



Cisco Virtual Switching System 1440

Tim Clegg



Introduction

Senior Consulting Engineer
Advanced Technology Services Team
Global Technology Resources, Inc. (GTRI)

My focus and specialty lies within:
routing & switching technologies
some data center technologies:
Server Load Balancing
Web Application Firewalls
WAAS (WAN optimization)

The majority of the clients I work with range from local to Federal government, local, national and multi-national enterprise and the dot-com industries.



Outline

- **Overview**

- What is Virtual Switching System (VSS)?
- Why Use VSS?
- VSS Components

- **Technical Deep-Dive**

- Hardware and Software Requirements
- Virtual Switch Link (VSL)
- Redundancy
- Dual-active
- Installation
- Best practices
- Limitations and Restrictions
- Scenarios



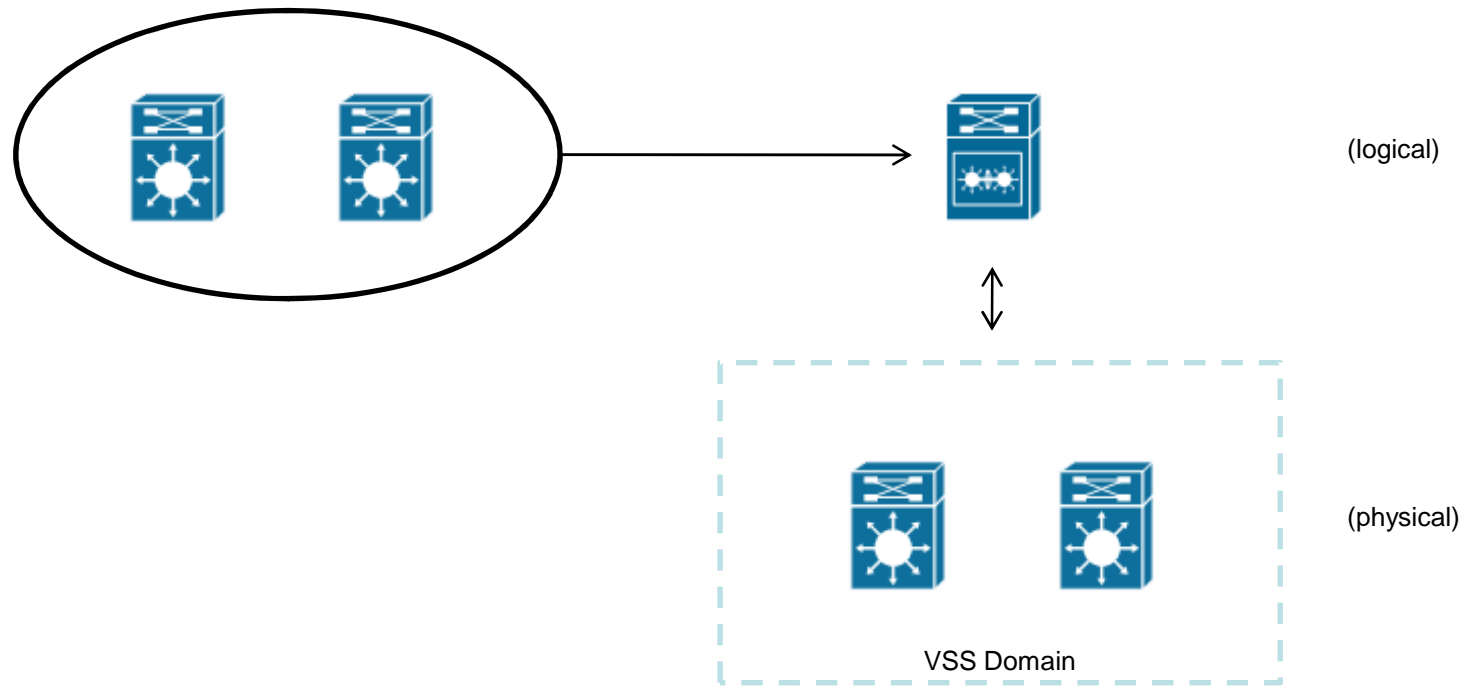
What we will not cover

- **Network Architecture**
 - VSS Placement
 - Principles of proper physical architecture (triangles, not squares, diverse paths, etc.)
- **Protocol Tuning**
 - Timers and protocol convergence issues
- **6500 Architecture**
 - Limitations with 6513s, differences in hardware architectures (63xx, 65xx, 67xx, etc.)

Overview

What is VSS?

VSS combines two physical switches into a single logical switch





Overview

Why Use VSS?

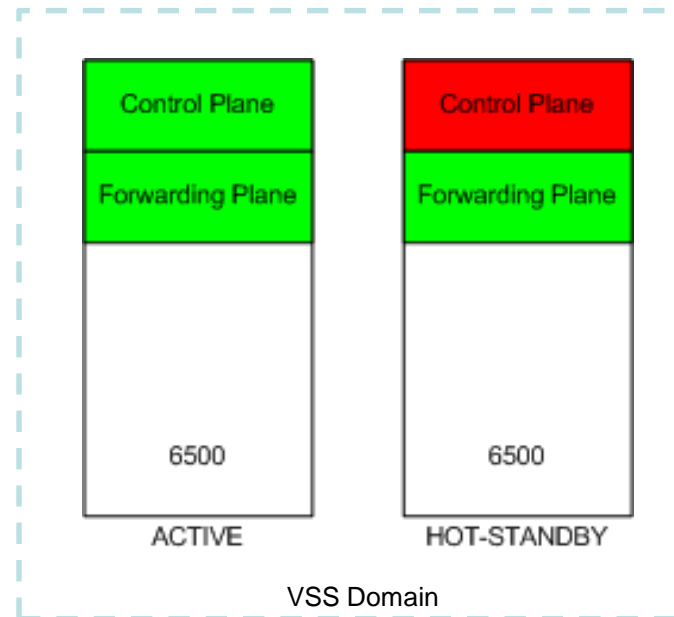
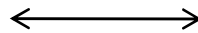
VSS allows us to

- Maximize network performance
- Increase network availability
- Simplify network architecture
- Reduce administrative burden
- Support virtualization

Overview

Why Use VSS?

- Maximize network performance
 - Single active control plane
 - Both forwarding planes are active





Overview

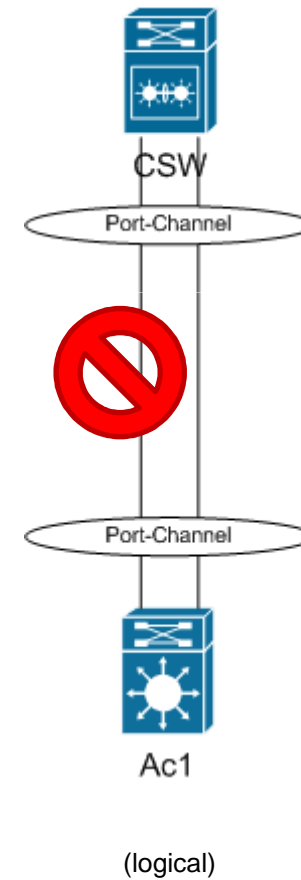
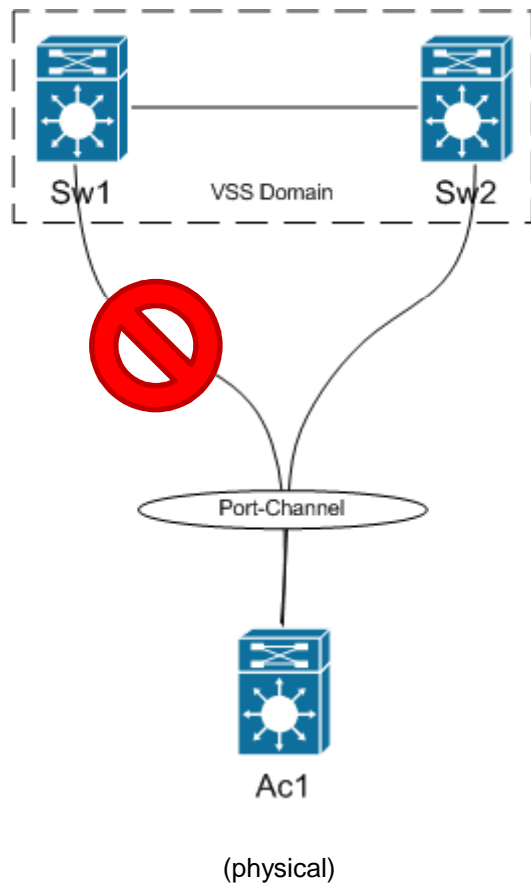
Why Use VSS?

- Increase network availability
 - Multichassis EtherChannel (MEC) allows us to
 - reduce the number of neighbor adjacencies
 - eliminate SPF, DUAL and STP calculations in the event of a
 - single link failure
 - VSS chassis failure
 - perform IOS and chassis upgrades with minimal disruption

Overview

Why Use VSS?

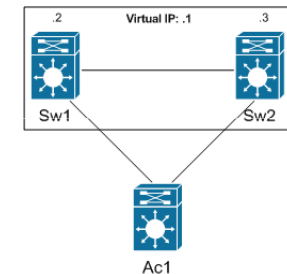
- Increase network availability



Overview

Why Use VSS?

- Simplify network architecture
 - Simple STP design
 - No need to have different root bridges so as to load-balance different VLANs across different links
 - No FHRP
 - HSRP, VRRP, GLBP, etc.
 - Fewer IP addresses used
 - Typically one IP is used for each physical switch, then another IP is used for the FHRP virtual IP (total of 3 IPs typically used)
 - Fewer neighbor adjacencies and smaller routing protocol databases





Overview

Why Use VSS?

- Reduce administrative burden
 - 1 active control plane = 1 logical switch
 - Using VSS there is a 50% reduction in the number of switches that must be managed
 - Time to properly prepare for a change window can be drastically reduced
 - Approximately 60% of network failures are caused by human error.
 - Reduce the number of times you and I have to touch the switch and we might reduce network failures



Overview

Why Use VSS?

- Support virtualization
 - 10Gbps
 - Is available with or without VSS
 - Multichassis EtherChannel (MEC)
 - Is only available with VSS
 - Doubles our bandwidth to/from VSS switches
 - Not just for switch-to-switch connections
 - Any device that supports EtherChannels (LAGs (Link Aggregation), etc.) can utilize MEC – no specific client requirements
 - Think about the Virtual Machine (VM) infrastructure
 - servers, blade servers, etc.



Overview

Why Use VSS?

Is my environment right for VSS?

High-bandwidth environments

Virtualization (VMs)

10Gbps

High-availability environments

Minimize network downtime

Redundant infrastructure

Reduction in number of devices to manage

Overview

Why Use VSS?

Financial Discussion

	Design #1	Design #2	Design #3	Design #4
# Chassis	1	1	2	2
# Sups in each chassis	1	2	1	2
Sup Redundancy?	No	Yes (intra-chassis)	Yes (inter-chassis)	Yes (intra & inter chassis)
Advantages	Inexpensive solution.	Redundant sups. Single device to manage.	Chassis redundancy. Sup redundancy. Additional power supply redundancy. VSS: single device to manage. VSS: maximized throughput.	Maximum amount of redundancy (chassis, sup and power supply).
Disadvantages	Lacks redundancy (sup & chassis).	Lacks chassis redundancy.	Non-VSS: multiple devices to manage. Non-VSS: sub-optimal throughput.	Multiple devices to manage.
Summary	Not recommended for most environments.	Common solution.	Perfect for VSS! If using VSS, this maximizes the equipment investment.	Can be the most expensive solution.



Overview

VSS Components

- Two 6500-series switch chassis'
 - Each populated with a single supervisor
 - No additional line cards are needed
 - Unless you need more than a couple of interfaces... 😊
 - Power supplies may be in combined or redundant mode
- Virtual Switch Link (VSL)
 - This is a regular 10Gbps interface which is configured as a VSL
 - The connection used for communication between the two chassis'
 - Think of this as the “stacking cable” between the two physical VSS switches



Overview

VSS Components

- The final end product is:
 - Two Catalyst 6500-series switches configured within the same VSS domain
 - Two sups (one in each chassis)
 - Two control planes
 - » 1 control plane is **active**
 - » 1 control plane is **hot-standby**
 - Two forwarding planes
 - » Both forwarding planes are **active**
 - Both switch chassis' within the VSS domain are connected and managed via the Virtual Switch Link (VSL)
 - Interface format
 - Pre-VSS: <type><module #>/<interface #>
 - Example: Gi1/25
 - Post-VSS: <type><chassis #>/<module #>/<interface #>
 - Example: Gi1/1/25 (switch 1, module 1, interface 25)

```
interface GigabitEthernet2/2/6
no switchport
no ip address
shutdown
```



Outline

- **Overview**
 - What is Virtual Switching System (VSS)?
 - Why Use VSS?
 - VSS Components
- **Technical Deep-Dive**
 - **Hardware and Software Requirements**
 - Virtual Switch Link (VSL)
 - Redundancy
 - Dual-active
 - Installation
 - Best practices
 - Limitations and Restrictions
 - Scenarios



Technical Deep-Dive

Hardware and Software Requirements

❑ Catalyst 6500-series Chassis

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

❑ Supervisor Engines

❑ VS-S720-10G-3C/XL

❑ IOS 12.2(33)SXH1 and later



Technical Deep-Dive

Hardware and Software Requirements

- ❑ Line Cards
 - ❑ WS-X6700-series (CEF720)
 - ❑ If a DFC is installed, it must be the same type as the system (DFC3C or DFC3CXL)
 - ❑ Mismatch can result in the line card being denied power until it is resolved
 - ❑ No support for non-6700-series line cards!
- ❑ VSL Supported Interfaces
 - ❑ only 10Gbps interfaces are supported
 - ❑ Two 10Gbps interfaces on sup
 - ❑ WS-X6708-10G-3C/XL
 - ❑ WS-X6716-10G-3C/XL
 - ❑ VSL support requires performance mode to be configured



Technical Deep-Dive

Hardware and Software Requirements

□ Service-Modules

Service Module	Minimum Cisco IOS Release	Minimum Module Release
Network Analysis Module (NAM-1 and NAM-2) (WS-SVC-NAM-1 and WS-SVC-NAM-2)	12.2(33)SXH1	3.6(1a)
Application Control Engine (ACE10 and ACE20) (ACE10-6500-K9 and ACE20-MOD-K9)	12.2(33)SXI	A2(1.3)
Intrusion Detection System Services Module (IDSM-2) (WS-SVC-IDSM2-K9)	12.2(33)SXI	6.0(2)E1
Wireless Services Module (WiSM) (WS-SVC-WISM-1-K9)	12.2(33)SXI	3.2.171.6
Firewall Services Module (FWSM) (WS-SVC-FWM-1-K9)	12.2(33)SXI	4.0.4



Outline

- **Overview**
 - What is Virtual Switching System (VSS)?
 - Why Use VSS?
 - VSS Components
- **Technical Deep-Dive**
 - Hardware and Software Requirements
 - **Virtual Switch Link (VSL)**
 - Redundancy
 - Dual-active
 - Installation
 - Best practices
 - Limitations and Restrictions
 - Scenarios



Technical Deep-Dive

Virtual Switch Link (VSL)

- Module ASICs encapsulate and decapsulate a Virtual Switch Header (VSH) to all traffic traversing the VSL
 - VSH is the same size (32-bytes) and format used internally within the 6500 (the internal compact header) and contains
 - Ingress port index
 - Destination port index
 - VLAN
 - Class of Service (CoS)
 - More...
- Only supported on 10Gbps interfaces



Technical Deep-Dive

Virtual Switch Link (VSL)

- The VSL carries the following kinds of traffic:
 - VSS-specific management protocols between the two VSS chassis'
 - Link Management Protocol (LMP)
 - Assists in communication of parameters necessary to establish communication between the two chassis
 - Tracks unidirectional links
 - Exchanges the chassis ID
 - Exchanges additional undocumented parameters
 - Role Resolution Protocol (RRP)
 - Assesses the hardware and software versions to determine if a VSS can be formed (determine if incompatibilities exist, etc.)
 - Resolves each chassis role (**active** and **hot-standby**)



Technical Deep-Dive

Virtual Switch Link (VSL)

- The VSL carries the following kinds of traffic:
 - Transit traffic
 - Normal transit traffic
 - Each chassis always prefers a local egress interface over a remote (located on the other chassis) egress interface
 - » Attempts to minimize VSL transit traffic
 - When a local egress interface doesn't exist, an egress interface on the other VSS chassis is used (assuming that one exists)
 - Service-module traffic
 - Traffic between service-modules within the two VSS chassis'
 - This can be normal transit traffic to the active module, control traffic between the two service-modules (state replication, config synchronization, etc.) and any other module-to-module communication
 - SPAN traffic



Technical Deep-Dive

Virtual Switch Link (VSL)

- The VSL is configured as a virtual-link interface

- Not a trunk
- Not a routed port
- Example:

```
int por200
description VSL Link for VSS
switch virtual link 1
no shut
!
int ra ten5/1 -2
description VSL Link for VSS
channel-group 200 mode on
no shut
!
```

- Use the VSLP ping to troubleshoot VSL connectivity

- Example:

```
VSS-SWITCH#ping vslp output int te1/5/4
```

```
Type escape sequence to abort.
```

```
Sending 5, 100-byte VSLP ping to peer-sup via output port 1/5/4, timeout is 2 seconds:
```

```
!!!!!
```

```
Success rate is 100 percent (5/5), round-trip min/avg/max = 12/12/12 ms
```

```
VSS-SWITCH#
```



Outline

- **Overview**
 - What is Virtual Switching System (VSS)?
 - Why Use VSS?
 - VSS Components
- **Technical Deep-Dive**
 - Hardware and Software Requirements
 - Virtual Switch Link (VSL)
 - **Redundancy**
 - Dual-active
 - Installation
 - Best practices
 - Limitations and Restrictions
 - Scenarios



Technical Deep-Dive

Redundancy

- Same features as is found in non-VSS platforms!
 - Nonstop Forwarding (NSF)
 - Enable per protocol
 - Allows for minimized downtime upon the loss of a routing protocol peering/adjacency
 - Essentially caches routing protocol data for a pre-determined amount of time allowing the remote peer to remediate the issue, allowing both peers to re-establish the adjacency
 - Stateful Switchover (SSO)
 - Syncs data between the **active** and **hot-standby** sups
 - Boot-environment
 - Running-configuration
 - Protocol state
 - Line card status
 - Lots more...
 - If one sup fails, the remaining (hot-standby) sup has a pretty complete picture and is ready to take over (switch to active) with minimal impact



Technical Deep-Dive

Redundancy

- Active Supervisor Engine Responsibilities
 - Handles all control-plane functions
 - Active console port
 - System administration (SSH/telnet, etc.)
 - Port management
 - OIR events
 - Etc.
 - Forwards (switches/routes) transit traffic
- Hot-Standby Supervisor Engine Responsibilities
 - Synchronizes with active sup
 - Waits for the active sup to fail (at which point it will take over)
 - Forwards (switches/routes) transit traffic



Outline

- **Overview**
 - What is Virtual Switching System (VSS)?
 - Why Use VSS?
 - VSS Components
- **Technical Deep-Dive**
 - Hardware and Software Requirements
 - Virtual Switch Link (VSL)
 - Redundancy
 - **Dual-active**
 - Installation
 - Best practices
 - Limitations and Restrictions
 - Scenarios

Technical Deep-Dive

Dual-Active

- A dual-active scenario occurs when the VSL is completely broken
 - Both switches will go active (the control planes)
 - This is why it's called dual-active – both are active at the same time
- Results in network instability
 - Both sups claim ownership of the same MAC address, IP address, etc.

Scenario	Normal		Dual-Active	
Switch #	1	2	1	2
Control-plane	Active	Hot-standby	Active	Active
Forwarding-plane	Active	Active	Active	Active



Technical Deep-Dive

Dual-Active

- Mitigate this scenario from happening by using:
 - Enhanced Port Aggregation Protocol (ePAgP)
 - VSLP Fast-Hello
 - Bidirectional Forwarding Detection (BFD)
- Flexible dual-active detection design options
 - Any and all of the methods/protocols may be used at the same time
 - Multiple interfaces can be configured for VSLP Fast-Hello
 - Allows for redundancy and flexibility in the detection of dual-active scenarios



Technical Deep-Dive

Dual-Active

- Dual-active Detection Methods and Protocols
 - Enhanced Port Aggregation Protocol (ePAgP)
 - Cisco recommends that this method be used
 - More “moving parts” than the other methods
 - Utilizes the switches directly connected to the VSS pair to detect the dual-active scenario
 - Requires hardware and software platform support from the switches connected to the VSS pair
 - Detection Performance:
 - Sub-second detection
 - IOS Support:
 - IOS 12.2(33)SXH1 and later
 - Available in all VSS-capable IOS versions

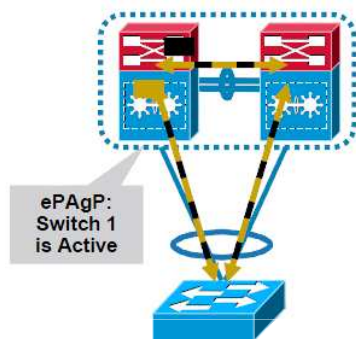
Non-VSS Switch ePAgP Requirements:

Platform	Software	Comments
6500	12.2(33)SXH	Sup720 and Sup32
45xx and 49xx	12.2(44)SG	
29xx, 35xx & 37xx	12.2(46)SE	37xx Stack – no support

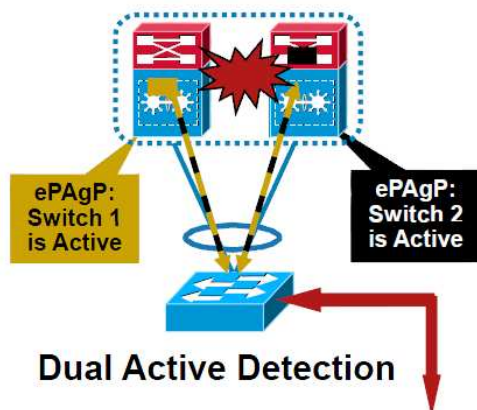
Technical Deep-Dive

Dual-Active

- Dual-active Detection Methods and Protocols
 - Enhanced Port Aggregation Protocol (ePAgP)



Normal Mode



Dual Active Detection

```
cr2-6500-vss#sh switch virtual dual-active summary
Pagg dual-active detection enabled: Yes
Bfd dual-active detection enabled: Yes
```

No interfaces excluded from shutdown in recovery mode

```
In dual-active recovery mode: Yes
Triggered by: PAgP detection
Triggered on interface: Gi2/0/10
Received id: 0019.a927.3000
Expected id: 0019.a924.e800
```

```
%PAGP_DUAL_ACTIVE-SP-3-RECOVERY_TRIGGER: PAgP running on Gi6/1 informing virtual switches of dual-active:
new active id 0019.a927.3000, old id 0019.a924.e800
```

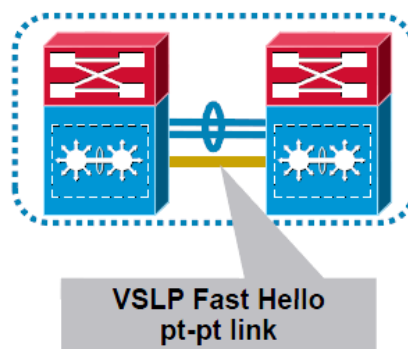
Technical Deep-Dive

Dual-Active

- Dual-active Detection Methods and Protocols

- VSLP Fast-Hello

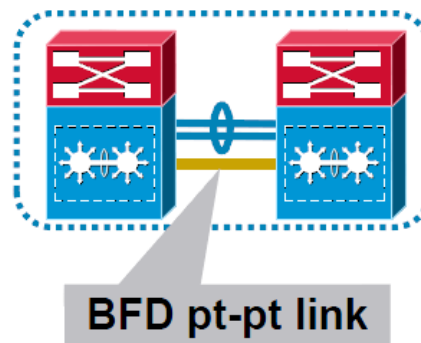
- Utilizes a secondary non-VSL connection between the two switch chassis' in the VSS domain
 - Simple, **fast**, cut-and-dry dual-active detection
 - Does not utilize any other non-VSS switches, so there are no hardware/software dependencies
 - Multiple interfaces can be configured for use as VSLP Fast-Hello dual-active detection interfaces
 - Detection Performance:
 - Sub-second detection
 - IOS Support:
 - IOS 12.2(33)SXI and later



Technical Deep-Dive

Dual-Active

- Dual-active Detection Methods and Protocols
 - Bidirectional Forwarding Detection (BFD)
 - Utilizes a secondary non-VSL connection between the two switch chassis' in the VSS domain
 - Simple, **slow**, cut-and-dry dual-active detection
 - Does not utilize any other non-VSS switches, so there are no hardware/software dependencies
 - Detection Performance:
 - Does not provide sub-second detection
 - IOS Support:
 - IOS 12.2(33)SXH1 and later
 - Like ePAgP, this is available in all VSS-capable IOS versions

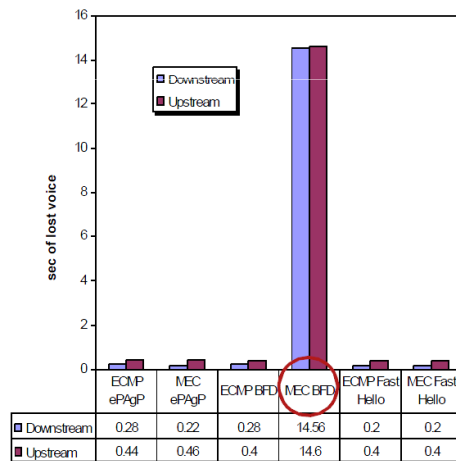


Technical Deep-Dive

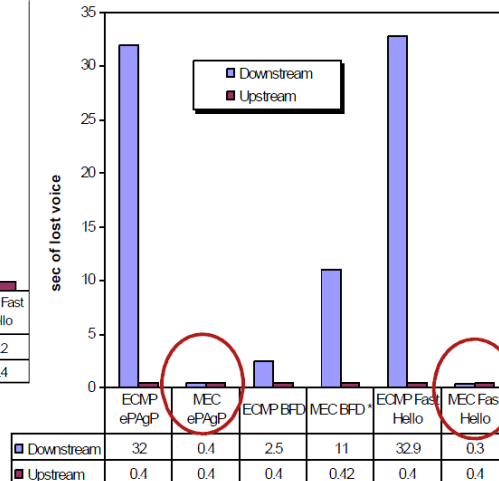
Dual-Active

- Each dual-active detection method/protocol has strengths and weaknesses
- There is no cut-and-dry one-size-fits-all solution
- Examine your environment, research the options and choose the best fit

VSS dual active convergence with EIGRP



VSS dual active convergence with OSPF



Dual ACTIVE	ePAgP	Fast Hello	BFD
EIGRP with ECMP Core			
EIGRP with MEC Core			⚠
OSPF with ECMP Core	⚠	⚠	
OSPF with MEC Core			⚠

= Sub-second Recovery ⚠ = Non-sub-second or variable recovery

* The MEC BFD convergence is variable



Technical Deep-Dive

Dual-Active

If a dual-active scenario is detected (using one or more of the previously mentioned protocols), the **originally active** switch will perform the following:

1. Generates messages, letting us know that a dual-active condition has been detected
2. Shuts down all non-VSL and non-excluded interfaces
 - Essentially removes the switch from the network so it doesn't cause problems
3. Once the VSL is restored (is operational):
 - If the **configuration has not changed**, it will **automatically reload**, restoring normal VSS behavior and operations
 - If a **configuration change has been detected**, the switch **requires manual intervention** to sync the configuration and reload the switch



Outline

- **Overview**
 - What is Virtual Switching System (VSS)?
 - Why Use VSS?
 - VSS Components
- **Technical Deep-Dive**
 - Hardware and Software Requirements
 - Virtual Switch Link (VSL)
 - Redundancy
 - Dual-active
 - **Installation**
 - Best practices
 - Limitations and Restrictions
 - Scenarios



Technical Deep-Dive

Installation

- **Verify Module Compatibility**
- Backup the configuration
- Migrate the existing configurations to a VSS-friendly configuration
 - Offline, well in advance of the VSS migration
- Inspect IOS version
 - Look for VSS-compatible IOS version
 - Use/upgrade to modular IOS?
- Erase configs (back to factory default)
- Configure VSS-specifics
 - Domain ID
 - Switch number, priorities and preemption
 - VSL
- Convert to VSS mode
- Accept VSS mode
- Confirm VSS conversion is successful

Technical Deep-Dive

Installation – Verify Module Compatibility

SWITCH 1

VSS-Sw1#show module

Mod	Ports	Card Type	Model	Serial No.
1	5	Supervisor Engine 720 10GE (Active)	VS-S720-10G	ZZZZZZZZZZ
2	4	CEF720 4 port 10-Gigabit Ethernet	WS-X6704-10GE	YYYYYYYYYY
3	48	CEF720 48 port 10/100/1000mb Ethernet	WS-X6748-GE-TX	XXXXXXXXXX
4	6	Firewall Module	WS-SVC-FWM-1	VVVVVVVVVV

Mod	MAC addresses	Hw	Fw	Sw	Status
1	0000.0000.a0c8 to 0000.0000.a0cf	2.0	8.5(2)	12.2(33)SXI	Ok
2	0000.0000.9454 to 0000.0000.9457	2.5	12.2(14r)S5	12.2(33)SXI	Ok
3	0000.0000.85be to 0000.0000.85ed	2.4	12.2(14r)S5	12.2(33)SXI	Ok
4	0000.0000.c0f6 to 0000.0000.c0fd	1.1	7.2(1)	4.0(3)	Ok

Mod	Sub-Module	Model	Serial	Hw	Status
1	Policy Feature Card 3	VS-F6K-PFC3C	BBBBBBBBBB1	1.0	Ok
1	MSFC3 Daughterboard	VS-F6K-MSFC3	BBBBBBBBBB2	1.0	Ok
2	Centralized Forwarding Card	WS-F6700-CFC	BBBBBBBBBB3	3.1	Ok
3	Centralized Forwarding Card	WS-F6700-CFC	BBBBBBBBBB4	3.1	Ok

Mod Online Diag Status

- 1 Pass
- 2 Pass
- 3 Pass
- 4 Pass

VSS-Sw1#

Technical Deep-Dive

Installation – Verify Module Compatibility

SWITCH 2

VSS-Sw2#show module

Mod	Ports	Card Type	Model	Serial No.
1	5	Supervisor Engine 720 10GE (Active)	VS-S720-10G	AAAAAAAAAAAA
2	4	CEF720 4 port 10-Gigabit Ethernet	WS-X6704-10GE	BBBBBBBBBBBB
3	48	CEF720 48 port 10/100/1000mb Ethernet	WS-X6748-GE-TX	CCCCCCCCCCCC
4	6	Firewall Module	WS-SVC-FWM-1	DDDDDDDDDD

Mod	MAC addresses	Hw	Fw	Sw	Status
1	1111.2222.9e78 to 1111.2222.9e7f	2.0	8.5(2)	12.2(33)SXI	Ok
2	1111.2222.bdcc to 1111.2222.bdcf	2.5	12.2(14r)S5	12.2(33)SXI	Ok
3	1111.2222.6030 to 1111.2222.605f	2.7	12.2(14r)S5	12.2(33)SXI	Ok
4	1111.2222.1c48 to 1111.2222.1c4f	1.1	7.2(1)	4.0(3)	Ok

Mod	Sub-Module	Model	Serial	Hw	Status
1	Policy Feature Card 3	VS-F6K-PFC3C	AAAAAAAAAAAA1	1.0	Ok
1	MSFC3 Daughterboard	VS-F6K-MSFC3	AAAAAAAAAAAA2	1.0	Ok
2	Centralized Forwarding Card	WS-F6700-CFC	AAAAAAAAAAAA3	3.1	Ok
3	Centralized Forwarding Card	WS-F6700-CFC	AAAAAAAAAAAA4	4.0	Ok

Mod Online Diag Status

- 1 Pass
- 2 Pass
- 3 Pass
- 4 Pass

VSS-Sw2#



Technical Deep-Dive

Installation

- ~~Verify Module Compatibility~~
- **Backup the configuration**
- Migrate the existing configurations to a VSS-friendly configuration
 - Offline, well in advance of the VSS migration
- Inspect IOS version
 - Look for VSS-compatible IOS version
 - Use/upgrade to modular IOS?
- Erase configs (back to factory default)
- Configure VSS-specifics
 - Domain ID
 - Switch number, priorities and preemption
 - VSL
- Convert to VSS mode
- Accept VSS mode
- Confirm VSS conversion is successful



Technical Deep-Dive

Installation – Backup Existing Configurations

SWITCH 1

```
VSS-Sw1#copy running-config startup-config  
Destination filename [startup-config]?
```

```
4507 bytes copied in 5.692 secs (792 bytes/sec)  
VSS-Sw1#copy startup-config disk0:old-startup-config  
Destination filename [old-startup-config]?
```

```
4507 bytes copied in 1.196 secs (3768 bytes/sec)  
VSS-Sw1#
```

SWITCH 2

```
VSS-Sw2#copy running-config startup-config  
Destination filename [startup-config]?
```

```
4507 bytes copied in 5.272 secs (855 bytes/sec)  
VSS-Sw2#copy startup-config disk0:old-startup-config  
Destination filename [old-startup-config]?
```

```
4507 bytes copied in 1.116 secs (4039 bytes/sec)  
VSS-Sw2#
```



Technical Deep-Dive

Installation

- ~~Verify Module Compatibility~~
- ~~Backup the configuration~~
- **Migrate the existing configurations to a VSS-friendly configuration**
 - Offline, well in advance of the VSS migration
- Inspect IOS version
 - Look for VSS-compatible IOS version
 - Use/upgrade to modular IOS?
- Erase configs (back to factory default)
- Configure VSS-specifics
 - Domain ID
 - Switch number, priorities and preemption
 - VSL
- Convert to VSS mode
- Accept VSS mode
- Confirm VSS conversion is successful



Technical Deep-Dive

Installation – Migrate Configs

- Best done manually
 - Very much like when migrating from hybrid to native mode
- Much of the config on switch 2 is lost during the conversion
- Pieces of the config that aren't lost during the conversion can result in instabilities if incorrect
 - If the same EtherChannel ID is in use on switch 1 as is used for the VSL on switch 2, we can have problems
- If the need arises to migrate from VSS to standalone mode, only the non-VSS config from switch 1 is used for both switch 1 and switch 2 after the migration process
 - You'll need to manually touch switch 2 – of course, you have pre-VSS config backups for switch 2 and can apply these, right?!



Technical Deep-Dive

Installation

- ~~Verify Module Compatibility~~
- ~~Backup the configuration~~
- ~~Migrate the existing configurations to a VSS-friendly configuration~~
 - Offline, well in advance of the VSS migration
- **Inspect IOS version**
 - Look for VSS-compatible IOS version
 - Use/upgrade to modular IOS?
- Erase configs (back to factory default)
- Configure VSS-specifics
 - Domain ID
 - Switch number, priorities and preemption
 - VSL
- Convert to VSS mode
- Accept VSS mode
- Confirm VSS conversion is successful



Technical Deep-Dive

Installation – Inspect IOS Version

- Ensure that the IOS version in use supports VSS!
- Consider using modular IOS
- Review the IOS release notes, Cisco Bug Toolkit, etc. to determine the best IOS release

SWITCH 1

```
VSS-Sw1#show version
Cisco IOS Software, s72033_rp Software (s72033_rp-ADVENTERPRISEK9_WAN-M), Version 12.2(33)SXI, RELEASE SOFTWARE (fc2)
...
ROM: System Bootstrap, Version 12.2(17r)SX5, RELEASE SOFTWARE (fc1)
...
VSS-Sw1#
```

SWITCH 2

```
VSS-Sw2#show version
Cisco IOS Software, s72033_rp Software (s72033_rp-ADVENTERPRISEK9_WAN-M), Version 12.2(33)SXI, RELEASE SOFTWARE (fc2)
...
ROM: System Bootstrap, Version 12.2(17r)SX5, RELEASE SOFTWARE (fc1)
...
VSS-Sw2#
```



Technical Deep-Dive

Installation

- ~~Verify Module Compatibility~~
- ~~Backup the configuration~~
- ~~Migrate the existing configurations to a VSS-friendly configuration~~
 - Offline, well in advance of the VSS migration
- ~~Inspect IOS version~~
 - Look for VSS-compatible IOS version
 - Use/upgrade to modular IOS?
- **Erase configs (back to factory default)**
- Configure VSS-specifics
 - Domain ID
 - Switch number, priorities and preemption
 - VSL
- Convert to VSS mode
- Accept VSS mode
- Confirm VSS conversion is successful



Technical Deep-Dive

Installation – Erase Configs

- Since we're not in a VSS setup now, we can safely use *write erase*

SWITCH 1

```
VSS-Sw1#write erase
Erasing the nvram filesystem will remove all configuration files! Continue? [confirm]
[OK]
Erase of nvram: complete
VSS-Sw1#
*Aug 17 14:24:49.335: %SYS-SP-7-NV_BLOCK_INIT: Initialized the geometry of nvram
VSS-Sw1#
VSS-Sw1#reload
Proceed with reload? [confirm]
...
```

SWITCH 2

```
VSS-Sw2#write erase
Erasing the nvram filesystem will remove all configuration files! Continue? [confirm]
[OK]
Erase of nvram: complete
VSS-Sw2#
*Aug 17 14:24:50.055: %SYS-SP-7-NV_BLOCK_INIT: Initialized the geometry of nvram
VSS-Sw2#
VSS-Sw2#reload
Proceed with reload? [confirm]
...
```



Technical Deep-Dive

Installation

- ~~Verify Module Compatibility~~
- ~~Backup the configuration~~
- ~~Migrate the existing configurations to a VSS-friendly configuration~~
 - Offline, well in advance of the VSS migration
- ~~Inspect IOS version~~
 - Look for VSS-compatible IOS version
 - Use/upgrade to modular IOS?
- ~~Erase configs (back to factory default)~~
- **Configure VSS-specifics**
 - Domain ID
 - Switch number, priorities and preemption
 - VSL
- Convert to VSS mode
- Accept VSS mode
- Confirm VSS conversion is successful



Technical Deep-Dive

Installation – Configure VSS-Specifics

- Verify the PFC mode is compatible with the hardware (configure it manually if needed)

SWITCH 1

```
VSS-Sw1#show platform hardware pfc mode  
PFC operating mode : PFC3C
```

```
VSS-Sw1#
```

SWITCH 2

```
VSS-Sw2#show platform hardware pfc mode  
PFC operating mode : PFC3C
```

```
VSS-Sw2#
```



Technical Deep-Dive

Installation – Configure VSS-Specifics

- Enable OOB MAC Synchronization

SWITCH 1

```
VSS-Sw1#show mac-address-table synchronize statistics
```

```
MAC Entry Out-of-band Synchronization Feature Statistics:
```

```
-----  
...  
Status of feature enabled on the switch      : off  
Default activity time                        : 160  
Configured current activity time           : 160  
...
```

```
VSS-Sw1#conf t
```

```
Enter configuration commands, one per line.  End with CNTL/Z.
```

```
VSS-Sw1(config)#mac-address-table synchronize
```

```
  % Current OOB activity time is [160] seconds
```

```
  % Recommended aging time for all vlans is atleast three times the activity interval  
and global aging time will be changed automatically if required
```

```
VSS-Sw1(config)#^Z
```

```
VSS-Sw1#show mac-address-table synchronize statistics
```

```
...  
Status of feature enabled on the switch      : on  
Default activity time                        : 160  
Configured current activity time           : 160  
...  
VSS-Sw1#
```



Technical Deep-Dive

Installation – Configure VSS-Specifics

- Enable OOB MAC Synchronization

SWITCH 2

```
VSS-Sw2#show mac-address-table synchronize statistics
```

```
MAC Entry Out-of-band Synchronization Feature Statistics:
```

```
-----  
...  
Status of feature enabled on the switch      : off  
Default activity time                        : 160  
Configured current activity time            : 160  
...
```

```
VSS-Sw2#conf t
```

```
Enter configuration commands, one per line.  End with CNTL/Z.
```

```
VSS-Sw2(config)#mac-address-table synchronize
```

```
  % Current OOB activity time is [160] seconds
```

```
  % Recommended aging time for all vlans is atleast three times the activity interval  
and global aging time will be changed automatically if required
```

```
VSS-Sw2(config)#^Z
```

```
VSS-Sw2#show mac-address-table synchronize statistics
```

```
...  
Status of feature enabled on the switch      : on  
Default activity time                        : 160  
Configured current activity time            : 160  
...  
VSS-Sw2#
```



Technical Deep-Dive

Installation – Configure VSS-Specifics

- VSS isn't configured yet...

SWITCH 1

```
VSS-Sw1#show switch virtual
Switch Mode : Standalone
  Not in Virtual Switch mode due to:
    Domain ID is not configured
VSS-Sw1#
```



Technical Deep-Dive

Installation – Configure VSS-Specifics

- VSS Domain ID

SWITCH 1

```
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 2

```
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
after the exec command 'switch convert mode virtual' is issued

vss-sw2(config-vs-domain)#
```



Technical Deep-Dive

Installation – Configure VSS-Specifics

- VSS Switch Number

SWITCH 1

```
vss-sw1(config-vs-domain)#switch 1
```

SWITCH 2

```
vss-sw2(config-vs-domain)#switch 2
```



Technical Deep-Dive

Installation – Configure VSS-Specifics

- VSS Switch Priorities

SWITCH 1

```
vss-sw1(config-vs-domain)#switch 1 priority 200  
vss-sw1(config-vs-domain)#switch 2 priority 150
```

SWITCH 2

```
vss-sw2(config-vs-domain)#switch 1 priority 200  
vss-sw2(config-vs-domain)#switch 2 priority 150
```



Technical Deep-Dive

Installation – Configure VSS-Specifics

- VSS Preemption

SWITCH 1

```
vss-sw1(config-vs-domain)#switch 1 preempt
```

Please note that Preempt configuration will make the ACTIVE switch with lower priority to reload forcefully when preempt timer expires.

The default preempt timer is 15 minutes. It is set to the approximate time required to bring up all linecards in a fully loaded chassis. Once the preempt timer starts, the standby switch with higher priority will take over as active after 15 minutes.

```
vss-sw1(config-vs-domain)#switch 2 preempt
```

Please note that Preempt configuration will make the ACTIVE switch with lower priority to reload forcefully when preempt timer expires.

The default preempt timer is 15 minutes. It is set to the approximate time required to bring up all linecards in a fully loaded chassis. Once the preempt timer starts, the standby switch with higher priority will take over as active after 15 minutes.

SWITCH 2

```
vss-sw2(config-vs-domain)#switch 1 preempt
```

Please note that Preempt configuration will make the ACTIVE switch with lower priority to reload forcefully when preempt timer expires.

The default preempt timer is 15 minutes. It is set to the approximate time required to bring up all linecards in a fully loaded chassis. Once the preempt timer starts, the standby switch with higher priority will take over as active after 15 minutes.

```
vss-sw2(config-vs-domain)#switch 2 preempt
```

Please note that Preempt configuration will make the ACTIVE switch with lower priority to reload forcefully when preempt timer expires.

The default preempt timer is 15 minutes. It is set to the approximate time required to bring up all linecards in a fully loaded chassis. Once the preempt timer starts, the standby switch with higher priority will take over as active after 15 minutes.



Technical Deep-Dive

Installation – Configure VSS-Specifics

- VSS Virtual MAC Address

SWITCH 1

```
vss-sw1(config-vs-domain)#mac-address use-virtual
```

SWITCH 2

```
vss-sw2(config-vs-domain)#mac-address use-virtual
```



Technical Deep-Dive

Installation – Configure VSS-Specifics

- VSL

SWITCH 1

```
vss-sw1(config)#int po1
vss-sw1(config-if)#switch virtual link 1
vss-sw1(config-if)#no shut
vss-sw1(config-if)#int ra ten1/4 -5
vss-sw1(config-if-range)#channel-group 1 mode on
vss-sw1(config-if-range)#no shut
```

SWITCH 2

```
vss-sw2(config)#int po2
vss-sw2(config-if)#switch virtual link 2
vss-sw2(config-if)#no shut
vss-sw2(config-if)#int ra ten1/4 -5
vss-sw2(config-if-range)#channel-group 2 mode on
vss-sw2(config-if-range)#no shut
```



Technical Deep-Dive

Installation

- ~~Verify Module Compatibility~~
- ~~Backup the configuration~~
- ~~Migrate the existing configurations to a VSS-friendly configuration~~
 - Offline, well in advance of the VSS migration
- ~~Inspect IOS version~~
 - Look for VSS-compatible IOS version
 - Use/upgrade to modular IOS?
- ~~Erase configs (back to factory default)~~
- ~~Configure VSS specifics~~
 - Domain ID
 - Switch number, priorities and preemption
 - VSL
- **Convert to VSS mode**
- Accept VSS mode
- Confirm VSS conversion is successful

Technical Deep-Dive

Installation – Convert to VSS Mode

SWITCH 1

```
vss-sw1#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.

```
Do you want to proceed? [yes/no]: yes
```

```
Converting interface names
```

```
Building configuration...
```

```
[OK]
```

```
Saving converted configuration to bootflash: ...
```

```
Destination filename [startup-config.converted_vs-20090817-145702]?
```

```
... <reboot process>
```

SWITCH 2

```
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.

```
Do you want to proceed? [yes/no]: yes
```

```
Converting interface names
```

```
Building configuration...
```

```
[OK]
```

```
Saving converted configuration to bootflash: ...
```

```
Destination filename [startup-config.converted_vs-20090817-145701]?
```

```
... <reboot process>
```

Your input
(enter) is
needed and
valued here!



Technical Deep-Dive

Installation

- ~~Verify Module Compatibility~~
- ~~Backup the configuration~~
- ~~Migrate the existing configurations to a VSS-friendly configuration~~
 - Offline, well in advance of the VSS migration
- ~~Inspect IOS version~~
 - Look for VSS-compatible IOS version
 - Use/upgrade to modular IOS?
- ~~Erase configs (back to factory default)~~
- ~~Configure VSS specifics~~
 - Domain ID
 - Switch number, priorities and preemption
 - VSL
- ~~Convert to VSS mode~~
- **Accept VSS mode**
- Confirm VSS conversion is successful



Technical Deep-Dive

Installation – Accept VSS Mode

SWITCH 1

```
vss-sw1#switch accept mode virtual
```

```
switch virtual domain 100
  switch 2 preempt
  switch 2 priority 150
interface Port-channel2
  switch virtual link 2
  no shutdown
interface TenGigabitEthernet2/1/4
  channel-group 2 mode on
interface TenGigabitEthernet2/1/5
  channel-group 2 mode on
```

This command will populate the above VSL configuration from the standby switch into the running configuration.

The startup configuration will also be updated with the new merged configuration if merging is successful.

```
Do you want to proceed? [yes/no]: yes
Merging the standby VSL configuration...
```

Please note that Preempt configuration will make the ACTIVE switch with lower priority to reload forcefully when preempt timer expires.

The default preempt timer is 15 minutes. It is set to the approximate time required to bring up all linecards in a fully loaded chassis. Once the preempt timer starts, the standby switch with higher priority will take over as active after 15 minutes.

```
*Aug 17 15:12:28.828: %VSLP-SW1_SP-5-VSL_UP: Ready for data traffic
```

```
Building configuration...
```

```
*Aug 17 15:12:40.375: %PFINIT-SW1_SP-5-CONFIG_SYNC: Sync'ing the startup configuration to the standby Router. [OK]
vss-sw1#
```



Technical Deep-Dive

Installation – Accept VSS Mode

SWITCH 2

```
vss-sw1-sdby>  
Standby console disabled
```

```
vss-sw1-sdby>  
Standby console disabled
```

```
vss-sw1-sdby>  
Standby console disabled
```



Technical Deep-Dive

Installation

- ~~Verify Module Compatibility~~
- ~~Backup the configuration~~
- ~~Migrate the existing configurations to a VSS-friendly configuration~~
 - Offline, well in advance of the VSS migration
- ~~Inspect IOS version~~
 - Look for VSS-compatible IOS version
 - Use/upgrade to modular IOS?
- ~~Erase configs (back to factory default)~~
- ~~Configure VSS specifics~~
 - Domain ID
 - Switch number, priorities and preemption
 - VSL
- ~~Convert to VSS mode~~
- ~~Accept VSS mode~~
- **Confirm VSS conversion is successful**



Technical Deep-Dive

Installation – Confirm Successful VSS Migration

SWITCH 1

```
vss-sw1#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
vss-sw1#
vss-sw1#
vss-sw1#show switch virtual role
```

Switch	Switch Number	Status	Preempt Oper(Conf)	Priority Oper(Conf)	Role	Session ID	
						Local	Remote
LOCAL	1	UP	TRUE (Y*)	200(200)	ACTIVE	0	0
REMOTE	2	UP	TRUE (Y*)	150(150)	STANDBY	330	6476

```
Standby configured preempt timer(switch 2): 15 minutes
Active configured preempt timer(switch 1): 15 minutes
```

```
In dual-active recovery mode: No
```

```
vss-sw1#
```



Technical Deep-Dive

Installation – Confirm Successful VSS Migration

SWITCH 1

```
vss-sw1#show redundancy
Redundant System Information :
-----
...
    Configured Redundancy Mode = sso
    Operating Redundancy Mode = sso
...
        Active Location = slot 1/1
        Current Software state = ACTIVE
...
Peer Processor Information :
-----
        Standby Location = slot 2/1
        Current Software state = STANDBY HOT
...
vss-sw1#
```



Technical Deep-Dive

Installation – Confirm Successful VSS Migration

SWITCH 1

```
vss-sw1#show vslp lmp neighbors
```

```
Instance #1:
```

```
LMP neighbors
```

```
Peer Group info:          # Groups: 1          (* => Preferred PG)
```

PG #	MAC	Switch	Ctrl Interface	Interfaces
*1	0000.0000.75c0	2	Te1/1/5	Te1/1/4, Te1/1/5

```
vss-sw1#
```

Technical Deep-Dive

Installation – Confirm Successful VSS Migration

SWITCH 1

```
vss-sw1#show vslp 1 lmp summary
```

```
LMP summary
```

```
Link info:          Configured: 2          Operational: 2
```

Interface	Flag	State	Peer Flag	Peer MAC	Peer Switch	Peer Interface	Timer(s)running (Time remaining)
Te1/1/4	vfsp	operational	vfsp	0000.0000.75c0	2	Te2/1/4	T4(360ms) T5(59.96s)
Te1/1/5	vfsp	operational	vfsp	0000.0000.75c0	2	Te2/1/5	T4(360ms) T5(59.96s)

```
Flags: v - Valid flag set      f - Bi-directional flag set  
       s - Negotiation flag set  p - Peer detected flag set
```

```
Timers: T4 - Hello Tx Timer   T5 - Hello Rx Timer
```

```
vss-sw1#
```



Technical Deep-Dive

Installation – Confirm Successful VSS Migration

SWITCH 1

```
vss-sw1#show vslp 1 rrp summary  
RRP Summary:
```

```
-----  
RRP information for Instance 1
```

```
-----  
Valid  Flags  Peer      Preferred  Reserved  
      Count      Peer      Peer
```

```
-----  
TRUE   V        1          1          1
```

```
-----  
Peer  Valid  Switch Status  Preempt  Priority  Role    Local  Remote  
Switch Group      Number      Oper(Conf) Oper(Conf)  Oper(Conf)  Oper(Conf)  SID    SID  
-----  
Local  0      TRUE   1      UP      Y (Y*)   200(200)  ACTIVE  0      0  
Remote 1      TRUE   2      UP      Y (Y*)   150(150)  STANDBY 330    6476
```

```
Peer 0 represents the local switch
```

```
Flags : V - Valid
```

```
vss-sw1#
```



Technical Deep-Dive

Installation – Confirm Successful VSS Migration

SWITCH 1

```
vss-sw1#ping vslp output interface ten1/1/4
```

```
Type escape sequence to abort.
```

```
Sending 5, 100-byte VSLP ping to peer-sup via output port 1/1/4, timeout is 2 seconds:
```

```
!!!!
```

```
Success rate is 100 percent (5/5), round-trip min/avg/max = 12/21/56 ms
```

```
vss-sw1#ping vslp output interface ten1/1/5
```

```
Type escape sequence to abort.
```

```
Sending 5, 100-byte VSLP ping to peer-sup via output port 1/1/5, timeout is 2 seconds:
```

```
!!!!
```

```
Success rate is 100 percent (5/5), round-trip min/avg/max = 12/19/48 ms
```

```
vss-sw1#ping vslp output interface ten2/1/4
```

```
% TenGigabitEthernet2/1/4 is not a local interface
```

```
vss-sw1#ping vslp output interface ten2/1/5
```

```
% TenGigabitEthernet2/1/5 is not a local interface
```

```
vss-sw1#
```

Technical Deep-Dive

Installation – Confirm Successful VSS Migration

Switch 1 (pre-VSS)

```
VSS-Sw1#dir ?
/all          List all files
/recursive    List files recursively
all-filesystems List files on all filesystems
bootflash:    Directory or file name
cns:          Directory or file name
const_nvram:  Directory or file name
dfc#2-bootflash: Directory or file name
dfc#3-bootflash: Directory or file name
disk0:        Directory or file name
flexwan-fpd:  Directory or file name
null:         Directory or file name
nvram:        Directory or file name
sup-bootdisk: Directory or file name
sup-bootflash: Directory or file name
sup-microcode: Directory or file name
system:       Directory or file name
tar:          Directory or file name
tmpsys:       Directory or file name
<cr>
```

VSS-Sw1#

Switch 1 (post-VSS)

```
vss-sw1#dir ?
/all          List all files
/recursive    List files recursively
all-filesystems List files on all filesystems
bootflash:    Directory or file name
cns:          Directory or file name
const_nvram:  Directory or file name
disk0:        Directory or file name
flexwan-fpd:  Directory or file name
null:         Directory or file name
nvram:        Directory or file name
revrcsf:      Directory or file name
slavebootflash: Directory or file name
slaveconst_nvram: Directory or file name
slavedisk0:   Directory or file name
slavenvram:   Directory or file name
slavercsf:    Directory or file name
slavesup-bootdisk: Directory or file name
slavesup-bootflash: Directory or file name
slavesystem:  Directory or file name
sup-bootdisk: Directory or file name
sup-bootflash: Directory or file name
sup-microcode: Directory or file name
sw1-slot1-bootflash: Directory or file name
sw1-slot1-const_nvram: Directory or file name
sw1-slot1-disk0: Directory or file name
sw1-slot1-nvram: Directory or file name
sw1-slot1-sup-bootdisk: Directory or file name
sw1-slot1-sup-bootflash: Directory or file name
sw1-slot2-dfc-bootflash: Directory or file name
sw1-slot3-dfc-bootflash: Directory or file name
sw2-slot1-bootflash: Directory or file name
sw2-slot1-const_nvram: Directory or file name
sw2-slot1-disk0: Directory or file name
sw2-slot1-nvram: Directory or file name
sw2-slot1-sup-bootdisk: Directory or file name
sw2-slot1-sup-bootflash: Directory or file name
sw2-slot2-dfc-bootflash: Directory or file name
sw2-slot3-dfc-bootflash: Directory or file name
system:       Directory or file name
tar:          Directory or file name
tmpsys:       Directory or file name
<cr>
```

vss-sw1#



Technical Deep-Dive

Installation – Success!

That's it!

Not really –

Don't forget to configure dual-active detection mechanisms



Outline

- **Overview**
 - What is Virtual Switching System (VSS)?
 - Why Use VSS?
 - VSS Components
- **Technical Deep-Dive**
 - Hardware and Software Requirements
 - Virtual Switch Link (VSL)
 - Redundancy
 - Dual-active
 - Installation
 - **Best practices**
 - Limitations and Restrictions
 - Scenarios



Technical Deep-Dive

Best Practices

The following best-practices have been compiled from the following sources:

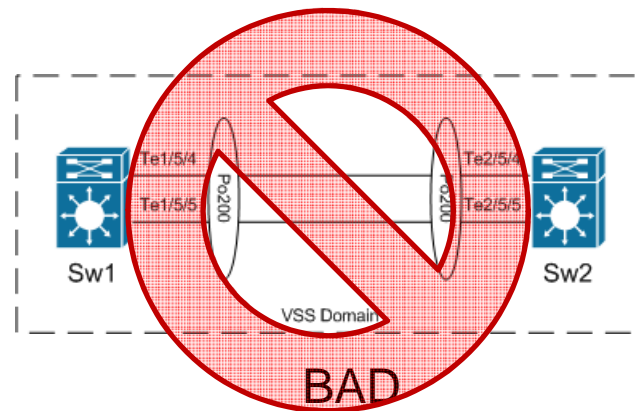
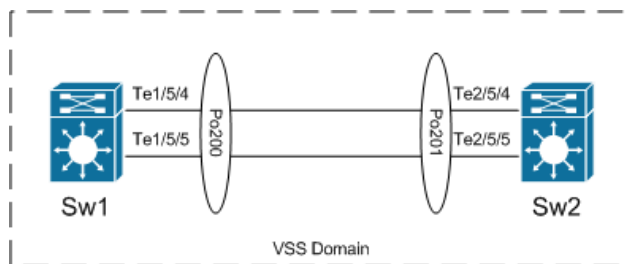
- Cisco Documents
 - Cisco Networkers/Live Presentations
 - Cisco Connection Online documents

- The school of hard knocks
 - Making it, breaking it and fixing it
 - Lab experience
 - Real-world experience

Technical Deep-Dive

Best Practices

- EtherChannels
 - Typically EtherChannel numbers are locally-significant to each switch
 - With VSS, all Multichassis EtherChannel (MEC) – that is, all MECs going to switches outside of the local VSS domain – share the same EtherChannel number. EtherChannel numbers are local to the VSS logical switch
 - Ensure that the VSL EtherChannel numbers are unique on each of the two switches within the VSS domain
 - If the numbers are the same on both ends of a VSL, when converting to VSS mode, the switches will not work properly
 - The switches merge the two configs together during the conversion process, and will have serious stability issues if the numbers are the same (it will allow it, but will not work properly)





Technical Deep-Dive

Best Practices

- Multichassis EtherChannels (MEC)
 - Dual-home all devices connecting to the VSS domain
 - Most environments are better off using MEC rather than Equal-Cost Multiple Paths (ECMP)
 - MEC simplifies the logical topology
 - Really beneficial in environments utilizing multicast (simplified RPF checks, etc.)
 - Tune the EtherChannel hashing algorithm
 - Not just on the VSS switches, but also on the other side!
 - Spend some time analyzing the hash results of the current algorithm to get an idea of whether or not it fits the traffic flows in your environment:

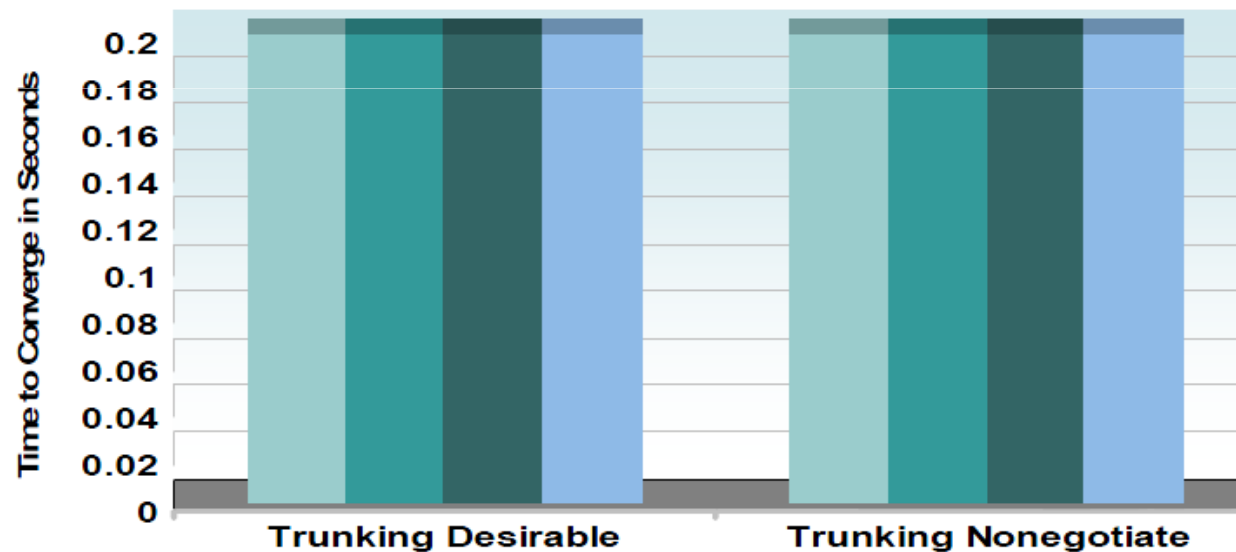
```
vss#sh etherchannel load-balance hash-result ?
interface Port-channel interface
ip IP address
ipv6 IPv6
l4port Layer 4 port number
mac Mac address
mixed Mixed mode: IP address and Layer 4 port number
mpls MPLS
```

```
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
```

Technical Deep-Dive

Best Practices

- Multichassis EtherChannels (MEC)
 - Cisco recommendations
 - Do not use *on* or *off* with PAgP, LACP or trunk negotiation
 - If using PAgP, use *desirable-desirable*
 - If using LACP, use *active-active*
 - Trunks should negotiate using *desirable-desirable*

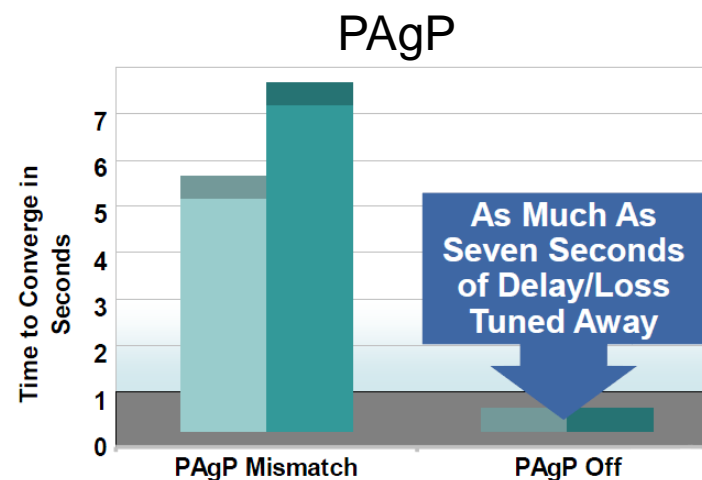
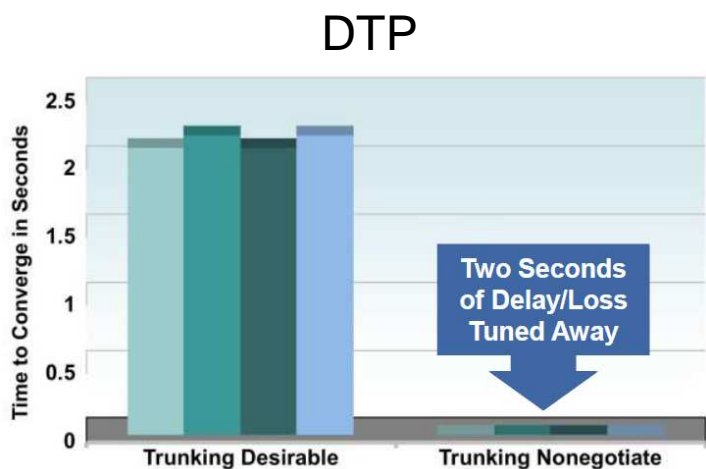


The results shown above is with BOTH channel and trunk mode being desirable on both ends

Technical Deep-Dive

Best Practices

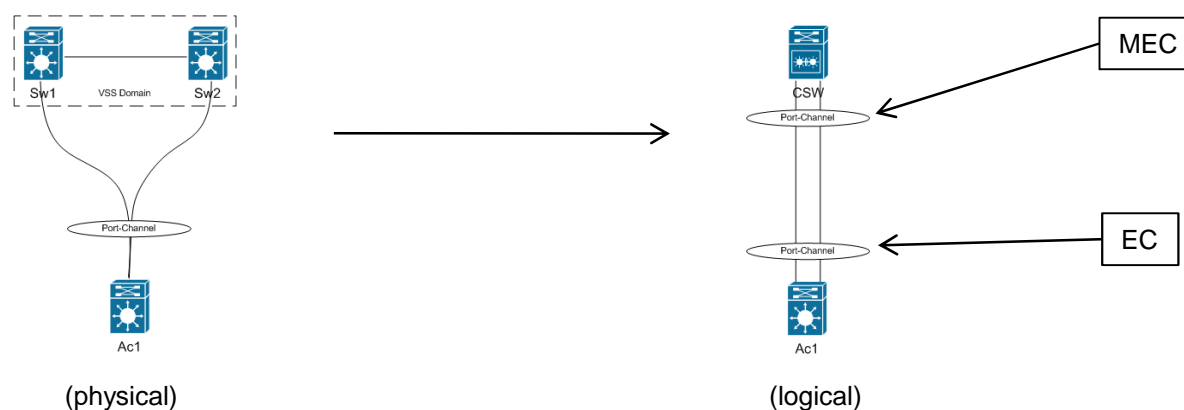
- Multichassis EtherChannels (MEC)
 - Tim's recommendations
 - Avoid PAgP, LACP & trunk negotiation (DTP) whenever possible
 - Use *on*
 - Disable DTP (*switchport nonegotiate*)
 - Cisco recommendations differ on whether or not to use PAgP, LACP, DTP, etc.
 - The fewer the gears in the machine makes my life easier
 - Less to configure
 - Less to troubleshoot
 - Faster convergence time
 - Determinism is key in stable networks
 - allowing the network to change the topology on its own is not desirable
 - So far, I've had great success with this recipe!



Technical Deep-Dive

Best Practices

- Multichassis EtherChannels (MEC)
 - Client switch requirements
 - There are none!*
 - As far as the non-VSS switch can tell, this is a regular EC
 - Remember, a pair of VSS switches looks like a single logical switch to the rest of the network
 - As long as device supports EtherChannels, it should work
 - Servers, switches, routers, etc. that support regular EC can all utilize MEC without a problem
 - VSS switches use the same port-channel number, but bind interfaces within both chassis' to create a MEC
 - This is automatic when the same port-channel is used!



* Except in the case of using ePAgP for dual-active detection



Technical Deep-Dive

Best Practices

- Configuration Mismatch
 - If either of the VSS switches are **not in normal operational mode** (one is **active** and the other is **hot-standby**), avoid making any configuration changes
 - Possible reasons why both switches may not be in normal operational mode:
 - Dual-active scenario
 - Incomplete VSS conversion (at an interim stage in the conversion process)
 - One of the chassis' has failed, been powered off or disconnected (VSL) from the other VSS chassis
 - Without both switches being fully operational, it results in a configuration mismatch
 - The switch which was down during the change comes back up, compares its last-known-good config and finds that it doesn't match the current config on the currently-active switch
 - This state requires manual intervention to manually sync the configurations and reload the switch
 - Even entering configuration mode (infamous *conf t*) will result in the switch *thinking* that a configuration change has been made – even if you immediately exit back to exec mode without making any changes
 - Moral of the story: avoid even entering config mode if both switches aren't up and fully operational



Technical Deep-Dive

Best Practices

- Erasing the configuration
 - We're all most likely familiar with the *write erase* command, which allows us to clear out an IOS device's configuration
 - Don't use *write erase* on a VSS system!
 - This command not only erases the startup-config, but it also deletes ROMMon variables which are needed for VSS initialization
 - VSS requires certain ROMMon variables in order to properly initialize
 - The *erase nvram* command deletes the contents of the NVRAM, but doesn't affect ROMMon
 - Sample scenario:
 - You want to erase a VSS switch configuration, keeping VSS intact, but all other configuration settings should be deleted.
 - This story has a happy ending if you use *erase nvram*!



Technical Deep-Dive

Best Practices

- Know your console port
 - With VSS, since the **active** switch handles all control-plane processes, this same switch is the only switch with an active console port.
 - The **hot-standby** switch console port is disabled
 - You can type whatever you'd like, but it'll tell you the same message:

```
VSS-SWITCH-sdby>enable  
Standby console disabled
```

```
VSS-SWITCH-sdby>conf t  
Standby console disabled
```
 - Remember that the **active** and **hot-standby** switches are determined by RRP (using the VSL)
 - Don't freak out if you see this – this means that you're not on the active switch console port – move your cable to the other switch console port in the same VSS domain!



Technical Deep-Dive

Best Practices

- Unique VSS Domain ID
 - Each pair of VSS switches must be within the same VSS domain
 - The domain ID is sometimes thought of as being locally significant to the two switches being configured
 - Not true – the VSS domain ID is used by different protocols
 - Used for identification of the pair of VSS switches
 - The VSS domain ID should be thought of as globally significant
 - Ensure that all pairs of VSS switches use unique VSS domain IDs



Technical Deep-Dive

Best Practices

- VSL
 - Size the VSL appropriately
 - Provide the VSL with at least the total uplink bandwidth of a single chassis'
 - Remember that other features can increase VSL traffic
 - SPAN
 - Service-modules
 - Single-homed devices
 - Ensure multiple links exist within the VSS EtherChannel
 - If only one interface will be used for the VSL, configure it in an EtherChannel
 - Allows for easy implementation of additional links into the VSL
 - Have the number of interfaces within the VSL in powers of 2
 - Results in better hash results / load-distribution across the EtherChannel
 - Link diversity
 - Have at least one interface within the VSL EtherChannel be on the sup
 - The sup comes up before any other line cards
 - Results in the VSS coming up faster
 - Have at least one interface within the VSL EtherChannel be on a line card
 - Guards the VSL against a hardware failure on the sup (X2 adapter, port ASIC, etc.)
 - Leave the VSL timers at the default settings



Technical Deep-Dive

Best Practices

- Dual-Active Detection
 - All methods are enabled by default
 - No interfaces are configured for them by default
 - Cisco recommendations
 - Use ePAgP when possible
 - If possible, use ePAgP & VSLP Fast-Hello
 - Tim's recommendations
 - Use VSLP Fast-Hello on multiple interfaces (on different line cards)
 - Disable all methods/protocols but VSLP Fast-Hello
 - Not all methods are equal
 - Spend time assessing your environment and the performance of the different methods based on your environment



Technical Deep-Dive

Best Practices

- Virtual MAC Address
 - By default, the VSS system uses the MAC address of the first-active switch for the VSS pair
 - It's best to use a virtual MAC address that never changes, no matter which switch comes up first in the VSS domain.
 - This can help avoid ARP table updates on end hosts during a switchover.
 - Virtual MAC is selected from the MAC address pool of 0008.e3ff.fc00 - 0008.e3ff.ffff



Technical Deep-Dive

Best Practices

- IOS

- Modular versus non-modular

- Modular is the future
- Go with modular

- Train selection

- Electric or steam? (SXI or SXH)
- I recommend IOS 12.2(33)SXI1 or later
 - The SXI train offers a lot of features, such as:
 - » eFSU
 - » If minor version mismatch (SXI vs SXI1 vs SXI2) occurs, the switches still operate in NSF/SSO, prior to SXI, the standby switch was in RPR mode (yuck!)
 - » SXI2 provides full MPLS feature parity with standalone (non-VSS) systems
 - » SXI2 offers unicast IPv6 support
 - » Greatly increased support for service-modules (ACE, FWSM, etc.)
 - » VSLP Fast-Hello
 - » The list goes on – check out the release notes for SXI, SXI1 and SXI2 – the SXI train really does offer a lot of features not found in SXH



Technical Deep-Dive

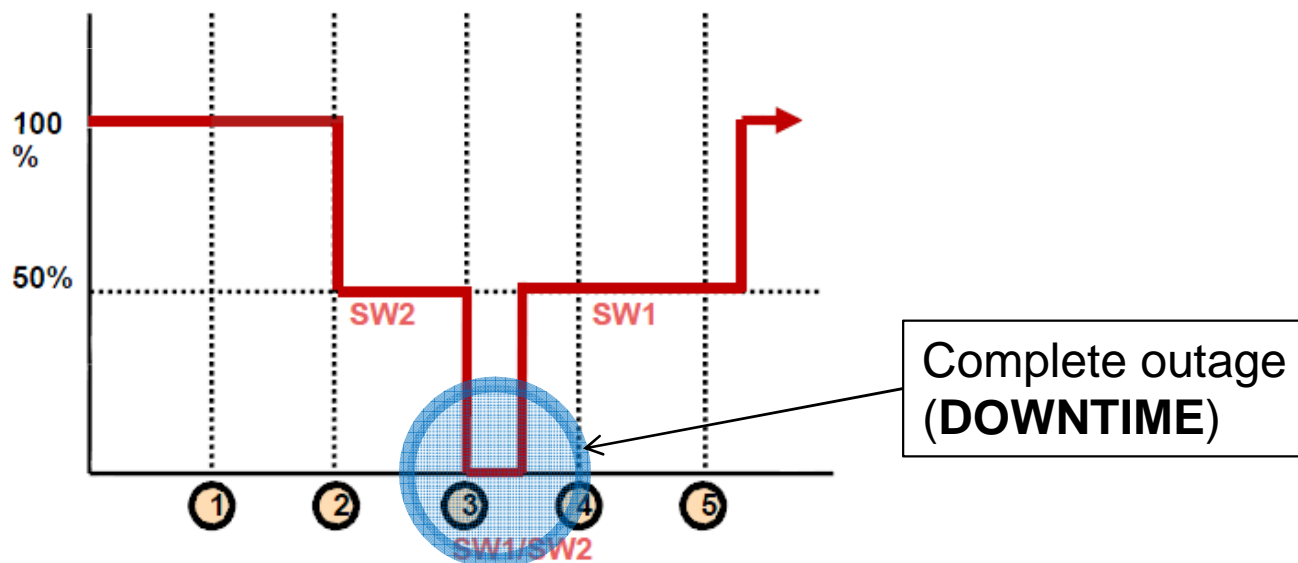
Best Practices

- IOS
 - Upgrades
 - Use In-Service Software Updates (ISSU) when possible, which allows for:
 - Fast Software Upgrade (FSU)
 - Enhanced Fast Software Upgrade (eFSU)
 - ISSU may be available in non-modular IOS (supporting upgrading the entire IOS image) and/or in modular IOS (allowing for patching of targeted IOS subsystems)
 - » Look at the IOS release notes, configuration guide, Feature Navigator, etc. for the particular IOS release you're looking at using
 - ISSU isn't a matter of changing the boot system statement(s), but has a structured upgrade process (with new CLI commands to support this)

Technical Deep-Dive

Best Practices

- IOS
 - Upgrades
 - Fast Software Upgrade (FSU)
 - Supported in IOS 12.2(SXH) and later
 - Requires an outage (**DOWNTIME**)
 - » Complete outage (several minutes)



Technical Deep-Dive

Best Practices

- IOS

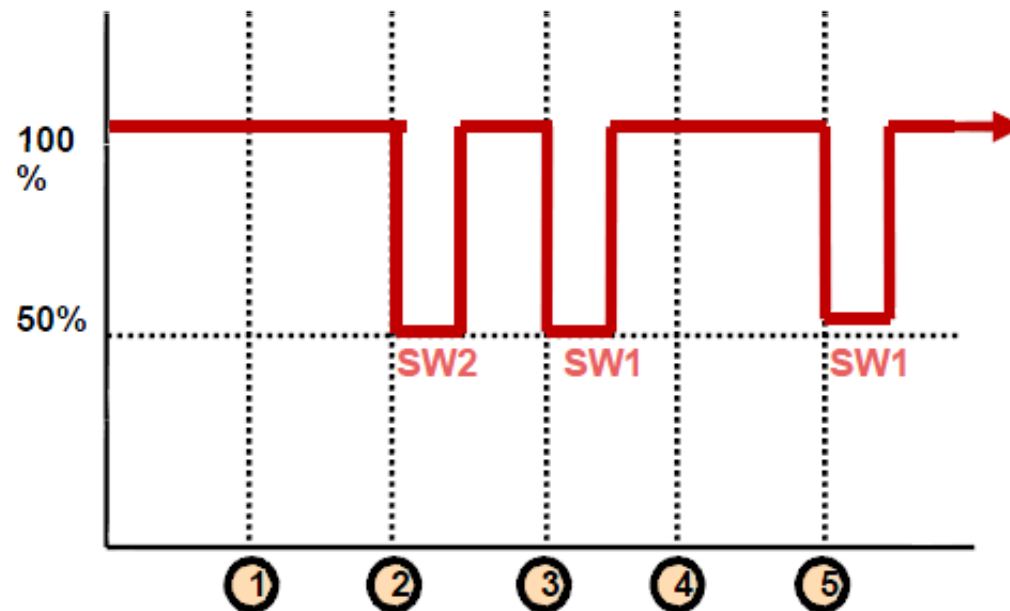
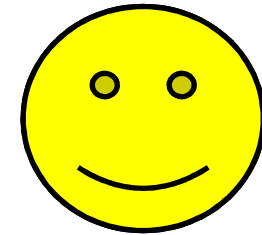
- Upgrades

- Enhanced Fast Software Upgrade (eFSU)

- Supported in IOS 12.2(SXI) and later

- **Near zero-downtime** upgrade

- » Lose 50% of throughput a couple of times, but no significant outage





Technical Deep-Dive

Best Practices

- Nonstop Forwarding
 - This isn't always configured by default!
 - EIGRP typically has NSF enabled by default
 - BGP, OSPF, IS-IS typically are not enabled by default
 - Refer to the IOS documentation to see if NSF is enabled by default
 - Look at how to enable it per protocol as it can differ between protocols
 - Remember to enable it for the routing protocols in use when it's not turned on by default

```
VSS#config t
VSS(config)#router ospf 10
VSS(config-router)#nsf

VSS#show ip ospf
Routing Process "ospf 10" with ID 192.168.2.1
Start time: 00:15:29.344, Time elapsed: 23:12:03.484
Supports only single TOS(TOS0) routes
External flood list length 0
Non-Stop Forwarding enabled
IETF NSF helper support enabled
Cisco NSF helper support enabled
Reference bandwidth unit is 100 mbps
```



Technical Deep-Dive

Best Practices

- Stateful Switchover
 - Typically this is enabled by default
 - Good idea to check that it's really configured and in place

```
VSS#show run
...
redundancy
main-cpu
auto-sync running-config
mode sso
...
```
 - If it doesn't exist, configure it!



Technical Deep-Dive

Best Practices

- Out-of-Band (OOB) MAC Address Synchronization
 - Syncs the MAC addresses between the Distributed Forwarding Cards (DFCs) in the line cards with the Policy Forwarding Card (PFC) in the supervisor engine
 - Without this enabled, the CAM across the DFCs could be out-of-sync, resulting in unicast flooding
 - To enable it

```
mac-address-table synchronize
```
 - Look at the settings

```
show mac-address-table synchronize statistics
```



Technical Deep-Dive

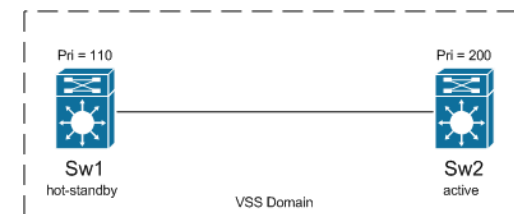
Best Practices

- VSS Conversion Process
 - Complete the entire VSS conversion process before doing anything else
 - Don't make any config changes (don't even enter config mode)
 - Remember that `switch accept mode virtual` is required to wrap things up
 - If things are altered prior to VSS being up and fully operational, it can result in config-mismatch

Technical Deep-Dive

Best Practices

- VSS Preemption
 - Preemption allows a switch with a higher priority to become active, demoting the currently active switch to hot-standby
 - Preemption is **not** enabled by default
 - Cisco recommendation
 - Do not use preemption
 - Only enable preemption on the switch with the highest priority
 - Tim's recommendation
 - Use preemption
 - Deterministic behavior – the same switch is always active in normal operation
 - Enable preemption on both switches
 - Downside is that it might result in some extra reloads
 - You choose which is best for your environment





Technical Deep-Dive

Best Practices

- VSS Switch Priority
 - Default priority is 100
 - Range can be from 1 (lowest) to 255 (highest)
 - Part of the decision-making process for determining the active switch (highest priority becomes active)
 - Cisco recommendations
 - Do not configure the switch priority
 - Tim's recommendations
 - Configure the switch priority
 - Configure both switches
 - Highest priority for active switch
 - » Example: Priority 200
 - Higher-than default priority for hot-standby switch
 - » Example: Priority 110



Technical Deep-Dive

Best Practices

- MAC Address Aging Timer
 - Cisco recommends that the CAM (MAC address table) aging timer be configured to 3-times (3x) the MAC sync timer

```
mac-address-table aging-time 480
```



Technical Deep-Dive

Best Practices

- Behavior – copy the config to both sups
 - Both chassis' automatically sync the config
 - It's a good idea when dealing with any files to ensure that it's copied to both the active and hot-standby sup
 - IOS images
 - Config files



Technical Deep-Dive

Best Practices

- Systemwide PFC Mode
 - Good idea to manually configure the PFC mode to eliminate any possible guess-work in terms of PFC compatibility
 - Remember that
 - PFC3CXL/DFC3CXL modules **can** fall back to PFC3C mode
 - PFC3C/DFC3C modules **will not** work in PFC3CXL mode
 - Check which mode you're in

```
vss#sh platform hardware pfc mode
PFC operating mode : PFC3C
Configured PFC operating mode : PFC3C
```
 - Best practice just to set it right off of the bat

```
vss#conf t
Enter configuration commands, one per line. End with CNTL/Z.
vss(config)#platform hardware vs1 pfc mode pfc3c
vss(config)#^Z
```



Technical Deep-Dive

Best Practices

- Spanning-Tree Protocol (STP)
 - A big **VSS selling point** is that we **no longer need STP**
 - True... on the pair of VSS chassis'
 - **We need STP for everywhere else in the network!**
 - **Do not disable STP** (even on the VSS switches)
 - It's a cheap insurance policy
 - If STP delays are a concern, follow STP best-practices and tune the protocol as needed
 - Consider enabling portfast on the trunks
 - As always, use this with caution – thoroughly assess the design prior to any changes
 - Use RPVST+/MSTP for best performance



Outline

- **Overview**
 - What is Virtual Switching System (VSS)?
 - Why Use VSS?
 - VSS Components
- **Technical Deep-Dive**
 - Hardware and Software Requirements
 - Virtual Switch Link (VSL)
 - Redundancy
 - Dual-active
 - Installation
 - Best practices
 - **Limitations and Restrictions**
 - Scenarios



Technical Deep-Dive

Limitations and Restrictions

- SPAN
 - A few limitations exist
 - As always, review the latest Cisco docs for the IOS release you'll be using
 - The VSL
 - Can be used as the source for a SPAN session, but the destination must be on the same physical switch chassis
 - All of you Wireshark gurus: the VSL header (VSH) is removed by the SPAN destination port, so forget about seeing it... ☺
 - Cannot be used as the source for a RSPAN, ERSPAN or Tx-only local SPAN session
 - Cannot be used as a SPAN destination

Virtual Switch System supports these SPAN capabilities per Virtual Switch domain.

Attribute	Value
Tx SPAN Sessions	14
Rx / Both SPAN Sessions	2
Total SPAN Sessions	16



Technical Deep-Dive

Limitations and Restrictions

- Quality of Service (QoS)
 - A few limitations exist
 - As always, review the latest Cisco docs for the IOS release you'll be using
 - Aggregate policer token buckets are not sync'd between the chassis' within a VSS domain
 - Need to take this into account when designing and implementing QoS
 - Do not try to alter the QoS settings for a VSL
 - Leave it alone – it has all of the settings needed when you enable the VSL
 - CoS is trusted by default
 - Priority queue is enabled
 - VSS control traffic and BPDUs are given priority on the VSL
 - Cannot configure a service-policy on the VSL
 - More limitations – just leave it alone!
 - SXI+ offers the ability to configure QoS policies on L2 interfaces
 - Allows us to configure QoS policies on L2, port-channel and L3 interfaces (SVI or physical routed interface)



Technical Deep-Dive

Limitations and Restrictions

- Weird SXI Behavior
 - Highly recommend testing/implementing SXI1 or SXI2a
 - The running-config isn't always the running-config
 - Read about it at <http://www.cciezone.com/?p=167>
 - Commands appeared in the running-config, however they weren't taking effect
 - Resolved the issue by re-entering the exact some commands
 - EIGRP tagging
 - SXI has a bug in which it uses a different tag format/version than previously recognized with SXH and other IOS platforms (12.4, etc.)
 - I'm not sure if this has to do with all IGP's or just EIGRP
 - Not a real pain point unless you're doing mutual redistribution (and using tags to prevent routing loops)
 - A real pain – skip to SXI1 to bypass the headache
 - Memory leaks
 - SXI2 has a known issue about memory leaks (can cause the switch to crash)
 - Look at SXI2a



Outline

- **Overview**
 - What is Virtual Switching System (VSS)?
 - Why Use VSS?
 - VSS Components
- **Technical Deep-Dive**
 - Hardware and Software Requirements
 - Virtual Switch Link (VSL)
 - Redundancy
 - Dual-active
 - Installation
 - Best practices
 - Limitations and Restrictions
 - **Scenarios**

Technical Deep-Dive

Scenarios – conversion with non-default configs

Converting to VSS with a Non-Default Switch Configuration

Switch 1	Switch 2
<pre>! spanning-tree mode rapid-pvst spanning-tree vlan 1-4094 priority 8192 ! interface TenGigabitEthernet1/5 switchport switchport trunk encapsulation dot1q switchport mode trunk switchport nonegotiate ! interface GigabitEthernet3/20 switchport switchport access vlan 20 switchport mode access ! interface Vlan33 ip address 10.252.33.2 255.255.255.0 standby 1 ip 10.252.33.1 standby 1 priority 250 standby 1 preempt ! mac-address-table synchronize mac-address-table aging-time 480 !</pre>	<pre>! spanning-tree mode rapid-pvst spanning-tree vlan 1-4094 priority 12288 ! interface TenGigabitEthernet1/5 switchport switchport trunk encapsulation dot1q switchport mode trunk switchport nonegotiate ! interface GigabitEthernet3/10 switchport switchport access vlan 10 switchport mode access ! interface Vlan33 ip address 10.252.33.3 255.255.255.0 standby 1 ip 10.252.33.1 standby 1 priority 200 standby 1 preempt ! mac-address-table synchronize mac-address-table aging-time 480 !</pre>



Technical Deep-Dive

Scenarios – conversion with non-default configs

After converting to VSS:

```
vss-sw1#sh run
...
spanning-tree mode rapid-pvst
spanning-tree vlan 1-4094 priority 8192
!
interface GigabitEthernet1/3/20
  switchport
  switchport access vlan 20
  switchport mode access
!
interface GigabitEthernet2/3/10
  no switchport
  no ip address
  shutdown
!
interface Vlan33
  ip address 10.252.33.2 255.255.255.0
  standby 1 ip 10.252.33.1
  standby 1 priority 250
  standby 1 preempt
!
...
vss-sw1#
```



Technical Deep-Dive

Scenarios – Config-mismatch

- We've converted to VSS mode on both switches
 - After the restart, before we continue with accepting the VSS mode, we look at the running-config on the active switch

```
vss-sw1#sh run
...
!
interface Port-channel1
 no switchport
 no ip address
 switch virtual link 1
 mls qos trust cos
 no mls qos channel-consistency
!
interface GigabitEthernet1/1/1
 no switchport
 no ip address
 shutdown
!
...
!
interface TenGigabitEthernet2/1/4
 no switchport
 no ip address
!
interface TenGigabitEthernet2/1/5
 no switchport
 no ip address
 shutdown
!
...
vss-sw1#
```

Technical Deep-Dive

Scenarios – Config-mismatch

```
vss-sw1#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
vss-sw1(config)#int po2
vss-sw1(config-if)#switch virtual link 2
vss-sw1(config-if)#no shut
vss-sw1(config)#int ten2/1/4
vss-sw1(config-if)#channel-group 2 mode on
*Aug 17 16:32:12.983: %VSLP-SW1_SP-5-VSL_UP: Ready for data traffic
vss-sw1(config-if)#int ten2/1/5
vss-sw1(config-if)#channel-group 2 mode on
vss-sw1(config-if)#end
vss-sw1#sh 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
Conversion completed       : No
Perform exec command 'switch accept mode virtual' to complete the VSS conversion
vss-sw1#sh switch virtual role

Switch  Switch Status  Preempt    Priority  Role      Session ID
      Number          Oper(Conf) Oper(Conf)          Local  Remote
-----
LOCAL   1      UP         TRUE (Y*)  250(250)  ACTIVE   0      0
REMOTE  2      UP         TRUE (Y*)  200(200)  STANDBY  2025   1923

Standby configured preempt timer(switch 2): 15 minutes
Active configured preempt timer(switch 1): 15 minutes

In dual-active recovery mode: No

vss-sw1#
```



Technical Deep-Dive

Scenarios – Config-mismatch

Switch 1

```
vss-sw1#redundancy reload shelf 2
Reload the entire remote shelf[confirm]
Preparing to reload remote shelf

vss-sw1#

...
*Aug 17 16:47:06.255: %VSLP-SW1_SP-5-VSL_UP: Ready for control traffic

vss-sw1#
*Aug 17 16:48:22.163: %VS_PARSE-SW1_SP-3-MISMATCH: STARTUP-CONFIG
*Aug 17 16:48:22.163: %VS_PARSE-SW1_SP-3-MISMATCH: Please use 'show switch virtual redundancy mismatch' for details
vss-sw1#
*Aug 17 16:48:22.179: %PFREDUN-SW1_SP-6-ACTIVE: Standby initializing for RPR mode
vss-sw1#
*Aug 17 16:48:23.867: %PFINIT-SW1_SP-5-CONFIG_SYNC: Sync'ing the startup configuration to the standby Router.
```

Switch 2

```
Cisco IOS Software, s72033_rp Software (s72033_rp-ADVENTERPRISEK9_WAN-M), Version 12.2(33)SXI, RELEASE SOFTWARE (fc2)
Technical Support: http://www.cisco.com/techsupport
Copyright (c) 1986-2008 by Cisco Systems, Inc.
Compiled Fri 07-Nov-08 04:01 by prod_rel_team
Image text-base: 0x40101328, data-base: 0x44559C50
```

Technical Deep-Dive

Scenarios – Config-mismatch

Switch 1

```
vss-sw1#sh 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
Conversion completed   : No
Perform exec command 'switch accept mode virtual' to complete the VSS conversion
vss-sw1#sh switch virtual role
```

Switch	Switch Number	Status	Preempt Oper(Conf)	Priority Oper(Conf)	Role	Session ID	
						Local	Remote
LOCAL	1	UP	TRUE (Y*)	250(250)	ACTIVE	0	0
REMOTE	2	UP	TRUE (Y*)	200(200)	STANDBY	2484	6115

```
Active configured preempt timer(switch 1): 15 minutes
```

```
In dual-active recovery mode: No
```

```
vss-sw1#
```



Technical Deep-Dive

Scenarios – Config-mismatch

Switch 1

```
vss-sw1#show redundancy
Redundant System Information :
-----
...
      Active Location = slot 1/1
      Current Software state = ACTIVE
...
      Standby Location = slot 2/1
      Current Software state = STANDBY COLD
...
vss-sw1#
```



Technical Deep-Dive

Scenarios – Config-mismatch

- Moral of the story:
 - Accept VSS mode!!!
 - Don't make config changes when the standby switch isn't fully operational (so that the config can be sync'd)

SWITCH 1

```
vss-sw1#switch accept mode virtual
```

```
switch virtual domain 100
  switch 2 preempt
  switch 2 priority 150
interface Port-channel2
  switch virtual link 2
  no shutdown
interface TenGigabitEthernet2/1/4
  channel-group 2 mode on
interface TenGigabitEthernet2/1/5
  channel-group 2 mode on
```

This command will populate the above VSL configuration from the standby switch into the running configuration.

The startup configuration will also be updated with the new merged configuration if merging is successful.

Do you want to proceed? [yes/no]: yes

Merging the standby VSL configuration...



Technical Deep-Dive

Scenarios – 6704 port for VSL

This won't work – nice try!

```
vss-sw2(config-if)#int ten 2/1
vss-sw2(config-if)#channel-group 2 mode on
Line card 2 is not a VSL capable module - port Te2/1 not added to VSL bundle.
Command rejected (Port-channel2, Te2/1): Interface configuration error as VSL port.

vss-sw2(config-if)#
```



Conclusion

- VSS is great for many environments
- Future-proof your network today, whether or not you plan on using VSS
- Once VSS is configured, life is pretty easy (fewer devices to manage, pretty self-contained, etc.)
- Be aware of some of the intricacies with dual-active detection and spend time designing the solution
- Don't take shortcuts – accept the VSS conversion, etc.
- Seek outside assistance if you're not comfortable
 - GTRI has several engineers who have VSS experience and can help

Thank you for coming!



Contact Information

Tim Clegg

Phone:

303-968-6529

Email:

tclegg@gtri.com

tim@cciezone.com