Deploying the Dell Force10 MXL into a Cisco Nexus Network Environment

A guide with basic examples for deploying the Dell Force10 MXL modular switch into the access layer of a Cisco Nexus network

Authors

Network Enabled Solutions Team
Jason Pearce, Manjesh Siddamurthy

Networking Enterprise Technologist
Kevin Horton

This document is for informational purposes only and may contain typographical errors and technical inaccuracies. The content is provided as is, without express or implied warranties of any kind.

December 2012| Rev 1.1

Contents

Introduction 5
  Content Overview 6
  Document Conventions 7
  Device Introduction 8
Dell Force10 MXL Deployment 12
  Initial Configuration 12
  Access Port Downlink Configuration 16
  Trunk Port Downlink Configuration 18
  Link Aggregation Group Configuration 20
  Trunk Port Uplink Configuration with MLAG at Top of Rack 21
  Trunk Port Uplink Configuration with Per-VLAN Spanning Tree 27
Spanning Tree Protocol Failover and Failback Observations 31
  Lab Environment 31
  Nexus MSTP and MXL MSTP Observations 32
  Nexus MSTP and MXL MSTP – Remainder of Test Observations 37
  Nexus MSTP and MXL MSTP – Greater than 64 Instances 38
  Nexus Rapid PVST+ and MXL PVST Observations 38
  Nexus Rapid PVST+ and MXL PVST – Remainder of Test Observations 42
  Summary – Spanning Tree Observations 42
Appendices 43
  Appendix A: Referenced Network Topology and Device Configurations 43
  Appendix B: Basic Terminology 54

Figures

Figure 1. Dell Force10 MXL Switch 5
Figure 2. Example Network Topology 7
Figure 3. MXL External Interfaces 8
Figure 4. FlexIO 40GbE QSFP+ Module 9
Figure 5. FlexIO 10GbE SFP+ Module 9
Figure 6. FlexIO 10GBASE-T Module 9
Figure 1. Topology Diagram: Top of Rack Nexus 5K using vPC 21
Figure 2. Topology Diagram: Top of Rack Nexus 5K using PVST+ 27
Figure 3. Topology Diagram: STP Interoperability Observations Lab 31
Figure 4. Topology Diagram: Detailed, Full Example Network 43

Tables

Table 1. Force10 MXL External Ethernet Interface Numbering 10
Table 2. Results – Spanning Tree Observations 42
Table 3. Devices and Firmware Versions Used in this Document 44

Introduction

This document is an easy-to-use guide of recommended basic deployment practices for the Dell Force10 MXL in the access layer of a Cisco Nexus network environment. The Dell Force 10 MXL—an I/O module for the Dell PowerEdge M1000e chassis—is a 10 and 40 gigabit Ethernet multilayer switch targeted at deployment in data center networks. The MXL provides 32 10GbE internal ports to server blades in an M1000e modular chassis and has multiple 10 and 40GbE external uplink and stacking options allowing it to serve well in many applications and environments.

Figure 1. Dell Force10 MXL Switch

The Dell Force10 MXL 10/40GbE switch brings a new level of connectivity and value to the blade server network switch environment. With 32 1/10GbE server facing ports, any transition from previous architectures will be a huge step forward. The 32 server facing ports provide connectivity on any of the available fabric slots (A, B, or C) for the M-Series blade servers with 1GbE or 10GbE KR-based network card capability. The switch offers 1/10/40GbE connectivity on the uplinks to interface with a top of rack switch, directly to a core switch, or directly to an iSCSI storage solution.

The MXL switch is an industry first, 40GbE capable, modular, and stackable blade switch for the PowerEdge M1000e chassis.

Content Overview

This document is broken up topically to allow the steps and concepts of MXL switch deployment to be addressed separately in a simple manner as detailed in Document Conventions. An introduction to the Force10 MXL hardware and its connectivity and management options is presented in Device Introduction. Once familiarized with the MXL switch, the out-of-box experience is continued with the first steps of MXL deployment as presented in Initial Configuration.

Link configuration is split into coverage of downlink configuration options (on the MXL, downlinks are generally its internal ports which connect to the server blades in a deployed M1000e chassis), configuration of a link aggregation group (LAG)—a set of ports configured to pass traffic together and behave as one link, and coverage of uplink configuration options (on the MXL these are generally its external ports, which connect to administratively designated ports on top-of-rack or end-of-row switches.

Downlink configuration for the MXL is covered for the two most common use cases: for VLAN unaware servers see Access Port Downlink Configuration and for VLAN aware servers see Trunk Port Downlink Configuration.

LAG configuration is then covered in Link Aggregation Group Configuration. While LAGs are sometimes configured on downlinks (dependent on application demands) they are recommended to use with uplinks.

Uplink configuration for the MXL will generally be done in one of two ways in a Cisco Nexus network environment: running PVST with discreet LAGs connected to separate top-of-rack switches as covered in Trunk Port Uplink Configuration with Per-VLAN Spanning Tree and running PVST with a single LAG connected across two top-of-rack switches that are using a Multi-chassis LAG as covered in Trunk Port Uplink Configuration with MLAG at Top of Rack (this uses the Cisco Nexus vPC feature).

To present the above configuration options as clearly as possible in this document, they have been integrated into a single example network. The full details of the example network, including overall topology and the complete configurations of the participating devices, are presented in Appendix A.

A glossary of applicable networking terms and concepts can be found in Appendix B. The definitions listed there are used throughout this document; so as unfamiliar terms are encountered in this document, refer to Appendix B for their definitions.

Additional materials referenced in this document and useful for configuration that is more advanced or specific than what this document covers are detailed in Appendix C.

Document Conventions

This document is intended as a quick reference guide for use during basic deployment of a Force10 MXL switch. As such, many details, configuration options, and specific features of the MXL are left out of this document.

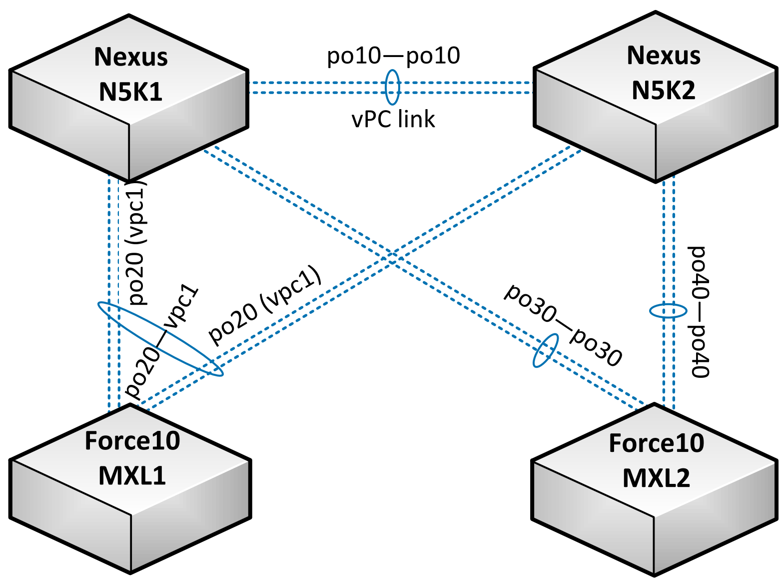
Example CLI Notation:

```
FTOS#configure
FTOS(conf)#hostname MXL1
MXL1(conf)#exit
MXL1#
```

As shown above, CLI examples are listed inside indented text blocks, use a monospace font, the user input is emboldened, and the portions of the commands that will likely need to be adjusted for local use are emboldened and italicized. Also note that each CLI example starts in privileged execution mode and where needed starts and ends with commands to move between modes. Following this practice each time is not required when entering multiple examples together (commands issued in the same mode can generally be strung together). But, it is presented in this manner to ease following individual examples in the document.

Example Network Topology:

Figure 2. Example Network Topology



As shown in the figure above, the network topology diagrams in this document are kept as simple as possible to communicate the need-to-know information for the topic on hand. Figure 2 above shows the example network that is explored in this document. While the example network topology does not mirror a production environment, individual parts of it do and the example topology does illustrate those parts more clearly. A more detailed topology diagram of this example network can be found in Appendix A.

Device Introduction

The Dell Force10 MXL is an I/O Module for the M1000e modular server chassis that is a line-speed, multilayer, 10 and 40Gb Ethernet switch that provides 32 internal 10GbE links—enabling full connectivity to M420 quarter-height server blades—and flexible 10 and 40Gb Ethernet options for external connectivity. With as much as 240 Gigabits of full-duplex external connectivity, it offers a favorable 1.33:1 over-subscription ratio of internal to external connectivity. With spanning-tree options of RSTP, MSTP, and PVST it can be natively integrated into most layer-2 network environments and optionally participate at layer-3 environments using OSPF or static routing. The MXL also offers Data-Center Bridging features enabling it to act as an FCoE transit switch (FIP snooping bridge). This document however only covers basic layer-2 deployment.

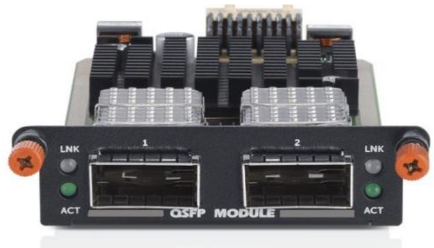
External Interfaces—Connectivity, Cabling, and Port Numbering

Figure 3. MXL External Interfaces

IO Bays 2 and 1 can each use one of the FlexIO modules detailed below. The fixed QSFP+ 40Gb Ethernet ports can be connected using Dell Force10 QSFP+ transceivers or Dell Force10 twinax cabling. Additionally there are special breakout fiber and twinax cables that allow each 40GbE port to instead be split into four 10GbE links that will connect with compatible 10GbE fiber transceivers or SFP+ ports respectively.

Between IO Bay 1 and the fixed QSFP+ ports is a USB type-A port that allows directly connecting a USB storage device to the MXL for copying files directly to and from the switch. Between the fixed QSFP+ ports and the MXL's latch hinge are the MXL's status indicator LEDs and a serial console interface that uses a physical USB type-A port—a required serial console cable is provided with each Force10 MXL switch. If the serial console cable is unavailable, the console may also be accessed via the M1000e CMC CLI "connect" command. Access to the serial console (either via the provided cable or the CMC) is required for manual out-of-box deployment of the MXL switch. See Management Connectivity for further details.

Figure 4. FlexIO 40GbE QSFP+ Module



The FlexIO 40GbE QSFP+ module provides two 40Gb Ethernet ports that can be connected using Dell Force10 QSFP+ transceivers or Dell Force10 twinax cabling. Additionally Dell offers breakout fiber and twinax cables that enable each 40GbE port to be split into four 10GbE links that will connect with compatible 10GbE fiber transceivers or SFP+ ports . This industry leading QSFP+ module provides the highest throughput available for each MXL FlexIO bay.

Figure 5. FlexIO 10GbE SFP+ Module

The FlexIO 10GbE SFP+ module provides four 10Gb Ethernet ports that can be connected using Dell Force10 SFP+ transceivers or Dell Force10 twinax cabling.

Figure 6. FlexIO 10GBASE-T Module

The FlexIO 10GBASE-T module provides four 10Gb Ethernet RJ45 ports that can be connected to other 10GBASE-T devices using standard Cat6A cabling.

Due to the availability of FlexIO modules with varying port and link count, the external interface numbering on the MXL can be difficult to understand. Table 1 details the external interface numbering in various modular configurations of the MXL switch. This numbering follows the potential10GbE links skipping numbers where available link count is less than the potential. Both the Ethernet interfaces and IO Bays are numbered from bottom to top when the MXL switch is installed in an M1000e modular chassis.

Table 1. Force10 MXL External Ethernet Interface Numbering

Module Type Link Speed	QSFP+ 40GbE	QSFP+ (breakout cables) 10GbE	SFP+ / 10GBASE-T 10GbE
IO Bay 2	53	56	n/a
		55	
		54	
		53	
	49	52	52
		51	51
		50	50
		49	49
IO Bay 1	45	48	n/a
		47	
		46	
		45	
	41	44	44
		43	43
		42	42
		41	41
Fixed QSFP+ Ports	37	40	n/a
		39	
		38	
		37	
	33	36	
		35	
		34	
		33	

Management Connectivity

The MXL switch provides a number of methods for connectivity to its management command-line interface (CLI).

Out-of-band console connectivity is available via serial on a physical USB type-A port on the face of the switch when used with the provided cable that is shipped with each MXL (see the previous section—External Interfaces—for help identifying the port). Serial terminal settings to access the console are set to 9600bps, no flow control, 8 character bits and 1 stop bit.

Out-of-band console access is also available via the CMC (Chassis Management Controller) “connect” command. A simple example is shown here using ssh to attach to the CMC console. For more information about accessing and using the CMC console, see the CMC User Guide.

```
~ % ssh root@172.25.188.40
root@172.25.188.40’s password:

Welcome to the CMC firmware version 4

$ connect switch-b1
connect: acquiring remote port.
Connected to remote port.
Escape character is '^\'.
MXL1>
```

Out-of-band network connectivity is provided through the M1000e CMC’s Ethernet connection and is configurable on the MXL’s CLI as the “managementethernet” interface. With the management Ethernet interface configured, the telnet or SSH services can be enabled to provide remote access to the command-line interface of the MXL. Example commands for configuring the management Ethernet interface and the telnet and SSH services of the MXL can be found in the Initial Configuration section of this document.

Dell Force10 MXL Deployment

Initial Configuration

This section provides a rapid introduction to some common MXL initial deployment tasks. For more detailed information on deploying the MXL see the Force10 MXL User Guide. The examples that are shown here start in privileged execution mode. On the serial console, privileged execution mode can be reached by using the “enable” command.

Configure the MXL Host Name

The configured hostname appears in the management prompt of the CLI and helps to quickly identify which device is being managed.

```
FTOS#configure
FTOS(conf)#hostname MXL1
MXL1(conf)#exit
MXL1#
```

Throughout this document two MXL switches are configured that have the host names MXL1 and MXL2. Their full configurations can be found in Appendix A.

Configure an IP Address and Default Gateway on the MXL Out-of-Band Management Interface

The out-of-band management Ethernet interface provides connectivity through the M1000e Chassis Management Controller. Only management traffic is sent or received on this interface.

```
MXL1#configure
MXL1(conf)#interface managementethernet 0/0
MXL1(conf-if-ma-0/0)#ip address 192.0.2.10/24

Proceed with Static IP [confirm yes/no]: yes
MXL1(conf-if-ma-0/0)#no shutdown
MXL1(conf-if-ma-0/0)#exit
MXL1(conf)#management route 0.0.0.0/0 192.0.2.1
MXL1(conf)#exit
MXL1#
```

The IP address of the management Ethernet interface can also be configured by the M1000e Chassis Management Controller. For further details see the CMC User Guide.

Add an Admin’s Username and Enable Password to the MXL configuration

In order to control access to the administration of the MXL, set username(s) and enable passwords.

```
MXL1#configure
MXL1(conf)#username NewUserName password NewUserPassword privilege 15
MXL1(conf)#no username root
```

```
MXL1(conf)#enable password NewEnablePassword
MXL1(conf)#exit
MXL1#
```

Configuring a username allows an individual to authenticate and is required for remote management. The root user (with password calvin) is automatically configured on the MXL during its first boot to enable rapid remote management during deployment. It is highly recommended to remove the default root user once local user accounts are configured.

Setting the enable password keeps unprivileged users and anyone with serial console access from changing the configuration of the switch. A user with privilege set to 15 runs in privileged execution mode and is not required to enter the enable password to manage the switch.

For a more secure remote management service it is recommended to use SSH.

Enable the SSH service on the MXL

SSH provides secure remote management connectivity to the MXL's CLI.

```
MXL1#configure
MXL1(conf)#ip ssh server enable
MXL1(conf)#exit
MXL1#
```

SSH requires keys in order to work properly.

You can determine if either an RSA or RSA1 key exists already by entering:

```
MXL1#
MXL1#show crypto key mypubkey rsa
ssh-rsa AAAAB3NzaC1yc2EAAAABIwAAAIEA7grssrAVe5qZM2hDGlxDBAolYCVFIWpeffW
BK1Ac1lvVsIKm+BjICf/bS16qwRuimdznFxNdUmru6hcbLNSe2m4c7mtdVI5D9gC6DYnHKH
OG9sqTkF46o2TQ5QsYV4cBZWvY69XF14XFzbAJBAJCcnZnD953pKm5VnMdzEf7YzE=
MXL1#
```

Or to check for existence of RSA1 Key:

```
MXL1#
MXL1#show crypto key mypubkey rsa1
ssh-rsa AAAAB3NzaC1yc2EAAAABIwAAAIEA7grssrAVe5qZM2hDGlxDBAolYCVFIWpeffW
BK1Ac1lvVsIKm+BjICf/bS16qwRuimdznFxNdUmru6hcbLNSe2m4c7mtdVI5D9gC6DYnHKH
OG9sqTkF46o2TQ5QsYV4cBZWvY69XF14XFzbAJBAJCcnZnD953pKm5VnMdzEf7YzE=
MXL1#
```

If no keys exist – or you wish to create new keys – then the following commands would be used with either RSA or RSA1 as the keyword in the command.

```
MXL1#configure
MXL1(conf)#crypto key gen rsa
Enter key size <1024-2048>. Default<1024>:
Generating SSHv2 RSA key.
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!
```

```
MXL1(conf)#exit
MXL1#
```

For additional information regarding SSH please refer to the MXL User Guide.

After configuring SSH and having confirmed the configuration is correct by logging in using SSH, it is recommended to disable Telnet for security reasons.

Disable the Telnet Service on the MXL

Telnet provides simple remote connectivity to the MXL's CLI and is enabled by default. It is however not as secure as SSH and is not allowed in many environments due to security concerns. The following example shows how to disable the telnet service on the MXL.

```
MXL1#configure
MXL1(conf)#no ip telnet server enable
MXL1(conf)#exit
MXL1#
```

Determine if Spanning Tree is already enabled

The goal is to run Per-VLAN Spanning Tree (PVST) but before PVST is configured, other spanning tree implementations must not already be enabled. In this example, RSTP is already running so it must be disabled before PVST can be enabled.

```
MXL1#show run | grep spanning
protocol spanning-tree rstp
MXL1#configure
MXL1#protocol spanning-tree rstp
MXL1(conf-rstp)#disable
MXL1(conf-rstp)#exit
MXL1(conf)#no protocol spanning-tree rstp
MXL1(conf)#exit
MXL1#
```

Enable Per-VLAN Spanning Tree on the MXL

Per-VLAN Spanning Tree (PVST) is an implementation of the spanning-tree protocol where a separate spanning tree is run inside of each VLAN. This allows network administrators to configure each VLAN with optimal root placement and active paths across a network topology. Most Cisco Nexus network environments run PVST and this document shows how to configure Dell Force10 MXL switches to operate natively in these environments by also running PVST.

```
MXL1#configure
MXL1(conf)#protocol spanning-tree pvst
MXL1(conf-pvst)#no disable
MXL1(conf-pvst)#exit
MXL1(conf)#exit
MXL1#
```

Split an MXL 40Gb Ethernet Interface into Four 10Gb Ethernet Links

The external QSFP+ 40Gb Ethernet ports can be configured as four separate 10Gb Ethernet links. Physical connectivity is enabled by an optical split fiber cable or a split twinax cable. When a 40GbE port is run in quad mode, it provides four 10Gb Ethernet interfaces that number sequentially starting with the port number of the 40GbE interface. In the following example this makes four 10GbE links numbered 33, 34, 35, and 36 from the single 40GbE port numbered 33.

```
MXL1#configure
MXL1(conf)#stack-unit 0 port 33 portmode quad
Please save and reload for the changes to take effect.
MXL1(conf)#exit
MXL1#copy running-config startup-config
File with same name already exist.
Proceed to copy the file [confirm yes/no]: yes
!
5678 bytes successfully copied

MXL1#reload

Proceed with reload [confirm yes/no]: yes
```

Note, as shown in the above example, splitting a 40GbE port into four 10GbE links requires saving the configuration and reloading the MXL to take effect.

Save the Current MXL Configuration Settings

Always remember to save your settings so that they are not lost in case the switch is restarted.

```
MXL1#copy running-config startup-config
File with same name already exist.
Proceed to copy the file [confirm yes/no]: yes
!
5678 bytes successfully copied
MXL1#
```

Access Port Downlink Configuration

Server's network interfaces' default configuration works with a single subnet available on the link and no VLAN tagging. To provide connectivity to this type of server interface, the matching switch interface that it connects to should be configured as an access port in the server's designated VLAN. Similar to the described server interface, an access port participates in only one VLAN and does not insert tags into the Ethernet frames on its link. To deploy an access port on an MXL running PVST follow these steps.

Configure a Downlink as an Access Port

```
MXL1#configure
MXL1(conf)#interface tengigabitethernet 0/1
MXL1(conf-if-te-0/1)#switchport
MXL1(conf-if-te-0/1)#spanning-tree pvst edge-port
MXL1(conf-if-te-0/1)#exit
MXL1(conf)#exit
MXL1#
```

The "switchport" setting enables a switch's Ethernet interface to participate in VLANs (prior to this command being run Force10 switch interfaces default to not forwarding traffic at all).

It is recommended that the "spanning-tree pvst edge-port" command should only be run on ports that will connect to servers or other end nodes and not on ports that will connect to other switches. This command designates a port as an expected edge of the spanning tree (only switches participate in spanning tree) and enables it to begin forwarding traffic as soon as it's connected (many seconds before the spanning-tree protocol would otherwise clear it for active use).

With the switchport feature enabled, the Ethernet interface is now ready for a VLAN to be configured for it.

Configure a VLAN for an Access Port Interface

```
MXL1#configure
MXL1(conf)#interface vlan 11
MXL1(conf-if-vl-11)#untagged tengigabitethernet 0/1
MXL1(conf-if-vl-11)#no shutdown
MXL1(conf-if-vl-11)#exit
MXL1(conf)#exit
MXL1#
```

As discussed above, an access port only participates in one VLAN at a time so if another VLAN were configured for this same port, it would remove the port from the first VLAN. For server interfaces that need access to multiple VLANs, see Trunk Port Downlink Configuration.

Now that a VLAN is configured for the access port, it is ready to be enabled. (To keep server network traffic where it is expected to be on the network, it is important to enable downlinks only after they are correctly configured.)

Enable the Configured Ethernet Interface

```
MXL1#configure
MXL1(conf)#interface tengigabitethernet 0/1
MXL1(conf-if-te-1)#no shutdown
```

```
MXL1(conf-if-te-1)#exit
MXL1(conf)#exit
MXL1#
```

For administrative convenience the above steps can be applied to a range of switch interfaces during configuration. In the following example, the downlinks of MXL1 in the example network are all configured as access ports with interfaces 1-8 on VLAN 11, interfaces 9-24 on VLAN 12, and interfaces 25-32 on VLAN 13.

Access Port CLI Example Using Interface Range

```
MXL1#configure
MXL1(conf)#interface range tengigabitethernet 0/1 - 32
MXL1(conf-if-range-te-0/1-32)#switchport
MXL1(conf-if-range-te-0/1-32)#spanning-tree pvst edge-port
MXL1(conf-if-range-te-0/1-32)#exit
MXL1(conf)#interface Vlan 11
MXL1(conf-if-vl-11)#untagged tengigabitethernet 0/1-8
MXL1(conf-if-vl-11)#no shutdown
MXL1(conf-if-vl-11)#exit
MXL1(conf)#interface Vlan 12
MXL1(conf-if-vl-12)#untagged tengigabitethernet 0/9-24
MXL1(conf-if-vl-12)#no shutdown
MXL1(conf-if-vl-12)#exit
MXL1(conf)#interface Vlan 13
MXL1(conf-if-vl-13)#untagged tengigabitethernet 0/25-32
MXL1(conf-if-vl-13)#no shutdown
MXL1(conf-if-vl-13)#exit
MXL1(conf)#interface range tengigabitethernet 0/1 - 32
MXL1(conf-if-range-te-0/1-32)#no shutdown
MXL1(conf-if-range-te-0/1-32)#exit
MXL1(conf)#exit
MXL1#
```

To see how the above commands affect the MXL’s running configuration, review the example MXL1 running configuration in Appendix A.

Trunk Port Downlink Configuration

Trunk ports can participate in multiple VLANs over one Ethernet interface and are often used for connection to virtualization hosts and other VLAN aware applications. To keep the traffic of the different VLANs from mixing, a numbered tag is inserted in each Ethernet frame (with the optional exception of the interface's "native" VLAN). To deploy a trunk port on an MXL running PVST follow these four steps.

Configure a Downlink as a Trunk Port

```
MXL2#configure
MXL2(conf)#interface tengigabitethernet 0/1
MXL2(conf-if-te-0/1)#portmode hybrid
MXL2(conf-if-te-0/1)#switchport
MXL2(conf-if-te-0/1)#spanning-tree pvst edge-port
MXL2(conf-if-te-0/1)#exit
MXL2(conf)#exit
MXL2#
```

The "switchport" setting enables a switch's Ethernet interface to participate in VLANs. A switchport enabled interface is referred to as a layer-2 interface. (Note: If the Ethernet interface had previously been configured as a layer-3 interface—which is one with an IP address directly configured on it—than the "no ip address" command would need to be run on the interface before the switchport feature could be enabled.)

The "portmode hybrid" command allows an Ethernet interface to carry both multiple tagged VLANs and a single untagged (also called a native) VLAN. If a given port is only expected to carry tagged VLANs or a single untagged VLAN, the "portmode hybrid" command may be omitted from its configuration.

The "spanning-tree pvst edge-port" command should only be run on ports that will connect to servers or other end nodes and never on ports that will connect to other switches. This command designates a port as an expected edge of the spanning tree (only switches participate in spanning tree) and enables it to begin forwarding traffic as soon as it's link is active (many seconds before the spanning-tree protocol would otherwise allow it to forward traffic).

With the switchport feature enabled, the Ethernet interface is now ready for one or more tagged VLANs to be configured for it.

Configure Tagged VLANs for a Trunk Port Interface

```
MXL2#configure
MXL2(conf)#interface vlan 11
MXL2(conf-if-vl-11)#tagged tengigabitethernet 0/1
MXL2(conf-if-vl-11)#no shutdown
MXL2(conf-if-vl-11)#exit
MXL2(conf)#interface vlan 12
MXL2(conf-if-vl-12)#tagged tengigabitethernet 0/1
MXL2(conf-if-vl-12)#no shutdown
MXL2(conf-if-vl-12)#exit
MXL2(conf)#exit
MXL2#
```

And the "portmode hybrid" command allows a trunk port to also carry a single untagged (or native) VLAN.

Configure a Native VLAN for a Trunk Port Interface

```
MXL2#configure
MXL2(conf)#interface vlan 13
MXL2(conf-if-vl-13)#untagged tengigabitethernet 0/1
MXL2(conf-if-vl-13)#no shutdown
MXL2(conf-if-vl-13)#exit
MXL2(conf)#exit
MXL2#
```

Now that the expected VLANs are configured for the trunk port, it is ready to be enabled. (To keep server network traffic where it is expected to be on the network, it is important to enable downlinks only after they are correctly configured.)

Enable a Configured Ethernet Interface

```
MXL2#configure
MXL2(conf)#interface tengigabitethernet 0/1
MXL2(conf-if-te-0/1)#no shutdown
MXL2(conf-if-te-0/1)#exit
MXL2(conf)#exit
MXL2#
```

For administrative convenience the above steps can be applied to a range of switch interfaces during configuration. In the following example, the downlinks of MXL2 in the example network are all configured as trunk ports with VLANs 11 and 12 tagged and VLAN 13 as their native VLAN.

Trunk Port CLI Example Using Interface Range

```
MXL2#configure
MXL2(conf)#interface range tengigabitethernet 0/1 - 32
MXL2(conf-if-range-te-0/1-32)#portmode hybrid
MXL2(conf-if-range-te-0/1-32)#switchport
MXL2(conf-if-range-te-0/1-32)#exit
MXL2(conf)#interface range vlan 11 - 12
MXL2(conf-if-range-vl-11-12)#tagged tengigabitethernet 0/1-32
MXL2(conf-if-range-vl-11-12)#no shutdown
MXL2(conf-if-range-vl-11-12)#exit
MXL2(conf)#interface Vlan 13
MXL2(conf-if-vl-13)#untagged tengigabitethernet 0/1-32
MXL2(conf-if-vl-13)#no shutdown
MXL2(conf-if-vl-13)#exit
MXL2(conf)#interface Range TenGigabitEthernet 0/1 - 32
MXL2(conf-if-range)#no shutdown
MXL2(conf-if-range)#exit
MXL2(conf)#exit
MXL2#
```

To see how the above commands affect the MXL's running configuration, review the example MXL2 running configuration in Appendix A.

Link Aggregation Group Configuration

As network switches interconnect more servers and other end nodes, a single active link between devices—especially between switches—often does not provide enough throughput or enough link resiliency for the applications running on the network. A common solution for this is to use link aggregation. A set of equal speed links connected between two devices can be configured to behave as a single connection by spreading traffic out over the participating links and keeping the aggregate connection active as long as there is at least one active member link for the group. This is called a LAG (Link Aggregation Group).

LAGs, while sometimes used between a server and a switch, are more often used between two switches as they are in the example network presented in this document. The MXL can support up to 16 links in a single LAG. The steps to configure one of the LAGs in the example network are shown below.

Configure the Link Aggregation Port Channel

```
MXL1#configure
MXL1(conf)#interface port-channel 20
MXL1(conf-if-po-20)#switchport
MXL1(conf-if-po-20)#exit
MXL1(conf)#exit
MXL1#
```

Configure the Link Aggregation Member Interfaces

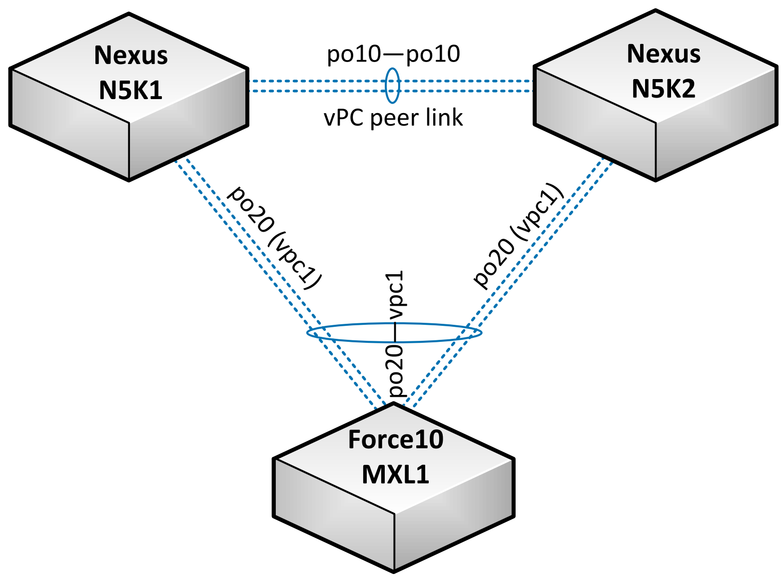
```
MXL1#configure
MXL1(conf)#interface range tengigabitethernet 0/33 - 36
MXL1(conf-if-range-te-0/33-36)#port-channel-protocol lacp
MXL1(conf-if-range-te-0/33-36-lacp)#port-channel 20 mode active
MXL1(conf-if-range-te-0/33-36-lacp)#exit
MXL1(conf-if-range-te-0/33-36)#no shutdown
MXL1(conf-if-range-te-0/33-36)#exit
MXL1(conf)#exit
MXL1#
```

! Note - observations have shown that the default hashing method on the Cisco Nexus may need to be changed in order to perform in the expected manner. In some instances the testing performed was able to show a marked bandwidth increase in a 4-port LAG by setting the "**port-channel load-balance ethernet source-dest-port**" global setting on a Cisco Nexus switch.

Trunk Port Uplink Configuration with MLAG at Top of Rack

The preferred deployment topology for an MXL in the access layer of a Cisco Nexus network is to use the Nexus vPC feature between two top-of-rack peer switches to provide a multi-chassis LAG connection to the MXL. While spanning-tree is still important to deploy to avoid loops forming due to switch or server misconfiguration or mishap, the vPC provides more resilient connectivity and better uplink throughput utilization for most applications. In this example, configuring the Nexus downlinks as vPC interfaces will cause the two separate Nexus switches to appear as a single logical switch to the MXL switch. This makes it possible to run active-active LAGs that are distributed across both Nexus switches from the MXL switch.

Figure 1. Topology Diagram: Top of Rack Nexus 5K using vPC



Deploying the multi-chassis vPC feature requires establishing the vPC peer relationship between two top-of-rack switches, configuring both halves of the multi-chassis LAG, and configuring the MXL's single uplink LAG. In addition, instructions are given for enabling per-VLAN spanning tree and VLAN membership configuration for the MXL.

All of the following configuration commands assume that the physical links between the involved switches are already in place.

Establish the Nexus Top-of-Rack vPC Peer Relationship

In the example network the top-of-rack switches N5K1 and N5K2 are vPC peers with a four-link LAG connected between them. The management address of each Nexus switch is used as the source and destination address in order to establish the keepalive connection between them. Following are the commands used to establish the peer relationship.

```
N5K1#configure
N5K1 (conf)#feature enable lacp
N5K1 (conf)#feature enable vpc
N5K1 (conf)#vpc domain 10
```

```
N5K1 (conf-vpc-10)#peer-keepalive destination 172.25.188.61 source
172.25.188.60
N5K1 (conf-vpc-10)#ip arp synchronize
N5K1 (conf-vpc-10)#exit
N5K1 (conf)#interface port-channel 10
N5K1 (conf-if-po-10)#switchport mode trunk
N5K1 (conf-if-po-10)#vpc peer-link
N5K1 (conf-if-po-10)#exit
N5K1 (conf)#interface Ethernet 1/11-14
N5K1 (conf-if-range)#switchport mode trunk
N5K1 (conf-if-range)#channel-group 10 mode active
N5K1 (conf-if-range)#exit
N5K1 (conf)#exit
N5K1#
```

This same command set is now run on N5K2 (except with the keepalive destination and source IP addresses reversed).

```
N5K2#configure
N5K2 (conf)#feature enable lacp
N5K2 (conf)#feature enable vpc
N5K2 (conf)#vpc domain 10
N5K2 (conf-vpc-10)#peer-keepalive destination 172.25.188.60 source
172.25.188.61
N5K2 (conf-vpc-10)#ip arp synchronize
N5K2 (conf-vpc-10)#exit
N5K2 (conf)#interface port-channel 10
N5K2 (conf-if-po-30)#switchport mode trunk
N5K2 (conf-if-po-30)#spanning-tree port type network
N5K2 (conf-if-po-30)#vpc peer-link
N5K2 (conf-if-po-30)#exit
N5K2 (conf)#interface Ethernet 1/11-12
N5K2 (conf-if-range)#switchport mode trunk
N5K2 (conf-if-range)#channel-group 10 mode active
N5K2 (conf-if-range)#exit
N5K2 (conf)#exit
N5K2#
```

In the example network the port-channels are configured to switchport trunk mode with the default setting allowing all available VLANs onto them.

Configure the Nexus vPC Multi-chassis LAG

```
N5K1#configure
N5K1 (conf)#interface port-channel 20
N5K1 (conf-if-po-20)#switchport mode trunk
N5K1 (conf-if-po-20)#vpc 20
N5K1 (conf-if-po-20)#exit
```

```
N5K1 (conf)#interface Ethernet 1/1-2
N5K1 (conf-if-range)#switchport mode trunk
N5K1 (conf-if-range)#channel-group 20 mode active
N5K1 (conf-if-range)#exit
N5K1 (conf)#exit
N5K1#
```

This same command set is now run on N5K2 to configure its half of the multi-chassis LAG.

```
N5K2#configure
N5K2 (conf)#interface port-channel 20
N5K2 (conf-if-po-20)#switchport mode trunk
N5K2 (conf-if-po-20)#vpc 20
N5K2 (conf-if-po-20)#exit
N5K2 (conf)#interface Ethernet 1/1-2
N5K2 (conf-if-range)#switchport mode trunk
N5K2 (conf-if-range)#channel-group 20 mode active
N5K2 (conf-if-range)#exit
N5K2 (conf)#exit
N5K2#
```

In the example network the port-channels are configured to switchport trunk mode and left with the default of allowing all available VLANs onto them.

Prepare the Nexus Per-VLAN Spanning Tree Instances

The Nexus' long method for deriving spanning-tree path cost is more compatible with the Force10 switch's spanning-tree implementation than the default short method. This setting is used in the example network because it provides a cleaner end result that is easier to understand. However, since in our example network the Force10 switches are only employed in an access switch role and do not provide further connectivity to other switches, this setting does not affect the calculated spanning trees' active paths.

Here are the commands for the N5K1 to enable the long pathcost method.

```
N5K1#configure
N5K1 (conf)#spanning-tree pathcost method long
N5K1 (conf)#exit
N5K1#
```

The same commands for the N5K2 switch follow.

```
N5K2#configure
N5K2 (conf)#spanning-tree pathcost method long
N5K2 (conf)#exit
N5K2#
```

Enable Per-VLAN Spanning Tree on the MXL

Most Cisco Nexus networks run per-VLAN spanning tree and the MXL is capable of natively participating in the version of the spanning tree protocol. Here are the commands required to enable it.

```
MXL1#configure
MXL1(conf)#protocol spanning-tree pvst
MXL1(conf-pvst)#no disable
MXL1(conf-pvst)#exit
MXL1(conf)#exit
MXL1#
```

Configure the MXL Uplink Port-Channel Links

On the MXL side of the Nexus vPC multi-chassis LAG, there is only a single LAG configured for uplink on both top-of-rack switches. Here are the commands used to configure the uplink LAG on MXL1 in the example network.

```
MXL1#configure
MXL1(conf)#interface port-channel 20
MXL1(conf-po-20)#switchport
MXL1(conf-po-20)#exit
MXL1(conf)#interface range tengigabitethernet 0/33 - 36
MXL1(conf-if-range-te-0/33-36)#port-channel-protocol lacp
MXL1(conf-if-range-te-0/33-36-lacp)#port-channel 20 mode active
MXL1(conf-if-range-te-0/33-36-lacp)#exit
MXL1(conf-if-range-te-0/33-36)#exit
MXL1(conf)#exit
MXL1#
```

Note that on the MXL using the 10Gb links 33 – 36 as in the example network requires splitting the 40Gb port 33 into four 10Gb links as covered in the Initial Deployment section.

Configure Tagged VLANs for the Trunk Port Uplinks

A substantial difference between the MXL's FTOS CLI and the Nexus CLI is that VLAN port membership is configured inside the VLAN interface (instead of inside the Ethernet interface). Here are the commands run to establish vlan membership for the uplink LAG on MXL1 in the example network.

```
MXL1#configure
MXL1(conf)#interface range vlan 11 - 13
MXL1(conf-if-range-vl-11-13)#tagged Port-channel 20
MXL1(conf-if-range-vl-11-13)#exit
MXL1(conf)#exit
MXL1#
```

Note that the VLANs are only added to the port-channel interface and not to the individual link member interfaces. On the MXL, individual links in a port-channel do not switch traffic when not attached to the port channel.

Enable the MXL Uplink Port-Channels

Now that the expected spanning-tree, port-channel, switchport, and VLAN settings are in place; the following commands to enable the link members of the uplink port-channel can be run.

```
MXL1#configure
MXL1(conf)#interface range tengigabitethernet 0/33 - 36
MXL1(conf-if-range-te-0/33-36)#no shutdown
MXL1(conf-if-range-te-0/33-36)#exit
MXL1(conf)#exit
MXL1#
```

These commands could also have been entered earlier in the process, but it is recommended practice to only enable links after they are configured in the expected manner.

Verify the MXL1's Spanning-tree State

Below is the output from MXL1 showing the general spanning-tree state of VLANs 11-13 and the forwarding status of its uplink LAG. The internal interface state has been snipped from the output to shorten the length and the font size is smaller to allow the formatted content to fit widthwise.

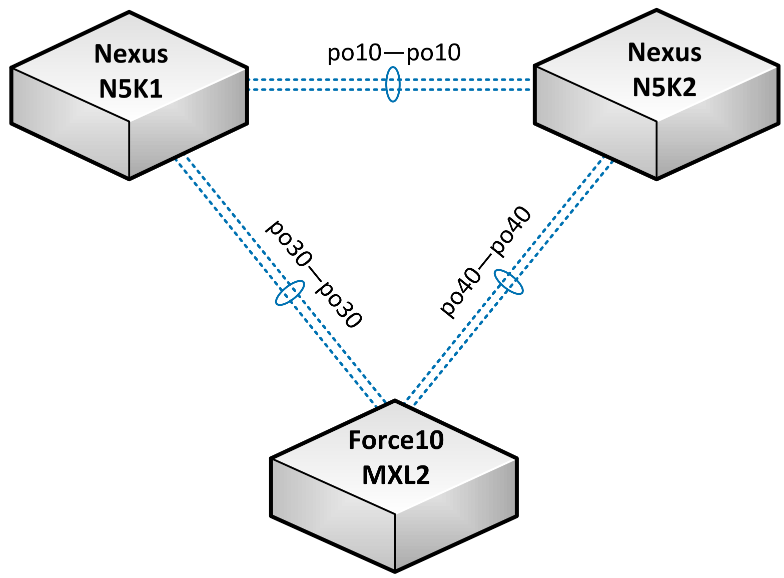
```
MXL1#show spanning-tree pvst vlan 11 brief
VLAN 11
Executing IEEE compatible Spanning Tree Protocol
Root ID    Priority 24587, Address 547f.ee53.3ec1
Root Bridge hello time 2, max age 20, forward delay 15
Bridge ID    Priority 32768, Address 001e.c9f1.053e
Configured hello time 2, max age 20, forward delay 15
Bpdu filter disabled globally

Interface                                                Designated
 Name       PortID   Prio Cost    Sts          Cost       Bridge ID          PortID
---------- -------- ---- ------  ----------- ------- -------------------- --------
Po 20      128.21   128  1400    FWD         1400    24587 0023.04ee.be0a  144.19
[...snipped...]
MXL1#show spanning-tree pvst vlan 12 brief
VLAN 12
Executing IEEE compatible Spanning Tree Protocol
Root ID    Priority 24588, Address 547f.ee56.5581
Root Bridge hello time 2, max age 20, forward delay 15
Bridge ID    Priority 32768, Address 001e.c9f1.053e
Configured hello time 2, max age 20, forward delay 15
Bpdu filter disabled globally

Interface                                                Designated
 Name       PortID   Prio Cost    Sts          Cost       Bridge ID          PortID
---------- -------- ---- ------  ----------- ------- -------------------- --------
Po 20      128.21   128  1400    FWD         1900    28684 0023.04ee.be0a  144.19
[...snipped...]
MXL1#show spanning-tree pvst vlan 13 brief
VLAN 13
Executing IEEE compatible Spanning Tree Protocol
Root ID    Priority 24589, Address 547f.ee53.3ec1
Root Bridge hello time 2, max age 20, forward delay 15
Bridge ID    Priority 32768, Address 001e.c9f1.053e
Configured hello time 2, max age 20, forward delay 15
Bpdu filter disabled globally

Interface                                                Designated
 Name       PortID   Prio Cost    Sts          Cost       Bridge ID          PortID
---------- -------- ---- ------  ----------- ------- -------------------- --------
```

```
Po 20      128.21   128  1400    FWD         1400    24589 0023.04ee.be0a  144.19
[...snipped...]
```

Trunk Port Uplink Configuration with Per-VLAN Spanning Tree

The availability of Per-VLAN spanning tree on the MXL allows it to natively integrate into the spanning-tree environment of a Cisco Nexus network. In the following example the MXL2 switch is configured for the PVST environment of the example network. As shown in the below diagram, it will have two LAGs that uplink it to two top-of-rack Nexus switches. Per-VLAN spanning tree will allow different VLANs to be active on each uplink LAG.

Figure 2. Topology Diagram: Top of Rack Nexus 5K using PVST+



Configure the Nexus Top-of-Rack Port-Channels

It is good practice to have the top-of-rack switches configured for the MXL connections before bringing the connections up. In the example network each top-of-rack Nexus switch provides a two-link LAG connection to MXL2. Here are the commands run to configure the LAG on N5K1.

```
N5K1#configure
N5K1 (conf)#feature enable lacp
N5K1 (conf)#interface port-channel 30
N5K1 (conf-if-po-30)#switchport mode trunk
N5K1 (conf-if-po-30)#exit
N5K1 (conf)#interface Ethernet 1/17 -18
N5K1 (conf-if-range)#switchport mode trunk
N5K1 (conf-if-range)#channel-group 30 mode active
N5K1 (conf-if-range)#exit
N5K1 (conf)#exit
N5K1#
```

This same command set is now run on N5K2 (except using N5K2's port-channel 40).

```
N5K2#configure
N5K2 (conf)#feature enable lacp
```

```
N5K2 (conf)#interface port-channel 40
N5K2 (conf-if-po-40)#switchport mode trunk
N5K2 (conf-if-po-40)#exit
N5K2 (conf)#interface Ethernet 1/17 - 18
N5K2 (conf-if-range)#switchport mode trunk
N5K2 (conf-if-range)#channel-group 40 mode active
N5K2 (conf-if-range)#exit
N5K2 (conf)#exit
N5K2#
```

Prepare the Nexus Per-VLAN Spanning Tree Instances

In the following CLI commands N5K1 is configured as primary or secondary root of the three VLANs on the example network.

```
N5K1#configure
N5K1 (conf)#spanning-tree pathcost method long
N5K1 (conf)#spanning-tree vlan 11,13 root secondary
N5K1 (conf)#spanning-tree vlan 12 root primary
N5K1 (conf)#exit
N5K1#
```

And now the equivalent commands for N5K2. Note that this time the primary and secondary roles are reversed.

```
N5K2#configure
N5K2 (conf)#spanning-tree pathcost method long
N5K2 (conf)#spanning-tree vlan 11,13 root primary
N5K2 (conf)#spanning-tree vlan 12 root secondary
N5K2 (conf)#exit
N5K2#
```

Enable Per-VLAN Spanning Tree on the MXL

The following commands enable per-VLAN spanning tree on MXL2 allowing it to integrate natively into the Nexus layer-2 switching environment.

```
MXL2#configure
MXL2(conf)#protocol spanning-tree pvst
MXL2(conf-pvst)#no disable
MXL2(conf-pvst)#exit
MXL2(conf)#exit
MXL2#
```

Configure the MXL Uplink Port-Channels

The following commands configure MXL2’s LAG interface for connection to N5K1.

```
MXL2#configure
MXL2(conf)#interface port-channel 30
MXL2(conf-if-po-30)#switchport
MXL2(conf-if-po-30)#exit
MXL2(conf)#exit
MXL2#
```

And now the commands that configure MXL2's LAG connection to N5K2.

```
MXL2#configure
MXL2(conf)#interface port-channel 40
MXL2(conf-if-po-40)#switchport
MXL2(conf-if-po-40)#no shutdown
MXL2(conf-if-po-40)#exit
MXL2(conf)#exit
MXL2#
```

Configure Tagged VLANs for the Trunk Port Uplinks

In the following commands, note that the MXL's VLAN membership is configured significantly differently than on the Nexus CLI—port membership is managed within the VLAN interface configuration.

```
MXL2#configure
MXL2(conf)#interface range vlan 11 - 13
MXL2(conf-if-range-vl-11-13)#tagged port-channel 30,40
MXL2(conf-if-range-vl-11-13)#no shutdown
MXL2(conf-if-range-vl-11-13)#exit
MXL2(conf)#exit
MXL2#
```

Enable the MXL Uplink Port-Channel Links

The final step in the CLI configuration of the uplink LAGs is to configure and enable the member interfaces of each LAG. Here are the commands for MXL2's connection to N5K1.

```
MXL2#configure
MXL2(conf)#interface range tengigabitethernet 0/33 - 34
MXL2(conf-if-range-te-0/33-34)#port-channel-protocol lacp
MXL2(conf-if-range-te-0/33-34-lacp)#port-channel 30 mode active
MXL2(conf-if-range-te-0/33-34-lacp)#exit
MXL2(conf-if-range-te-0/33-36)#no shutdown
MXL2(conf-if-range-te-0/33-36)#exit
MXL2(conf)#exit
MXL2#
```

And here are the commands for configuring and enabling MXL2's LAG members for its connection to N5K2.

```
MXL2#configure
MXL2(conf)#interface range tengigabitethernet 0/35 - 36
MXL2(conf-if-range-te-0/35-36)#port-channel-protocol lacp
MXL2(conf-if-range-te-0/35-36-lacp)#port-channel 40 mode active
MXL2(conf-if-range-te-0/35-36-lacp)#exit
MXL2(conf-if-range-te-0/33-36)#no shutdown
MXL2(conf-if-range-te-0/35-36)#exit
MXL2(conf)#exit
MXL2#
```

Verify the MXL2's Spanning-tree State

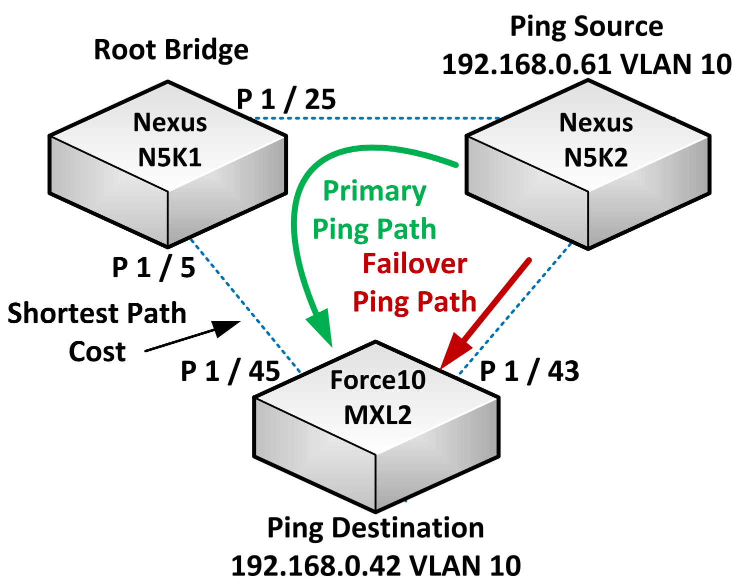
Below is the output from MXL2 showing the general spanning-tree state for VLANs 11-13 and the forwarding status of its uplink LAGs. The internal interface state has been snipped from the output to shorten the length and the font size is smaller to allow the formatted content to fit widthwise.

```
MXL2#show spanning-tree pvst vlan 11 brief
VLAN 11
Executing IEEE compatible Spanning Tree Protocol
Root ID    Priority 24587, Address 547f.ee53.3ec1
Root Bridge hello time 2, max age 20, forward delay 15
Bridge ID    Priority 32768, Address 001e.c9f1.051e
Configured hello time 2, max age 20, forward delay 15
Bpdu filter disabled globally

Interface                                                Designated
 Name      PortID   Prio Cost    Sts          Cost       Bridge ID          PortID
---------- -------- ---- ------  ----------- -------  -------------------- --------
Po 30      128.31   128  1800    BLK         1800     28683 547f.ee56.5581  144.29
Po 40      128.41   128  1800    FWD         1800     24587 547f.ee53.3ec1  144.39
[...snipped...]
MXL2#show spanning-tree pvst vlan 12 brief
VLAN 12
Executing IEEE compatible Spanning Tree Protocol
Root ID    Priority 24588, Address 547f.ee56.5581
Root Bridge hello time 2, max age 20, forward delay 15
Bridge ID    Priority 32768, Address 001e.c9f1.051e
Configured hello time 2, max age 20, forward delay 15
Bpdu filter disabled globally

Interface                                                Designated
 Name      PortID   Prio Cost    Sts          Cost       Bridge ID          PortID
---------- -------- ---- ------  ----------- -------  -------------------- --------
Po 30      128.31   128  1800    FWD         1800     24588 547f.ee56.5581  144.29
Po 40      128.41   128  1800    BLK         1800     28684 547f.ee53.3ec1  144.39
[...snipped...]
MXL2#show spanning-tree pvst vlan 13 brief
VLAN 13
Executing IEEE compatible Spanning Tree Protocol
Root ID    Priority 24589, Address 547f.ee53.3ec1
Root Bridge hello time 2, max age 20, forward delay 15
Bridge ID    Priority 32768, Address 001e.c9f1.051e
Configured hello time 2, max age 20, forward delay 15
Bpdu filter disabled globally

Interface                                                Designated
 Name      PortID   Prio Cost    Sts          Cost       Bridge ID          PortID
---------- -------- ---- ------  ----------- -------  -------------------- --------
Po 30      128.31   128  1800    BLK         1800     28685 547f.ee56.5581  144.29
Po 40      128.41   128  1800    FWD         1800     24589 547f.ee53.3ec1  144.39
[...snipped...]
```

Spanning Tree Protocol Failover and Failback Observations

The primary role of any Spanning Tree Protocol is to prevent loops from occurring in a network topology. The secondary role is to re-converge after a topology change (link failure) as quickly as possible. The Force10 MXL blade switch supports Spanning Tree Protocol (IEEE 802.1d), Rapid Spanning Tree Protocol (802.1w), Multiple Spanning Tree Protocol (IEEE 802.1s) and Per VLAN Spanning Tree Protocol. Some observations regarding the interoperability of Nexus switches and the Force10 MXL blade switch using spanning-tree protocols are presented below. Figure 3 depicts the switch configuration used during these observations.

Figure 3. Topology Diagram: STP Interoperability Observations Lab



Lab Environment

The configuration is composed of two Nexus switches. Nexus switches can run in 2 STP modes, either MSTP or Rapid PVST+. The behavior of both modes was observed. A single Force10 MXL blade switch was used in the configuration.

Please note: The 2 Nexus switches used for the STP observations were not the same as the ones used earlier in this document and as a result have different MAC addresses etc.

Cisco's vPC was not used in the configuration. If vPC had been configured on the Nexus downlink ports to the Force10 MXL blade switch, then there would have been a preference to have a LAG configured on the two MXL ports so that both links could be used in an Active-Active LACP scenario. In that case, when spanning tree information is observed from the Force10 MXL blade switch's perspective, the Port Channel's role would be Root, but since there is only one logical path (the port channel) there would be no interface designated as an Alternate. By not using vPC, a triangle of 3 independent switches was created and STP was forced to block at least one of the Force10 MXL blade switch links in order to guarantee a loop free topology.

In the configuration for this lab, the N5K1 was configured as the Root Bridge Primary and the N5K2 was configured as the Root Bridge Secondary. Since N5K1 was the Root Bridge, the link between it and the

Force10 MXL blade switch was the link with the lowest path cost and should, from the Force10 MXL's perspective, have always been in the Forwarding Status and Root Role when it was available.

In this lab 6 different tests were performed for the MSTP to MSTP testing, and then again for the Rapid PVST+ to PVST testing. Each test was performed while a series of pings were in progress. The pings originated from the N5K2 and were destined to a Switched Virtual Interface configured on the Force10 MXL2 blade switch. The pings were executed with an interval of 1 second and a count of 1000, causing the system to execute 1000 pings, unless stopped, and to do so every 1 second. While the pings were in progress, various tests involving failovers and failbacks were performed allowing the lab environment to be monitored for the number of pings that were "lost" before the network could successfully re-converge. By determining how many pings were "lost" and knowing that one per second was being transmitted, it was possible to measure the failover and failback times with a granularity of 1 second. The tests that were performed are as follows:

1) **Failover: Local Interface Failure Simulation**. In this test a shutdown was issued on the Force10 MXL blade switch's uplink interface which currently had a status of Forwarding and was in the Root Role – that is, its interface 1/45. This simulated a failure of that interface. This should cause the interface 1/43 to be placed in the Root Role with a Status of Forwarding.
2) **Failback: Local Interface**. In this test a no shutdown was issued on the Force10 MXL2's interface 1/45. Since this interface has the lowest path cost to the Root Bridge it should immediately be placed in the Role of Root with a Status of Forwarding and the Force10 MXL2's interface 1/43 should be placed in the Role of Alternate with a Status of Blocking.
3) **Failover: Uplink Failure Simulation (Uplink Fast).** In this test a shutdown was issued on N5K1's interface 1/5. This should cause the Force10 MXL2 blade switch to place its interface 1/45 into a Status and Role of DIS(carding) and its interface 1/43 in to the role of Root with a Status of Forwarding.
4) **Failback: Uplink Failure Simulation (Uplink Fast).** In this test a no shutdown was issued on N5K1's interface 1/5. This should cause the Force10 MXL2 blade switch to failover to its interface 1/45 with a Status of Forwarding and a Role of Root since it has the lowest path cost to the Root Bridge.
5) **Failover: Backbone Failure Simulation (Backbone Fast).** In this test a shutdown was issued on N5K1's interface 1/25. This should cause MXL2 to failover to its interface 1/43 interface with a Role of Root.
6) **Failback: Backbone Failure Simulation (Backbone Fast).** In this test a no shut command was issued on NX5K1's interface 1/25. This should cause STP on MXL2 to place its interface 1/43 into a Role of Alternate and its interface 1/45 into the Role of Root since it has the lowest path cost to the Root Bridge.

The agenda then was to run each of these 6 tests in 2 environments:

1) MSTP running on the Nexus Switches and MSTP running on the MXL
2) Rapid PVST+ running on the Nexus Switches and PVST running on the MXL

Using the ping command set at 1 second intervals, it was possible to tell how many seconds it took for the Spanning Tree to re-converge after each failover and failback by counting the number of pings that were "lost" – the number of pings that did not result in a response from the destination address.

Nexus MSTP and MXL MSTP Observations

As mentioned previously the Nexus switches can run in one of two Spanning Tree modes, MSTP or Rapid PVST+

Here is the command for configuring the Nexus Switches to run MSTP:

```
N5K2#configure
N5K2 (conf)#spanning-tree mode mst
N5K2 (conf)#exit
N5K2#
```

The process for configuring the MXL switch to run MSTP.

First, check to see if STP is already configured. If it is, then disable and remove it from the configuration, before configuring MSTP.

Determine if Spanning Tree is already enabled

In this example, RSTP was already running so it was disabled before configuring MSTP.

```
MXL2#show run | grep spanning
protocol spanning-tree rstp
MXL2#configure
MXL2#protocol spanning-tree rstp
MXL2(conf-rstp)#disable
MXL2(conf-rstp)#exit
MXL2(conf)#no protocol spanning-tree rstp
MXL2(conf)#exit
MXL2#
```

Enable MSTP Spanning Tree on the MXL

Here are the commands to configure MSTP:

```
MXL2#configure
MXL2(conf)#protocol spanning-tree mstp
MXL2(conf-pvst)#no disable
MXL2(conf-pvst)#exit
MXL2(conf)#exit
MXL2#
```

Verify that Spanning Tree is Running as Expected

What is shown below as a result of the show spanning-tree command is:

1) The MXL's MSTP has converged with the Nexus switches and has accepted N5K1 as the Root Bridge with its Priority of 24576
2) Interface 1/43 has a status of Blocking and is in the Role of Alternate
3) Interface 1/45 has a status of Forwarding and is in the Role of Root.
4) Given that 1/45 has the lowest path cost to the Root Bridge, this is the expected topology

MXL2#show spanning-tree msti 0 brief

MSTI 0 VLANs mapped 1-4094

```
Executing IEEE compatible Spanning Tree Protocol

Root ID    Priority 24576, Address 547f.ee7a.7301

Root Bridge hello time 2, max age 20, forward delay 15, max hops 19

Bridge ID    Priority 32768, Address 001e.c9f1.0153

Configured hello time 2, max age 20, forward delay 15, max hops 20

Bpdu filter disabled globally

CIST regional root ID Priority 24576, Address 547f.ee7a.7301

CIST external path cost 0


Interface                              Designated

 Name      PortID   Prio Cost    Sts       Cost       Bridge ID       PortID

---------- -------- ---- ------- ----------- ------- -------------------- --------

Po 1       128.2    128  200000  DIS       2000    32768 001e.c9f1.0153 128.2

Te 1/1     128.186  128  2000    DIS       2000    32768 001e.c9f1.0153 128.186

Te 1/2     128.187  128  200000  DIS       2000    32768 001e.c9f1.0153 128.187

...

...

Te 1/30    128.215  128  200000  DIS       2000    32768 001e.c9f1.0153 128.215

Te 1/31    128.216  128  200000  DIS       2000    32768 001e.c9f1.0153 128.216

Te 1/32    128.217  128  200000  DIS       2000    32768 001e.c9f1.0153 128.217

Te 1/43    128.228  128  2000    BLK       2000    28672 547f.ee6c.21c1 128.131

Te 1/45    128.230  128  2000    FWD       2000    24576 547f.ee7a.7301 128.133


Interface                                        Bpdu

 Name      Role   PortID   Prio Cost    Sts       Cost    Link-type Edge Filter Boundary

---------- ------ -------- ---- ------- ----------- ------- --------- ---- ------ --------

Po 1       Dis    128.2    128  200000  DIS       2000    P2P      No   No     No

Te 1/1     Dis    128.186  128  2000    DIS       2000    P2P      No   No     No
```

Te 1/2	Dis	128.187	128	200000	DIS	2000	P2P	No	No	No
...										
...										
Te 1/30	Dis	128.215	128	200000	DIS	2000	P2P	No	No	No
Te 1/31	Dis	128.216	128	200000	DIS	2000	P2P	No	No	No
Te 1/32	Dis	128.217	128	200000	DIS	2000	P2P	No	No	No
Te 1/43	Altr	128.228	128	2000	BLK	2000	P2P	No	No	No
Te 1/45	Root	128.230	128	2000	FWD	2000	P2P	No	No	No

MSTP Test - Failover: Local Interface Failure Simulation

After logging in to both the MXL2 and the N5K2, a ping was issued to MXL2 once every second from the N5K2. While the pings were in progress, interface 1/45 on MXL2 was shut down and an observation was made regarding how many pings were lost. As shown below, only one ping was lost. This means that the re-convergence took approximately 1 second to occur.

```
NX5K2# ping 192.168.0.42 interval 1 count 1000 PING 192.168.0.42 (192.168.0.42): 56 data bytes Request 0 timed out

64 bytes from 192.168.0.42: icmp_seq=1 ttl=254 time=3.228 ms

64 bytes from 192.168.0.42: icmp_seq=2 ttl=254 time=7.365 ms

64 bytes from 192.168.0.42: icmp_seq=3 ttl=254 time=5.275 ms

64 bytes from 192.168.0.42: icmp_seq=4 ttl=254 time=3.046 ms

64 bytes from 192.168.0.42: icmp_seq=5 ttl=254 time=7.768 ms

64 bytes from 192.168.0.42: icmp_seq=6 ttl=254 time=3.042 ms

64 bytes from 192.168.0.42: icmp_seq=7 ttl=254 time=3.031 ms

64 bytes from 192.168.0.42: icmp_seq=8 ttl=254 time=3.025 ms

64 bytes from 192.168.0.42: icmp_seq=9 ttl=254 time=3.051 ms

Request 10 timed out

64 bytes from 192.168.0.42: icmp_seq=11 ttl=254 time=4.052 ms

64 bytes from 192.168.0.42: icmp_seq=12 ttl=254 time=7.738 ms

64 bytes from 192.168.0.42: icmp_seq=13 ttl=254 time=3.07 ms

64 bytes from 192.168.0.42: icmp_seq=14 ttl=254 time=3.032 ms
```

```
64 bytes from 192.168.0.42: icmp_seq=15 ttl=254 time=3.035 ms
```

Verify that Spanning Tree is running as expected after the **Local Interface Failure Simulation**

What is shown below as a result of the show spanning-tree command is:

1) Interface 1/43 has a status of Forwarding and is in the Role of Root
2) Interface 1/45 has a status of Disabled and is in the Role of Discarding.
3) Given that 1/43 was previously in the Role of Alternate and with the failure of interface 1/45 has been assigned the Role of Root - this is the expected topology

```
MXL2#show spanning-tree msti 0 brief

MSTI 0 VLANs mapped  1-4094

Executing IEEE compatible Spanning Tree Protocol

Root ID    Priority 24576, Address 547f.ee7a.7301

Root Bridge hello time 2, max age 20, forward delay 15, max hops 18

Bridge ID    Priority 32768, Address 001e.c9f1.0153

Configured hello time 2, max age 20, forward delay 15, max hops 20

Bpdu filter disabled globally

CIST regional root ID Priority 24576, Address 547f.ee7a.7301

CIST external path cost 0


Interface                                    Designated

 Name      PortID   Prio Cost    Sts       Cost       Bridge ID       PortID

---------- -------- ---- ------- ----------- ------- -------------------- --------

Po 1       128.2     128  200000  DIS        4000    32768 001e.c9f1.0153 128.2

Te 1/1     128.186  128  2000    DIS        4000    32768 001e.c9f1.0153 128.186

Te 1/2     128.187  128  200000  DIS        4000    32768 001e.c9f1.0153 128.187

Te 1/3     128.188  128  2000    DIS        4000    32768 001e.c9f1.0153 128.188

...

Te 1/30    128.215  128  200000  DIS        4000    32768 001e.c9f1.0153 128.215
```

```
Te 1/31    128.216  128  200000  DIS       4000    32768 001e.c9f1.0153 128.216

Te 1/32    128.217  128  200000  DIS       4000    32768 001e.c9f1.0153 128.217

Te 1/43    128.228  128  2000    FWD       4000    28672 547f.ee6c.21c1 128.131

Te 1/45    128.230  128  2000    DIS       4000    32768 001e.c9f1.0153 128.230


Interface                                          Bpdu

 Name      Role   PortID   Prio Cost    Sts       Cost   Link-type Edge Filter Boundary

---------- ------ -------- ---- ------- ----------- ------- --------- ---- ------ --------

Po 1      Dis    128.2    128  200000  DIS       4000    P2P      No   No     No

Te 1/1     Dis    128.186  128  2000    DIS       4000    P2P      No   No     No

Te 1/2     Dis    128.187  128  200000  DIS       4000    P2P      No   No     No

Te 1/3     Dis    128.188  128  2000    DIS       4000    P2P      No   No     No

...

Te 1/30    Dis    128.215  128  200000  DIS       4000    P2P      No   No     No

Te 1/31    Dis    128.216  128  200000  DIS       4000    P2P      No   No     No

Te 1/32    Dis    128.217  128  200000  DIS       4000    P2P      No   No     No

Te 1/43    Root   128.228  128  2000    FWD       4000    P2P      No   No     No

Te 1/45    Dis    128.230  128  2000    DIS       4000    P2P      No   No     No

MXL2#
```

Nexus MSTP and MXL MSTP – Remainder of Test Observations

After the Local Port Failover and Failback scenarios were observed, the following 4 scenarios were also observed.

1) **Failover: Uplink Failure Simulation (Uplink Fast).** In this test a shutdown was issued on N5K1's interface 1/5. This caused MXL2 to place its interface 1/45 into a status of Discarding and its interface 1/43 in to the Role of Root. This scenario was observed with only a single ping being lost during the failover, indicating that the re-convergence took about 1 second

2) **Failback: Uplink Failure Simulation (Uplink Fast).** In this test a no shutdown was issued on N5K1's interface 1/5. This caused STP on MXL2 to place its interface 1/43 into a Role of Alternate and its interface 1/45 in to the Role of Root since it has the lowest path cost to the

Root Bridge. This scenario was observed with only a single ping being lost during the failback, indicating that the re-convergence took about 1 second.

3) **Failover: Backbone Failure Simulation (Backbone Fast**). In this test a shutdown was issued on N5K1's interface 1/25. This caused MXL2 to place its interface 1/43 interface into the Role of Designated. This scenario was observed with only a single ping being lost, indicating that the re-convergence took about 1 second.

4) **Failback: Backbone Failure Simulation (Backbone Fast).).** In this test a no shutdown was issued on NX5K1's interface 1/25. This caused STP on MXL2 to place its interface 1/43 into a Role of Alternate. This scenario was observed with only a single ping being lost, indicating that the re-convergence took about 1 second.

Nexus MSTP and MXL MSTP – Greater than 64 Instances

The Force10 MXL supports up to 64 MSTIs (Multiple Spanning Tree Instances). MSTI 0 is reserved for the CIST (Common Internal Spanning Tree) leaving Multiple Spanning Tree Instances 1 – 63 on which VLANs can be assigned. In the event that the Nexus switches are using more than 64 Instances, then the question arises regarding how to configure and interoperate with the Force10 MXL blade switch. The best practice is that VLANs assigned to instances greater than 63 on the Nexus switches, should be left to default to MSTI 0 (Common Internal Spanning Tree) on the Force10 MXL blade switch. The previous six tests were conducted in 2 MSTP environments:

1) **VLANs on Multiple Spanning Tree Instances in the 1 - 63 range.**

2) **VLANs left on Multiple Spanning Tree Instance 0 (CIST) as the Instance numbers as assigned on the Nexus switches could not be matched since they were greater than 63.**

The failover / failback results described above were observed in **both** environments.

Nexus Rapid PVST+ and MXL PVST Observations

As previously mentioned, the Nexus switches can run in one of two Spanning Tree modes, MSTP or Rapid PVST+

Here is the command for configuring the Nexus Switches to run Rapid PVST+:

```
N5K2#configure
N5K2 (conf)#spanning-tree mode rapid-pvst
N5K2 (conf)#exit
N5K2#
```

The process for configuring the MXL switch to run MSTP:

First, check to see if STP is already configured. If it is, then disable and remove it from the configuration, before configuring PVST.

Determine if Spanning Tree is already enabled

In this example, RSTP is already running so it was disabled before configuring PVST.

```
MXL2#show run | grep spanning
protocol spanning-tree rstp
MXL2#configure
MXL2#protocol spanning-tree rstp
```

```
MXL2(conf-rstp)#disable
MXL2(conf-rstp)#exit
MXL2(conf)#no protocol spanning-tree rstp
MXL2(conf)#exit
MXL2#
```

Enable PVST Spanning Tree on the MXL

Here are the commands to configure PVST:

```
MXL2#configure
MXL2(conf)#protocol spanning-tree pvst
MXL2(conf-pvst)#no disable
MXL2(conf-pvst)#exit
MXL2(conf)#exit
MXL2#
```

Verify that Spanning Tree is Running as Expected

What is show below as a result of the show spanning-tree command is:

1) The MXL's STP has converged with the Nexus switches and has accepted N5K1 as the Root Bridge with its Priority of 24576
2) Interface 1/43 has a status of Blocking and is in the Role of Alternate
3) Interface 1/45 has a status of Forwarding and is in the Role of Root.
4) Given that 1/45 has the lowest Path Cost to the Root Bridge, this is the expected topology

```
MXL2# show spanning-tree pvst vlan 10 brief

VLAN 10

Executing IEEE compatible Spanning Tree Protocol

Root ID    Priority 24586, Address 547f.ee7a.7301

Root Bridge hello time 2, max age 20, forward delay 15

Bridge ID    Priority 32768, Address 001e.c9f1.0153

Configured hello time 2, max age 20, forward delay 15

Bpdu filter disabled globally


Interface                                   Designated

 Name      PortID   Prio Cost    Sts       Cost      Bridge ID      PortID

---------- -------- ---- ------  ----------- ------- ------------------- --------
```

```
Te 1/43    128.228  128  2000    BLK       2000    28682 547f.ee6c.21c1  128.131

Te 1/45    128.230  128  2000    FWD        2000    24586 547f.ee7a.7301  128.133


Interface

 Name      Role   PortID   Prio Cost    Sts       Cost    Link-type Edge BpduFilter

---------- ------ -------- ---- ------- ----------- ------- --------- ---- ----------

Te 1/43    Altr   128.228  128  2000    BLK       2000    P2P      No   No

Te 1/45    Root   128.230  128  2000    FWD        2000    P2P      No   No

MXL2#
```

PVST Test - Failover: Local Interface Failure Simulation

After logging in to both the MXL2 and the N5K2, a ping was issued to MXL2 once every second from the NX5K2. While the pings were in progress, interface 1/45 on MXL2 was shut down an observation was made regarding how many ping requests were lost. As shown below, only one ping was lost. This means that the re-convergence took approximately 1 second to occur.

```
NX5K2# ping 192.168.0.42 interval 1 count 1000 PING 192.168.0.42 (192.168.0.42): 56 data
bytes Request 0 timed out

64 bytes from 192.168.0.42: icmp_seq=1 ttl=254 time=3.228 ms

64 bytes from 192.168.0.42: icmp_seq=2 ttl=254 time=7.365 ms

64 bytes from 192.168.0.42: icmp_seq=3 ttl=254 time=5.275 ms

64 bytes from 192.168.0.42: icmp_seq=4 ttl=254 time=3.046 ms

64 bytes from 192.168.0.42: icmp_seq=5 ttl=254 time=7.768 ms

64 bytes from 192.168.0.42: icmp_seq=6 ttl=254 time=3.042 ms

64 bytes from 192.168.0.42: icmp_seq=7 ttl=254 time=3.031 ms

64 bytes from 192.168.0.42: icmp_seq=8 ttl=254 time=3.025 ms

64 bytes from 192.168.0.42: icmp_seq=9 ttl=254 time=3.051 ms

64 bytes from 192.168.0.42: icmp_seq=10 ttl=254 time=3.07 ms

64 bytes from 192.168.0.42: icmp_seq=11 ttl=254 time=4.052 ms

64 bytes from 192.168.0.42: icmp_seq=12 ttl=254 time=7.738 ms

64 bytes from 192.168.0.42: icmp_seq=13 ttl=254 time=3.07 ms
```

```
Request 10 timed out

64 bytes from 192.168.0.42: icmp_seq=14 ttl=254 time=3.032 ms

64 bytes from 192.168.0.42: icmp_seq=15 ttl=254 time=3.035 ms

...
```

Verify that Spanning Tree is running as expected after the **Local Interface Failure Simulation**

What is shown below as a result of the show spanning-tree command is:

1) Interface 1/43 has a status of Forwarding and is in the Role of Root
2) Interface 1/45 has a status of Disabled and is in the Role of Discarding.
3) Given that 1/43 was previously in the Role of Alternate and with the failure of interface 1/45 has been assigned the Role of Root - this is the expected topology

```
MXL-R#show spanning-tree pvst vlan 10 brief

VLAN 10

Executing IEEE compatible Spanning Tree Protocol

Root ID    Priority 24586, Address 547f.ee7a.7301

Root Bridge hello time 2, max age 20, forward delay 15

Bridge ID    Priority 32768, Address 001e.c9f1.0153

Configured hello time 2, max age 20, forward delay 15

Bpdu filter disabled globally


Interface                                   Designated

 Name       PortID   Prio Cost     Sts        Cost       Bridge ID        PortID

---------- -------- ---- ------  ----------- ------- -------------------- --------

Te 1/43     128.228  128  2000     FWD        2002    28682 547f.ee6c.21c1  128.131

Te 1/45     128.230  128  2000     DIS        2002    32768 001e.c9f1.0153  128.230


Interface

 Name       Role    PortID   Prio Cost    Sts        Cost    Link-type Edge BpduFilter

---------- ------ -------- ---- ------- ----------- ------- --------- ---- ----------
```

Te 1/43 Root 128.228 128 2000 FWD 2002 P2P No No

Te 1/45 Dis 128.230 128 2000 DIS 2002 P2P No No

Nexus Rapid PVST+ and MXL PVST – Remainder of Test Observations

After the Local Port Failover and Failback scenarios were observed, the following 4 scenarios were also observed

1) **Failover: Uplink Failure Simulation (Uplink Fast).** In this test a shutdown was issued on N5K1's interface 1/5. This caused MXL2 to place its interface 1/45 into a status of Discarding and its interface 1/43 in to the role of Root. This scenario was observed with only a single ping being lost during the failover, indicating that the re-convergence took about 1 second
2) **Failback: Uplink Failure Simulation (Uplink Fast).** In this test a no shutdown on N5K1's interface 1/5 was issued. This caused PVST on MXL2 to place its interface 1/43 into a Role of Alternate and its interface 1/45 in to the Role of Root since it has the lowest path cost to the Root Bridge. This scenario was observed with only a single ping being lost during the failback, indicating that the re-convergence took about 1 second.
3) **Failover: Backbone Failure Simulation (Backbone Fast**). In this test a shutdown on N5K1's interface 1/25 was issued. This caused MXL2 to place its interface 1/43 interface into the Role of Designated. This scenario was observed with only a single ping being lost, indicating that the re-convergence took about 1 second.
4) **Failback: Backbone Failure Simulation (Backbone Fast).** In this test a no shutdown was issued on NX5K1's interface 1/25. This caused PVST on MXL2 to place its interface 1/43 into a role of Alternate. This scenario was observed with only a single ping being lost, indicating that the re-convergence took about 1 second.

Summary – Spanning Tree Observations

The first and primary role of any Spanning Tree Protocol is to prevent loops. The second role is to recognize network topology changes (link failures) and re-converge the network as quickly as possible. In each of the scenarios observed, the Dell Force10 MXL switch indeed proved that it is more than capable of doing just that and doing so in conjunction with Nexus switches.

Table 2. Results – Spanning Tree Observations

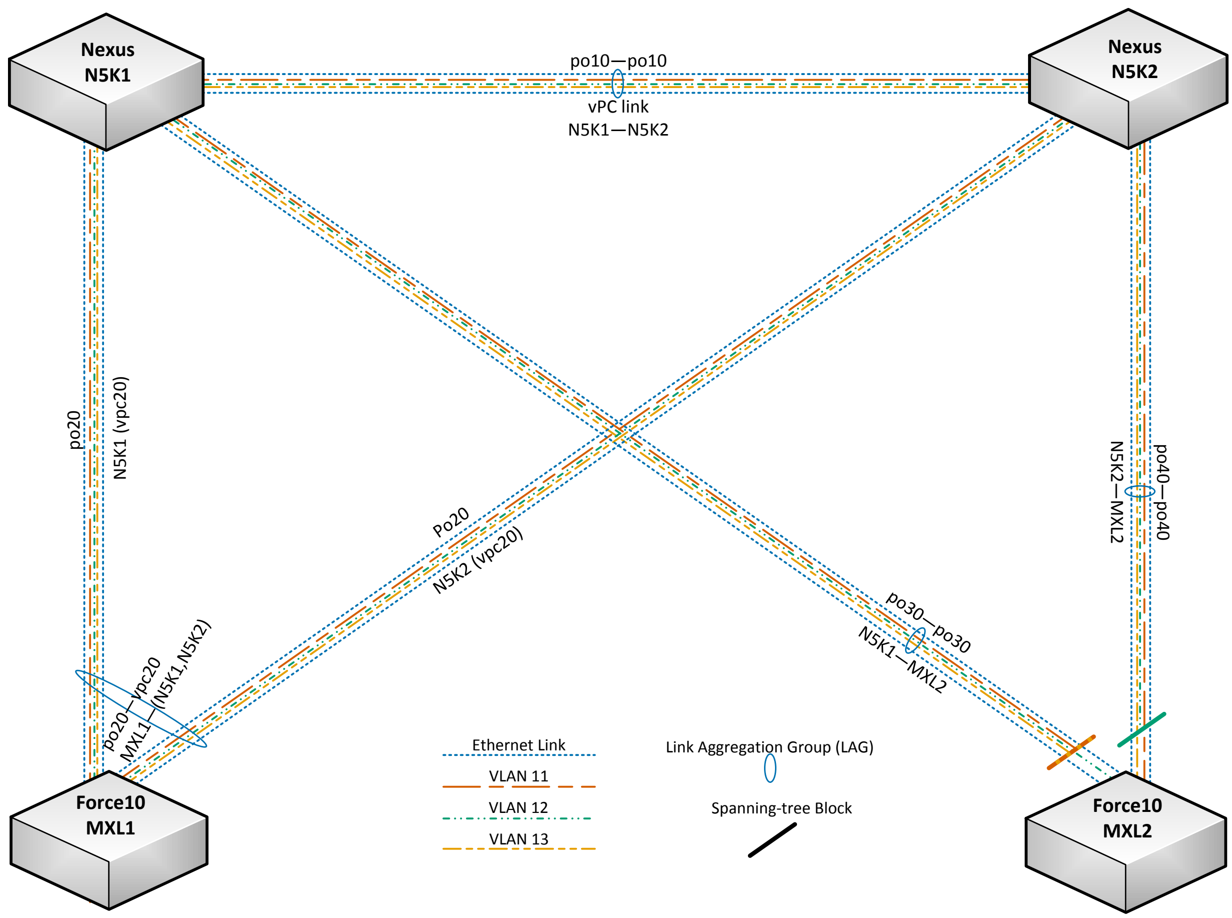
Scenario	Nexus MSTP / MXL MSTP	Nexus Rapid PVST+ / MXL PVST
Local Port Failover (Uplink Fast)	**1s**	**1s**
Local Port Failback (Uplink Fast)	**1s**	**1s**
Uplink Failover (Uplink Fast)	**1s**	**1s**

Uplink Failback (Uplink Fast)	**1s**	**1s**
Upstream Failover (Backbone Fast)	**1s**	**1s**
Upstream Failback (Backbone Fast)	**1s**	**1s**

Appendices

Appendix A: Referenced Network Topology and Device Configurations

Figure 4. Topology Diagram: Detailed, Full Example Network



N5K1 is the primary root of VLAN 12's spanning tree instance and the secondary root of VLANs 11 and 13's. Its port-channel 10 uses Ethernet interfaces 1/17-20, port-channel 20 uses Ethernet interfaces 1/1-2, and its port-channel 30 uses Ethernet interfaces 1/25-26.

N5K2 is the primary root of VLANs 11 and 13's spanning tree instances and the secondary root of VLAN 12's. Its port-channel 10 uses Ethernet interfaces 1/17-20, port-channel 20 uses Ethernet interfaces 1/1-2, and its port-channel 40 uses Ethernet interfaces 1/25-26.

MXL1 has a single northbound LAG that connects to a multi-chassis LAG (a Nexus vPC) provided by N5K1 and N5K2 and that actively passes traffic for all VLANs on all participating LAG members. Its port-channel 20 uses TenGigabitEthernet interfaces 0/33-36.

MXL2 has two northbound LAGs—one for N5K1 and one for N5K2. On MXL2, spanning tree blocks VLANs 11 and 13 on the connection to N5K1 and blocks VLAN 12 on the connection to N5K2. Its port-channel 30 uses TenGigabitEthernet interfaces 0/33-34 and port-channel 40 uses TenGigabitEthernet interfaces 0/35-36.

Documented Device Details

This document presents the deployment of two Force10 MXL switches attached to two top-of-rack Nexus 5548UP switches. The firmware revisions used are detailed in Table 2 below.

Table 3. Devices and Firmware Versions Used in this Document

Device Model	Firmware Version
Dell Force10 MXL	8.3.16.1
Cisco Nexus 5548UP	5.2(1)N1(1)

On the following pages is located the running configurations of the switches in the completed example network. The MXL1 and MXL2 configurations are listed side-by-side for ease of comparison between their configuration differences. All of the configurations presented have had their username entries removed.

MXL1 Running Configuration

```
! Version 8.3.16.1
!
boot system stack-unit 0 primary system: B:
!
redundancy auto-synchronize full
!
hostname MXL1
!
protocol spanning-tree pvst
 no disable
!
stack-unit 0 provision MXL-10/40GbE
!
stack-unit 0 port 33 portmode quad
!
interface TenGigabitEthernet 0/1
 no ip address
 switchport
 spanning-tree pvst edge-port
 no shutdown
!
interface TenGigabitEthernet 0/2
 no ip address
 switchport
 spanning-tree pvst edge-port
 no shutdown
!
interface TenGigabitEthernet 0/3
 no ip address
 switchport
 spanning-tree pvst edge-port
 no shutdown
!
interface TenGigabitEthernet 0/4
 no ip address
 switchport
 spanning-tree pvst edge-port
 no shutdown
!
interface TenGigabitEthernet 0/5
 no ip address
 switchport
 spanning-tree pvst edge-port
 no shutdown
!
interface TenGigabitEthernet 0/6
 no ip address
 switchport
 spanning-tree pvst edge-port
 no shutdown
!
interface TenGigabitEthernet 0/7
 no ip address
 switchport
 spanning-tree pvst edge-port
 no shutdown
!
interface TenGigabitEthernet 0/8
 no ip address
 switchport
 spanning-tree pvst edge-port
 no shutdown
!
```

MXL2 Running Configuration

```
! Version 8.3.16.1
!
boot system stack-unit 0 primary system: B:
!
redundancy auto-synchronize full
!
hostname MXL2
!
protocol spanning-tree pvst
 no disable
!
stack-unit 0 provision MXL-10/40GbE
!
stack-unit 0 port 33 portmode quad
!
interface TenGigabitEthernet 0/1
 no ip address
 portmode hybrid
 switchport
 spanning-tree pvst edge-port
 no shutdown
!
interface TenGigabitEthernet 0/2
 no ip address
 portmode hybrid
 switchport
 no shutdown
!
interface TenGigabitEthernet 0/3
 no ip address
 portmode hybrid
 switchport
 spanning-tree pvst edge-port
 no shutdown
!
interface TenGigabitEthernet 0/4
 no ip address
 portmode hybrid
 switchport
 spanning-tree pvst edge-port
 no shutdown
!
interface TenGigabitEthernet 0/5
 no ip address
 portmode hybrid
 switchport
 spanning-tree pvst edge-port
 no shutdown
!
interface TenGigabitEthernet 0/6
 no ip address
 portmode hybrid
 switchport
 spanning-tree pvst edge-port
 no shutdown
!
interface TenGigabitEthernet 0/7
 no ip address
 portmode hybrid
 switchport
 spanning-tree pvst edge-port
 no shutdown
!
interface TenGigabitEthernet 0/8
 no ip address
 portmode hybrid
 switchport
 spanning-tree pvst edge-port
 no shutdown
!
```

```
interface TenGigabitEthernet 0/9
 no ip address
 switchport
 spanning-tree pvst edge-port
 no shutdown
!
interface TenGigabitEthernet 0/10
 no ip address
 switchport
 spanning-tree pvst edge-port
 no shutdown
!
interface TenGigabitEthernet 0/11
 no ip address
 switchport
 spanning-tree pvst edge-port
 no shutdown
!
interface TenGigabitEthernet 0/12
 no ip address
 switchport
 spanning-tree pvst edge-port
 no shutdown
!
interface TenGigabitEthernet 0/13
 no ip address
 switchport
 spanning-tree pvst edge-port
 no shutdown
!
interface TenGigabitEthernet 0/14
 no ip address
 switchport
 spanning-tree pvst edge-port
 no shutdown
!
interface TenGigabitEthernet 0/15
 no ip address
 switchport
 spanning-tree pvst edge-port
 no shutdown
!
interface TenGigabitEthernet 0/16
 no ip address
 switchport
 spanning-tree pvst edge-port
 no shutdown
!
interface TenGigabitEthernet 0/17
 no ip address
 switchport
 spanning-tree pvst edge-port
 no shutdown
!
interface TenGigabitEthernet 0/18
 no ip address
 switchport
 spanning-tree pvst edge-port
 no shutdown
!
```

```
interface TenGigabitEthernet 0/9
 no ip address
 portmode hybrid
 switchport
 spanning-tree pvst edge-port
 no shutdown
!
interface TenGigabitEthernet 0/10
 no ip address
 portmode hybrid
 switchport
 spanning-tree pvst edge-port
 no shutdown
!
interface TenGigabitEthernet 0/11
 no ip address
 portmode hybrid
 switchport
 spanning-tree pvst edge-port
 no shutdown
!
interface TenGigabitEthernet 0/12
 no ip address
 portmode hybrid
 switchport
 spanning-tree pvst edge-port
 no shutdown
!
interface TenGigabitEthernet 0/13
 no ip address
 portmode hybrid
 switchport
 spanning-tree pvst edge-port
 no shutdown
!
interface TenGigabitEthernet 0/14
 no ip address
 portmode hybrid
 switchport
 spanning-tree pvst edge-port
 no shutdown
!
interface TenGigabitEthernet 0/15
 no ip address
 portmode hybrid
 switchport
 spanning-tree pvst edge-port
 no shutdown
!
interface TenGigabitEthernet 0/16
 no ip address
 portmode hybrid
 switchport
 spanning-tree pvst edge-port
 no shutdown
!
interface TenGigabitEthernet 0/17
 no ip address
 portmode hybrid
 switchport
 spanning-tree pvst edge-port
 no shutdown
!
interface TenGigabitEthernet 0/18
 no ip address
 portmode hybrid
 switchport
 spanning-tree pvst edge-port
 no shutdown
!
```

```
interface TenGigabitEthernet 0/19
 no ip address
 switchport
 spanning-tree pvst edge-port
 no shutdown
!
interface TenGigabitEthernet 0/20
 no ip address
 switchport
 spanning-tree pvst edge-port
 no shutdown
!
interface TenGigabitEthernet 0/21
 no ip address
 switchport
 spanning-tree pvst edge-port
 no shutdown
!
interface TenGigabitEthernet 0/22
 no ip address
 switchport
 spanning-tree pvst edge-port
 no shutdown
!
interface TenGigabitEthernet 0/23
 no ip address
 switchport
 spanning-tree pvst edge-port
 no shutdown
!
interface TenGigabitEthernet 0/24
 no ip address
 switchport
 spanning-tree pvst edge-port
 no shutdown
!
interface TenGigabitEthernet 0/25
 no ip address
 switchport
 spanning-tree pvst edge-port
 no shutdown
!
interface TenGigabitEthernet 0/26
 no ip address
 switchport
 spanning-tree pvst edge-port
 no shutdown
!
interface TenGigabitEthernet 0/27
 no ip address
 switchport
 spanning-tree pvst edge-port
 no shutdown
!
interface TenGigabitEthernet 0/28
 no ip address
 switchport
 spanning-tree pvst edge-port
 no shutdown
!
```

```
interface TenGigabitEthernet 0/19
 no ip address
 portmode hybrid
 switchport
 spanning-tree pvst edge-port
 no shutdown
!
interface TenGigabitEthernet 0/20
 no ip address
 portmode hybrid
 switchport
 spanning-tree pvst edge-port
 no shutdown
!
interface TenGigabitEthernet 0/21
 no ip address
 portmode hybrid
 switchport
 spanning-tree pvst edge-port
 no shutdown
!
interface TenGigabitEthernet 0/22
 no ip address
 portmode hybrid
 switchport
 spanning-tree pvst edge-port
 no shutdown
!
interface TenGigabitEthernet 0/23
 no ip address
 portmode hybrid
 switchport
 spanning-tree pvst edge-port
 no shutdown
!
interface TenGigabitEthernet 0/24
 no ip address
 portmode hybrid
 switchport
 spanning-tree pvst edge-port
 no shutdown
!
interface TenGigabitEthernet 0/25
 no ip address
 portmode hybrid
 switchport
 spanning-tree pvst edge-port
 no shutdown
!
interface TenGigabitEthernet 0/26
 no ip address
 portmode hybrid
 switchport
 spanning-tree pvst edge-port
 no shutdown
!
interface TenGigabitEthernet 0/27
 no ip address
 portmode hybrid
 switchport
 spanning-tree pvst edge-port
 no shutdown
!
interface TenGigabitEthernet 0/28
 no ip address
 portmode hybrid
 switchport
 spanning-tree pvst edge-port
 no shutdown
!
```

```
interface TenGigabitEthernet 0/29
 no ip address
 switchport
 spanning-tree pvst edge-port
 no shutdown
!
interface TenGigabitEthernet 0/30
 no ip address
 switchport
 spanning-tree pvst edge-port
 no shutdown
!
interface TenGigabitEthernet 0/31
 no ip address
 switchport
 spanning-tree pvst edge-port
 no shutdown
!
interface TenGigabitEthernet 0/32
 no ip address
 switchport
 spanning-tree pvst edge-port
 no shutdown
!
interface TenGigabitEthernet 0/33
 no ip address
!
 port-channel-protocol LACP
  port-channel 20 mode active
 no shutdown
!
interface TenGigabitEthernet 0/34
 no ip address
!
 port-channel-protocol LACP
  port-channel 20 mode active
 no shutdown
!
interface TenGigabitEthernet 0/35
 no ip address
!
 port-channel-protocol LACP
  port-channel 20 mode active
 no shutdown
!
interface TenGigabitEthernet 0/36
 no ip address
!
 port-channel-protocol LACP
  port-channel 20 mode active
 no shutdown
!
interface fortyGigE 0/37
 no ip address
 shutdown
!
interface fortyGigE 0/41
 no ip address
 shutdown
!
interface fortyGigE 0/45
 no ip address
 shutdown
!
```

```
interface TenGigabitEthernet 0/29
 no ip address
 portmode hybrid
 switchport
 spanning-tree pvst edge-port
 no shutdown
!
interface TenGigabitEthernet 0/30
 no ip address
 portmode hybrid
 switchport
 spanning-tree pvst edge-port
 no shutdown
!
interface TenGigabitEthernet 0/31
 no ip address
 portmode hybrid
 switchport
 spanning-tree pvst edge-port
 no shutdown
!
interface TenGigabitEthernet 0/32
 no ip address
 portmode hybrid
 switchport
 spanning-tree pvst edge-port
 no shutdown
!
interface TenGigabitEthernet 0/33
 no ip address
!
 port-channel-protocol LACP
  port-channel 30 mode active
 no shutdown
!
interface TenGigabitEthernet 0/34
 no ip address
!
 port-channel-protocol LACP
  port-channel 30 mode active
 no shutdown
!
interface TenGigabitEthernet 0/35
 no ip address
!
 port-channel-protocol LACP
  port-channel 40 mode active
 no shutdown
!
interface TenGigabitEthernet 0/36
 no ip address
!
 port-channel-protocol LACP
  port-channel 40 mode active
 no shutdown
!
interface fortyGigE 0/37
 no ip address
 shutdown
!
interface fortyGigE 0/41
 no ip address
 shutdown
!
interface fortyGigE 0/45
 no ip address
 shutdown
!
```

```
interface fortyGigE 0/49
 no ip address
 shutdown
!
interface fortyGigE 0/53
 no ip address
 shutdown
!
interface ManagementEthernet 0/0
 ip address 172.25.188.43/16
 no shutdown
!
interface Port-channel 20
 no ip address
 switchport
 no shutdown
!
interface Vlan 1
 no ip address
 shutdown
!
interface Vlan 11
 no ip address
 tagged Port-channel 20
 untagged TenGigabitEthernet 0/1-8
 no shutdown
!
interface Vlan 12
 no ip address
 tagged Port-channel 20
 untagged TenGigabitEthernet 0/9-24
 no shutdown
!
interface Vlan 13
 no ip address
 tagged Port-channel 20
 untagged TenGigabitEthernet 0/25-32
 no shutdown
!
management route 0.0.0.0/0 172.25.188.254
no ip telnet server enable
!
ip ssh server enable
!
protocol lldp
!
line console 0
line vty 0
line vty 1
line vty 2
line vty 3
line vty 4
line vty 5
line vty 6
line vty 7
line vty 8
line vty 9
!
End
```

```
interface fortyGigE 0/49
 no ip address
 shutdown
!
interface fortyGigE 0/53
 no ip address
 shutdown
!
interface ManagementEthernet 0/0
 ip address 172.25.188.44/16
 no shutdown
!
interface Port-channel 30
 no ip address
 switchport
 no shutdown
!
interface Port-channel 40
 no ip address
 switchport
 no shutdown
!
interface Vlan 1
 ip address dhcp
 no shutdown
!
interface Vlan 11
 no ip address
 tagged TenGigabitEthernet 0/1-32
 tagged Port-channel 30,40
 no shutdown
!
interface Vlan 12
 no ip address
 tagged TenGigabitEthernet 0/1-32
 tagged Port-channel 30,40
 no shutdown
!
interface Vlan 13
 no ip address
 tagged Port-channel 30,40
 untagged TenGigabitEthernet 0/1-32
 no shutdown
no ip telnet server enable
!
ip ssh server enable
!
protocol lldp
!
line console 0
line vty 0
line vty 1
line vty 2
line vty 3
line vty 4
line vty 5
line vty 6
line vty 7
line vty 8
line vty 9
!
end
```

N5K1 Running Configuration

```
version 5.2(1)N1(1)
hostname N5K1

no feature telnet
cfs eth distribute
feature lacp
feature vpc
feature lldp

banner motd #Nexus 5000 Switch
#

no ip domain-lookup
logging event link-status default
class-map type qos class-fcoe
class-map type queuing class-fcoe
  match qos-group 1
class-map type queuing class-all-flood
  match qos-group 2
class-map type queuing class-ip-multicast
  match qos-group 2
class-map type network-qos class-fcoe
  match qos-group 1
class-map type network-qos class-all-flood
  match qos-group 2
class-map type network-qos class-ip-multicast
  match qos-group 2

vrf context management
vlan 1,11-13
spanning-tree pathcost method long
spanning-tree vlan 1,12 priority 24576
spanning-tree vlan 11,13 priority 28672
vpc domain 10
  peer-keepalive destination 172.25.188.61 source 172.25.188.60
  ip arp synchronize
port-profile default max-ports 512


interface port-channel10
  switchport mode trunk
  spanning-tree port type network
  vpc peer-link

interface port-channel20
  switchport mode trunk
  vpc 20

interface port-channel30
  switchport mode trunk

interface Ethernet1/1
  switchport mode trunk
  channel-group 20 mode active

interface Ethernet1/2
  switchport mode trunk
  channel-group 20 mode active

interface Ethernet1/3

interface Ethernet1/4

interface Ethernet1/5

interface Ethernet1/6

interface Ethernet1/7
```

```
interface Ethernet1/8

interface Ethernet1/9

interface Ethernet1/10

interface Ethernet1/11

interface Ethernet1/12

interface Ethernet1/13

interface Ethernet1/14

interface Ethernet1/15

interface Ethernet1/16

interface Ethernet1/17
  switchport mode trunk
  channel-group 10 mode active

interface Ethernet1/18
  switchport mode trunk
  channel-group 10 mode active

interface Ethernet1/19
  switchport mode trunk
  channel-group 10 mode active

interface Ethernet1/20
  switchport mode trunk
  channel-group 10 mode active

interface Ethernet1/21

interface Ethernet1/22

interface Ethernet1/23

interface Ethernet1/24

interface Ethernet1/25
  switchport mode trunk
  channel-group 30 mode active

interface Ethernet1/26
  switchport mode trunk
  channel-group 30 mode active

interface Ethernet1/27

interface Ethernet1/28

interface Ethernet1/29

interface Ethernet1/30

interface Ethernet1/31

interface Ethernet1/32

interface mgmt0
  ip address 172.25.188.60/16
line console
line vty
boot kickstart bootflash:/n5k-ks.521N10.116.bin
boot system bootflash:/n5k.521N10.116.bin
```

N5K2 Running Configuration

```
version 5.2(1)N1(1)
hostname N5K2

no feature telnet
cfs eth distribute
feature lacp
feature vpc
feature lldp

banner motd #Nexus 5000 Switch
#

no ip domain-lookup
logging event link-status default
class-map type qos class-fcoe
class-map type queuing class-fcoe
  match qos-group 1
class-map type queuing class-all-flood
  match qos-group 2
class-map type queuing class-ip-multicast
  match qos-group 2
class-map type network-qos class-fcoe
  match qos-group 1
class-map type network-qos class-all-flood
  match qos-group 2
class-map type network-qos class-ip-multicast
  match qos-group 2

vrf context management
vlan 1,11-13
spanning-tree pathcost method long
spanning-tree vlan 1,12 priority 28672
spanning-tree vlan 11,13 priority 24576
vpc domain 10
  peer-keepalive destination 172.25.188.60 source 172.25.188.61
  ip arp synchronize
port-profile default max-ports 512


interface port-channel10
  switchport mode trunk
  spanning-tree port type network
  vpc peer-link

interface port-channel20
  switchport mode trunk
  vpc 20

interface port-channel40
  switchport mode trunk

interface Ethernet1/1
  switchport mode trunk
  channel-group 20 mode active

interface Ethernet1/2
  switchport mode trunk
  channel-group 20 mode active

interface Ethernet1/3

interface Ethernet1/4

interface Ethernet1/5

interface Ethernet1/6

interface Ethernet1/7

interface Ethernet1/8
```

```
interface Ethernet1/9

interface Ethernet1/10

interface Ethernet1/11

interface Ethernet1/12

interface Ethernet1/13

interface Ethernet1/14

interface Ethernet1/15

interface Ethernet1/16

interface Ethernet1/17
  switchport mode trunk
  channel-group 10 mode active

interface Ethernet1/18
  switchport mode trunk
  channel-group 10 mode active

interface Ethernet1/19
  switchport mode trunk
  channel-group 10 mode active

interface Ethernet1/20
  switchport mode trunk
  channel-group 10 mode active

interface Ethernet1/21

interface Ethernet1/22

interface Ethernet1/23

interface Ethernet1/24

interface Ethernet1/25
  switchport mode trunk
  channel-group 40 mode active

interface Ethernet1/26
  switchport mode trunk
  channel-group 40 mode active

interface Ethernet1/27

interface Ethernet1/28

interface Ethernet1/29

interface Ethernet1/30

interface Ethernet1/31

interface Ethernet1/32

interface mgmt0
  ip address 172.25.188.61/16
line console
line vty
boot kickstart bootflash:/n5k-ks.521N10.116.bin
boot system bootflash:/n5k.521N10.116.bin
```

Appendix B: Basic Terminology

Bridging

Bridging—commonly called switching—is frame-by-frame layer-2 forwarding of Ethernet traffic with forwarding decisions generally based on each frame's source and destination MAC address. A simple bridge has three available actions to perform on received traffic—filter, forward, or flood. An Ethernet bridge is very simple and relies on bridging protocols like spanning-tree to not send data frames in a loop, causing a storm.

Channel Group

See LAG.

CLI

Command Line Interface (CLI) is the text-based console interface that is used for entering management and configuration commands into devices like the Dell Force10 MXL switch. The MXL's CLI can be accessed via telnet, SSH, an externally-accessible serial connection, and also from the CMC's CLI.

CMC

Chassis Management Controller (CMC) is the embedded management interface of the Dell PowerEdge™ M1000e blade server chassis. Among other functions, the CMC provides network and console access to installed IO modules including the Dell Force10 MXL switch.

Filter, Flood, Forward

Data frames received by Ethernet switches may be filtered (meaning discarded or dropped) according to defined behavior, automatic protocols, or administrative configuration; flooded (meaning sent out all other links) if the data frame's destination MAC address is unknown; or forwarded (meaning sent to one other link) if the destination MAC address is already learned. A destination for a MAC address may be manually configured on a link's interface; but, most MAC address destinations are learned dynamically based on source MAC addresses of already received data frames—when a bridge receives a data frame, it remembers the source MAC address of the frame on the received link for five minutes and will then selectively forward data frames destined for that MAC address over that link as a learned destination.

IOM

IO module (IOM) refers to the modules at the rear of the Dell PowerEdge M1000e chassis that will receive and transmit I/O (Ethernet, FC, Infiniband, etc.) from the blade servers located at the front of the chassis. The Dell Force10 MXL switch is as an IOM for the M1000e blade server chassis.

LACP

Link Aggregation Control Protocol (LACP) is the protocol used to ensure that the multiple links in a LAG do not form loops due to misconfiguration or device misbehavior. It is recommended practice to always use LACP on configured LAGs.

LAG

Link Aggregation Group (LAG) is a configured bundle of Ethernet links that are treated as the same logical Ethernet link. There are multiple terms that apply to LAGs including channel group, port channel, trunk, and even some server Ethernet interface teaming involves a collection of links that would be considered a LAG. However while channel group and port channel always apply to LAG use, trunk and teaming do not.

LAN

A Local Area Network (LAN) is a term for a network that services a limited area from the size of a single table to a large as an office building. They are generally interconnected using Ethernet switches. The term is sometimes applied to a network involving a single broadcast domain and sometimes applied to a network involving multiple broadcast domains separated into VLANs (and often rejoined via routing).

Link

Link is a term in networking that refers to a connection made between two nodes in a network. In Ethernet networking it is generally used to refer to a direct connection between two ports.

MAC Address

Media Access Control Address (MAC Address) is a layer-2 node identifier. In Ethernet bridging, MAC addresses are used for source and destination identification. They can also be used as system identifiers since vendor-assigned (or burned-in) MAC addresses are globally unique. An Ethernet MAC address is 48 bits long and generally written in groupings of two hexadecimal digits often separated by colons or hyphens like this: 00:1e:c9:00:cb:01. But, are sometimes written in groupings of four hexadecimal digits separated by periods like this: 001e.c900.cb01

MLAG

Multi-chassis Link Aggregation Group (MLAG) is a LAG implementation that connects across multiple switches on one side of the logical aggregated link. This requires shared management of the logical link between the switches sharing a side and situational forwarding of frames that differs from the forwarding that would be present across the multiple LAGs that an MLAG replaces. There is no standard method for implementing an MLAG but multiple vendor specific methods. Nexus vPC and Force10 VLT are examples of MLAG implementations.

MSTP

Multiple Spanning-Tree Protocol (MSTP) is a standards-based modified version of the rapid spanning tree protocol that carries multiple spanning tree instances within its rapid spanning-tree protocol packet. Spanning-tree instance 0 is assigned to the common rapid spanning-tree instance and additional instances above 0 may be configured. For each spanning-tree instance, a root switch is elected and unique active and backup links can be chosen providing potentially unique traffic paths on the network per instance. Each VLAN can then be assigned to a spanning-tree instance allowing active traffic on separate VLANs to potentially utilize separate paths across the network. In order for interconnected switches to participate together in more than the common spanning-tree instance, they must have the same MSTP configuration (a checksum of this configuration is included the spanning-tree protocol packet and must be the same between switches in order for them to participate). MSTP is originally defined in the IEEE 802.1s standard and is included in 802.1q IEEE Virtual LANs standard.

Out-of-Band
An out-of-band interface provides management connectivity to a device without participating in or relying on a device's in-band (normal-use) data interfaces. On a switch this means that an out-of-band interface does not send or receive traffic from the switched links—neither bridged nor routed. Common out-of-band interface types are Ethernet and serial console—often both are presented with RJ-45 (8P8C) connectors although on IO modules in the Dell PowerEdge M1000e chassis the serial connector is sometimes a physical USB type-A port requiring a special cable.

Port Channel
See LAG.

PVST
Per-VLAN Spanning-Tree (PVST) is a vendor specific implantation of the spanning tree protocol that maintains separate instances for each VLAN, passing that instance's protocol frames within the VLAN it manages. This method simplifies deployment of multiple VLANs and is popularly used where available.

RSTP
Rapid Spanning-Tree Protocol (RSTP) is a standards-based modified version of the basic spanning tree protocol that allows for much faster convergence times of spanning tree instances and provides for special administratively-assigned port states that improve behavior in certain circumstances. RSTP is originally defined in the IEEE 802.1w standard and is included in 802.1d IEEE Ethernet bridging standard.

Spanning Tree
Spanning Tree refers to a family of layer-2 management protocols used by Ethernet bridges to establish a loop-free forwarding topology. At layer-2, Ethernet is a very simple technology that without intervening protocols or configuration can easily forward traffic in endless loops—see Bridging for an explanation. The spanning-tree protocols provide a standard way for an interconnected set of Ethernet bridging devices to only use links that will not cause traffic flows to loop (Ethernet switches—being very fast bridges—can forward traffic very quickly and looping traffic flows will rapidly saturate all available bandwidth with unwanted, repeated traffic). In a spanning tree, a single bridge is elected the root bridge—either by lowest assigned priority or by having the lowest of the presented MAC addresses. Once a root bridge is elected, every other bridge keeps its one link with the lowest path cost to the root bridge active. Links with redundant paths are then blocked by switches that don't have the lowest path cost for that link. As a result, with spanning tree each non-root bridge effectively has only one active link between it and the root bridge and the topology of these unblocked links draws a tree to the root bridge. Spanning tree has a long and involved history on Ethernet and there are many different implementations with different timings and protocol feature sets—see RSTP, MSTP, and PVST.

Storm
Ethernet Storms—often called broadcast storms—are a descriptive term for excessive (and generally unwanted) data frames being continuously sent to all available links. Because Ethernet bridges dynamically learn destinations based on recently received traffic, do not limit the number of times that data frames can be forwarded across bridges, and do not recognize when data frames have been forwarded back to the same bridge; without the oversight provided by bridging management protocols, looped flows of data frames can quickly cause a storm. A storm can quickly saturate all available bandwidth—catastrophically affecting a network's performance and reliability. To keep potential storms from happening, Ethernet bridging generally relies on spanning-tree protocols to establish loop-free topologies and MAC learning to keep traffic selectively forwarded.

STP

Spanning-Tree Protocol (STP)—see Spanning Tree.

Switching

Switching, in an Ethernet context, is a specific technology but the term has largely been generalized to mean layer-2 Ethernet bridging. Ethernet switching is in fact the employment of ASIC technologies to implement Ethernet traffic forwarding and filtering in specialized circuits and memory structures designed for high throughput, low latency, and low cost performance. Generally an Ethernet switch will at least function as a layer-2 bridge but more advanced models have multilayer capabilities including layer-3 routing and multilayer filtering, logic, and frame modification.

Switchport

Switchport is a configuration term used to denote an Ethernet switch's link interface that is configured for layer-2 bridging (participating in one or more VLANs).

ToR

Top of Rack (ToR) is a term for a switch that is actually positioned at the top of a server rack in a data center.

Trunk

Trunk is an ambiguous term in Ethernet networking that can apply to a LAG—a group of multiple links acting as one or to a switchport interface of an Ethernet switch configured in trunk mode to pass multiple VLANs across the one link.

VLAN

Virtual Local Area Network (VLAN) is a single layer-2 network (also called a broadcast domain as broadcast traffic does not escape a VLAN on its own). Multiple VLANs can be passed between switches using switchport trunk interfaces. When passed across trunk links, frames in a VLAN are prefixed with the number of the VLAN that they belong to—a twelve bit value that allows just over 4000 differently numbered VLANs.