bypass
```

As a result, you can access the NAM through a "session" directly to the module that also includes the switch ID:

```
vss#session switch 1 slot 3 processor 1
The default escape character is Ctrl-^, then x.
You can also type 'exit' at the remote prompt to end the session
Trying 127.0.0.31 ... Open
Cisco Network Analysis Module (WS-SVC-NAM-2)
Linux 2.6.10-nam (localhost.localdomain) (pts/2)
localhost.localdomain login: root
Password:
Last login: Thu Aug 9 14:24:38 2007 from 127.0.0.51 on pts/1
Linux localhost.localdomain 2.6.10-nam #1 SMP Mon Jan 22 14:34:15
PST 2007 i686 GNU/Linux
Cisco Network Analysis Module (WS-SVC-NAM-2) Console, 3.6(1a)
Copyright (c) 1999-2007 by cisco Systems, Inc.
WARNING! Default password has not been changed!
root@localhost.localdomain#
```

From this point onward, all configuration is performed directly on the NAM, it is identical to that performed in standalone mode, and it can be operated or configured through the HTTP interface. (See Figure 16.)

Figure 16. Cisco NAM Configuration and Monitoring



With 12.2(33)SXI, VSS supports four additional service modules. These include:

- WS-SVC-FWSM-1-K9 (4.0.4)
- ACE-10/ACE-20-6500-K9 (A2(1.2))
- WS-SVC-WISM-1K9 (6.0(2)E1)
- WS-SVC-IDSM2-K9 (3.2.171.6)

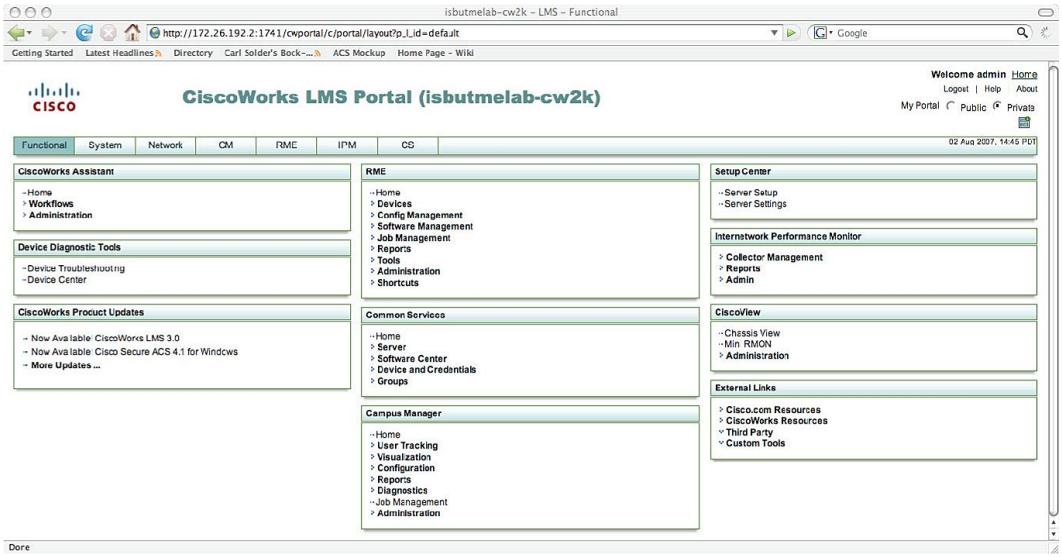
Service modules can be placed in either of the physical chassis that comprise a VSS system. VSS active and standby roles are independent of the service module redundancy role. For more information, please refer to this paper—Integrate Cisco Service Modules with Cisco 6500 VSS: http://www.cisco.com/en/US/products/ps9336/products_tech_note09186a0080a7c72b.shtml

Managing Cisco Virtual Switching System Using CiscoWorks LMS

The CiscoWorks LAN Management Solution (LMS) is a suite of powerful management tools that simplify the configuration, administration, monitoring, and troubleshooting of Cisco networks. CiscoWorks LMS 3.0.1 adds support for the management of a Cisco Virtual Switching System environment by supporting the following functions (Figure 17):

- Conversion from standalone environment to Cisco Virtual Switching System environment
- Inventory collection and reporting
- Configuration collection and archive management
- Image upgrade and patch management
- CiscoView chassis management and monitoring

Figure 17. CiscoWorks LMS 3.0.1



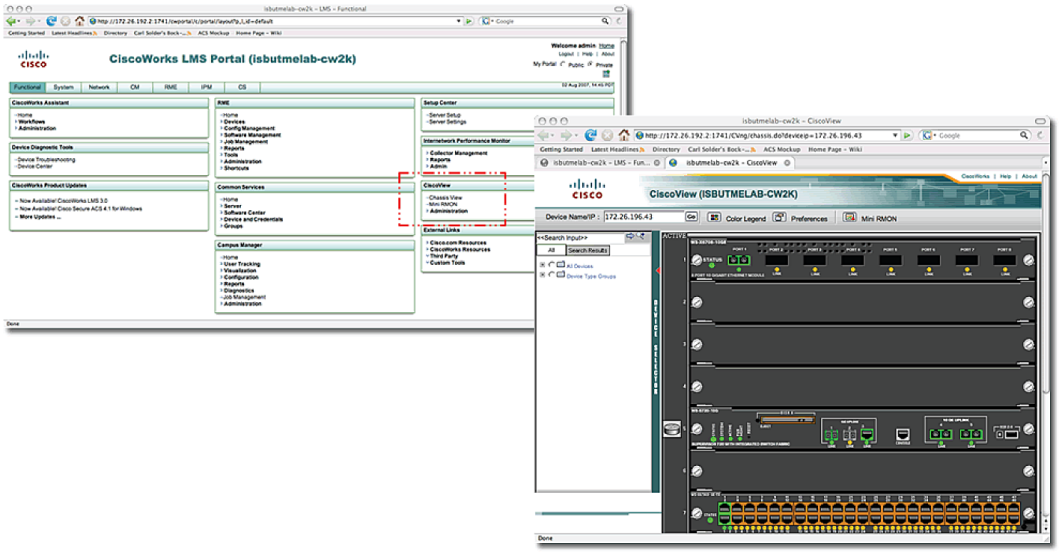
The following section provides an overview of the supported features and functions.

CiscoView Chassis Management and Monitoring

CiscoView allows for a graphical management interface that gives you real-time views of networked devices. The views displayed in CiscoView deliver a continuously updated physical picture of device configuration and performance conditions.

From a Cisco Virtual Switching System perspective, both members of the VSS can be displayed individually when you provide the IP address or host name of the system, both of which you can access under the Chassis View link under the CiscoView section from the main CiscoWorks LMS 3.0.1 launch page. To display the active virtual switch chassis, supply an IP address for the Cisco Virtual Switching System to be monitored (Figure 18).

Figure 18. CiscoWorks LMS with CiscoView



You can also manage the standby chassis with the same IP address or host name as well as within the same view by simply toggling between the active virtual switch and the standby virtual switch selector on the chassis itself (Figure 19).

Figure 19. CiscoView with Active and Standby Switch Views

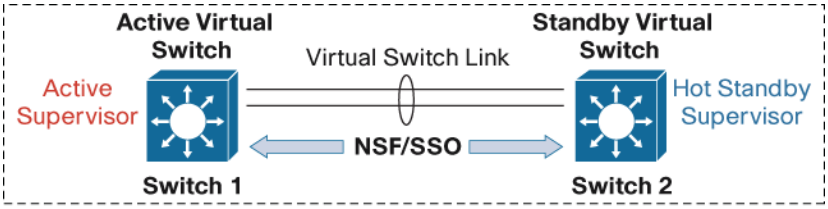
High Availability

Central to the high-availability model of a Cisco Virtual Switching System are the concepts of NSF/SSO and RPR. The intricacies of these protocols are beyond the scope of this paper; you can find more information on these protocols in the Cisco Catalyst 6500 documentation materials as well as in the following white paper:

Nonstop Forwarding and Stateful Switchover on the Cisco Catalyst 6500
http://www.cisco.com/en/US/prod/collateral/switches/ps5718/ps708/prod_white_paper0900aecd801c5cd7.html

The high-availability model of the system changes when you integrate two chassis together into a single network entity. In order to take advantage of the existing innovations in NSF/SSO technologies, Cisco Virtual Switching System has implemented a high-availability model that uses this redundancy framework for an inter-chassis environment (Figure 20).

Figure 20. Interchassis NSF/SSO in a Cisco Virtual Switching System Environment



In an SSO system, “high availability-aware” protocols and features may synchronize events and state information from the active supervisor engine to the hot-standby supervisor engine. From a redundancy framework viewpoint, the active supervisor engine acts as a server, whereas the standby supervisor engine acts as the client. Information that is “high availability-aware” will be statefully synchronized between these entities such that in the event of a failover, the standby supervisor engine does not need to re-learn this information, resulting in a minimal amount of outage time.

As Figure 19 shows, the supervisor engine in the active virtual switch (switch 1 in the figure) assumes the role as the active supervisor engine, whereas the supervisor engine in the standby virtual switch (switch 2) assumes the role as the hot-standby supervisor engine. You can verify this situation with the following command:

```
VSS#sh switch virtual redundancy
My Switch Id = 1
Peer Switch Id = 2
```

```
Last switchover reason = none
Configured Redundancy Mode = sso
Operating Redundancy Mode = sso
Switch 1 Slot 1 Processor Information :
-----------------------------------------------
Current Software state = ACTIVE
Uptime in current state = 1 day, 19 hours, 30 minutes
Image Version = Cisco IOS Software, s72033_rp Software (s72033_rp-
ADVENTERPRISEK9_WAN_DBG-VM), Version 12.2(SIERRA_INTEG_070530) INTERIM SOFTWARE
Synced to V122_32_8_11, 12.2(32.8.11)SR on rainier, Weekly 12.2(32.8.11)SX85
Technical Support: http://www.cisco.com/techsupport
Copyright (c) 1986-2007 by Cisco Systems, Inc.
Compiled Thu 31-May-07 02:23 by kchristi
BOOT = sup-bootdisk:s72033-adventerprisek9_wan_dbg-vz.SIERRA_INTEG_070530,1;
CONFIG_FILE =
BOOTLDR =
Configuration register = 0x2102
Fabric State = ACTIVE
Control Plane State = ACTIVE
Switch 2 Slot 1 Processor Information :
-----------------------------------------------
Current Software state = STANDBY HOT (switchover target)
Uptime in current state = 1 day, 19 hours, 30 minutes
Image Version = Cisco IOS Software, s72033_rp Software (s72033_rp-
ADVENTERPRISEK9_WAN_DBG-VM), Version 12.2(SIERRA_INTEG_070530) INTERIM SOFTWARE
Synced to V122_32_8_11, 12.2(32.8.11)SR on rainier, Weekly 12.2(32.8.11)SX85
Technical Support: http://www.cisco.com/techsupport
Copyright (c) 1986-2007 by Cisco Systems, Inc.
Compiled Thu 31-May-07 02:23 by kchristi
BOOT = sup-bootdisk:s72033-adventerprisek9_wan_dbg-vz.SIERRA_INTEG_070530,1;
CONFIG_FILE =
BOOTLDR =
Configuration register = 0x2102
Fabric State = ACTIVE
Control Plane State = STANDBY
```

As this output indicates, if a failure occurs in the active supervisor engine or the active virtual switch, a SSO switchover is invoked, and the hot-standby supervisor engine in switch 2 assumes the role as the active supervisor engine for the Cisco Virtual Switching System. This switchover should take approximately 50 ms.

Intrachassis Availability

The initial release of the Cisco Virtual Switching System supports only a single supervisor per chassis. If a second, or redundant, supervisor is installed in an individual chassis then the redundant supervisor will not fully boot. The redundant supervisor will stop the boot process at the ROMMON stage. In this configuration any device connected to the chassis in a single-homed, or single-attach, manner must rely on the availability of the single supervisor. Therefore the recommendation for connecting to the VSS is to always dual-attach devices.

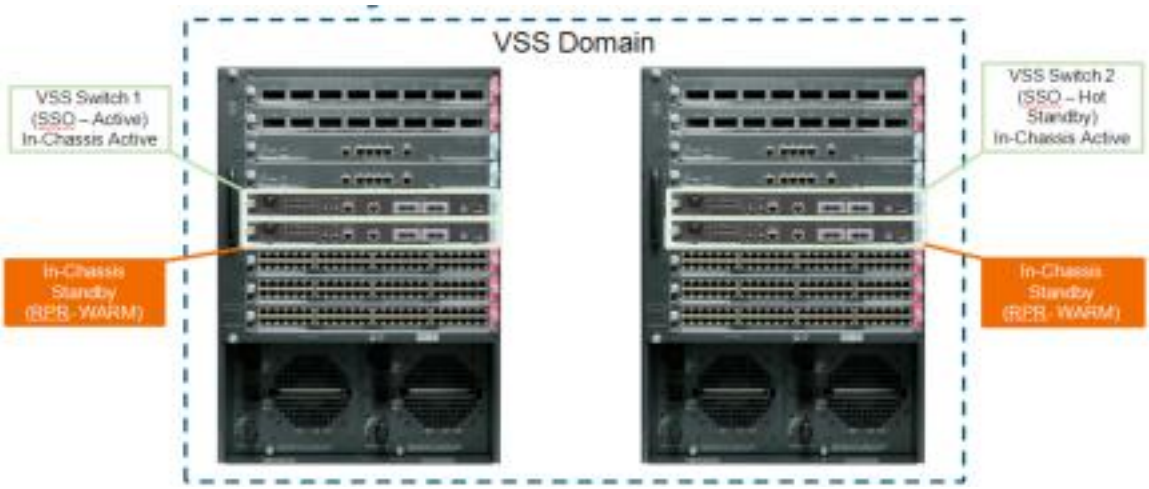
As a result of the single supervisor per chassis support the recovery period for replacing a failed supervisor module is undeterministic in that the recover process requires manual intervention in order to install and initialize a new supervisor in the chassis.

Beginning in the 12.2(33)SXI4 software release, Quad-Sup Uplink Forwarding is supported which allows for a redundant supervisor to fully boot Cisco IOS Software, thereby providing a deterministic recovery option for redundant supervisors in a VSS chassis.

Quad-Sup Uplink Forwarding

Quad-Sup Uplink Forwarding is a supervisor redundancy feature developed just for the Virtual Switching System. The Quad-Sup Uplink Forwarding feature allows the In-chassis standby supervisor to fully boot Cisco IOS Software and provide full functionality of its uplink ports. In addition, the supervisor will perform some basic synchronization with the Active supervisor in the local chassis. From the local chassis perspective whichever supervisor boots first will become the “In-chassis Active” supervisor while the second supervisor will be become the “In-chassis Standby” supervisor. See Figure 21.

Figure 21. Virtual Switching System with Quad-Sup Uplink Forwarding



With the In-chassis Standby supervisor fully booted the uplink ports are fully operational. The uplink ports can be used as part of the VSL port-channel interfaces or other connectivity just like ports on any other line card.

RPR-WARM Redundancy Mode

The In-chassis Standby supervisor runs a new redundancy mode called “RPR- Warm,” stated “RPR minus Warm.” The RPR- Warm redundancy mode is only available in the Virtual Switching System.

In addition to fully booting the in-chassis standby supervisor and providing fully operational uplink ports, the RPR-Warm redundancy mode also provides synchronization of the necessary information to allow the In-chassis Standby supervisor to reload and take over as the in-chassis active supervisor if needed. The RPR- Warm redundancy mode synchronizes the following key variables and data structures:

- Startup-config
- Vlan.dat
- BOOT ROMMON variable
- CONFIG_FILE ROMMON variable
- BOOTLDR ROMMON variable
- DIAG ROMMON variable
- SWITCH_NUMBER ROMMON variable

It is important to note that the RPR- Warm redundancy mode is not a stateful redundancy mode as it applies to the local chassis. In other words if the in-chassis active supervisor does fail then the In-chassis standby supervisor will detect the failure and reload the local chassis. Subsequently the former in-chassis standby supervisor will boot as the in-chassis active supervisor.

During the local chassis reload the line cards will also reload. For devices connected to the Virtual Switching System in a dual-homed manner using a multi-chassis Etherchannel connection or using Layer 3 Equal Cost Multipath links, only the interfaces attached to the chassis performing the reload will be affected. Based on the peer devices ability to detect the loss of link on the interfaces associated with reloading chassis, traffic will switched to the remaining active chassis in the Virtual Switching System. Typically for Multi-chassis Etherchannel or Layer 3 Equal Cost links this is a hardware based subsecond convergence event.

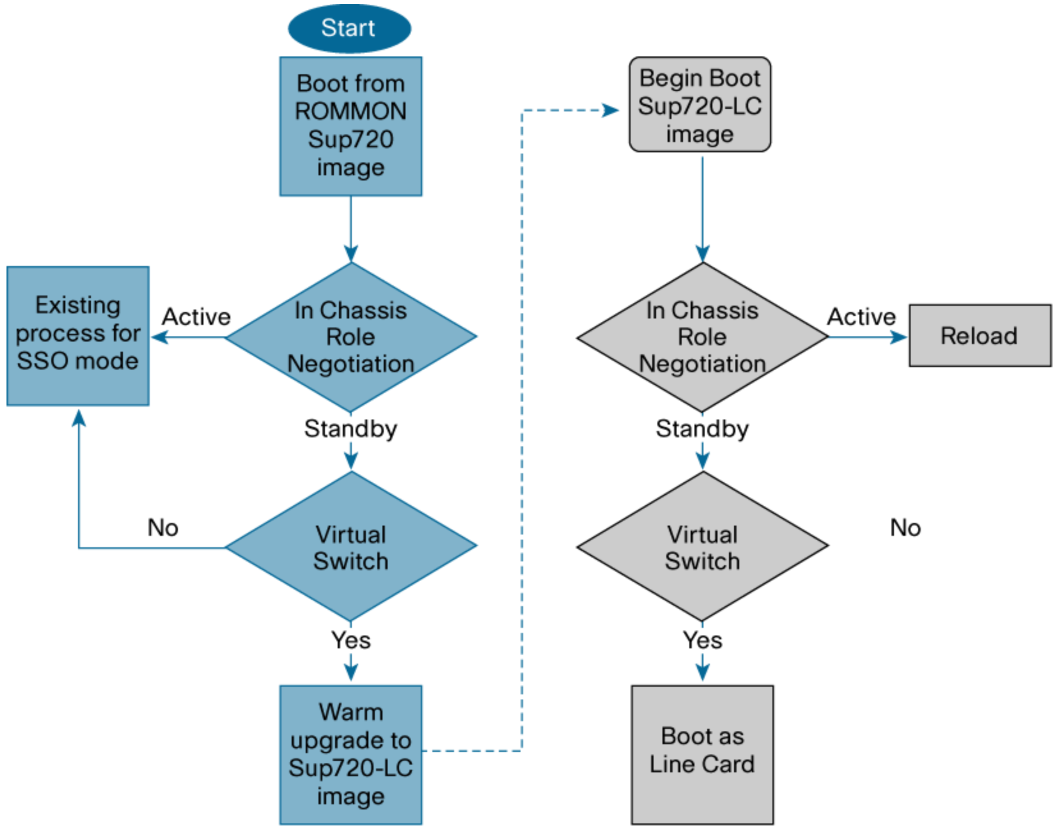
In-chassis Standby Boot Process

The boot process for the second supervisor installed in VSS chassis is different compared to the boot process for a second supervisor in a standalone chassis. Early in the boot process the supervisor will perform a role negotiation with the existing supervisor in the chassis. Once the supervisor determines that it will become the standby supervisor for the chassis it will then detect if the system is configured for the VSS. If the VSS configuration is detected the supervisor will then extract and boot a new software image dedicated to an In-chassis standby supervisor in a VSS. The new software image will proceed to boot and reach the RPR-WARM redundancy mode.

As shown in Figure 22, the second supervisor proceeds to boot to the RPR-WARM redundancy mode by loading a different Cisco IOS Software image. The new software image is specifically developed for a supervisor module operating as the VSS in-chassis standby role. The new image is called the “Sup-LC” image, as in Supervisor-Line Card. The Sup-LC image file is extracted out of the image already running on the supervisor in much the same way as a line card extracts its image file from the Cisco IOS Software image running on the active supervisor. Therefore there are no additional requirements to copy a separate image to the file system of the switch.

Once the supervisor successfully loads the Sup-LC image the supervisor will primarily operate as DFC-enabled line card, In addition the supervisor will perform synchronization of key supervisor subsystems so that if needed the supervisor may reload and assume the role of the in-chassis active supervisor.

Figure 22. Boot Process for the Redundant Supervisor in a VSS Chassis



The only requirement to support Quad-Sup Uplink Forwarding is that both supervisor modules must be configured to boot the 12.2(33)SXI4 or newer software version. The installation process for the redundant supervisor assumes that the in-chassis active supervisor is already configured and converted to the virtual switching mode.

The following screenshot shows the console output at the stage where the redundant supervisor has detected the virtual switch configuration from the in-chassis active supervisor and subsequently extracts the Sup-LC image file and starts to boot as the In-chassis standby.

```
System detected Virtual Switch configuration...
 Interface TenGigabitEthernet 2/5/4 is member of PortChannel 2
 Interface TenGigabitEthernet 2/5/5 is member of PortChannel 2

*Apr  5 20:27:50.747: %SYS-3-LOGGER_FLUSHING: System pausing to ensure console debugging
output.

Firmware compiled 02-Mar-10 17:41 by integ Build [100]

*Apr  5 20:27:50.747: %PFREDUN-6-STANDBY: Initializing as STANDBY processor for this
switch!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
Decompressing the image :
########################################################################################
########################################################################################
# [OK]

 Launching the SPLC image!
                Restricted Rights Legend
```

Once the in-chassis standby has fully loaded the Sup-LC image file, the supervisor will operate very much like a DFC-enabled line card. The Sup-LC image file runs on the supervisor switch processor. Therefore the console

interface will appear as a DFC line card as well. The redundancy mode of the supervisor module can be monitored using the existing show commands.

The following screenshot provides an abbreviated output from the “show switch virtual redundancy” CLI with Quad-Sup Uplink Forwarding enabled.

Configuration Synchronization

When the redundancy-framework progression between the active supervisor engine and standby supervisor engine is reached, the configuration is synchronized between active virtual switch and standby virtual switch. The configuration file contains the configuration for the entire Cisco Virtual Switching System and overwrites the configuration that exists on the standby virtual switch. Both the bulk configuration synchronization and the incremental configuration synchronization are sent through internal control messages across the VSL. As a result, the configuration in the standby virtual switch NVRAM is overwritten.

Virtual Switch Priorities and Switch Preemption

Because the Cisco Virtual Switching System consists of two chassis merged into a single entity, you can designate a particular physical switch to prefer an active role while its peer prefers the standby role. This designation is usually determined by the following by default:

- If switches are initiated at different times, then the switch that is initiated first becomes the active virtual switch.
- If the switches are initiated simultaneously, the switch with the lower switch ID becomes the active virtual switch.

You can alter the default behavior by using the Virtual Switch Priorities feature and the Switch Preemption function.

Virtual Switch Priorities

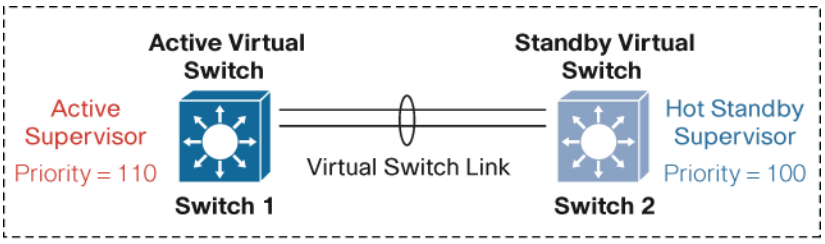
Virtual Switch Priorities are assigned to each member of the Cisco Virtual Switching System under the virtual switch configuration mode. By influencing the weighting of the priorities of each switch, you can deterministically define which physical switch will prefer the active virtual switch role or the standby virtual switch role.

A sample configuration follows:

```
vss#conf t
Enter configuration commands, one per line. End with CNTL/Z.
vss(config)#switch virtual domain 10
vss(config-vs-domain)#switch 1 priority 110
*Jul 7 08:59:11.913: %VSLP-SW1_SPSTBY-5-RRP_RT_CFG_CHANGE: Configured priority value
is different from operational value.
Change will take effect after config is saved and switch is reloaded.
vss(config-vs-domain)#switch 2 priority 100
vss(config-vs-domain)#^Z
vss#
```

Notice from this configuration that the higher-priority value (110) assumes the active virtual switch role and the default priority is set to 100 (Figure 23).

Figure 23. Virtual Switch Priorities



After you save the configuration, you can verify the roles of each virtual switch member with the following command:

```
vss#sh switch virtual role
Switch Switch Status Preempt Priority Role Session ID
Number Oper(Conf) Oper(Conf) Local Remote
-----------------------------------------------------------------
LOCAL 1 UP FALSE(N) 110(110) ACTIVE 0 0
REMOTE 2 UP FALSE(N) 100(100) STANDBY 6179 1085
In dual-active recovery mode: No
```

Note that the switch priorities affect role determination if both virtual switches are initiated simultaneously. If either switch (regardless of priority) is initiated prior to the subsequent switch, it always assumes the role of the active virtual switch. This behavior changes only if the Switch Preemption feature is configured.

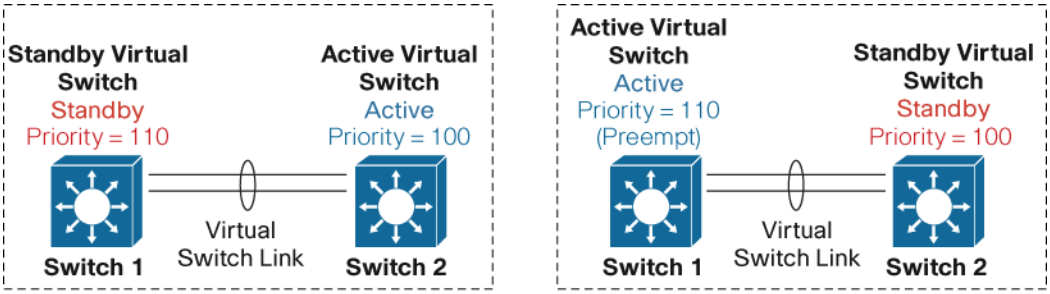
Switch Preemption

As mentioned previously, you can determine virtual switch roles by boot order, switch ID, and switch priorities. However, after you determine the roles, you cannot change them without manual intervention. It may be desirable, however, to always configure a particular physical switch to assume the active virtual switch role. You can achieve this configuration with the Switch Preemption feature.

Switch Preemption works by comparing the Virtual Switch Priorities of the two individual switches after they are both brought online. If the virtual switch with the higher priority has the Switch Preemption feature configured and the

current role of the higher-priority virtual switch is in standby mode, then the current active virtual switch performs a SSO switchover after a preconfigured period of time so that the virtual switch with the higher switch priority takes over as the active virtual switch (Figure 24).

Figure 24. Virtual Switch Preemption



You should enable Switch Preemption only on the switch that has the higher switch priority. With the following command you can enable this function under the virtual switch configuration mode. You can specify an optional timer value from 5 to 20 minutes, where this number represents the number of minutes the current active virtual switch waits after it establishes communication with the peer standby virtual switch through the VSL. The default and minimum time is set to 5 minutes. This timer is important, because it takes a variable amount of time after VSL initialization to initialize the remaining modules and establish network connectivity.

```
vss#conf t
Enter configuration commands, one per line. End with CNTL/Z.
vss(config)#switch virtual domain 10
vss(config-vs-domain)#switch 2 preempt 7
vss(config-vs-domain)#^Z
vss#
```

Use the following command to verify the configuration:

```
vss#show switch virtual role
Switch Switch Status Preempt Priority Role Session ID
Number Oper(Conf) Oper(Conf) Local Remote
-----------------------------------------------------------------
LOCAL 1 UP FALSE(N ) 100(100) ACTIVE 0 0
REMOTE 2 UP TRUE (Y*) 120(120) STANDBY 1170 1366
Standby operational preempt timer(switch 2): 7 minutes
Standby will takeover as active in approx. : 4 minutes
Standby configured preempt timer(switch 2): 7 minutes
In dual-active recovery mode: No
```

Preemption should only be configured if there is a compelling requirement to do so. If preemption is enabled, the convergence time will be longer due to the fact that the switch will experience an additional reload. This additional reload is necessary in order for the new switch to go from an active to a standby state.

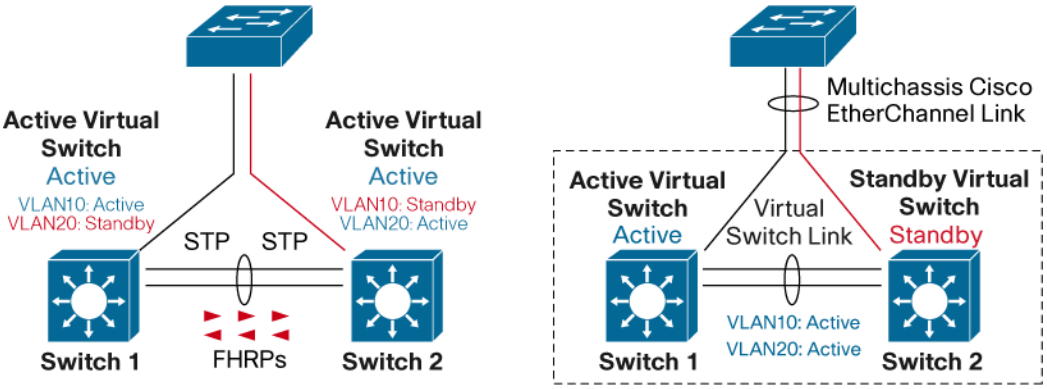
First Hop Redundancy Protocols

First Hop Redundancy Protocols (Hot Standby Router Protocol [HSRP], Gateway Load Balancing Protocol [GLBP], and Virtual Router Redundancy Protocol [VRRP]) provide default gateway redundancy for devices when there are two or more redundant routing nodes but the attached devices are not learning the network topology through Layer 3 routing protocols.

Typically, in a standalone environment, these First Hop Redundancy Protocols (FHRPs) are required to provide a single default gateway that is both redundant and highly available. These protocols typically designate an active forwarder for a given pair of Layer 3 interfaces by using respective hello protocols. Additionally, a separate instance of these hello protocols is run for each pair of interfaces for which the FHRP is configured.

In a Cisco Virtual Switching System environment, the use of FHRPs to provide default gateway redundancy has been obviated in most environments because a single interface and router MAC address are shared across both virtual switches (Figure 25).

Figure 25. First Hop Redundancy Protocols

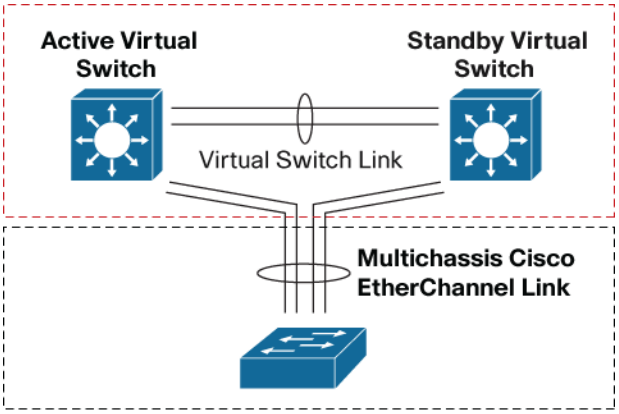


You can enable FHRP to another network device or Cisco Virtual Switching System with the FHRP frames redirected to the route processor of the active virtual switch for processing.

Failure Scenarios

This section assumes the topology shown in Figure 26.

Figure 26. Failure Scenarios



In the topology of Figure 26, the Cisco Virtual Switching System is deployed in the distribution layer of the network, where we will examine the effects that different failures will have on this setup.

Active Supervisor Engine Failure

The standby supervisor engine can detect the failure of the active supervisor engine using one of the following methods:

- Redundancy framework heartbeats sent across the VSL
- VSL Protocol (VSLP)
- Cisco Generic Online Diagnostics (GOLD) failure event
- CDL-based hardware assistance
- Full VSL link down

Upon detecting the failure of the active supervisor, the hot-standby supervisor engine performs an SSO switchover and assumes the role of the active supervisor. The newly active supervisor engine simulates an an online insertion and removal (OIR) removed event for all modules in the previous active chassis to remove those cards from the running chassis inventory. Subsequently, if the VSS is running Quad-Sup Uplink Forwarding or if the failed supervisor is capable of performing a reload the failed chassis will be reloaded. This chassis will then become the VSS Hot Standby (Figure 27).

The effect on the data path is that all the modules on the previous active virtual switch chassis are brought down, resulting in a slight traffic disruption for those traffic flows that were destined to the associated VSS chassis. The duration of disruption depends on the network configuration.

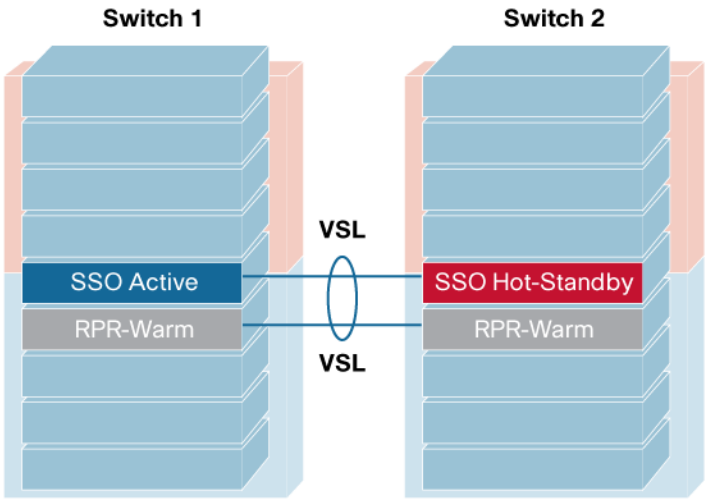
The best practice recommendation[1] dictates that devices connect to the VSS using two or more interfaces in a redundant fashion (across both chassis), such is the case when using multichassis etherchannel or redundant layer 3 equal cost paths across both switches. If there are redundant connections made across the VSS then theoutage duration is typically a sub-second event...

For the multichassis Cisco EtherChannel links, the remote endpoint of the link detects the failure of the active virtual switch ports through LACP or PAgP or a physical link-down situation and uses the links connecting to the standby virtual switch instead.

For network designs where redundant connections cannot be made across the two switches the outage duration depends on how quickly the physical supervisor module can be replaced. The Quad-Sup Uplink Forwarding feature introduced in the 12.2(33)SXI4 feature greatly reduces this outage time as well as automates the configuration and recovery. If the VSS is using Quad-Sup Uplink Forwarding then the chassis experiencing the supervisor failure simply performs a reload and the former in-chassis standby supervisor will then boot as the in-chassis active supervisor module. See the High Availability section of this document for more details on Quad-Sup Uplink Forwarding.

[1] Cisco Campus 3.0 Virtual Switching System Design Guide
http://www.cisco.com/en/US/docs/solutions/Enterprise/Campus/VSS30dg/campusVSS_DG.html

Figure 27. Quad-Sup Uplink Forwarding



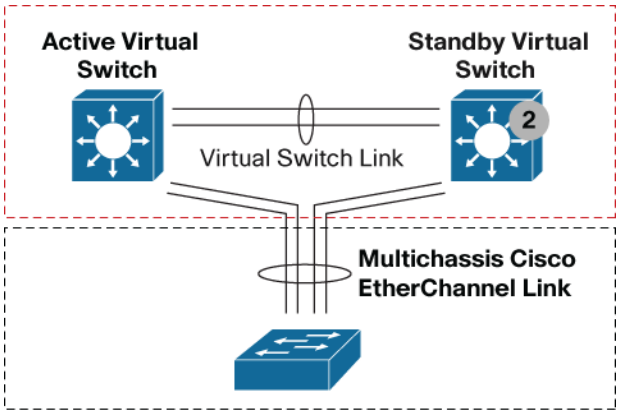
Hot-Standby Supervisor Engine Failure

You can detect the failure of the hot-standby supervisor engine in the following ways:

- Redundancy framework heartbeats sent across the VSL by the active supervisor
- Full VSL link-down situation
- Cisco Generic Online Diagnostics GOLD failure event

Upon detecting the failure of the hot-standby supervisor engine, the active supervisor engine simulates an OIR removed event for all modules in the standby virtual switch. This simulation is performed because the modules on the remote chassis have no connectivity because VSL communication to the modules is proxied by the local supervisor-engine CPU (Figure 28).

Figure 28.



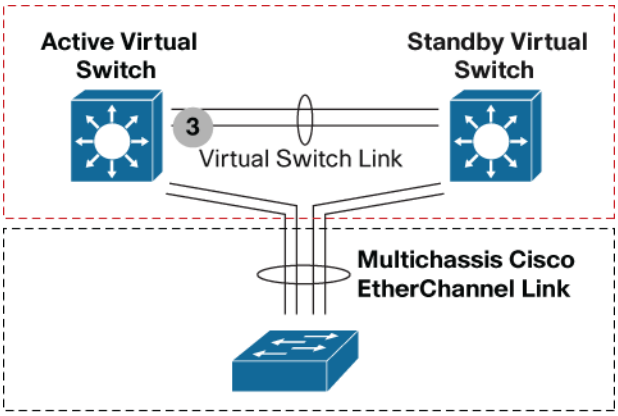
The effect on the data path is that all line cards on the standby virtual switch are brought down. Assuming that adjacent devices are dual-homed to both the active virtual switch and standby virtual switch, only those flows being forwarded through the standby virtual switch are affected, and the time to recovery depends on the mechanism used to detect the link failure (Cisco EtherChannel technology, Layer 3 load balancing, or Spanning Tree Protocol). Upon detecting that the interface connected to the standby virtual switch has failed, traffic resorts to using the link to the active supervisor engine. Those data flows passing through the active virtual switch are not affected. The control plane is not affected because the control-plane functions remain on the active supervisor engine on the active virtual switch. The control plane experiences a removal of all of the standby virtual switch interfaces.

If the standby Virtual Switch contains an in-chassis standby supervisor or in the case of a single supervisor configuration the hot-standby supervisor engine can reinitialize, it will reload. Upon bootup the chassis proceeds with VSL initialization and enters into standby chassis role with all its interfaces becoming operational again.

VSL Single-Link Failure

The failure of a single VSL link is discovered by the active supervisor engine, either through a link-down event or through the failure of periodic VSLP messages sent across the link to check the VSL link state (Figure 29).

Figure 29.



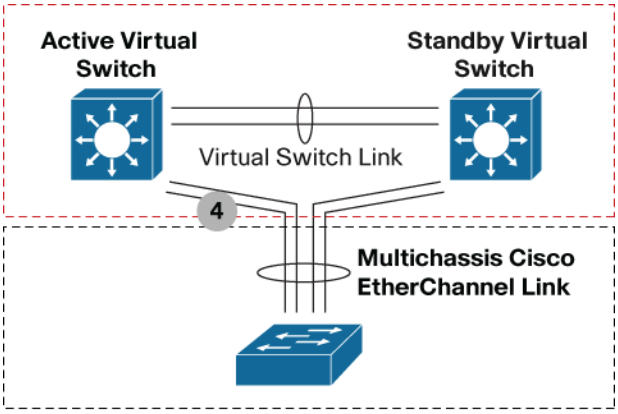
The index values, RBH, and fabric programming for selecting the VSL link will need to be automatically updated to reflect the removal of a link from the VSL. The active supervisor engine sends all of these messages.

Availability is not affected for those data flows that do not use the VSL. For those traffic flows that use the VSL, traffic outage is estimated to be approximately 50 to 100 ms. Notice that the duration of time is slightly faster than that for other multichassis Cisco EtherChannel links because VSL always takes advantage of the adaptive Cisco EtherChannel load-balancing algorithm.

Single Link Failure Within a Multichassis Cisco EtherChannel Link

The failure of a single link within the multichassis Cisco EtherChannel link that is not the last link connecting to a chassis is recognized by either the multichassis Cisco EtherChannel link control protocol (PAgP or LACP) or the link-down event (Figure 30).

Figure 30.



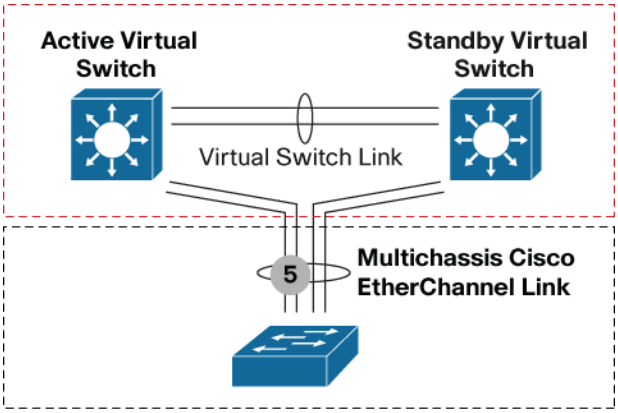
This link failure causes the RBH values to be redistributed among the remaining multichassis Cisco EtherChannel link ports in the local chassis that is the same as a link failure in a standard Cisco EtherChannel link using a standalone Cisco Catalyst 6500. The endpoint (host, switch, router, and so on) on the other end of the multichassis Cisco EtherChannel link detects the link failure and adjusts its load-balancing algorithms to avoid the failed link.

Availability is not affected for those data flows that do not use the failed link. For those traffic flows that do use the failed link, the effect consists of the time it takes to detect the link failure and reprogram the indices within the system, estimated to be approximately 50 to 200 ms. You can achieve faster convergence time by using the adaptive Cisco EtherChannel load-balancing algorithm.

All Links to a Single Chassis Within the Multichassis Cisco EtherChannel Link Fail

When all of the links connecting to a single chassis that is part of the same multichassis Cisco EtherChannel link fail, the port bundle is converted from a multichassis Cisco EtherChannel link to a standard Cisco EtherChannel link and is treated as a single-homed port. The links connecting to the peer chassis remain functional. From this point onward, all traffic from the failed link chassis destined for the Cisco EtherChannel link are sent to the remote chassis through the VSL (Figure 31).

Figure 31.



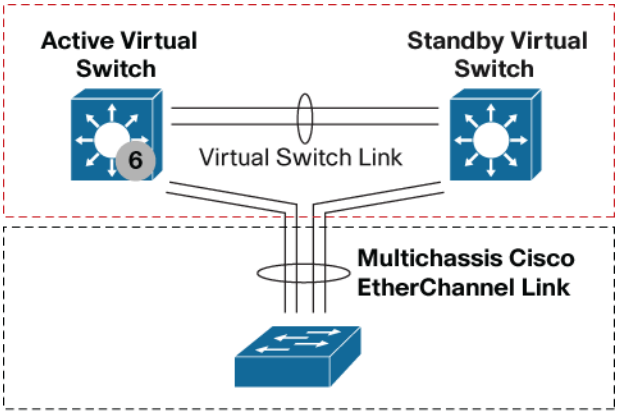
The control protocols managing the Cisco EtherChannel link (PAgP or LACP) continue to originate from the active supervisor engine and are sent out of the standby virtual switch ports through the VSL. The endpoint (host, switch, router, and so on) on the other end of the multichassis Cisco EtherChannel link detects the link failure and adjusts its load-balancing algorithms to avoid the failed link.

Availability is not affected for those data flows that do not use the failed link. For those traffic flows that do use the failed link, the effect consists of the time it takes to detect the link failure and reprogram the indices within the system.

Active Virtual Switch Chassis Failure

The failure of the active virtual switch chassis is handled and detected in the same manner as the failure of the active supervisor engine (Figure 32).

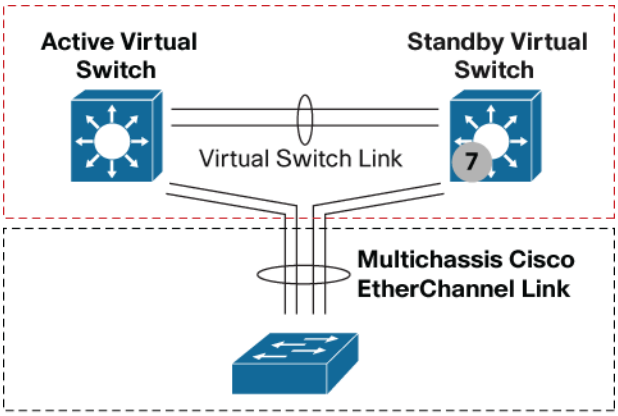
Figure 32.



Standby Virtual Switch Chassis Failure

The failure of the standby virtual switch chassis is handled and detected in the same manner as a failure of the hot-standby supervisor engine (Figure 33).

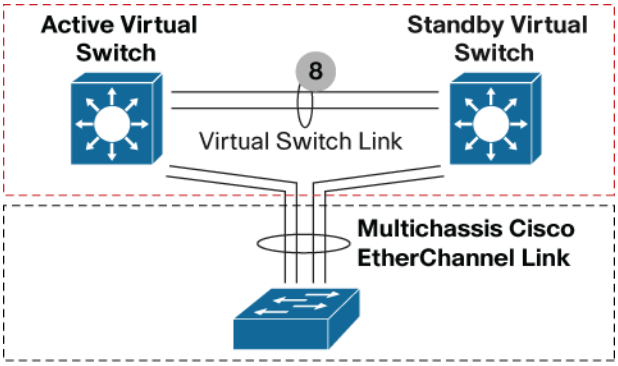
Figure 33.



Complete VSL Failure (Dual Active)

The active supervisor engine discovers the failure of the VSL either through a link-down event or through the failure of the periodic VSLP messages sent across the member links to check the VSL link status. From the perspective of the active virtual switch chassis, the standby virtual switch is lost. The standby virtual switch chassis also views the active virtual switch chassis as failed and transitions to active virtual switch state through an SSO switchover. This scenario is known as a dual-active scenario (Figure 34).

Figure 34.



In this case, each virtual switch assumes the role as the active virtual switch and each virtual switch controls only its local ports. However, there will most likely also be some global Layer 2 and Layer 3 configuration, and the interface configuration for the multichassis Cisco EtherChannel links will be applied on both chassis . Duplication of this configuration can possibly have adverse effects to the network topology and traffic.

At Layer 3, any virtual interfaces (for example, port channels, SVIs, loopbacks, and so on) are duplicated on both chassis, causing duplicate IP addresses on the network. Any secure communications such as SSH and the cryptography feature set have the same set of keys on both chassis. At Layer 2, the spanning tree has the same bridge ID in both switches, possibly causing conflict. In general, this condition causes the same effect as when two routers or switches within a network contain the same configuration file.

To avoid this disruptive scenario, you should configure the VSL as a multiple-link port channel and spread it across all the available supervisor engines and modules within the chassis. You should also run the individual members of the VSL across separate physical paths when possible.

In some circumstances this configuration may not be possible; Cisco Virtual Switching System has different mechanisms to address this dual-active scenario:

- Configuration of the VSL failure-detection feature
- Detection of a dual-active scenario
- Action taken to resolve the situation
- Recovery behavior upon restoring the VSL

Detection Mechanisms and Configuration

Because of the challenges to distinguish a remote chassis power failure and a VSL failure, each chassis attempts to detect its peer chassis in order to avoid the dual-active scenario. In a dual-active scenario, we must assume that the VSL links cannot be used in any way to detect the failure. The only remaining options are to use alternative paths that may or may not exist between the two chassis.

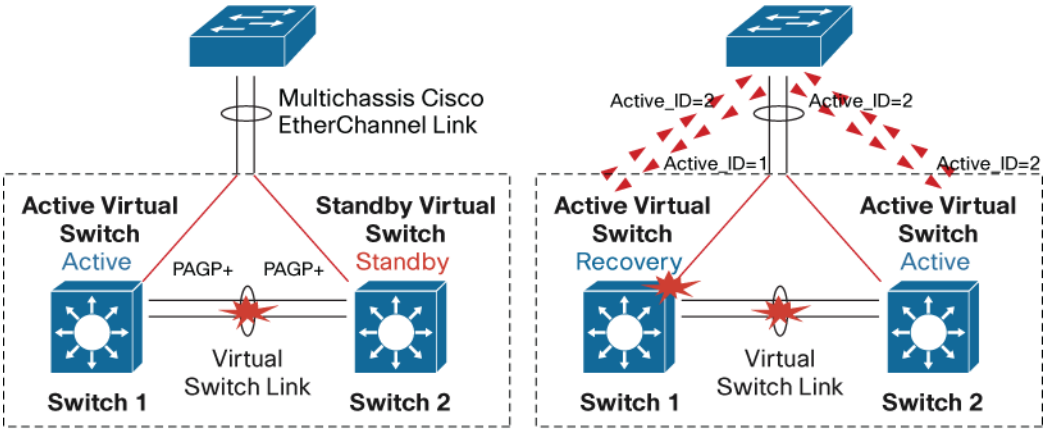
Currently, there are currently three mechanisms for detecting a dual-active scenario.

- Enhanced PAgP
- Layer 3 BFD
- Fast-hello

Enhanced PAgP

With the introduction of Cisco Virtual Switching System in the first software release, an enhancement to the PAgP protocol (Enhanced PAgP or PAgP+) has been implemented to assist in the dual-active detection. A list of all software releases for respective switch platforms supporting Enhanced PAgP is included in Table 1 (Figure 35).

Figure 35. Dual-Active Detection Mechanisms



The result of this detection is that the standby virtual switch (switch 2) always transitions to become an active virtual switch and the active virtual switch (switch 1) always enters into recovery mode.

Upon the detection of VSL going down on switch 2, the switch will immediately transmit a PAgP message on all port channels enabled for Enhanced PAgP dual-active detection, with a Type-Length-Value (TLV) containing its own Active ID = 2. When the access switch receives this PAgP message on any member of the port channel, it detects that it has received a new active ID value and considers such a change as an indication that it should consider switch 2 to be the new active virtual switch. In turn, the access switch modifies its local active ID value to Active ID = 2 and immediately sends a message to both virtual switches on all members of the port channel with the new Active ID = 2 to indicate that it now considers switch 2 to be the active virtual switch.

Form this point onward, the access switch sends TLVs containing Active ID = 2 to the virtual switches in all its regularly scheduled PAgP messages.

Use the following commands to configure the Cisco Virtual Switching System to take advantage of dual-active detection using Enhanced PAgP:

```
vss#conf t
Enter configuration commands, one per line. End with CNTL/Z.
vss(config)#switch virtual domain 10
vss(config-vs-domain)#dual-active detection pagp
vss(config-vs-domain)#dual-active trust channel-group 20
vss(config-vs-domain)#
```

To verify the configuration and make sure that Enhanced PAgP is compatible with its neighbors, issue the following command:

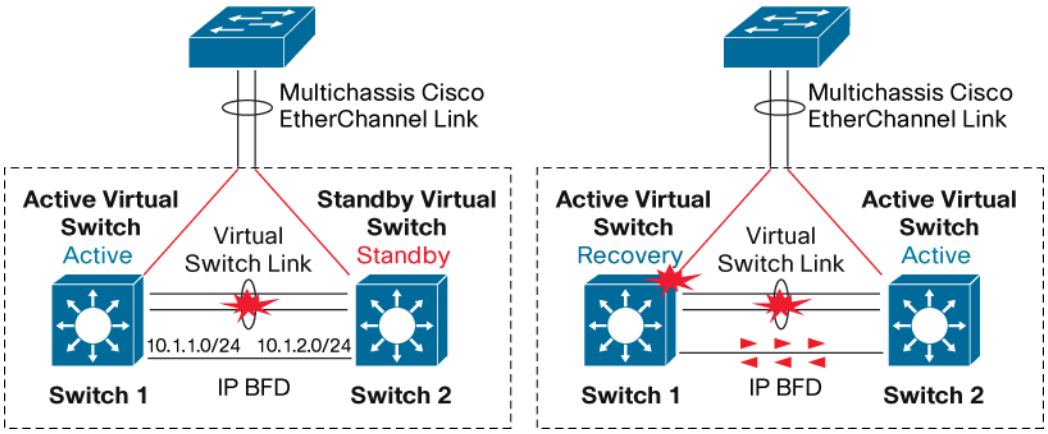
```
vss#sh switch virtual dual-active pagp
PAgP dual-active detection enabled: Yes
PAgP dual-active version: 1.1
```

```
Channel group 10 dual-active detect capability w/nbrs
Dual-Active trusted group: No
Dual-Active Partner Partner Partner
Port Detect Capable Name Port Version
Gi1/8/1 No SAL0802SHG 5/2 N/A
Gi2/8/1 No SAL0802SHG 5/1 N/A
Channel group 20 dual-active detect capability w/nbrs
Dual-Active trusted group: Yes
Dual-Active Partner Partner Partner
Port Detect Capable Name Port Version
Te1/1/1 Yes vs-access-2 Te5/1 1.1
Te2/1/1 Yes vs-access-2 Te5/2 1.1
```

Layer 3 BFD

If no Enhanced PAgP neighbors are available to assist in dual-active detection, then another method is required to perform this function; use of a dedicated Layer 3 direct link heartbeat mechanism between the virtual switches is an inexpensive way to determine whether or not a dual-active scenario has occurred (Figure 36).

Figure 36. Dual-Active Detection Using IP-BFD



Bidirectional Forwarding Detection (BFD) assists in the fast detection of a failed VSL, bringing in natively the benefits that BFD offers, such as subsecond timers and pseudo-preemption. To take advantage of this feature, you must first configure BFD on the selected interfaces that will be participating in IP-BFD dual-active detection, noting that these interfaces must be directly connected to each other:

```
vss#conf t
Enter configuration commands, one per line. End with CNTL/Z.
vss(config)#int gig 1/5/1
vss(config-if)#ip address 10.1.1.1 255.255.255.0
vss(config-if)#bfd interval 100 min_rx 100 multiplier 50
vss(config-if)#no shutdown
vss(config-if)#int gig 2/5/1
```

```
vss(config-if)#ip address 10.1.2.1 255.255.255.0
vss(config-if)#bfd interval 100 min_rx 100 multiplier 50
vss(config-if)#no shutdown
vss(config-if)#exit
```

Note that in a Cisco Virtual Switching System environment, both interfaces are seen to be Layer 3 routed interfaces on the same logical router and hence require different network addresses even though they are directly connected together.

To enable IP-BFD for dual-active detection, use the following configuration:

```
vss(config)#switch virtual domain 10
vss(config-vs-domain)#dual-active detection bfd
vss(config-vs-domain)#dual-active pair interface gig1/5/1 interface gig2/5/1 bfd
adding a static route 10.1.2.0 255.255.255.0 Gi1/5/1 for this dual-active pair
adding a static route 10.1.1.0 255.255.255.0 Gi2/5/1 for this dual-active pair
vss#show switch virtual dual-active bfd
Bfd dual-active detection enabled: Yes
Bfd dual-active interface pairs configured:
interface-1 Gi1/5/1 interface-2 Gi2/5/1
```

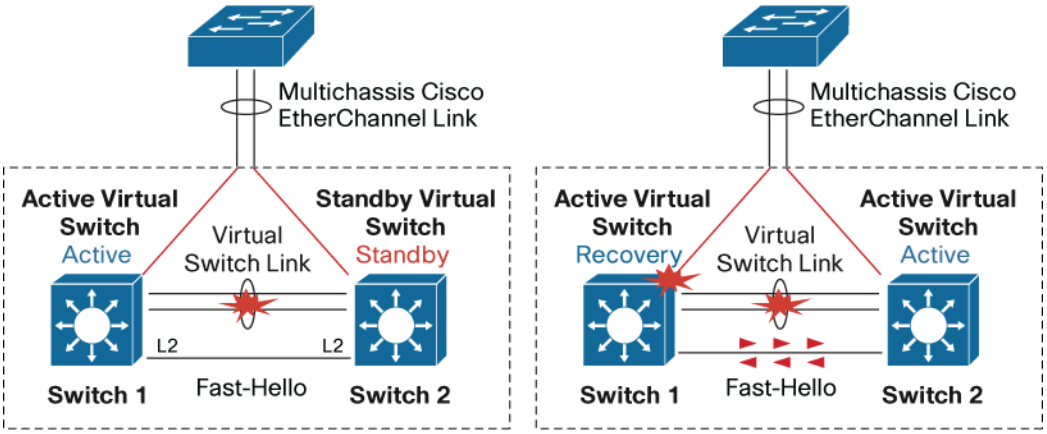
Note that by configuring these commands, static routes are automatically added for the remote addresses and are installed in the Routing Information Base (RIB) only if a dual-active scenario occurs. As a result, no packets are forwarded between the switches through the heartbeat interfaces until the VSL is brought down.

When the VSL does go down, a unique internal MAC address (selected from the pool of MAC addresses reserved for the line card) is assigned for each of the local interfaces, and sending BFD heartbeat packets brings up BFD neighbors. If the standby virtual switch has taken over as active, a BFD “adjacency-up” event is generated, indicating that a dual-active situation has occurred.

Fast-Hello Detection

With 12.2(33)SXI, a Fast-Hello detection mechanism was introduced. This is similar to IP-BFD, as it is a dedicated direct link heartbeat between the VSS switches. However, Fast-Hello is a L2 connection and you can configure up to 4 non-VSL links. The two chassis will periodically exchange fast-hello heartbeat packets that contain the switch state. When the VSL fails, the fast hello packets are no longer received on each switch, thus indicating that a dual-active scenario has occurred (Figure 37).

Figure 37. Dual-Active Detection Using Fast-Hello



In order for fast-hello to operate, the interfaces must be enabled with this protocol. Unlike IP-BFD, that only exchanges hello packets after the VSL fails, Fast-Hello exchanges heartbeat messages throughout the period that the VSL remains up, thereby reducing the time it takes to detect the dual-active scenario.

```
vss(config)# interface fastethernet 1/2/40
vss(config-if)# dual-active fast-hello
WARNING: Interface FastEthernet1/2/40 placed in restricted config mode. All extraneous
configs removed!
```

Up to four interfaces (directly connected) on each chassis can be configured. These interfaces must be physical interfaces, no logical ports (i.e. SVI).

```
vss(config-if)# no shutdown
vss(config-if)# exit
vss(config)# exit
vss# show run interface fastethernet 1/2/40
interface FastEthernet1/2/40
no switchport
no ip address
dual-active fast-hello
end
```

Once fast-hello has been configured, the existing configuration on the interface will be automatically removed and will only be restricted for fast-hello configurations. Also, UDLD will be disabled on fast-hello pairs.

```
vss(config)# switch virtual domain 10
vss(config-vs-domain)# dual-active detection fast-hello
vss(config-vs-domain)# exit
```

Action upon Dual-Active Detection

Upon detecting the dual-active condition, the original active chassis enters into recovery mode and brings down all of its interfaces except the VSL and nominated management interfaces, effectively removing the device from the network.

To nominate specific interfaces to be excluded from being brought down during dual-active detection recovery, use the following commands:

```
vss(config)#switch virtual domain 10
vss(config-vs-domain)#dual-active exclude interface gigabitEthernet 1/5/3
WARNING: This interface should only be used for access to the switch when in
dual-active recovery mode and should not be configured for any other purpose
vss(config-vs-domain)#dual-active exclude interface gigabitEthernet 2/5/3
WARNING: This interface should only be used for access to the switch when in
dual-active recovery mode and should not be configured for any other purpose
vss(config-vs-domain)#
```

To verify this configuration is correct, issue the following commands:

```
vss#sh switch virtual dual-active summary
Pagp dual-active detection enabled: Yes
Ip bfd dual-active detection enabled: Yes
Interfaces excluded from shutdown in recovery mode:
Gi1/5/3
Gi2/5/3
In dual-active recovery mode: No
```

You will see the following messages on the active virtual switch to indicate that a dual-active scenario has occurred:

```
*Jun 26 16:06:36.157: %VSLP-SW2_SPSTBY-3-VSLP_LMP_FAIL_REASON: Port 5/4: Link down
*Jun 26 16:06:36.782: %VSLP-SW1_SP-3-VSLP_LMP_FAIL_REASON: Port 5/4: Link down
*Jun 26 16:06:36.838: %VSL-SW1_SP-5-VSL_CNTRL_LINK: vsl_new_control_link NEW VSL
Control Link 5/5
*Jun 26 16:06:37.037: %VSLP-SW1_SP-3-VSLP_LMP_FAIL_REASON: Port 5/5: Link down
*Jun 26 16:06:37.097: %VSL-SW1_SP-2-VSL_STATUS: -========= VSL is DOWN =========-
```

The following messages on the standby virtual switch console indicate that a dual-active scenario has occurred:

```
*Jun 26 16:06:36.161: %VSL-SW2_SPSTBY-5-VSL_CNTRL_LINK: vsl_new_control_link NEW VSL
Control Link 5/5
*Jun 26 16:06:37.297: %VSLP-SW2_SPSTBY-3-VSLP_LMP_FAIL_REASON: Port 5/5: Link down
*Jun 26 16:06:37.297: %VSL-SW2_SPSTBY-2-VSL_STATUS: -========= VSL is DOWN =========-
*Jun 26 16:06:37.301: %PFREDUN-SW2_SPSTBY-6-ACTIVE: Initializing as Virtual Switch
ACTIVE processor
*Jun 26 16:06:37.353: %SYS-SW2_SPSTBY-3-LOGGER_FLUSHED: System was paused for 00:00:00
to ensure console debugging output.
*Jun 26 16:06:37.441: %DUALACTIVE-SP-1-VSL_DOWN: VSL is down - switchover, or possible
dual-active situation has occurred
```

Recovery from Dual-Active Scenario

If a VSL flap occurs, the system recovers automatically. Upon a link-up event from any of the VSL links, the previous active supervisor engine that is now in recovery mode reloads itself, allowing it to initialize as the hot-standby supervisor engine. If the peer chassis is not detected because the VSL is down again, the dual-active detection mechanism determines whether or not the peer chassis is active. If the peer chassis is detected, this event is treated as another VSL failure event and the chassis once again enters into recovery mode.

When the VSL is restored, the following messages are displayed on the console and the switch in recovery mode (previous active virtual switch) reloads:

```
*Jun 26 16:23:34.877: %DUALACTIVE-1-VSL_RECOVERED: VSL has recovered during dual-active situation: Reloading switch 1
*Jun 26 16:23:34.909: %SYS-5-RELOAD: Reload requested Reload Reason: Reload Command.
<…snip…>
***
*** --- SHUTDOWN NOW ---
***
*Jun 26 16:23:42.012: %SYS-SW1_SP-5-RELOAD: Reload requested
*Jun 26 16:23:42.016: %OIR-SW1_SP-6-CONSOLE: Changing console ownership to switch processor
*Jun 26 16:23:42.044: %SYS-SW1_SP-3-LOGGER_FLUSHED: System was paused for 00:00:00 to ensure console debugging output.
System Bootstrap, Version 8.5(1)
Copyright (c) 1994-2006 by cisco Systems, Inc.
<…snip…>
```

After the chassis reloads, it reinitializes and the supervisor engine enters into standby virtual switch mode. If Switch Preemption is configured to prioritize this chassis to become active, it assumes this role after the preempt timer expires.

Finally, traffic convergence associated with a VSL failure scenario involves the dual-active detection mechanisms, the recovery mode operations and finally the restoration period which involves a reload of the previously active VSS chassis. In each one of the three previous stages there is a possibility for some traffic disruption. Typically the disruption will be similar to the NSF/SSO switchover scenarios described previously but other factors from the overall network design can have an effect as well. These include the types of connections to the VSS, either L2 or L3 Multichassis Etherchannels or Equal Cost Paths. For more detailed analysis on convergence times associated with the VSS failure scenarios please refer to following document:

Campus 3.0 Virtual Switching System Design Guide
http://www.cisco.com/en/US/docs/solutions/Enterprise/Campus/VSS30dg/campusVSS_DG.html

Quality of Service

Quality of service (QoS) handling on the Cisco Catalyst 6500 Series Switches can be separated into two distinct areas of responsibility: port-based QoS features and forwarding-engine (PFC or DFC) features. Both areas operate together to make sure of differentiated servicing of traffic throughout the system.

In a Cisco Virtual Switching System environment, proper QoS handling becomes even more important because of the following reasons:

- Control traffic between the two Cisco Virtual Switching System switches (active virtual switch and standby virtual switch) should be prioritized and not be dropped.
- The existence of the VSL and also multichassis Cisco EtherChannel links represents potential points of congestion that must be properly accounted for.

Additionally, the nature of dual-homing logical connections across different chassis and forwarding engines represents a change in the way features such as policing and marking work; these features also need to be properly accounted for.

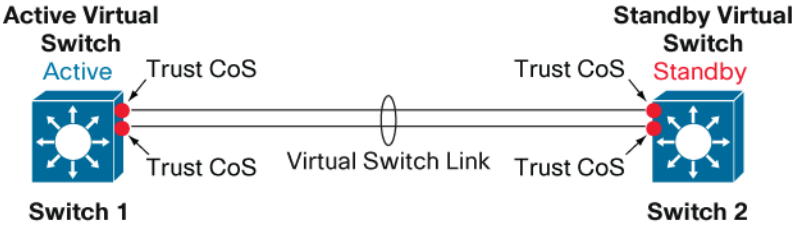
VSL as a Congestion Point

In terms of a system-level perspective, the VSL can be viewed as a backplane connection that bonds the two virtual switch chassis together into a single, logical entity. Although provisions have been made to Cisco EtherChannel and Equal Cost Multipath (ECMP) hashing mechanisms (refer to the section “Cisco Etherchannel Concepts”), under certain circumstances (in the case of single-homed connections whether by design or if failures occur) it may be possible to oversubscribe the links that form the VSL.

VSL should always consist of at least 2 ports of 10 Gigabit Ethernet connections, but because of hash inefficiencies it may be possible to oversubscribe a single VSL member port. Therefore, correct prioritization needs to occur on the VSL.

Correct prioritization is accomplished by always provisioning the VSL as a link that is in trust-CoS mode, provisioning that not only maintains the internal differentiated services code point (DSCP) markings set by either ingress switch but also sets up default receive and transmit queues and properly assigns the appropriate colored frames to the correct queues (Figure 38).

Figure 38. Virtual Switch Link QoS



The following output shows that port channel 2 is configured as a VSL, and it has the QoS configuration of trust CoS programmed by default. Also note that removing or modifying this trust command is not permitted:

```
interface Port-channel2
no switchport
no ip address
switch virtual link 2
mls qos trust cos
end
vss#conf t
Enter configuration commands, one per line. End with CNTL/Z.
vss(config)#int po2
```

```
vss(config-if)#no mls qos trust cos
HWIF-QOS: QoS configs are not allowed on VSL Portgroup
vss(config-if)#mls qos trust dscp
HWIF-QOS: QoS configs are not allowed on VSL Portgroup
```

The following output shows that for a member port of the VSL, the relevant CoS-queue mappings have already been provisioned for both ingress and egress queues, even without QoS globally enabled:

```
vss#sh queueing int te2/5/4
Interface TenGigabitEthernet2/5/4 queueing strategy: Weighted Round-Robin
Port QoS is enabled
Trust state: trust COS
Extend trust state: not trusted [COS = 0]
Default COS is 0
Queueing Mode In Tx direction: mode-cos
Transmit queues [type = 1p3q4t]:
Queue Id Scheduling Num of thresholds
---------------------------------------
01 WRR 04
02 WRR 04
03 WRR 04
04 Priority 01
WRR bandwidth ratios: 100[queue 1] 150[queue 2] 200[queue 3]
queue-limit ratios: 50[queue 1] 20[queue 2] 15[queue 3] 15[Pri Queue]
queue tail-drop-thresholds
--------------------------
1 70[1] 100[2] 100[3] 100[4]
2 70[1] 100[2] 100[3] 100[4]
3 100[1] 100[2] 100[3] 100[4]
queue random-detect-min-thresholds
----------------------------------
1 40[1] 70[2] 70[3] 70[4]
2 40[1] 70[2] 70[3] 70[4]
3 70[1] 70[2] 70[3] 70[4]
queue random-detect-max-thresholds
----------------------------------
1 70[1] 100[2] 100[3] 100[4]
2 70[1] 100[2] 100[3] 100[4]
3 100[1] 100[2] 100[3] 100[4]
```

```
WRED disabled queues:
queue thresh cos-map
---------------------------------------
1 1 0
1 2 1
1 3
1 4
2 1 2
2 2 3 4
2 3
2 4
3 1 6 7
3 2
3 3
3 4
4 1 5
Queueing Mode In Rx direction: mode-cos
Receive queues [type = 2q4t]:
Queue Id Scheduling Num of thresholds
-----------------------------------------
01 WRR 04
02 WRR 04
WRR bandwidth ratios: 10[queue 1] 90[queue 2]
queue-limit ratios: 80[queue 1] 20[queue 2]
queue tail-drop-thresholds
--------------------------
1 70[1] 80[2] 90[3] 100[4]
2 100[1] 100[2] 100[3] 100[4]
queue random-detect-min-thresholds
----------------------------------
1 40[1] 40[2] 50[3] 50[4]
2 100[1] 100[2] 100[3] 100[4]
queue random-detect-max-thresholds
----------------------------------
1 70[1] 80[2] 90[3] 100[4]
2 100[1] 100[2] 100[3] 100[4]
WRED disabled queues: 2
```

```
queue thresh cos-map
---------------------------------------
1 1 0 1
1 2 2 3
1 3 4
1 4 6 7
2 1 5
2 2
2 3
2 4
<…snip…>
```

A restriction has been imposed, however, that does not permit you to modify QoS settings on the VSL ports in the initial release of software. Hence, you can modify only the default queue, drop threshold, and buffer depth settings.

```
vss(config)#int te2/5/4
vss(config-if)#priority-queue cos-map 1 2
HWIF-QOS: QoS configs are not allowed on VSL Portgroup
```

Additionally, policy maps used for classification or policing are also forbidden on the VSL and its respective members. This feature will be addressed in future software releases.

Control Traffic over VSL

Multiple types of control traffic must be parsed between the two virtual switches, including VSLP and other inband messages. These special control frames are tagged with a special bit internal to the system, indicating that they require specialized treatment and are automatically assigned to the priority queue of the interface for expedited delivery. As a result, no extra configuration is required.

The priority queue is always serviced first, prior to any other Deficit Weighted Round Robin (DWRR) queues.

Using Supervisor Engine 720-10G VSS 10 Gigabit Ethernet Uplink Ports as VSL Interfaces

You can use the 10 Gigabit Ethernet uplink ports on the Supervisor Engine 720-10G VSS to form a VSL. In addition to the two 10 Gigabit Ethernet uplink ports, there are also three additional Gigabit Ethernet ports: two Small Form-Factor Pluggable (SFP) interfaces and a 10/100/1000 RJ-45 interface. You cannot use these ports as VSL interfaces.

If you use only the 10 Gigabit Ethernet ports, you can optimize the queue structure to take full advantage of an 8q4t queue structure on receive and 1p7q4t queue structure on transmit. However, if you use both the Gigabit Ethernet interfaces and 10 Gigabit Ethernet interfaces concurrently, then the 10 Gigabit Ethernet interfaces take on the queue structure of the Gigabit Ethernet interfaces, which is 4q4t on receive and 1p3q4t on transmit.

If you want to use only the 10 Gigabit Ethernet ports, you must shut down the Gigabit Ethernet ports and globally configure an extra CLI on the system:

```
VSS(config)#mls qos 10g-only
Error: following ports have to be shut to enable 10g-only mode:
Gi1/5/1 Gi1/5/2 Gi1/5/3
Command Rejected!
```

```
VSS(config)#interface range gigabitEthernet 1/5/1 - 3
VSS(config-if-range)#shut
VSS(config-if-range)#exit
VSS(config)#mls qos 10g-only
HWIF-QOS: Queuing qos cfg (wrr-queue/rcv-queue/priority-queue) will be reset to
default on Supervisor Slot 5 interfaces!
VSS(config)#
VSS#sh interfaces tenGigabitEthernet 1/5/4 capabilities | include QOS
QOS scheduling: rx-(8q4t), tx-(1p7q4t)
QOS queueing mode: rx-(cos,dscp), tx-(cos,dscp)
```

Applying Policies

Classification or policing policies are applied to the system through the Modular QoS CLI (MQC) mechanisms, which use class maps and policy maps. Each policy map can use multiple class maps to make up a policy map, and you can define these policy classes for different types of traffic flows.

On the Cisco Catalyst 6500, you can define up to 255 class maps per policy map, with a total of 1024 class maps per system, implying that across the Virtual Switching System a maximum of 1024 class maps can be supported.

MQC in Cisco IOS Software allows the separation of class maps from policy maps and the separation of these maps from the application on an interface. The initial release of software also has a limitation in that it allows for only a limited number of interfaces to be indexed uniquely for QoS purposes. As a result, you can apply QoS policies only on Layer 3 interfaces (SVIs, physical interfaces, port channels, and so on) and on Layer 2 Cisco EtherChannel links.

In a Cisco Virtual Switching System, application of policies on physical Layer 2 interfaces is now supported in 12.2(33)SXI and above.

Policing

Policing is the process of inspecting whether traffic on a given port or within a VLAN has exceeded a predefined rate. If that traffic is out of profile (that is, the rate of the traffic stream exceeds the predefined rate), either excess data can be dropped or its priority value marked down.

Two types of policers are supported on the Cisco Catalyst 6500 Series Switches: aggregate policers and microflow policers. Although you can implement both types, they are subject to different caveats in a Cisco Virtual Switching System environment. The next section addresses some of these caveats relating to policers.

Aggregate Policing

Aggregate policers limit the amount of traffic received or transmitted in or out a port. The aggregate policer applies to all traffic on a port or VLAN that matches a specified QoS ACL. If the aggregate policer is applied to a single interface, it counts all matching traffic (that matches the classifying ACL) coming into the interface toward the policer. If the aggregate policer is applied to a VLAN, then all of the matching traffic coming in any of the ports in that VLAN is counted toward the stated rate.

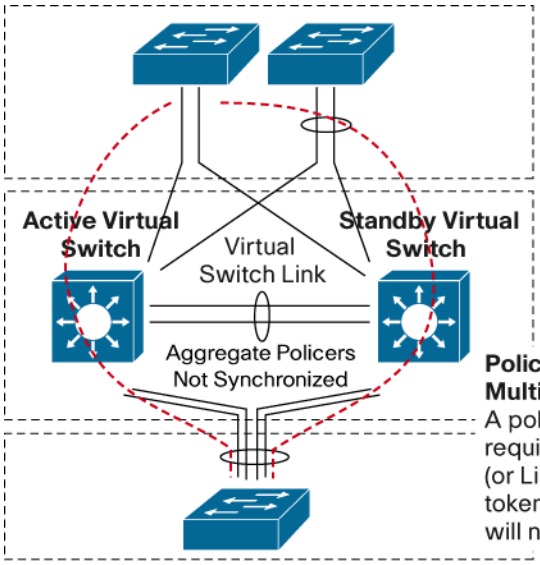
There are two forms of aggregate policers:

- **Per-interface aggregate policers**: These policers are applied to an individual interface using the police command within a policy-map class. You can apply these map classes to multiple interfaces, but the policer polices each interface separately.
- **Named aggregate policers or shared aggregate**: These policers are applied to a group of ports and police traffic across all interfaces cumulatively. Name aggregates are applied using the mls qos aggregate police command.

The policing function is typically handled by the ingress forwarding engine (either PFC or DFC). A critical restriction to implementing aggregate policers in a Cisco Virtual Switching System environment is the current lack of distributed aggregate policing capabilities across different forwarding engines. That is, if a policer is required to span across multiple forwarding engines, each forwarding engine keeps track of its own token-bucket quota and hence generally results in the under-policing of traffic. This situation usually manifests itself when applying policers on the following types of interfaces (Figure 39):

- VLAN interfaces that consist of member ports that belong to multiple forwarding engines
- Port-channel interfaces that consist of member ports that belong to multiple forwarding engines
- Shared aggregate policers that consist of member ports that belong to multiple forwarding engines

Figure 39. Aggregate Policing Within Cisco Virtual Switching System



Microflow Policing and User-Based Rate Limiting

Microflow policing allows you to police individual traffic flows at a given rate. Depending on the flow mask used (whether it is a unique source or destination MAC address, source or destination IP address, or TCP/User Datagram Protocol [UDP] port numbers), you can use microflow policing to limit the amount of data sent or received for that flow on a port or VLAN basis. In the microflow definition, you can either drop packets that exceed the prescribed rate limit or have their DSCP value marked down.

User Based Rate Limiting (UBRL) is a form of microflow policing that also supports the policing of individual flows. The primary difference is that you can specify a source-only flow or destination-only flow rather than the full source or destination address of the packet.

For both microflow policing and UBRL, the NetFlow table on either the PFC or DFC is used to track the individual flows as well as maintain the flow statistics and—most importantly—track the rate of ingress traffic for each individual flow by implementing a separate token bucket for each NetFlow entry. Cisco Virtual Switching System also has the restriction that each forwarding engine is responsible for the calculation of each flow independently and cannot be synchronized across multiple forwarding engines.

As a result, only flows that always arrive on the same forwarding engine are policed correctly; otherwise they are under-policed. Generally, this situation allows only the following flow masks for use on multichassis Cisco EtherChannel link interfaces:

- **Source and destination**: Source and destination IP address
- **Interface, source, and destination**: Input interface, source, and destination IP address
- **Full**: Source, destination IP address, IP, and TCP/UDP source and destination ports if present
- **Interface, full**: Input interface, source, destination IP address, IP, and TCP/UDP source and destination ports if present

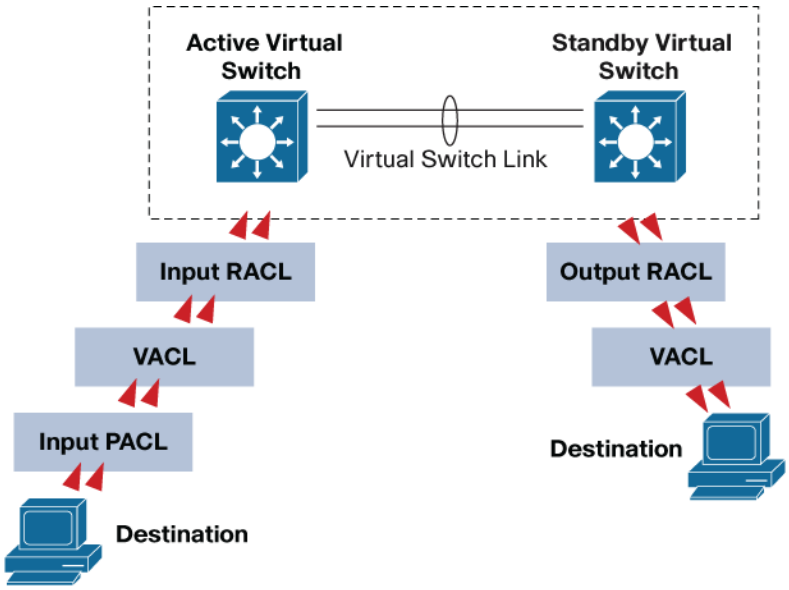
As a result, UBRL does not yield the desired behavior if applied to multichassis Cisco EtherChannel link interfaces or other distributed Cisco EtherChannel interfaces because they are source-only or destination-only by nature.

Access Control Lists

This section examines the way access lists are modified in the Cisco Virtual Switching System environment. Essentially three types of ACLs are supported in a Cisco Catalyst 6500 system as of Cisco IOS Software Release 12.2(33)SXH (Figure40):

- Router ACLs (RACLs)
- VLAN ACLs (VACLs)
- Port-based ACLs (PACLs)

Figure 40. Access-List Processing



All of these ACLs are compiled by the system and programmed into hardware-based ternary content addressable memory (TCAM) on the system PFCs or DFCs. Within a Cisco Virtual Switching System environment, these ACLs

are compiled by the active route processor for the entire system (on the active virtual switch) and programmed to all PFCs and DFCs in the system.

Router ACLs

Router ACLs refers to all ACLs that are applied to interfaces that also have an IP address specified, including Layer 3 physical routed interfaces, Layer 3 SVIs, as well as port-channel interfaces. Directional by nature, RACLs apply only to traffic that is routed through those specific interfaces.

In a Cisco Virtual Switching System environment, RACLs do not change significantly because they can be applied to all Layer 3 interfaces across the entire system (on switch 1, switch 2, or both). Global TCAM show commands, however, have been extended to account for the switch keyword. For example:

```
vss#sh tcam counts switch 1
Used Free Percent Used Reserved
---- ---- ------------ --------
Labels:(in) 4 4092 0
Labels:(eg) 2 4094 0
ACL_TCAM
--------
Masks: 77 4019 1 72
Entries: 49 32719 0 576
QOS_TCAM
--------
Masks: 22 4074 0 18
Entries: 22 32746 0 144
LOU: 0 128 0
ANDOR: 0 16 0
ORAND: 0 16 0
ADJ: 3 2045 0
vss#sh tcam counts switch 2
Used Free Percent Used Reserved
---- ---- ------------ --------
Labels:(in) 4 4092 0
Labels:(eg) 2 4094 0
ACL_TCAM
--------
Masks: 77 4019 1 72
Entries: 49 32719 0 576
QOS_TCAM
--------
Masks: 22 4074 0 18
```

```
Entries: 22 32746 0 144 LOU: 0 128 0
ANDOR: 0 16 0
ORAND: 0 16 0
ADJ: 3 2045 0
```

VLAN ACLs

VACLs refers to all ACLs that are applied to Layer 2 VLANs directly and affect both traffic that is switched within the VLAN for which the VACL is applied and traffic that is routed through the VLAN. VACLs are bidirectional.

In a Cisco Virtual Switching System environment, VACLs do not change significantly because they can be applied across VLANs that are local to a particular virtual switch as well as across the entire Cisco Virtual Switching System. Global TCAM show commands have also been extended to account for the switch keyword.

Port-Based ACLs

PACLs refers to those ACLs that are applied directly to a physical port that is also configured as a Layer 2 switchport. Note that when an IP address is applied to such an interface, the ACL becomes a RACL. PACLs are directional by nature, and only ingress PACLs are supported.

For software releases prior to 12.2(33)SXI4 there are some changes made to the way PACLs are applied in a Cisco Virtual Switching System environment. They relate to the current software restriction that does not allow the system to consecutively address more than 2000 ports from a Layer 2 ACL indexing perspective. This limitation implies that PACLs cannot be applied to physical orphan ports—ports that exist on a single chassis only. You can apply PACLs only on Layer 2 Cisco EtherChannel links or multichassis Cisco EtherChannel links. This behavior is evidenced by the CLI not being available on physical Layer 2 interfaces:

```
vss(config)#int gig 1/5/2
vss(config-if)#switchport
vss(config-if)#ip ?
Interface IP configuration subcommands:
admission Apply Network Admission Control
arp Configure ARP features
auth-proxy Apply authenticaton proxy
<…snip…>
vss(config)#int port-channel 102
vss(config-if)#switchport
vss(config-if)#ip ?
Interface IP configuration subcommands:
access-group Specify access control for packets
admission Apply Network Admission Control
arp Configure ARP features
auth-proxy Apply authenticaton proxy
<…snip…>
```

PACLs on physical layer 2 interfaces are supported in VSS beginning in the 12.2(33)SXI4 software.

Americas Headquarters
Cisco Systems, Inc.
San Jose, CA

Asia Pacific Headquarters
Cisco Systems (USA) Pte. Ltd.
Singapore

Europe Headquarters
Cisco Systems International BV
Amsterdam, The Netherlands

Cisco has more than 200 offices worldwide. Addresses, phone numbers, and fax numbers are listed on the Cisco Website at www.cisco.com/go/offices.

Printed in USA C11-429338-03 05/10