


**Part No. 060378-10, Rev. C  
December 2013**

# **OmniSwitch AOS Release 7 Data Center Switching Guide**

Alcatel·Lucent 

[www.alcatel-lucent.com](http://www.alcatel-lucent.com)

**This user guide documents AOS Release 7.3.2 for the OmniSwitch 10K and  
AOS Release 7.3.3 for the OmniSwitch 6900.  
The functionality described in this guide is subject to change without notice.**

Copyright © 2013 by Alcatel-Lucent. All rights reserved. This document may not be reproduced in whole or in part without the express written permission of Alcatel-Lucent.

Alcatel-Lucent® and the Alcatel-Lucent logo are registered trademarks of Alcatel-Lucent. Xylan®, OmniSwitch®, OmniStack®, and Alcatel-Lucent OmniVista® are registered trademarks of Alcatel-Lucent.

OmniAccess™, Omni Switch/Router™, PolicyView™, RouterView™, SwitchManager™, VoiceView™, WebView™, X-Cell™, X-Vision™, and the Xylan logo are trademarks of Alcatel-Lucent.



26801 West Agoura Road  
Calabasas, CA 91301  
(818) 880-3500 FAX (818) 880-3505  
support@ind.alcatel.com

**Service & Support Contact Information**

North America: 800-995-2696  
Latin America : 877-919-9526  
EMEA : +800 00200100 (Toll Free) or +1(650)385-2193  
Asia Pacific: +65 6240 8484  
Web: service.esd.alcatel-lucent.com  
Email: esd.support@alcatel-lucent.com

# Contents

	<b>About This Guide</b> .....	vii
	Supported Platforms .....	vii
	Who Should Read this Manual? .....	vii
	When Should I Read this Manual? .....	vii
	What is in this Manual? .....	viii
	What is Not in this Manual? .....	viii
	How is the Information Organized? .....	viii
	Documentation Roadmap .....	ix
	Related Documentation .....	xi
	Technical Support .....	xii
<b>Chapter 1</b>	<b>Understanding Data Center Switching</b> .....	1-1
	In This Chapter .....	1-1
	Data Center Switching Components .....	1-2
	OmniSwitch Pod .....	1-2
	Data Center Mesh .....	1-2
	Virtual Network Profile .....	1-3
	Network Virtualization Technology .....	1-3
	Data Center Mesh Configuration Example .....	1-4
<b>Chapter 2</b>	<b>Configuring Data Center Bridging</b> .....	2-1
	In This Chapter .....	2-1
	DCB Specifications .....	2-2
	DCB Defaults .....	2-3
	DCB Port Defaults .....	2-4
	Data Center Bridging Overview .....	2-5
	Priority-Based Flow Control Overview .....	2-5
	Enhanced Transmission Selection Overview .....	2-7
	Data Center Bridging Exchange Overview .....	2-10
	Interaction with Other Features .....	2-13
	Converged Enhanced Ethernet DCBX .....	2-13
	QoS .....	2-13
	Shortest Path Bridging .....	2-14
	SNMP .....	2-14
	Virtual Chassis .....	2-14
	Using DCB Profiles .....	2-16

Configuring DCB Profiles .....	2-22
Creating a Custom Profile .....	2-23
Changing the Profile Assignment .....	2-23
Configuring Support for Legacy Pause Frames .....	2-24
Configuring DCBx Port Parameters .....	2-25
Multicast and Unicast Traffic Distribution .....	2-26
Non-Default Profile .....	2-26
Multicast Source PFC on OmniSwitch 6900 .....	2-28
Verifying the DCB Configuration .....	2-29
<b>Chapter 3</b>	
<b>Configuring Shortest Path Bridging</b> .....	3-1
In This Chapter .....	3-2
Shortest Path Bridging Specifications .....	3-3
SPB-M Parameter Defaults .....	3-4
SPB-M Interface Defaults .....	3-4
SPB-M Service Defaults .....	3-5
Shortest Path Bridging Overview .....	3-6
SPB-M Shortest Path Trees .....	3-8
SPB Services .....	3-12
Sample SPB-M Network Topology .....	3-13
IP over SPB-M .....	3-15
Interaction With Other Features .....	3-18
Backbone VLANs (VLAN Manager) .....	3-18
Dynamic Service Access Points (UNP) .....	3-18
Hardware .....	3-18
Link Aggregation .....	3-19
OAM .....	3-19
Quality of Service (QoS) .....	3-19
VRF .....	3-20
Quick Steps for Configuring SPB-M .....	3-21
Quick Steps for Configuring the SPB-M Backbone .....	3-21
Quick Steps for Configuring SPB Services .....	3-22
Sample Command Configuration .....	3-22
Configuring SPB-M .....	3-24
Configure the SPB-M Backbone (ISIS-SPB) .....	3-24
Configure SPB-M Services .....	3-24
SPB Configuration Guidelines .....	3-25
Configuring BVLANS .....	3-26
Configuring SPB Interfaces .....	3-28
Configuring Global ISIS-SPB Parameters .....	3-29
Creating a SPB Service .....	3-35
Configuring Service Access Points (SAPs) .....	3-38
Configuring IP over SPB .....	3-44

Verifying the SPB Backbone and Services .....	3-53
Verifying the ISIS-SPB Backbone Configuration .....	3-53
Verifying the SPB Service Configuration .....	3-53
<b>Chapter 4      Configuring FIP Snooping .....</b>	<b>4-1</b>
In This Chapter .....	4-1
FIP Snooping Specifications .....	4-2
FIP Snooping Defaults .....	4-3
Terms and Definitions .....	4-4
FIP Snooping Overview .....	4-5
FCoE Initialization Protocol .....	4-6
OmniSwitch FIP Snooping ACLs .....	4-8
Interaction with Other Features .....	4-12
Data Center Bridging (DCB) .....	4-12
802.1AB Link Layer Discovery Protocol .....	4-13
Loop Avoidance .....	4-13
Universal Network Profile (UNP) .....	4-13
QoS .....	4-13
Multiple VLAN Registration Protocol (MVRP) .....	4-14
Configuring FIP Snooping .....	4-15
Configuration Guidelines .....	4-16
Configuring Lossless DCB for FCoE .....	4-17
Configuring Global FCoE Parameters .....	4-19
Configuring FCoE VLANs .....	4-21
Configuring FCoE Ports .....	4-22
Configuration Example .....	4-23
Verifying the FIP Snooping Configuration .....	4-25
<b>Chapter 5      Configuring an FCoE/FC Gateway .....</b>	<b>5-1</b>
In This Chapter .....	5-2
FCoE/FC gateway Specifications .....	5-4
FCoE/FC gateway Overview .....	5-5
OmniSwitch FCoE/FC Gateway Fabric .....	5-5
Using the N_Port Proxy Mode .....	5-7
Using the F_Port Proxy Mode .....	5-12
Using the E_Port Proxy Mode .....	5-15
FIP Packet Processing .....	5-18
Interaction with Other Features .....	5-19
FIP Snooping .....	5-19
Data Center Bridging (DCB) .....	5-19
Hardware .....	5-20
FCoE/FC Gateway Configuration Guidelines .....	5-21
Configuring FC Ports .....	5-22
Assigning a DCB Profile for FC Ports .....	5-22

	Configuring the FC Port Type and Mode .....	5-22
	Configuring an N_Port Proxy Operation .....	5-23
	Configuring N_Port Proxy Load Balancing .....	5-24
	Configuring an F_Port Proxy Operation .....	5-25
	Configuring an E_Port Proxy Operation .....	5-26
	FCoE/FC Gateway Configuration Examples .....	5-28
	Example 1: Multiple Gateway Operations on the Same Fabric .....	5-28
	Example 2: OmniSwitch Gateway with NPIV Host .....	5-31
	Verifying the FCoE/FC Gateway Configuration .....	5-33
<b>Chapter 6</b>	<b>Virtual Machine Classification</b> .....	6-1
	In This Chapter .....	6-1
	UNP (vNP) Specifications .....	6-2
	EVB Specifications .....	6-2
	Server Virtualization Overview .....	6-3
	Classifying Virtual Machines .....	6-4
	UNP Overview .....	6-4
	Using EVB .....	6-8
	Tracking Virtual Machines .....	6-13
<b>Appendix A</b>	<b>Software License and Copyright Statements</b> .....	A-1
	Alcatel-Lucent License Agreement .....	A-1
	ALCATEL-LUCENT SOFTWARE LICENSE AGREEMENT .....	A-1
	Third Party Licenses and Notices .....	A-4
	<b>Index</b> .....	Index-1

# About This Guide

This *OmniSwitch AOS Release 7 Data Center Switching Guide* describes how to set up and monitor supported protocols and software features that comprise the Alcatel-Lucent data center switching architecture. Some of the features described in this guide are purchased as an add-on package to the base switch software.

## Supported Platforms

The information in this guide applies only to OmniSwitch 10K and OmniSwitch 6900 switches.

## Who Should Read this Manual?

The audience for this user guide are network administrators and IT support personnel who need to configure, maintain, and monitor switches and routers in a live network. However, anyone wishing to gain knowledge on how software features supporting data center switching are implemented in the OmniSwitch 10K and OmniSwitch 6900 switches will benefit from the material in this configuration guide.

## When Should I Read this Manual?

Read this guide as soon as you are ready to integrate your OmniSwitch into your network and you are ready to set up the data center switching protocols and features. You should already be familiar with the basics of managing a single OmniSwitch as described in the *OmniSwitch AOS Release 7 Switch Management Guide*.

The topics and procedures in this manual assume an understanding of the OmniSwitch directory structure and basic switch administration commands and procedures.

## What is in this Manual?

This switching guide includes a “Data Center Switching Introduction” chapter that provides a description of the Alcatel-Lucent OmniSwitch data center switching architecture and software features. In addition to this introduction, this guide also includes information about configuring the following features;

- Data Center Bridging (DCB) protocols.
- Shortest Path Bridging MAC (SPBM), including SPBM support of Provider Backbone Bridging (PBB) encapsulation and services.
- Universal Network Profile (UNP), specifically for configuring Virtual Network Profiles (vNP) to manage virtual machines within and between data centers.
- Edge Virtual Bridging (EVB) for managing virtual machines created and managed on servers also running the EVB protocol.

## What is Not in this Manual?

The configuration procedures in this manual use Command Line Interface (CLI) commands in all examples. CLI commands are text-based commands used to manage the switch through serial (console port) connections or via Telnet sessions. Procedures for other switch management methods, such as web-based (WebView or OmniVista) or SNMP, are outside the scope of this guide.

For information on WebView and SNMP switch management methods consult the *OmniSwitch AOS Release 7 Switch Management Guide*. Information on using WebView and OmniVista can be found in the context-sensitive on-line help available with those network management applications.

This guide provides overview material on software features, how-to procedures, and application examples that will enable you to begin configuring your OmniSwitch. It is not intended as a comprehensive reference to all CLI commands available in the OmniSwitch. For such a reference to all OmniSwitch AOS Release 7 CLI commands, consult the *OmniSwitch CLI Reference Guide*.

## How is the Information Organized?

Each chapter in this guide includes sections that will satisfy the information requirements of casual readers, rushed readers, serious detail-oriented readers, advanced users, and beginning users.

**Quick Information.** Most chapters include a *specifications table* that lists RFCs and IEEE specifications supported by the software feature. In addition, this table includes other pertinent information such as minimum and maximum values and sub-feature support. Some chapters include a *defaults table* that lists the default values for important parameters along with the CLI command used to configure the parameter. Many chapters include *Quick Steps* sections, which are procedures covering the basic steps required to get a software feature up and running.

**In-Depth Information.** All chapters include *overview sections* on software features as well as on selected topics of that software feature. *Topical sections* may often lead into *procedure sections* that describe how to configure the feature just described. Many chapters include *tutorials* or *application examples* that help convey how CLI commands can be used together to set up a particular feature.



# Documentation Roadmap

The OmniSwitch user documentation suite was designed to supply you with information at several critical junctures of the configuration process. The following section outlines a roadmap of the manuals that will help you at each stage of the configuration process. Under each stage, we point you to the manual or manuals that will be most helpful to you.

## Stage 1: Using the Switch for the First Time

**Pertinent Documentation:** *OmniSwitch Getting Started Guide*  
*Release Notes*

A hard-copy *OmniSwitch 10K Getting Started Guide* is included with your switch; this guide provides all the information you need to get your switch up and running the first time. It provides information on unpacking the switch, rack mounting the switch, installing NI modules, unlocking access control, setting the switch's IP address, and setting up a password. It also includes succinct overview information on fundamental aspects of the switch, such as hardware LEDs, the software directory structure, CLI conventions, and web-based management.

At this time you should also familiarize yourself with the Release Notes that accompanied your switch. This document includes important information on feature limitations that are not included in other user guides.

## Stage 2: Gaining Familiarity with Basic Switch Functions

**Pertinent Documentation:** *OmniSwitch Hardware Users Guide*  
*OmniSwitch AOS Release 7 Switch Management Guide*

Once you have your switch up and running, you will want to begin investigating basic aspects of its hardware and software. Information about switch hardware is provided in the *OmniSwitch 10K Hardware Guide*. This guide provides specifications, illustrations, and descriptions of all hardware components, such as chassis, power supplies, Chassis Management Modules (CMMs), Network Interface (NI) modules, and cooling fans. It also includes steps for common procedures, such as removing and installing switch components.

This guide is the primary users guide for the basic software features on a single switch. This guide contains information on the switch directory structure, basic file and directory utilities, switch access security, SNMP, and web-based management. It is recommended that you read this guide before connecting your switch to the network.

## Stage 3: Integrating the Switch Into a Network

**Pertinent Documentation:** *OmniSwitch AOS Release 7 Network Configuration Guide*  
*OmniSwitch AOS Release 7 Advanced Routing Configuration Guide*  
*OmniSwitch AOS Release 7 Data Center Switching Guide*

When you are ready to connect your switch to the network, you will need to learn how the OmniSwitch implements fundamental software features, such as 802.1Q, VLANs, Spanning Tree, and network routing protocols. The *OmniSwitch AOS Release 7 Network Configuration Guide* contains overview information, procedures, and examples on how standard networking technologies are configured on the OmniSwitch.

The *OmniSwitch AOS Release 7 Advanced Routing Configuration Guide* includes configuration information for networks using advanced routing technologies (OSPF and BGP) and multicast routing protocols (DVMRP and PIM-SM).

The *OmniSwitch AOS Release 7 Data Center Switching Guide* includes configuration information for data center networks using virtualization technologies (SPBM and UNP) and Data Center Bridging protocols (PFC, ETC, and DCBX).

### **Anytime**

The *OmniSwitch CLI Reference Guide* contains comprehensive information on all CLI commands supported by the switch. This guide includes syntax, default, usage, example, related CLI command, and CLI-to-MIB variable mapping information for all CLI commands supported by the switch. This guide can be consulted anytime during the configuration process to find detailed and specific information on each CLI command.

## Related Documentation

The following are the titles and descriptions of all the related OmniSwitch user manuals:

- *OmniSwitch 10K and OmniSwitch 6900 Getting Started Guides*

Describes the hardware and software procedures for getting an OmniSwitch up and running. Also provides information on fundamental aspects of OmniSwitch software architecture.
- *OmniSwitch 10K and OmniSwitch 6900 Getting Started Guides*

Complete technical specifications and procedures for all OmniSwitch chassis, power supplies, fans, and Network Interface (NI) modules.
- *OmniSwitch CLI Reference Guide*

Complete reference to all CLI commands supported on the OmniSwitch. Includes syntax definitions, default values, examples, usage guidelines and CLI-to-MIB variable mappings.
- *OmniSwitch AOS Release 7 Switch Management Guide*

Includes procedures for readying an individual switch for integration into a network. Topics include the software directory architecture, image rollback protections, authenticated switch access, managing switch files, system configuration, using SNMP, and using web management software (WebView).
- *OmniSwitch AOS Release 7 Network Configuration Guide*

Includes network configuration procedures and descriptive information on all the major software features and protocols included in the base software package. Chapters cover Layer 2 information (Ethernet and VLAN configuration), Layer 3 information (routing protocols, such as RIP and IPX), security options (authenticated VLANs), Quality of Service (QoS), link aggregation, and server load balancing.
- *OmniSwitch AOS Release 7 Advanced Routing Configuration Guide*

Includes network configuration procedures and descriptive information on all the software features and protocols included in the advanced routing software package. Chapters cover multicast routing (DVMRP and PIM-SM), Open Shortest Path First (OSPF), and Border Gateway Protocol (BGP).
- *OmniSwitch AOS Release 7 Data Center Switching Guide*

Includes an introduction to the OmniSwitch data center switching architecture as well as network configuration procedures and descriptive information on all the software features and protocols that support this architecture. Chapters cover Shortest Path Bridging MAC (SPBM), Data Center Bridging (DCB) protocols, Virtual Network Profile (vNP), and the Edge Virtual Bridging (EVB) protocol.
- *OmniSwitch Transceivers Guide*

Includes SFP and XFP transceiver specifications and product compatibility information.
- *Technical Tips, Field Notices*

Includes information published by Alcatel's Customer Support group.
- *Release Notes*

Includes critical Open Problem Reports, feature exceptions, and other important information on the features supported in the current release and any limitations to their support.

# Technical Support

An Alcatel-Lucent service agreement brings your company the assurance of 7x24 no-excuses technical support. You'll also receive regular software updates to maintain and maximize your Alcatel-Lucent product's features and functionality and on-site hardware replacement through our global network of highly qualified service delivery partners.

With 24-hour access to Alcatel-Lucent's Service and Support web page, you'll be able to view and update any case (open or closed) that you have reported to Alcatel-Lucent's technical support, open a new case or access helpful release notes, technical bulletins, and manuals.

Access additional information on Alcatel-Lucent's Service Programs:

Web: [service.esd.alcatel-lucent.com](http://service.esd.alcatel-lucent.com)

Phone: 1-800-995-2696

Email: [esd.support@alcatel-lucent.com](mailto:esd.support@alcatel-lucent.com)

# 1 Understanding Data Center Switching

Alcatel-Lucent helps enterprises address the challenges and ongoing transformation of data center networks while delivering a high-quality user experience for new real-time applications, greater agility in deploying new applications, seamless integration of public cloud services, and reduced data center costs. To deliver on these capabilities, Alcatel-Lucent provides a unique blueprint for data center switching that brings together three core innovations:

- **OmniSwitch Pod.** A unique architecture concept for top-of-rack switches that can provide server-to-server connectivity without the need for a core switch to carry traffic. The pod is a highly dense architecture that ensures low latency and high performance between servers connected to the same pod.
- **Data Center Mesh.** The mesh consists of pods connected to each other and to core switches to bring together thousands of server-facing ports with low aggregate end-to-end latency. Implementing the data center mesh architecture allows enterprises to create virtual data centers supporting defined work-groups or departments.
- **Virtual Network Profile (vNP).** A type of Universal Network Profile (UNP) that the administrator configures specifically for virtual machine classification. A virtual machine is bound to a vNP to ensure consistent application of network access controls and QoS policies when a virtual machine is initially detected or moves to a different network location.

## In This Chapter

This chapter contains an overview of the OmniSwitch components and features of the Alcatel-Lucent Data Center Switching solution. It provides a general example for configuring the related data center software applications through the Command Line Interface (CLI). CLI commands are used in the configuration examples; for more details about the syntax of commands, see the *OmniSwitch CLI Reference Guide*.

The following topics are included in this chapter:

- [“Data Center Switching Components” on page 1-2.](#)
- [“Data Center Mesh Configuration Example” on page 1-4.](#)

# Data Center Switching Components

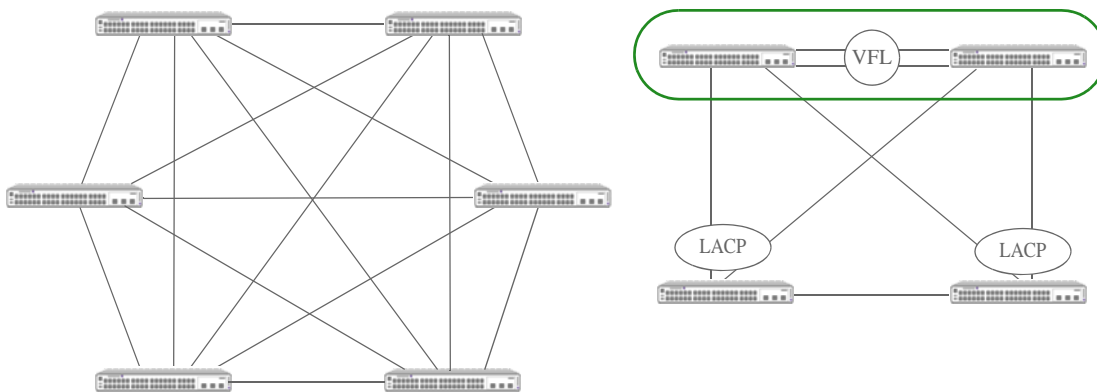
Key components of the OmniSwitch data center switching framework include an OmniSwitch pod, the data center mesh, and virtual network profiles (vNP). The OmniSwitch also supports the use of other network virtualization technology features, such as Shortest Path Bridging (SPB), Data Center Bridging (DCB), and Virtual Chassis.

## OmniSwitch Pod

The Alcatel-Lucent OmniSwitch pod is an architecture concept for top-of-rack switches that provides server-to-server (east-west and north-south) connectivity without the need for a core switch to carry traffic. The pod is a highly dense architecture that ensures low latency and high performance between servers connected to the same pod.

The architecture of the OmniSwitch pod provides the scalability necessary to handle changing data center demands. A pod may initially only consist of one or two such switches, but as the demand to handle more and more traffic grows, additional switches are easily added into the pod configuration.

Figure 1 shows some examples of OmniSwitch pod architectures.

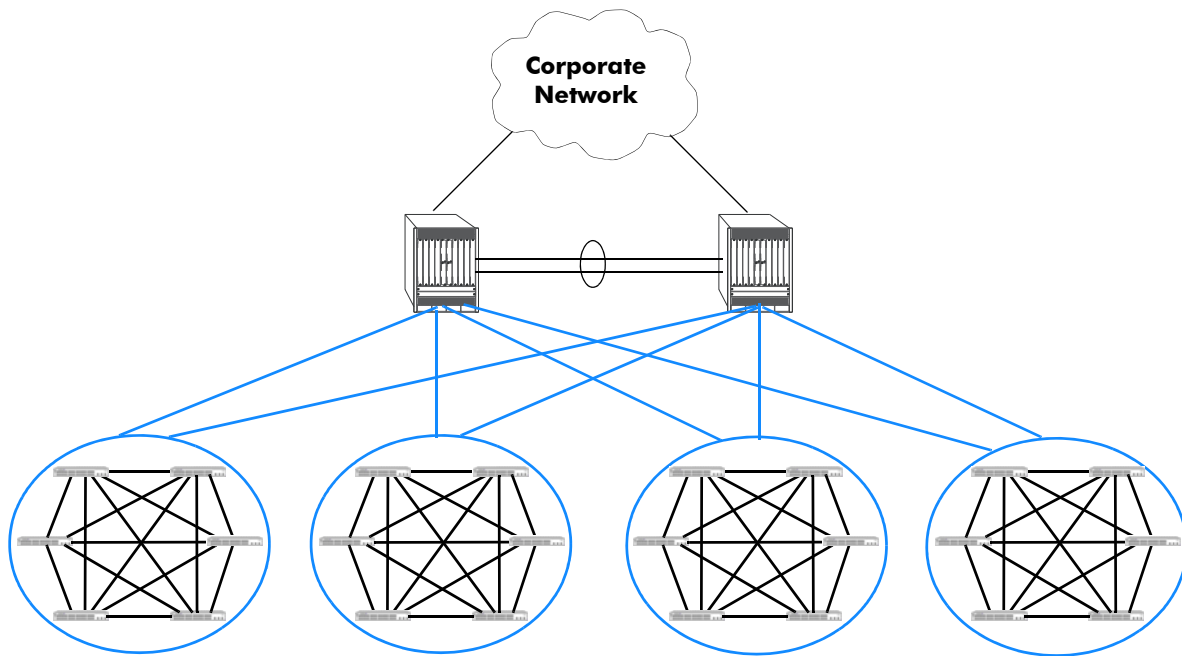


**Figure 1: OmniSwitch Pod Examples**

Each pod is a switching fabric where every switch could be connected to every other switch. When pods are then connected together with other pods and core switches, a data center mesh is formed.

## Data Center Mesh

The Alcatel-Lucent data center mesh consists of pods connected to each other and to core switches to bring together thousands of server-facing ports with low aggregate end-to-end latency and high resiliency. Implementing the data center mesh architecture allows enterprises to create virtual data centers supporting virtualized workgroups or departments. Figure 2 shows an example of a data center mesh that consists of four OmniSwitch 6900 pods and two OmniSwitch 10K core switches:



**Figure 2: Alcatel-Lucent Data Center Mesh**

## Virtual Network Profile

The Virtual Network Profile (vNP) feature facilitates the discovery and movement of virtual machines (VMs). A vNP is a type of Universal Network Profile (UNP) that is configured specifically for machine classification, especially VMs. A virtual machine is bound to a vNP to ensure consistent application of network access controls and QoS policies when a virtual machine is initially detected or moved to a different network location.

A vNP classifies VMs in the same manner as any other device connected to a UNP port. Once a VM is assigned to a vNP, the VM traffic is bound to the VLAN or service defined in the profile. In addition, any optional QoS policies associated with the profile are also applied to the VM traffic.

## Network Virtualization Technology

OmniSwitch data center switching leverages the use of the following AOS Release 7 software features to facilitate a network virtualization solution:

- **Shortest Path Bridging MAC (SPBM)** to virtualize the data center mesh.

SPBM extends Layer 2 across the data center mesh while maintaining a loop-free network. All connections between all switches in the topology remain active (no blocking of redundant links).

SPBM uses the Provider Backbone Bridge (PBB) network model to encapsulate (using IEEE 802.1ah headers) and tunnel customer traffic through the network backbone. The shortest path trees upon which the PBB network infrastructure operates are determined using a version of the Intermediate System-to-Intermediate System (IS-IS) link state protocol that supports TLV extensions for SPB (ISIS-SPB).

For more information about SPBM, see [Chapter 3, “Configuring Shortest Path Bridging.”](#)

- **Data Center Bridging (DCB)** protocols to convert Ethernet into a lossless transport to support a reliable storage area network fabric within the data center mesh. DCB protocols are implemented using embedded profiles in the same manner that QoS QSet profiles are applied.

The DCB profiles are based on the 802.1Q-REV/D1-5 standard to define how the switch classifies different traffic types and priority mappings and then groups those types into traffic classes. Profiles also specify the Traffic Class Flow (TCF), which is LL (lossless; PFC initiated upstream) or nL (lossy; PFC not initiated upstream).

For more information about DCB, see [Chapter 2, “Configuring Data Center Bridging.”](#)

- **Virtual Chassis (VC)** is a group of chassis managed through a single management IP address.

VC provides both node level and link level redundancy for devices connecting to the aggregation layer via standard 802.3ad link aggregation mechanisms. See [Chapter 9, “Configuring Virtual Chassis.”](#) for more information.

## Data Center Mesh Configuration Example

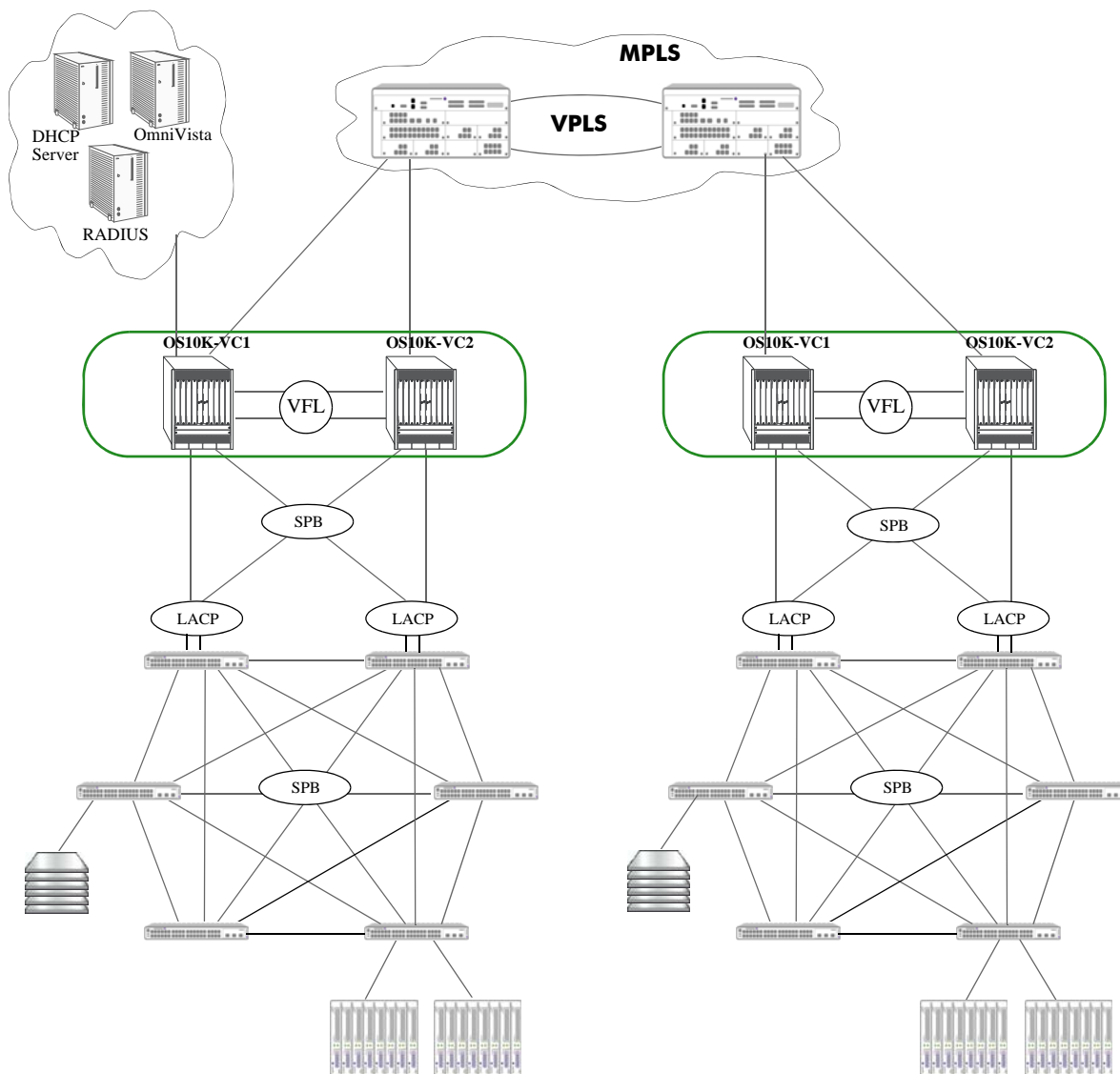
Traditional data center networks consist of a 3-tier approach: access, aggregation, and core. The high-speed, low-latency capability of the OmniSwitch 6900 provides the ability to combine the access and aggregation tiers.

The application example in this section consists of two data centers that incorporate the use of the OmniSwitch data center switching mesh architecture. Each pod shown in Figure 5 interconnects six OS6900 switches delivering 240 server-facing ports while maintaining low latency between servers in the same pod.

In this example, each OmniSwitch 6900 pod is connected to two OmniSwitch 10K core switches to form the data center mesh. In turn, the OmniSwitch 10K core switches connect to Alcatel-Lucent Service Routers (7750) platforms to provide interconnectivity between the two data centers through an MPLS cloud over Layer 2 virtual LAN services.

The following diagram illustrates the example data center switching network that implements the OmniSwitch pod/mesh architecture combined with supported virtualization technologies.





**Figure 5: Data Center Mesh Example**

In this data center mesh configuration example:

- Two separate data center configurations are depicted, each using a single OS6900 pod and two OS10K switches for the core.
- Each pod consists of six switches. The pod is formed by connecting each switch to every one of the other six switches. In other words, each switch is configured with five links that connect the switch to the pod.
- The core tier for each data center is made up of two OS10K switches. Virtual Chassis (VC) is configured for each pair of OS10Ks to create a chassis group that is managed through a single management IP address. VC provides both node level and link level redundancy for the 10K chassis group connecting to the aggregation (pod) layer via standard 802.3ad link aggregation mechanisms.

The following configuration snapshot shows a sample VC configuration for the core switches:

```
OS10K-VC1-> show configuration vcm-snapshot chassis-id 1
! Virtual Chassis Manager:
virtual-chassis chassis-id 1 configured-chassis-id 1
virtual-chassis chassis-id 1 vf-link 0 create
virtual-chassis chassis-id 1 vf-link 0 member-port 1/2/1
```

- Shortest Path Bridging MAC (SPBM) is used in each data center mesh (pod plus core) to define sets of loop-free shortest path trees (SPTs) through the network. Each switch serves as the SPT root for all traffic entering the switch, thus allowing the switch to provide the shortest path to every other switch.

The bridging methodology that allows each switch to serve as its own root is enforced through the use of SPBM backbone VLANs (BVLANS). The BVLAN is a transport VLAN for SPBM services and SPT calculations.

The following configuration snapshot shows a sample SPBM bridging configuration. Each switch that will participate in the SPBM domain (pod and core) must have the same BVLAN configuration. SPBM merges the core and pod together into the same virtual mesh domain.

```
OS6900-1-> show configuration snapshot spb-isis
! SPB-ISIS:
spb isis bvlan 4001 ect-id 1
spb isis bvlan 4002 ect-id 2
spb isis control-bvlan 4001
spb isis interface port 1/1/2-5
spb isis interface port 1/1/7
spb isis interface port 2/1/2-5
spb isis interface linkagg 2
spb isis admin-state enable
```

- SPBM Ethernet services are bound to the SPT bridging configuration and are used to encapsulate and tunnel data through the data center mesh. SPBM services are configured on any OmniSwitch in the mesh that will originate or terminate a service. This usually is on switches that are connected to host devices or that connect to other networks where traffic will enter or leave the mesh.

The OmniSwitch Service Manager feature is used to build the SPBM service architecture layer within the SPBM mesh domain. Provisioning a service requires the configuration of three logical entities: a service instance identifier (I-SID), service access port, and a service access point (SAP).

The following configuration snapshot shows a sample SPB service configuration:

```
OS6900-1-> show configuration snapshot svcmgr
! SVCMgr:
service access port 1/11
service access port 1/12
service access port 1/13
service spb 1001 isid 1001 bvlan 4001 admin-state enable
service spb 1002 isid 1002 bvlan 4001 admin-state enable
service spb 1001 sap port 1/11:1001 admin-state enable
service spb 1001 sap port 1/12:1001 admin-state enable
service spb 1001 sap port 1/13:1001 admin-state enable
service spb 1002 sap port 1/11:1002 admin-state enable
service spb 1002 sap port 1/12:1002 admin-state enable
service spb 1002 sap port 1/13:1002 admin-state enable
```

- When a physical switch port comes up, a QSet instance (a set of eight queues) is automatically associated with the port for unicast traffic. In addition, a default Data Center Bridging profile (DCP 8) is automatically assigned to the QSI.

The DCP 8 profile applies strict priority scheduling with lossy traffic class management to each of the

eight egress port queues. However, ports connected to devices running critical applications that require lossless Ethernet, such as the storage and server hosts shown in Figure 5, can be assigned to one of the other 10 pre-defined profiles or a user-configured profile that will provide the necessary class of service for that device.

The following configuration snapshot shows a sample DCB configuration. In this sample, a custom profile (DCP 11) is created with custom settings for the traffic classes. DCP 11 is then assigned to ports 1/1 and 1/9-10, replacing the default DCP 8 assignment. In addition, ports 1/12 and 1/34-35 are assigned to DCP 7 and 9. All other ports on the switch are using default DCP 8.

```
OS6900-1-> show configuration snapshot vfc
! Virtual Flow Control:
qos qsp dcb 11 import qsp dcb "dcp-9"
qos qsp dcb "dcp-11" tc 1 pfc flow-type nLL
qos qsp dcb "dcp-11" tc 0 pfc flow-type nLL
qos qsp dcb "dcp-11" tc 2 pfc flow-type nLL
qos qsp dcb "dcp-11" tc 4 pfc flow-type nLL
qos qsp dcb "dcp-11" tc 5 pfc flow-type nLL
qos qsp dcb "dcp-11" tc 6 pfc flow-type nLL
qos qsp dcb "dcp-11" tc 7 pfc flow-type nLL
qos qsi port 1/1 qsp dcb "dcp-11"
qos qsi port 1/9-10 qsp dcb "dcp-11"
qos qsi port 1/12 qsp dcb "dcp-7"
qos qsi port 1/34-35 qsp dcb "dcp-9"
```

- Virtual Network Profile (vNPs) are configured on each switch that is connected to a server to facilitate VM discovery and mobility within the local data center or mobility between the local and remote data center. In addition, Universal Network Profile (UNP) functionality is enabled on each of the server connections to activate vNP classification of VMs.

The vNPs are used to assign VMs to SPBM services that span the data center mesh. When a VM is discovered or migrates to a new location, the assigned vNP applies the necessary access controls and any QoS policies specifically defined for the VM.

The following configuration snapshot shows a sample vNP configuration.

```
OS6900-1-> show configuration snapshot da-unp
! DA-UNP:
unp customer-domain 1 description "AT&T"
unp customer-domain 2 description "Sprint"
unp port 1/15 port-type spb-access
unp port 1/15 unp-customer-domain 1
unp port 1/16 port-type spb-access
unp port 1/16 unp-customer-domain 1
unp port 1/17 port-type spb-access
unp port 1/17 unp-customer-domain 2
unp port 1/18 port-type spb-access
unp port 1/18 unp-customer-domain 2
unp spb-profile unp_for_spb1 tag-value 115:150 isid 6000 bvlan 4001
unp classification vlan-tag 125 spb-profile unp_for_spb1
unp classification vlan-tag 225 spb-profile unp_for_spb1
```



# 2 Configuring Data Center Bridging

Data Center Bridging (DCB) is a set of standards that extend Ethernet capabilities to support the convergence of storage and data in today's virtualized networks. The Alcatel-Lucent implementation of DCB for the OmniSwitch supports the following DCB standards:

- **Priority-Based Flow Control (PFC)**—based on the IEEE 802.1Qbb standard, PFC pauses traffic based on congestion priority instead of blocking the entire link when congestion occurs. Allows lossless and lossy traffic with different priorities on the same physical port.
- **Enhanced Transmission Selection (ETS)**—based on the IEEE 802.1Qaz standard, ETS groups related traffic into priority groups (traffic classes) to which bandwidth guarantees and scheduling are applied.
- **Data Center Bridging Exchange (DCBx)**—exchanges and negotiates PFC and ETS information between two directly connected peer devices.

This implementation of the DCB enhanced Ethernet protocols uses embedded profiles to apply the PFC, ETS, and DCBx configuration to traffic flows. This approach is similar to how QSet profiles (QSPs) are used to apply the QoS configuration for bandwidth management and egress port queue scheduling.

## In This Chapter

This chapter describes DCB in general and how DCB profiles and port configurations are applied to the switch. It provides information about configuring DCB through the Command Line Interface (CLI). CLI commands are used in the configuration examples; for more details about the syntax of commands, see the *OmniSwitch CLI Reference Guide*.

The following topics and procedures are included in this chapter:

- [“DCB Specifications” on page 2-2.](#)
- [“DCB Defaults” on page 2-3.](#)
- [“Data Center Bridging Overview” on page 2-5.](#)
- [“Interaction with Other Features” on page 2-13](#)
- [“Using DCB Profiles” on page 2-16](#)
- [“Configuring DCB Profiles” on page 2-22](#)
- [“Configuring DCBx Port Parameters” on page 2-25.](#)
- [“Multicast and Unicast Traffic Distribution” on page 2-26.](#)

## DCB Specifications

The DCB functionality described in this chapter is supported on the OmniSwitch 10K and OmniSwitch 6900 switches, unless otherwise stated in the following DCB Specifications table or specifically noted within any other section of this chapter. Note that any maximum limits provided in the DCB Specifications table are subject to available system resources.

Platforms Supported	OmniSwitch 6900 and the following OmniSwitch 10K modules: <ul style="list-style-type: none"> <li>• OS10K-QNI-U8 (8 x 40G)</li> <li>• OS10K-QNI-U4 (4 x 40G)</li> <li>• OS10K-XNI-U32E (32 x 10G)</li> <li>• OS10K-XNI-U16E (16 x 10G)</li> <li>• OS10K-XNI-U16L (8 x 10G, 8 x 1G)</li> </ul>
OmniSwitch Software License	Data Center
IEEE Standards Supported	802.1Qbb— <i>Priority-based Flow Control</i> 802.1Qaz D2.5— <i>Enhanced Transmission Selection</i> 802.1Qaz D2.5— <i>Data Center Bridging Exchange</i> <i>Converged Enhanced Ethernet DCBX v.1.01</i> 802.1Q-REV/D1.5— <i>Media Access Control (MAC) Bridges and Virtual Bridged Local Area Networks</i>
Maximum number of DCB profiles	128 profiles: <ul style="list-style-type: none"> <li>• Profiles 1–11 are predefined, with profile 8 serving as the default profile for all ports.</li> <li>• Profiles 12–128 are reserved for user-defined (custom) profiles.</li> </ul>
Maximum number of lossless queues (priorities)	110 total per switch (OmniSwitch 6900) 8 per-port (OmniSwitch 10K)
DCB TLVs supported	ETS Configuration ETS Recommendation PFC Configuration Application Priority

## DCB Defaults

Traffic class management and related QoS functions are implemented using a Queue Set (QSet) framework. Each port and link aggregate is associated with a set of eight egress queues, referred to as a Queue Set Instance (QSI). Each QSI is associated with DCB profile 8 (DCP 8) by default.

A DCP defines both the Data Center Bridging Exchange (DCBX) control parameters and the traffic class management parameters that are applied to the eight queues associated with the QSI. See [“Configuring DCB Profiles” on page 2-22](#) for more information.

The following are the default DCBX and traffic class management settings applied with DCP 8:

DCB Profile 8 (Default Profile)							
Profile Control :: DCBX Configuration							
PFC :: Enable TLV : [*Yes/No] Willing : [*Yes/No] Capability : 8				Recommendation :: Enable TLV : [*Yes/No] Recommended Profile : [DCB_Def_Profile_1/NULL]			
ETS :: Enable TLV : [*Yes/No] Willing : [*Yes/No] Max TC : 3				Application :: Enable TLV : [Yes/*No] Defense Mode :: [*Enable/Disable] Default MTU :: [9216]			
Profile Data :: Max Traffic Classes (TC) : 8							
Traffic Class (TC)	Traffic Type (TT)	TC Description (TD)	TC Priority Map (TP)	TC Bandwidth (%)		TC Flow Control (TF)	TC Scheduler (TS)
0	BE	Best Effort	0	0	100	Lossy	Strict Priority
1	BK	Background	1	0	100	Lossy	Strict Priority
2	EE	Excellent Effort	2	0	100	Lossy	Strict Priority
3	CA	Critical Applications	3	0	100	Lossy	Strict Priority
4	VI	Video (<100ms latency)	4	0	100	Lossy	Strict Priority
5	VO	Voice (<10ms latency)	5	0	100	Lossy	Strict Priority
6	IC	Internetwork Control	6	0	100	Lossy	Strict Priority
7	NC	Network Control	7	0	100	Lossy	Strict Priority

**Note.** QSet profiles and DCB profiles are mutually exclusive on the same port. If the OmniSwitch Data Center software license is installed, then DCB profiles are used by default. If this license is not installed, then QSet profiles are used by default. See [“Using DCB Profiles” on page 2-16](#) for more information.

## DCB Port Defaults

Parameter Description	Command	Default
The DCBx administrative status for the port.	<b>qos qsi dcb dcbx admin-state</b>	Enabled
The transmission status for the PFC configuration TLV	<b>qos qsi dcb dcbx pfc tlv</b>	Enabled
The PFC defense mode status	<b>qos qsi dcb dcbx pfc defense</b>	Enabled
The PFC “willing” to negotiate status	<b>qos qsi dcb dcbx pfc willing</b>	Yes
The transmission status for the ETS configuration TLV	<b>qos qsi dcb dcbx ets config-tlv</b>	Enabled
The transmission status for the ETS recommended TLV	<b>qos qsi dcb dcbx ets recommended TLV</b>	Enabled
The ETS “willing” to negotiate status	<b>qos qsi dcb dcbx ets willing</b>	Yes



# Data Center Bridging Overview

Convergence of data and storage into Ethernet has become more prevalent due to the combined data and storage bandwidth requirements of virtualized networks. Data Center Bridging (DCB) technology combined with the Alcatel-Lucent Data Center Mesh architecture provides a unified framework upon which different types of network traffic are converged into a single transport layer.

For example, each of the following traffic types can coexist in a converged network without imposing serious restrictions on the performance of the other traffic types:

- **Storage Area Network (SAN) traffic**—A SAN provides access to a shared storage solutions. Requires transmission of traffic with no packet loss (Lossless).
- **Local Area Network (LAN) traffic**—A LAN provides end-user access to server-based network applications, such as email, database, and web servers. Requires best effort transmission of traffic for important applications. Can tolerate some traffic loss (Lossy).
- **Inter-Process Communication (IPC) traffic**—IPC refers to the mechanisms or activities that provide communications and data sharing between applications. Requires the highest priority for low latency transmission.

The Alcatel-Lucent OmniSwitch supports the following Data Center Bridging (DCB) protocols to provide a single, shared infrastructure for reliable delivery of data and storage over Ethernet:

- **Priority-Based Flow Control (PFC)**—Based on the IEEE 802.1Qbb standard, PFC pauses traffic based on congestion priority instead of blocking the entire link when congestion occurs. Allows lossless and lossy traffic with different priorities on the same physical port.
- **Enhanced Transmission Selection (ETS)**—Based on the IEEE 802.1Qaz standard, ETS provides a common framework for dynamic bandwidth management. ETS groups related traffic into priority groups (Traffic Classes) to which bandwidth guarantees and scheduling are applied.
- **Data Center Bridging Exchange (DCBX)**—Based on the IEEE 802.1Qaz standard, DCBX uses the Link Layer Discovery Protocol (LLDP) to exchange and negotiate PFC and ETS information between two directly connected peer switches. Enabled by default, DCBX is responsible for auto-negotiation and auto-configuration of link parameters for DCB functions.

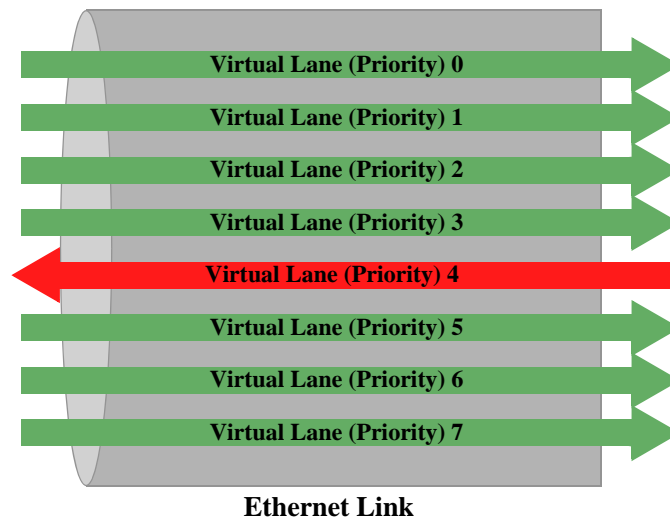
## Priority-Based Flow Control Overview

Priority-based Flow Control (PFC) is the evolution of the IEEE 802.3 Ethernet PAUSE mechanism that uses pause frames to signal the other end of a link to stop sending traffic when buffer congestion is detected on any one of the ingress port queues. The problem with the 802.3x approach is that when congestion occurs for only one type of traffic on a specific queue, the transmission of all traffic on the port is stopped.

PFC offers a more granular approach that is needed to support the convergence of various traffic types over the same link. Instead of pausing all traffic when ingress port congestion occurs, PFC only pauses traffic tagged with a specific Class of Service (CoS) priority value while allowing traffic assigned to other priority values to continue (or get dropped when congestion occurs) on the same link.

In essence, using PFC divides an Ethernet link into eight virtual lanes. Each virtual lane represents one of the CoS priority values (0–7). When buffer congestion occurs for traffic marked with one of these priority values, PFC initiates pause frames back to the source of that specific traffic. The source then holds back the traffic until the congestion clears (bandwidth allocated for that priority becomes available). As shown in Figure 1, PFC detected congestion for traffic marked with priority 4, so only that traffic is

paused and packet loss is avoided. All other traffic marked with different priorities continues to forward on the link until congestion occurs for those priorities as well.



**Figure 1: PFC Example**

To determine how traffic is treated when assigned to one of these eight priority values, the pre-defined OmniSwitch DCB profiles use the following general IEEE recommendations for mapping traffic types to the CoS priority values:

- Priority 1, BK—Background {Bulk transfers}
- Priority 0, BE—Best Effort {unprioritized user applications}
- Priority 2, EE—Excellent Effort {CEO's best effort}
- Priority 3, CA—Critical Applications {guaranteed minimum bandwidth}
- Priority 4, VI—Video {low latency < 100ms}
- Priority 5, VO—Voice {low latency < 10ms}
- Priority 6, IC—Internetwork Control {guaranteed delivery}
- Priority 7, NC—Network Control {guaranteed delivery}

## Using PFC for OmniSwitch Flow Control

The main purpose of PFC is to provide a lossless transport for traffic sensitive to packet loss, such as storage area network (SAN), LAN backup, or management traffic, while at the same time allowing packet-drop congestion management for other traffic types on the same link. The OmniSwitch implementation of DCB accomplishes this through the use of embedded profiles that define the PFC configuration for each priority queue associated with each switch port.

There are 10 pre-defined profiles available, and the ability to create additional custom profiles is also supported. Each profile specifies which priority queue requires lossless (no packet drop) or lossy (packet drop) congestion management. When a switch port comes up, a single, pre-defined DCB profile is associated with that port by default. Currently DCB profile 8 is the default profile applied to each port. This profile defines lossy flow control for all eight queues (virtual lanes).

The following general steps are required to configure a lossless transport lane for a specific type of traffic:

- 1** Select a pre-defined DCB profile or create a custom profile that defines lossless flow control for a specific CoS priority value.
- 2** Determine the end-to-end lossless data path through the OmniSwitch network and apply the profile identified in Step 1 to each port in that path.
- 3** Make sure that the traffic requiring lossless transmission is marked with the same CoS priority value to which the DCB profile will apply lossless flow control. The switch will only pause frames sent with the same priority value as the lossless priority value specified in the DCB profile.
- 4** Make sure that all ingress ports are configured as QoS trusted ports with the default classification set to 802.1p.

For more information about DCB profiles, see [“Using DCB Profiles” on page 2-16](#).

For more information about CoS 802.1p priority bit classification and marking, see [“How Traffic is Classified and Marked” on page 25-6 in Chapter 25, “Configuring QoS,” of the \*OmniSwitch AOS Release 7 Network Configuration Guide\*](#).

## Enhanced Transmission Selection Overview

Priority-based Flow Control (PFC) uses the Class of Service (COS) 802.1p priority values (0–7) to determine which traffic to pause when congestion occurs. Enhanced Transmission Selection (ETS) groups these priorities (traffic types) into Traffic Classes and then allocates a percentage of minimum bandwidth to each class.

Traffic Classes are groups of priorities that have similar traffic handling requirements (for example, LAN, SAN, and IPC). The amount of bandwidth assigned to each Traffic Class is a minimum guarantee of available bandwidth. Available bandwidth is defined as the amount of link bandwidth remaining after Traffic Classes not subject to ETS (for example, Strict Priority is used instead) or vendor-specific requirements are serviced. In addition,

- Available bandwidth for a Traffic Class is configurable in 1% increments.
- The maximum deviation of available bandwidth is 10% when all of the following is true:
  - > Only ETS traffic is flowing.
  - > No PFC pause frames have been received.
  - > All ETS Traffic Classes are offered enough load to consume their share of allocated bandwidth.
- Any unused portion of the minimum bandwidth is made available for use by other classes, but only until the original Traffic Class needs the bandwidth again. At such time, PFC (if enabled) will pause the required traffic to minimize packet loss until the bandwidth is regained.

## Using ETS for OmniSwitch Bandwidth Allocation

The OmniSwitch implementation of DCB uses embedded profiles to define the ETS and PFC configuration for each priority queue associated with each switch port. ETS combines the CoS priorities used by PFC into Traffic Classes. Each DCB profile defines the ETS grouping of such priority values into Traffic Classes to which the specified bandwidth allocation, flow control (lossless or lossy), and scheduling are applied.

ETS requires a minimum of three Traffic Classes per port but supports a maximum of eight Traffic Classes per port. The use of DCB profiles to apply the ETS configuration facilitates the correct mapping of traffic types and priorities into Traffic Classes according to the IEEE 802.1Q-REV/D1-5 standard.

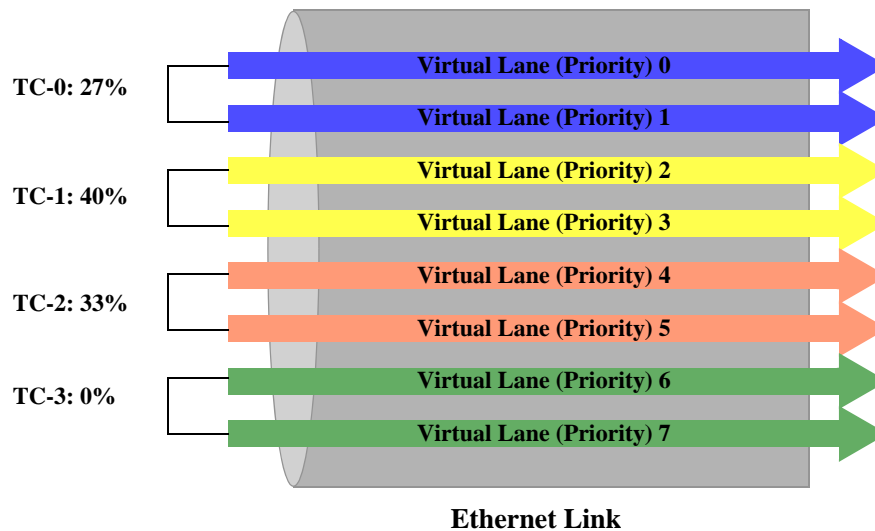
There are 10 pre-defined DCB profiles available, and the ability to create additional custom profiles is also supported. The following table shows the grouping of traffic types into a Traffic Class (TC) as applied by the pre-defined DCB profiles (see [“Using DCB Profiles” on page 2-16](#)).

**Table 1: DCB Profile Traffic Classes**

DCB Profile	Traffic Type Allocation	TC Priority Map
1	TC-0: Best Effort, Background, Excellent Effort, Critical Applications	0, 1, 2, 3
	TC-1: Voice, Video	5, 4
	TC-2: Network Control, Internetwork Control	7, 6
2	TC-0: Best Effort, Background	0, 1,
	TC-1: Critical Applications, Excellent Effort	3, 2
	TC-2: Voice, Video	5, 4
	TC-3: Network Control, Internetwork Control	7, 6
3	TC-0: Best Effort, Background	0, 1,
	TC-1: Critical Applications, Excellent Effort	3, 2
	TC-2: Voice, Video	5, 4
	TC-3: Internetwork Control	6
	TC-4: Network Control	7
4	TC-0: Background	1
	TC-1: Best Effort	0
	TC-2: Critical Applications, Excellent Effort	3, 2
	TC-3: Voice, Video	5, 4
	TC-4: Internetwork Control	6
5	TC-0: Background	1
	TC-1: Best Effort	0
	TC-2: Excellent Effort	2
	TC-3: Critical Applications	3
	TC-4: Voice, Video	5, 4
	TC-5: Internetwork Control	6
6	TC-6: Network Control	7
	TC-0: Background	1
	TC-1: Best Effort	0
	TC-2: Excellent Effort	2
	TC-3: Critical Applications	3
	TC-4: Video	4
	TC-5: Voice	5
TC-6: Internetwork Control	6	
	TC-7: Network Control	7

<b>DCB Profile</b>	<b>Traffic Type Allocation</b>	<b>TC Priority Map</b>
7	<b>TC-0:</b> Best Effort	0
	<b>TC-1:</b> Background	1
	<b>TC-2:</b> Excellent Effort	2
	<b>TC-3:</b> Critical Applications	3
	<b>TC-4:</b> Video	4
	<b>TC-5:</b> Voice	5
	<b>TC-6:</b> Internetwork Control	6
	<b>TC-7:</b> Network Control	7
8	<b>TC-0:</b> Best Effort	0
	<b>TC-1:</b> Background	1
	<b>TC-2:</b> Excellent Effort	2
	<b>TC-3:</b> Critical Applications	3
	<b>TC-4:</b> Video	4
	<b>TC-5:</b> Voice	5
	<b>TC-6:</b> Internetwork Control	6
	<b>TC-7:</b> Network Control	7
9	<b>TC-0:</b> Background	1
	<b>TC-1:</b> Best Effort	0
	<b>TC-2:</b> Excellent Effort	2
	<b>TC-3:</b> Critical Applications	3
	<b>TC-4:</b> Video	4
	<b>TC-5:</b> Voice	5
	<b>TC-6:</b> Internetwork Control	6
	<b>TC-7:</b> Network Control	7
10	<b>TC-0:</b> Background	1
	<b>TC-1:</b> Best Effort	0
	<b>TC-2:</b> Excellent Effort	2
	<b>TC-3:</b> Critical Applications	3
	<b>TC-4:</b> Video	4
	<b>TC-5:</b> Voice	5
	<b>TC-6:</b> Internetwork Control	6
	<b>TC-7:</b> Network Control	7

Figure 2 further illustrates ETS Traffic Class groupings by showing the Traffic Class minimum bandwidth guarantees as defined and applied by DCB profile 2:



**Figure 2: ETS Traffic Classes (DCB Profile 2)**

In the Figure 2 example,

- Traffic marked with priorities 0 and 1 is grouped into Traffic Class 0, which is assigned a minimum bandwidth guarantee of 27% of the available bandwidth.
- Traffic marked with priorities 2 and 3 is grouped into TC-1, which is assigned a minimum bandwidth guarantee of 40% of the available bandwidth.
- Traffic marked with priorities 4 and 5 is grouped into TC-2, which is assigned a minimum bandwidth guarantee of 33% of the available bandwidth.
- Traffic marked with priorities 6 and 7 is grouped into TC-3, but the TC-3 minimum bandwidth is set to zero. This is because DCB profile 2 applies Strict Priority scheduling to TC-3, not ETS. As a result, virtual lanes (priority queues) 6 and 7 are serviced first then ETS allocates the remaining bandwidth for TC-0, TC-1, and TC-2 priority queues.

For more information about DCB profiles, see [“Using DCB Profiles” on page 2-16](#).

For more information about 802.1p CoS priority bit classification and marking, see [“How Traffic is Classified and Marked” on page 25-6 in Chapter 25, “Configuring QoS,” of the \*OmniSwitch AOS Release 7 Network Configuration Guide\*](#).

## Data Center Bridging Exchange Overview

The Data Center Bridging Capabilities Exchange Protocol (DCBX) communicates the DCB capabilities of each OmniSwitch and exchanges PFC and ETS configuration details with directly connected peer devices. In addition, DCBx functionality helps to ensure a consistent configuration across the network.

- DCBx discovers the DCB capabilities of directly connected peers (for example, does the port on the other end of the link support PFC and is it active).

- Mis-configurations are detected when DCB configuration parameter values are exchanged between two peers and the parameters are not the same on both peers.
- When a configuration mismatch is detected and both peers are “willing”, an automatic negotiation takes place to reconcile a common operational configuration for both peers.

DCBX uses the IEEE 802.1AB Link Layer Discover Protocol (LLDP) to discover and exchange information between two link peers. There are specific LLDP Type-Length-Values (TLVs) that DCBX defines for exchanging PFC and ETS attribute values, such as:

- Is PFC in use on the peer switch.
- What are the priority values that PFC manages.
- Which traffic type priority values are mapped to a specific Traffic Class.
- What are the minimum bandwidth guarantees defined for each Traffic Class.

An OmniSwitch is considered a DCB-capable switch if the Alcatel-Lucent OmniSwitch Data Center software license is installed. When this license is installed, DCBX is automatically enabled on each switch port and DCB profiles apply the PFC and ETS configuration to the QSet instance (QSI) for each port.

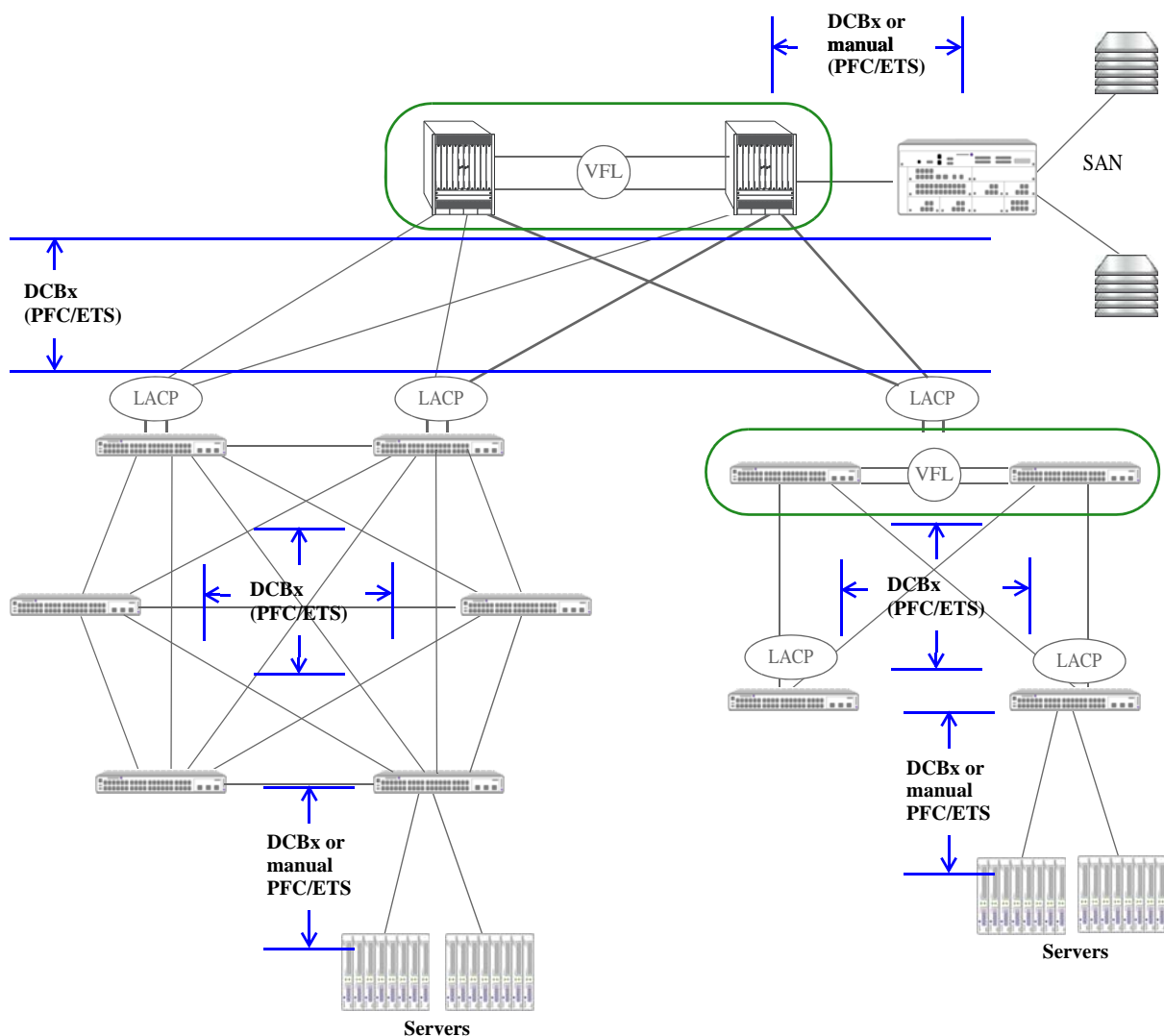
The following default DCBX configuration is applied to each DCBX-enabled port:

- PFC and ETS TLVs are enabled. The following TLVs are supported:
  - > PFC Configuration
  - > ETS Configuration
  - > ETS Recommendation
- The “willing” bit setting is enabled. This setting indicates that the switch is willing to negotiate changes to the operational DCBX configuration on the local port to match the DCBX configuration on the remote port.
- PFC defense mode is enabled. When PFC negotiations fail and the defense mode is enabled, no PFC functionality is applied to any priority but traffic flow is allowed to continue. However, if the defense mode is disabled when negotiations fail, the local PFC configuration is applied.

## DCBx in the Data Center Mesh

The diagram in Figure 3 shows where DCBx is enabled in a sample OmniSwitch Data Center Mesh topology. In this example,

- The OmniSwitch Data Center Software License is installed on each OmniSwitch to enable the DCB features (PFC, ETS, and DCBx) on each switch port.
- DCBx negotiates PFC and ETS parameters between the switches on every port. Manual configuration between an OmniSwitch and a non-OmniSwitch device (servers or switches from other vendors) may require manual configuration of DCB parameters on those devices.
- The servers could be sending SAN traffic at a priority negotiated for lossless and could use a different priority for iSCSI SAN traffic. Note that any application requiring lossless access to the network is supported as long as the traffic is stamped with the correct priority that is negotiated for lossless.



**Figure 3: DCBx Example**

For more information about DCBx, see [“Configuring DCBx Port Parameters”](#) on page 2-25.

For more information about DCB profiles, see [“Using DCB Profiles”](#) on page 2-16.



# Interaction with Other Features

This section contains important information about how other OmniSwitch features interact with Data Center Bridging (DCB) features. Refer to the specific chapter for each feature to get more detailed information about how to configure and use the feature.

## Converged Enhanced Ethernet DCBX

The OmniSwitch implementation of Data Center Bridging supports two versions of the DCB Exchange protocol (DCBX):

- IEEE 802.1Qaz DCBX
- Converged Enhanced Ethernet (CEE) DCBX version 1.0.1

By default, a DCB port will use the IEEE 802.1Qaz version of DCBX until the port detects the peer switch is using the CEE version. When this occurs, the switch will automatically stop 802.1Qaz DCBX on the port and start using CEE DCBX.

It is possible to configure the DCB port to use only a specific version of DCBX (no auto-detection) using the **qos qsi dcb dcbx version** command. For example, the following command changes the DCBX version for the specified port to CEE:

```
-> qos qsi port 1/10 dcb dcbx version cee
```

To change the DCBX version to IEEE 802.1Qaz, use the **qos qsi dcb dcbx version** command with the **ieee** parameter. For example:

```
-> qos qsi port 1/11 dcb dcbx version ieee
```

To change the port back to automatically detecting the peer version, use the **qos qsi dcb dcbx version** command with the **auto** parameter. For example:

```
-> qos qsi port 1/11 dcb dcbx version auto
```

---

**Note.** Using DCB profile 11 (DCP 11) is highly recommended when connecting an OmniSwitch port to equipment from other vendors. For more information see [“Using DCB Profiles” on page 2-16](#) and [“Configuring DCB Profiles” on page 2-22](#).

---

## QoS

- This implementation of DCB provides enhanced QoS congestion and bandwidth allocation to support multiple traffic types on the same Ethernet link. However, DCB profiles are used to apply the DCB configuration instead of QoS profiles. These two profile types are mutually exclusive on the same port; only one or the other profile type is applied to a port at any given time.
- Even though a port is only associated with a single DCB profile or a single QoS profile, it is possible to have a mixture of both profile types on different ports within the same switch. For example, when a DCB-licensed switch boots up, all ports are associated with a DCB profile by default, but the profile association for any port can be changed to a QoS profile without disrupting the DCB profile associations of other ports.
- The DCB protocols, Priority-based Flow Control (PFC) and Enhanced Transmission Selection (ETS), use the Class of Service (CoS) 802.1p markings found in the frame header to define traffic groups.

These markings are the result of QoS classification that occurs prior to and separately from the application of DCB functionality. DCB does not replace existing QoS classification and enqueueing mechanisms.

For more information about DCB and QoS profiles, see [“Using DCB Profiles” on page 2-16](#) and [“Multicast and Unicast Traffic Distribution” on page 2-26](#).

For more information about 802.1p CoS priority bit classification and marking, see [“How Traffic is Classified and Marked” on page 25-6 in Chapter 25, “Configuring QoS,” of the \*OmniSwitch AOS Release 7 Network Configuration Guide\*](#).

## Shortest Path Bridging

In order to support DCBX on an SPB access port (for example, between a server and an OmniSwitch), create a Layer 2 profile to allow the access port to participate as an LLDP (802.1ab) protocol peer to the server. If this is not done, then LLDP packets will be treated as traffic and DCBX will not be negotiated between the access port and server.

The following is an example of an SPB configuration that creates Layer 2 profile “DCBX” and associates that profile with access port 1/1/6:

```
-> show configuration snapshot svcmgr
! SVCMGR:
service l2profile DCBX 802.1ab peer
service access port 1/1/6 l2profile DCBX
```

## SNMP

If SNMP performs sets on the standard MIBs at the port level and the port is associated with:

- A custom DCB profile that is only associated with that one port, then the changes are made on that profile.
- A custom DCB profile that is associated with multiple ports, a new custom profile is created with the changes and the new profile is associated only with that port. The new profile is accessible through the OmniSwitch Command Line Interface (CLI).
- A default DCB profile, a new custom profile is created with the changes and the new profile is associated only with that port. The new profile is accessible through the OmniSwitch CLI.

For more information about DCB profiles, see [“Using DCB Profiles” on page 2-16](#).

## Virtual Chassis

- DCBx is not supported on VF-Links.
- Only two profiles can be applied on the VF-links: DCB 8 (applied by default, all priorities are lossy) and DCB 9 (all priorities are lossless, PFC is enabled).

## PFC Support Across the VFL

To support PFC across the VFL of Virtual Chassis, the links have to be configured in a certain manner. Only one VF Link is allowed from each port group based on the type of module being used to create the VFL. See Table 2 for the port group requirements:

**Table 2: OS10K Module Port Groups**

<b>Module</b>	<b>Port Grouping</b>
OS10K-XNI-U32E	Port Group 1 (Ports 1-16) Port Group 2 (Ports 17-32)
OS10K-XNI-U16E/U16L	Port Group 1 (Ports 1-16)
OS10K-QNI-U8E	Port Group 1 (Ports 1-4) Port Group 2 (Ports 5-8)
OS10K-QNI-U4E	Port Group 1 (Ports 1-4)
<b>Note:</b> The OS6900 platform and the OS10K-XNI-U32S module are not affected	

- If a chassis is operating in virtual chassis mode and the Data Center license is activated, then this port group requirement will be enforced. If there is no Data Center license installed this requirement is not needed.
- If the **vcsetup.cfg** file contains a configuration that does not adhere to this requirement, then on boot up only one VF-link from each group of ports listed above will be enabled. The remaining ports of the same port group will not be enabled. There is no guarantee as to which port will be selected to be enabled.
- During run time configuration if there is already a port from the port group that is a member of the VFL, then no additional ports from the port group can be added to the VFL and an error will be displayed if attempted.
- When configuring PFC over VFL on an OmniSwitch 6900, the VFL should not have more than eight ports comprising the VFL.

## Using DCB Profiles

The DCB configuration for PFC, ETS, and DCBX is applied to switch ports through DCB profiles the same way that QSet profiles apply the QoS queue management configuration to switch ports. However, only DCB profiles or QSet profiles are allowed at any given time.

---

**Note.** QSet profiles and DCB profiles are mutually exclusive. If the OmniSwitch Data Center software license is installed, then DCB profiles are used. If this license is not installed, then QSet profiles are used.

---

- If the OmniSwitch Data Center software license is not installed, the switch boots up as a “non-DCB” switch and QSet profiles are applied to switch ports.
- If the OmniSwitch Data Center software license is installed, the switch boots up as a DCB switch and profiles are applied as follows:
  - If there is no existing DCB configuration, the default DCB profile (DCP 8) is applied to all switch ports. This occurs even if the port was previously assigned to a non-default QSP (for example, QSP 2, 3, or 4).
  - If there is an existing DCB configuration, the profiles are applied based on that configuration.
- If a switch boots up in the DCB mode and no DCB configuration is present (only default DCP 8), the switch will start DCBX by default to make sure each port is auto-configurable via DCBX to match the peer configuration. This provides a “plug-and-play” installation process that allows a switch running the default DCB configuration to automatically adapt to the network.

The DCB profiles are based on the 802.1Q-REV/D1-5 standard to define how the switch classifies different traffic types and priority mappings and then groups those types into traffic classes. Profiles also specify the Traffic Class Flow (TCF), which is LL (lossless; PFC initiated upstream) or nL (lossy; PFC not initiated upstream).

There are 11 pre-defined DCB profiles (DCP 1–11) available that represent common applications of the DCB standards (see [“Pre-Defined DCB Profiles \(Unicast\)” on page 2-16](#)). Creating custom profiles is also allowed; a maximum of 128 (including the 11 pre-defined) profiles per switch is supported.

### Pre-Defined DCB Profiles (Unicast)

A DCB profile defines a set of PFC, ETS, and DCBx parameters that are applied to different traffic classes. Using profiles simplifies the DCB configuration process across the network. Once the appropriate pre-defined profile is selected or a custom profile is created for a particular set of traffic classes, the profile is then applied to each OmniSwitch that lies in the transport path for the specified set of traffic classes.

This section contains the Traffic Class priority mappings and the related DCB configuration parameters for each of the eleven pre-defined OmniSwitch DCB profiles. By default, each QSet port instance is associated with DCP 8. See [“Multicast and Unicast Traffic Distribution” on page 2-26](#) for more information.

DCB Profile 1							
Profile Control :: DCBX Configuration							
PFC :: Enable TLV : [*Yes/No] Willing : [*Yes/No] Capability : 8				Recommendation :: Enable TLV : [*Yes/No] Recommended Profile : [DCB_Def_Profile_1/NULL]			
ETS :: Enable TLV : [*Yes/No] Willing : [*Yes/No] Max TC : 3				Application :: Enable TLV : [Yes/*No] Defense Mode :: [*Enable/Disable] Default MTU :: [9216]			
Profile Data :: Max Traffic Classes (TC) : 3							
Traffic Class (TC)	Traffic Type (TT)	TC Description (TD)	TC Priority Map (TP)	TC Bandwidth (%)		TC Flow Control (TF)	TC Scheduler (TS)
0	BE	Best Effort	0, 1, 2, 3	33	100	Lossy	ETS
1	VO	Voice (<10ms latency)	5, 4	67	100	Loss Less	ETS
2	NC	Network Control	7, 6	0	100	Lossy	Strict Priority

DCB Profile 2							
Profile Control :: DCBX Configuration							
PFC :: Enable TLV : [*Yes/No] Willing : [*Yes/No] Capability : 8				Recommendation :: Enable TLV : [*Yes/No] Recommended Profile : [DCB_Def_Profile_1/NULL]			
ETS :: Enable TLV : [*Yes/No] Willing : [*Yes/No] Max TC : 3				Application :: Enable TLV : [Yes/*No] Defense Mode :: [*Enable/Disable] Default MTU :: [9216]			
Profile Data :: Max Traffic Classes (TC) : 4							
Traffic Class (TC)	Traffic Type (TT)	TC Description (TD)	TC Priority Map (TP)	TC Bandwidth (%)		TC Flow Control (TF)	TC Scheduler (TS)
0	BE	Best Effort	0, 1	27	100	Lossy	ETS
1	CA	Critical Applications	3, 2	40	100	Lossy	ETS
2	VO	Voice (<10ms latency)	5, 4	33	100	Loss Less	ETS
3	NC	Network Control	7, 6	0	100	Lossy	Strict Priority

DCB Profile 3							
Profile Control :: DCBX Configuration							
PFC :: Enable TLV : [*Yes/No] Willing : [*Yes/No] Capability : 8				Recommendation :: Enable TLV : [*Yes/No] Recommended Profile : [DCB_Def_Profile_1/NULL]			
ETS :: Enable TLV : [*Yes/No] Willing : [*Yes/No] Max TC : 3				Application :: Enable TLV : [Yes/*No] Defense Mode :: [*Enable/Disable] Default MTU :: [9216]			
Profile Data :: Max Traffic Classes (TC) : 5							
Traffic Class (TC)	Traffic Type (TT)	TC Description (TD)	TC Priority Map (TP)	TC Bandwidth (%) Min	TC Bandwidth (%) Max	TC Flow Control (TF)	TC Scheduler (TS)
0	BE	Best Effort	0, 1	27	100	Lossy	ETS
1	CA	Critical Applications	3, 2	40	100	Lossy	ETS
2	VO	Voice (<10ms latency)	5, 4	33	100	Loss Less	ETS
3	IC	Internetwork Control	6	0	100	Lossy	Strict Priority
4	NC	Network Control	7	0	100	Lossy	Strict Priority

DCB Profile 4							
Profile Control :: DCBX Configuration							
PFC :: Enable TLV : [*Yes/No] Willing : [*Yes/No] Capability : 8				Recommendation :: Enable TLV : [*Yes/No] Recommended Profile : [DCB_Def_Profile_1/NULL]			
ETS :: Enable TLV : [*Yes/No] Willing : [*Yes/No] Max TC : 3				Application :: Enable TLV : [Yes/*No] Defense Mode :: [*Enable/Disable] Default MTU :: [9216]			
Profile Data :: Max Traffic Classes (TC) : 6							
Traffic Class (TC)	Traffic Type (TT)	TC Description (TD)	TC Priority Map (TP)	TC Bandwidth (%) Min	TC Bandwidth (%) Max	TC Flow Control (TF)	TC Scheduler (TS)
0	BK	Background	1	7	100	Lossy	ETS
1	BE	Best Effort	0	20	100	Lossy	ETS
2	CA	Critical Applications	3, 2	40	100	Lossy	ETS
3	VO	Voice (<10ms latency)	5, 4	33	100	Loss Less	ETS
4	IC	Internetwork Control	6	0	100	Lossy	Strict Priority
5	NC	Network Control	7	0	100	Lossy	Strict Priority

DCB Profile 5							
Profile Control :: DCBX Configuration							
PFC :: Enable TLV : [*Yes/No] Willing : [*Yes/No] Capability : 8				Recommendation :: Enable TLV : [*Yes/No] Recommended Profile : [DCB_Def_Profile_1/NULL]			
ETS :: Enable TLV : [*Yes/No] Willing : [*Yes/No] Max TC : 3				Application :: Enable TLV : [Yes/*No] Defense Mode :: [*Enable/Disable] Default MTU :: [9216]			
Profile Data :: Max Traffic Classes (TC) : 7							
Traffic Class (TC)	Traffic Type (TT)	TC Description (TD)	TC Priority Map (TP)	TC Bandwidth (%)		TC Flow Control (TF)	TC Scheduler (TS)
				Min	Max		
0	BK	Background	1	7	100	Lossy	ETS
1	BE	Best Effort	0	20	100	Lossy	ETS
2	EE	Excellent Effort	2	25	100	Lossy	ETS
3	CA	Critical Applications	3	15	100	Lossy	ETS
4	VO	Voice (<10ms latency)	5, 4	33	100	Loss Less	ETS
5	IC	Internetwork Control	6	0	100	Lossy	Strict Priority
6	NC	Network Control	7	0	100	Lossy	Strict Priority

DCB Profile 6							
Profile Control :: DCBX Configuration							
PFC :: Enable TLV : [*Yes/No] Willing : [*Yes/No] Capability : 8				Recommendation :: Enable TLV : [*Yes/No] Recommended Profile : [DCB_Def_Profile_1/NULL]			
ETS :: Enable TLV : [*Yes/No] Willing : [*Yes/No] Max TC : 3				Application :: Enable TLV : [Yes/*No] Defense Mode :: [*Enable/Disable] Default MTU :: [9216]			
Profile Data :: Max Traffic Classes (TC) : 8							
Traffic Class (TC)	Traffic Type (TT)	TC Description (TD)	TC Priority Map (TP)	TC Bandwidth (%)		TC Flow Control (TF)	TC Scheduler (TS)
				Min	Max		
0	BK	Background	1	7	100	Lossy	ETS
1	BE	Best Effort	0	20	100	Lossy	ETS
2	EE	Excellent Effort	2	15	100	Lossy	ETS
3	CA	Critical Applications	3	25	100	Lossy	ETS
4	VI	Video (<100ms latency)	4	15	100	Loss Less	ETS
5	VO	Voice (<10ms latency)	5	18	100	Loss Less	ETS
6	IC	Internetwork Control	6	0	100	Lossy	Strict Priority
7	NC	Network Control	7	0	100	Lossy	Strict Priority

DCB Profile 7							
Profile Control :: DCBX Configuration							
PFC :: Enable TLV : [*Yes/No] Willing : [*Yes/No] Capability : 8				Recommendation :: Enable TLV : [*Yes/No] Recommended Profile : [DCB_Def_Profile_1/NULL]			
ETS :: Enable TLV : [*Yes/No] Willing : [*Yes/No] Max TC : 3				Application :: Enable TLV : [Yes/*No] Defense Mode :: [*Enable/Disable] Default MTU :: [9216]			
Profile Data :: Max Traffic Classes (TC) : 8							
Traffic Class (TC)	Traffic Type (TT)	TC Description (TD)	TC Priority Map (TP)	TC Bandwidth (%)		TC Flow Control (TF)	TC Scheduler (TS)
0	BE	Best Effort	0	0	100	Loss Less	Strict Priority
1	BK	Background	1	0	100	Loss Less	Strict Priority
2	EE	Excellent Effort	2	0	100	Loss Less	Strict Priority
3	CA	Critical Applications	3	0	100	Loss Less	Strict Priority
4	VI	Video (<100ms latency)	4	0	100	Loss Less	Strict Priority
5	VO	Voice (<10ms latency)	5	0	100	Loss Less	Strict Priority
6	IC	Internetwork Control	6	0	100	Loss Less	Strict Priority
7	NC	Network Control	7	0	100	Loss Less	Strict Priority

DCB Profile 8 (Default Profile)							
Profile Control :: DCBX Configuration							
PFC :: Enable TLV : [*Yes/No] Willing : [*Yes/No] Capability : 8				Recommendation :: Enable TLV : [*Yes/No] Recommended Profile : [DCB_Def_Profile_1/NULL]			
ETS :: Enable TLV : [*Yes/No] Willing : [*Yes/No] Max TC : 3				Application :: Enable TLV : [Yes/*No] Defense Mode :: [*Enable/Disable] Default MTU :: [9216]			
Profile Data :: Max Traffic Classes (TC) : 8							
Traffic Class (TC)	Traffic Type (TT)	TC Description (TD)	TC Priority Map (TP)	TC Bandwidth (%)		TC Flow Control (TF)	TC Scheduler (TS)
0	BE	Best Effort	0	0	100	Lossy	Strict Priority
1	BK	Background	1	0	100	Lossy	Strict Priority
2	EE	Excellent Effort	2	0	100	Lossy	Strict Priority
3	CA	Critical Applications	3	0	100	Lossy	Strict Priority
4	VI	Video (<100ms latency)	4	0	100	Lossy	Strict Priority
5	VO	Voice (<10ms latency)	5	0	100	Lossy	Strict Priority
6	IC	Internetwork Control	6	0	100	Lossy	Strict Priority
7	NC	Network Control	7	0	100	Lossy	Strict Priority



DCB Profile 9							
Profile Control :: DCBX Configuration							
PFC :: Enable TLV : [*Yes/No] Willing : [*Yes/No] Capability : 8				Recommendation :: Enable TLV : [*Yes/No] Recommended Profile : [DCB_Def_Profile_1/NULL]			
ETS :: Enable TLV : [*Yes/No] Willing : [*Yes/No] Max TC : 3				Application :: Enable TLV : [Yes/*No] Defense Mode :: [*Enable/Disable] Default MTU :: [9216]			
Profile Data :: Max Traffic Classes (TC) : 8							
Traffic Class (TC)	Traffic Type (TT)	TC Description (TD)	TC Priority Map (TP)	TC Bandwidth (%)		TC Flow Control (TF)	TC Scheduler (TS)
0	BK	Background	1	12	100	Loss Less	ETS
1	BE	Best Effort	0	12	100	Loss Less	ETS
2	EE	Excellent Effort	2	12	100	Loss Less	ETS
3	CA	Critical Applications	3	12	100	Loss Less	ETS
4	VI	Video (<100ms latency)	4	13	100	Loss Less	ETS
5	VO	Voice (<10ms latency)	5	13	100	Loss Less	ETS
6	IC	Internetwork Control	6	13	100	Loss Less	ETS
7	NC	Network Control	7	13	100	Loss Less	ETS

DCB Profile 10							
Profile Control :: DCBX Configuration							
PFC :: Enable TLV : [*Yes/No] Willing : [*Yes/No] Capability : 8				Recommendation :: Enable TLV : [*Yes/No] Recommended Profile : [DCB_Def_Profile_1/NULL]			
ETS :: Enable TLV : [*Yes/No] Willing : [*Yes/No] Max TC : 3				Application :: Enable TLV : [Yes/*No] Defense Mode :: [*Enable/Disable] Default MTU :: [9216]			
Profile Data :: Max Traffic Classes (TC) : 8							
Traffic Class (TC)	Traffic Type (TT)	TC Description (TD)	TC Priority Map (TP)	TC Bandwidth (%)		TC Flow Control (TF)	TC Scheduler (TS)
0	BK	Background	1	12	100	Lossy	ETS
1	BE	Best Effort	0	12	100	Lossy	ETS
2	EE	Excellent Effort	2	12	100	Lossy	ETS
3	CA	Critical Applications	3	12	100	Lossy	ETS
4	VI	Video (<100ms latency)	4	13	100	Lossy	ETS
5	VO	Voice (<10ms latency)	5	13	100	Lossy	ETS
6	IC	Internetwork Control	6	13	100	Lossy	ETS
7	NC	Network Control	7	13	100	Lossy	ETS

DCB Profile 11							
Profile Control :: DCBX Configuration							
PFC :: Enable TLV : [*Yes/No] Willing : [*Yes/No] Capability : 8				Recommendation :: Enable TLV : [*Yes/No] Recommended Profile : [DCB_Def_Profile_1/NULL]			
ETS :: Enable TLV : [*Yes/No] Willing : [*Yes/No] Max TC : 3				Application :: Enable TLV : [Yes/*No] Defense Mode :: [*Enable/Disable] Default MTU :: [9216]			
Profile Data :: Max Traffic Classes (TC) : 8							
Traffic Class (TC)	Traffic Type (TT)	TC Description (TD)	TC Priority Map (TP)	TC Bandwidth (%) Min	TC Bandwidth (%) Max	TC Flow Control (TF)	TC Scheduler (TS)
0	BE	Best Effort	0	12	100	Loss Less	ETS
1	BK	Background	1	12	100	Loss Less	ETS
2	EE	Excellent Effort	2	12	100	Loss Less	ETS
3	CA	Critical Applications	3	12	100	Loss Less	ETS
4	VI	Video (<100ms latency)	4	13	100	Loss Less	ETS
5	VO	Voice (<10ms latency)	5	13	100	Loss Less	ETS
6	IC	Internetwork Control	6	13	100	Loss Less	ETS
7	NC	Network Control	7	13	100	Loss Less	ETS

The Windows driver for an Intel NIC has a fixed number of traffic classes (eight) and the priority to TC mapping uses a linear mapping. To support ETS BW allocations with eight traffic classes and linear priority to traffic class mapping, use DCB profile 11 (DCP 11).

## Configuring DCB Profiles

The default DCB profile (DCP 8) is automatically assigned to each QSet instance when a port goes active or a port joins a LAG. It is only necessary to assign a different profile if DCP 8 attributes are not sufficient.

Consider the following when configuring DCB profiles:

- DCB profiles 1–10 are predefined profiles that are not modifiable and cannot be deleted from the switch configuration.
- Creating a custom profile is allowed by importing one of the pre-defined profiles into a new profile ID between 11 and 128, then modifying the new profile attributes as necessary.
- There is only one DCP assigned to each QSet instance and only one QSet instance per port or link aggregate (LAG). However, a LAG may show multiple QSet instances, one for each port that is a member of the LAG.
- When a port leaves a LAG, the default DCP 8 profile is associated with the QSet instance for that port. In other words, if the QSet instance for a port was associated with DCP 4 when the port joined the LAG, the port is associated with DCP 8 when it leaves the LAG.

## Creating a Custom Profile

The **qos qsp dcb import** command is used to create a custom profile using one of the 10 pre-defined DCB profiles as a template. For example, the following command creates profile 20 (DCP 20) using pre-defined DCP 5 as the template for the new profile:

```
-> qos qsp dcb 20 import qsp dcb 5
```

Once the custom profile is created, the following traffic class attributes for the profile are configurable using the **qos qsp dcb tc** command:

TC Attribute	Command Parameters	Description
PFC flow control	<b>pfc flow-type nll   ll</b>	Changes the traffic class to lossy or loss less.
PFC link delay	<b>pfc link-delay 10-100</b>	Sets the actual headroom value for the traffic class. Setting an incorrect value for this command may result in traffic loss. In most cases, the profile default value of 0 for lossy and 52 for lossless is sufficient.
Bandwidth	<b>min-bw   max-bw</b>	Configures the minimum and maximum bandwidth for the traffic class.
Recommended bandwidth	<b>recommended bw</b>	Configures the recommended minimum bandwidth for the traffic class.

## Guidelines for Modifying Custom Profiles

- 1 Changing the priority-to-TC mapping is not allowed.
- 2 Changing the TC scheduler type (SP, ETS) is not allowed.
- 3 Enabling or disabling PFC is allowed on any TC in the custom profile. The PFC status applies to all priorities within the specific TC.
- 4 Changing Weights (Min Rate) on any of the ETS TC is allowed, with ?ETS\_TC\_Weights <= 100%Available\_PR (after SP allocation).
- 5 Changing PIR (shaper)/Max Rate is allowed on all TC Classes/Types (essential for high Priority SP (runaway) classes).
- 6 When a custom profile is modified, the changes are applied to all ports that are associated with that custom profile. To apply specific changes to a single port (QSet instance), import a custom or default profile into a new custom profile, make the necessary changes, then apply the new custom profile to the port.

## Changing the Profile Assignment

To assign a different profile to a specific QSet instance (QSI), use the **qos qsi qsp dcb** command. For example, the following commands assign DCP 2 to port 1/2 and DCP 3 to ports 2/1-10 and linkagg 5:

```
-> qos qsi port 1/2 qsp dcb 2
-> qos qsi port 2/1-10 qsp dcb 3
-> qos qsi linkagg 5 qsp dcb 3
```

To view the DCB profile configuration for the switch, use the **show qos qsp dcb** command.

```
-> show qos qsp dcb
```

Legends: Prio TC Map:

Represents the priority to traffic class mapping;  
begins with priority 0 on the left and displays the  
traffic class it belongs to.

#	Name	Priority TC Map	PFC Cap	ETS		Template-DCP #	Name	802.3x Pause-Ready
				Max TC	TC			
1	dcp-1	00001122	8	8	8	1	dcp-1	No
2	dcp-2	00112233	8	8	8	2	dcp-2	No
3	dcp-3	00112234	8	8	8	3	dcp-3	No
4	dcp-4	10223345	8	8	8	4	dcp-4	No
5	dcp-5	10234456	8	8	8	5	dcp-5	No
6	dcp-6	10234567	8	8	8	6	dcp-6	No
7	dcp-7	01234567	8	8	8	7	dcp-7	No
8	dcp-8	01234567	8	8	8	8	dcp-8	No
9	dcp-9	10234567	8	8	8	9	dcp-9	No
10	dcp-10	10234567	8	8	8	10	dcp-10	No
20	dcp-20	10234567	8	8	8	10	dcp-10	No

See the *OmniSwitch CLI Reference Guide* for more information about related **show** commands.

## Configuring Support for Legacy Pause Frames

Configuring support for legacy PAUSE frames on a DCB OmniSwitch is useful when the switch needs to interoperate with a server that does not support PFC. Support for legacy PAUSE frames is configurable by creating a custom profile with the 802.3x pause attribute enabled.

- 1 Create a custom profile based on any pre-defined profile and enable PAUSE. For example:

```
-> qos qsp dcb 12 import qsp dcb dcp-9 802.3x-pause
```

- 2 Disable DCBX and PFC TLVs on the port that will support the PAUSE frames. For example:

```
-> qos qsi port 1/1/5 dcb dcbx admin-state disable
-> qos qsi port 1/1/5 dcb dcbx pfc tlv disable
```

- 3 Enable pause on the port. For example:

```
-> interfaces 1/1/5 pause tx-and-rx
```

- 4 Apply the custom profile, “dcp-9” to the port. For example:

```
-> qos qsi port 1/1/5 qsp dcb 11
```

When this type of custom profile is applied on an ingress port, a legacy PAUSE frame is sent on the ingress port back to the server during congestion on the egress.

## Configuring DCBx Port Parameters

By default, the Data Center Bridging Exchange (DCBX) protocol is enabled on all switch ports when the switch boots up with the OmniSwitch Data Center software license. To disable DCBX for a port or link aggregate, use the **qos qsi dcb dcbx admin-state** command. For example:

```
-> qos qsi port 1/10 dcb dcbx admin-state disable
-> qos qsi linkagg 2 dcb dcbx admin-state disable
```

In addition to configuring the status of DCBX on a port, the following DCBX port attributes are also configurable using the **qos qsi dcb dcbx pfc** and **qos qsi dcb dcbx ets** commands:

- The “willing” bit setting (Yes or No) for PFC and ETS. By default, this parameter is set to Yes, which means that when a profile configuration mismatch occurs between two directly connected devices, each device will negotiate common values for PFC settings. The ETS settings from the other device are accepted by each device.
- The status of configuration TLV transmission for PFC and ETS (enabled by default).
- The status of the recommended TLV for ETS (enabled by default).

To verify the DCBX port configuration, use the **show qos qsi dcb dcbx** command. For example:

```
-> show qos qsi dcb dcbx
```

Port	DCP	Name	DCBX	Stats	PFC	ETS					
						Admin	Admin	Defense	PFC TLV	PFC Will	Cfg TLV
1/1	8	dcp-8	Ena	Dis	Ena	Ena	Yes	Ena	Ena	Yes	Dis
1/2	8	dcp-8	Ena	Dis	Ena	Ena	Yes	Ena	Ena	Yes	Dis
1/3	8	dcp-8	Ena	Dis	Ena	Ena	Yes	Ena	Ena	Yes	Dis
1/4	8	dcp-8	Ena	Dis	Ena	Ena	Yes	Ena	Ena	Yes	Dis
1/5	8	dcp-8	Ena	Dis	Ena	Ena	Yes	Ena	Ena	Yes	Dis
1/6	8	dcp-8	Ena	Dis	Ena	Ena	Yes	Ena	Ena	Yes	Dis
1/7	8	dcp-8	Ena	Dis	Ena	Ena	Yes	Ena	Ena	Yes	Dis
1/8	8	dcp-8	Ena	Dis	Ena	Ena	Yes	Ena	Ena	Yes	Dis
10	8	dcp-8	Ena	Dis	Ena	Ena	Yes	Ena	Ena	Yes	Dis
vfl	8	dcp-8	Ena	Dis	Ena	Ena	Yes	Ena	Ena	Yes	Dis

See the *OmniSwitch CLI Reference Guide* for more information about the **qos qsi dcb dcbx** and related **show** commands.

# Multicast and Unicast Traffic Distribution

The following Class of Service (CoS) model for unicast and multicast traffic is applied when either the default QSet profile (QSP 1) or the default DCB profile (DCP 8) is the active profile for the port.

Cos 0 - Lower Priority MC (0-3) = 10

Cos 1 - Higher Priority MC (4-7) = 52

Cos 3 - All Other Unicast UC(0-7) =108

Cos 7 - CPU Generated Packets = 127 (maximum weight)

For example:

- When sending two streams of 100% MC Lower Priority and 100% MC Higher Priority, the distribution should be 10 and 50 packets, which is approximately 17% of Lower Priority MC and 83% of Higher Priority.
- When sending Lower Priority MC 100% and UC 100%, the distribution is 9% of MC and 91% of UC.
- When sending Higher Priority MC 100% and UC 100%, the distribution is 32% of MC and 68% of UC.

## Non-Default Profile

The CoS model implemented also applies for non-default QSet profiles (QSP 2, 3, and 4), except on the OmniSwitch 6900 and the following OmniSwitch 10K modules:

- OS10K-QNI-U8 (8 x 40G)
- OS10K-QNI-U4 (4 x 40G)
- OS10K-XNI-U32E (32 x 10G)
- OS10K-XNI-U16E (16 x 10G)
- OS10K-XNI-U16L (8 x 10G, 8 x 1G)

However, for non-default QSet profiles (QSP 2–4) and non-default DCB profiles (DCP 1–7, 9–128) on the OmniSwitch 6900 and the OmniSwitch 10K modules listed above, the multicast and unicast queue mapping is as follows:

## Strict Priority Profiles (for example, DCP 7)

Queues	Priority	Precedence
UC7	7	Highest
MC3	7, 6	
UC6	6	
UC5	5	
MC2	5, 4	
UC4	4	
UC3	3	
MC1	3, 2	
UC2	2	
UC1	1	
MC0	1, 0	
UC0	0	Lowest

## Weighted Round Robin (WRR) Profiles

Queues	Priority	Weight
UC7	7	W7
MC3	7, 6	Avg(W7,W6)
UC6	6	W6
UC5	5	W5
MC2	5, 4	Avg(W5,W4)
UC4	4	W4
UC3	3	W3
MC1	3, 2	Avg(W3,W2)
UC2	2	W2
UC1	1	W1
MC0	1, 0	Avg(W1,W0)
UC0	0	W0

Note: W<sub>n</sub> = Weight of UC<sub>n</sub>

Avg(W<sub>n</sub>,W<sub>m</sub>) = Average of Weights of UC<sub>n</sub> & UC<sub>m</sub>

## Profile with a Mix of Strict Priority and WRR

Unicast queues configured as Strict Priority will inherit behavior from the Strict Priority model, and unicast queues configured as WRR will inherit behavior from the WRR model. Multicast queues will always follow the behavior that the corresponding unicast queues are following. For example:

- If UC7 and UC6 are Strict Priority, then the MC3 (priority 6 and 7) will also use Strict Priority.
- If UC7 and UC6 are Weighted Round Robin, then MC3 (priority 6 and 7) will also use Weighted Round Robin. The weight of MC3 will be the average of the weights for UC6 and UC7.

For DCB profile ETS behavior, where a Traffic Class (TC) can have more than one priority, multicast queues will follow the corresponding unicast queue behavior. For example:

DCB Profile 1:

TC 0 ( Priority 0 -3)

TC 1 ( Priority 4 -5)

TC 2 ( Priority 6 -7)

TC 0 has UC0 through UC3 in Round Robin, so MC0 (priority 0 and 1) and MC1 (priority 2 and 3) will also participate in the Round Robin behavior of TC 0.

## Multicast Source PFC on OmniSwitch 6900

Ingress admission control on the OmniSwitch 6900 does not distinguish between unicast and multicast traffic. Therefore, a multicast source connected to a port which is PFC aware will react to congestion thereby pausing transmission. This will affect multicast hosts not in the congestion path.

When a multicast source is attached to a port on a OmniSwitch 6900, make sure that PFC is not enabled for that particular priority on the ingress. This can be done by configuring the port to use DCP 8 (all priorities are lossless) or for instance, DCB-1 (priority 4 and 5 are lossless, so multicast may be sent at any other priority other than priority 4 or 5).

If multicast sources are configured to react to PFC, it will affect subscribers not in the congestion path.



## Verifying the DCB Configuration

Displaying the Data Center Bridging (DCB) configuration is helpful to verify the actual configuration on each switch in the data center mesh topology. To display information about the DCB configuration, use the **show** commands listed in this section.

<b>show qos qsp dcb</b>	Displays the configured DCB profiles and the traffic classes associated with the DCB profile.
<b>show qos qsi dcb dcbx</b>	Displays the Data Center Bridging Exchange (DCBX) port configuration and status.
<b>show qos qsi dcb ets</b>	Displays the DCB Enhanced Transmission Selection (ETS) port configuration and status.
<b>show qos qsi dcbx pfc</b>	Displays the DCB Priority-based Flow Control (PFC) port configuration and status.
<b>show qos pfc-lossless-usage</b>	Displays the PFC lossless priority usage for the switch.
<b>show qos qsi dcb pfc stats</b>	Displays PFC port statistics.
<b>show configuration snapshot vfc</b>	Displays the current running configuration for DCB, QSP, PFC, and ETS.



# 3 Configuring Shortest Path Bridging

The Alcatel-Lucent OmniSwitch supports Shortest Path Bridging MAC (SPB-M), as defined in the IEEE 802.1aq standard. SPB-M uses the Provider Backbone Bridge (PBB) network model to encapsulate (using IEEE 802.1ah headers) and tunnel customer traffic through the network backbone. The shortest path trees upon which the PBB network infrastructure operates are determined using a version of the Intermediate System-to-Intermediate System (IS-IS) link state protocol that supports TLV extensions for SPB (ISIS-SPB).

Incorporating SPB-M into the data center infrastructure provides the following benefits:

- Transparently extends Layer 2 connections (VLAN segments) across a large virtual service Layer 2 backbone network.
- Maintains a loop-free network while providing efficient use of available bandwidth, especially in a mesh topology. All connections between all switches in the topology remain active (no blocking of redundant links).
- A shortest path is automatically calculated between each bridge and every other bridge in the data center mesh, resulting in low latency and sub-second convergence times needed to support critical data center bridging requirements.
- Can process a large number of customer MAC addresses without overrunning provider network resources. Customer MAC addresses are only learned on Backbone Edge Bridges (BEB), where customer traffic is then encapsulated and tunneled through the network core infrastructure. Backbone Core Bridges (BCB) do not have to learn any customer MAC addresses.
- Provides a clear separation of customer traffic (between different customers and between the provider network domain). Entry points for customer traffic are clearly defined on the participating BEBs. Customer traffic is identified and associated with a specific service instance bound to the PBB infrastructure.
- Integration with Virtual Machine Network Profiles (vNPs) to support virtual machine (VM) discovery and mobility.

## In This Chapter

This chapter provides an overview about how Shortest Path Bridging MAC (SPB-M) works and how to configure SPB-M through the Command Line Interface (CLI). CLI commands are used in the configuration examples; for more details about the syntax of commands, see the *OmniSwitch CLI Reference Guide*.

This chapter includes the following topics:

- [“Shortest Path Bridging Specifications” on page 3-3.](#)
- [“SPB-M Parameter Defaults” on page 3-4.](#)
- [“SPB-M Interface Defaults” on page 3-4.](#)
- [“SPB-M Service Defaults” on page 3-5.](#)
- [“Shortest Path Bridging Overview” on page 3-6.](#)
- [“IP over SPB-M” on page 3-15.](#)
- [“Interaction With Other Features” on page 3-18.](#)
- [“Quick Steps for Configuring SPB-M” on page 3-21.](#)
- [“Configuring SPB-M” on page 3-24.](#)
- [“Verifying the SPB Backbone and Services” on page 3-53.](#)

# Shortest Path Bridging Specifications

The Shortest Path Bridging functionality described in this chapter is supported on the OmniSwitch 10K and OmniSwitch 6900, unless otherwise stated in the following specifications table or specifically noted within any other section of this chapter. Note that any maximum limits provided in this table are subject to available system resources.

Platforms Supported	OmniSwitch 10K, 6900
OmniSwitch Software License	Advanced
IEEE Standards Supported	802.1aq/D3.6: Draft February 10, 2011— <i>Virtual Bridged Local Area Networks-Amendment 9: Shortest Path Bridging</i> 802.1ah/D4.2: DRAFT March 26, 2008— <i>Virtual Bridged Local Area Networks–Amendment 6: Provider Backbone Bridging</i>
IETF Internet-Drafts Supported	draft-ietf-isis-ieee-aq-05.txt— <i>ISIS Extensions Supporting IEEE 802.1aq Shortest Path Bridging</i> IETF draft— <i>IP/IPVPN services with IEEE 802.1aq SPBB networks</i> IETF draft— <i>IP/IPVPN services with IEEE 802.1aq SPB networks</i>
SPB mode supported	SPB-M (MAC-in-MAC)
IP over SPB-M	IPv4 (VPN-Lite and L3 VPN) VRF-to-ISID mapping (one-to-one, one-to-many)
Maximum number of ISIS-SPB instances per switch.	1
Maximum number of BVLANS per switch	4
Number of equal cost tree (ECT) algorithm IDs supported.	16 (Can select any ID between 1 and 16 to assign to a BVLAN)
Maximum number of service instance identifiers (I-SIDs) per switch	1K
Maximum number of VLANs or SVLANs per I-SID	4K
Maximum number of SAPS	OS10K - 8K OS6900 -X20/X40 - 4K OS6900-T20/T40 - 4K (VC Mode) OS6900-T20/T40 - 8K (Standalone Mode)
Maximum Transmission Unit (MTU) size for SPB services.	9K (not configurable at this time)

## SPB-M Parameter Defaults

Parameter Description	Command	Default
ISIS-SPB status for the switch.	<b>spb isis admin-state</b>	Disabled
Equal Cost Tree (ECT) ID number for the backbone VLAN (BVLAN).	<b>spb isis bvlan ect-id</b>	1 or next available ECT ID number on the local switch.
Control BVLAN for the switch.	<b>spb isis control-bvlan</b>	None
The BVLAN tandem multicast mode (only applies to associated SPB services running in tandem mode).	<b>spb isis bvlan tandem-multicast-mode</b>	Source and Group (S, G)
Priority value for the ISIS-SPB instance.	<b>spb isis bridge-priority</b>	32768
Wait time intervals, in milliseconds, for shortest path first (SPF) calculations.	<b>spb isis spf-wait</b>	maximum wait: 1000 initial wait : 100 second wait : 300
Wait time intervals, in milliseconds, for link state PDU (LSP) transmissions.	<b>spb isis lsp-wait</b>	maximum wait: 1000 initial wait : 0 second wait : 300
Graceful restart status for the switch.	<b>spb isis graceful-restart</b>	Disabled
Graceful restart helper status for the switch.	<b>spb isis graceful-restart helper</b>	Disabled

## SPB-M Interface Defaults

Parameter Description	Command	Default
SPB interface status	<b>spb isis interface</b>	Disabled
SPB interface time interval between each hello packet transmission.	<b>spb isis interface hello-interval</b>	9 seconds
SPB interface hello multiplier used to determine hello packet hold time.	<b>spb isis interface hello-multiplier</b>	3
SPB interface link cost to reach the peer bridge.	<b>spb isis interface metric</b>	10

## SPB-M Service Defaults

By default, there are no SPB-M service components configured for the switch. However, when a service is created, the following default values apply:

Parameter Description	Command	Default
SPB service administrative status.	<b>service spb admin-state</b>	Disabled
SPB service multicast replication mode.	<b>service spb multicast-mode</b>	Head-end
SPB service VLAN translation.	<b>service spb vlan-xlation</b>	Disabled
SPB service maximum transmission unit (MTU) value.	Not configurable at this time	9194
SPB service statistics collection.	<b>service spb stats</b>	Disabled
SPB service description.	<b>service spb description</b>	None.
Default profile automatically applied to access ports.	<b>service access l2profile</b>	def-access-profile
Layer 2 profile that specifies how control packets are processed on service access ports.	<b>service l2profile</b>	def-access-profile: STP, GVRP, MVRP = tunnel 802.3ad = peer 802.1x, 802.1ab, AMAP = drop CSCO PDU, VLAN, uplink = drop
VLAN translation for the service access port.	<b>service access vlan-xlation</b>	Disabled
Service access point (SAP) administrative status.	<b>service spb sap admin-state</b>	Disabled
SAP encapsulation.	<b>service spb sap</b>	0 (untagged traffic).
SAP trust mode.	<b>service spb sap trusted</b>	Trusted
SAP statistics collection.	<b>service spb sap stats</b>	Disabled
SAP description.	<b>service spb sap description</b>	None

# Shortest Path Bridging Overview

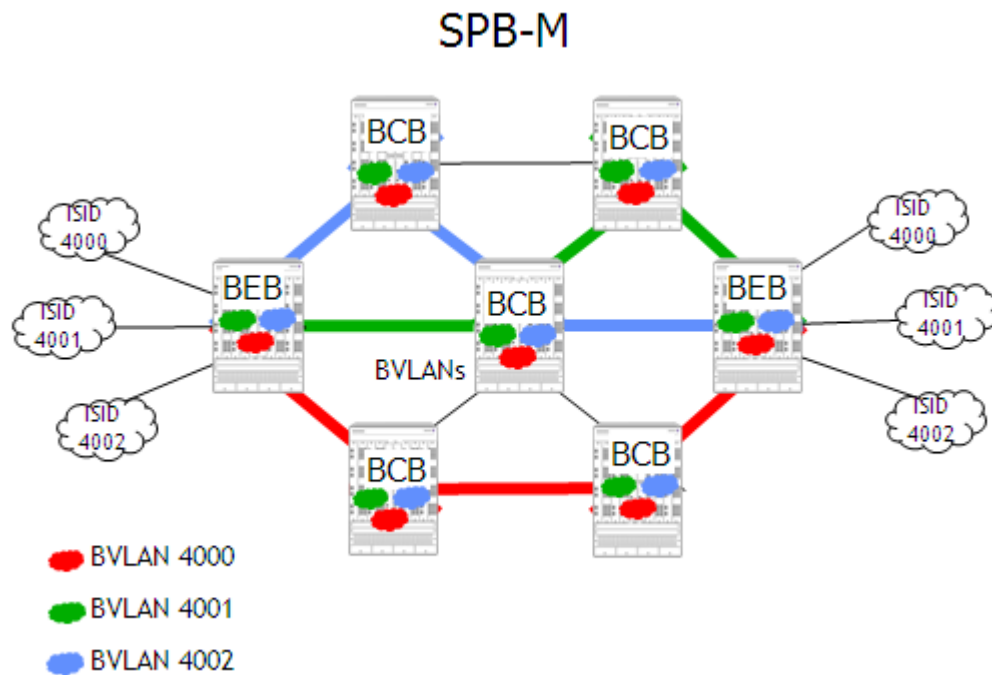
The Alcatel-Lucent OmniSwitch implementation of Shortest Path Bridging (SPB) supports SPB MAC (SPB-M) as defined in the IEEE 802.1aq standard. SPB-M is defined for use in Provider Backbone Bridge (PBB) networks as specified in the IEEE 802.1ah standard.

SPB-M provides a mechanism to automatically define a shortest path tree (SPT) bridging configuration through a Layer 2 Ethernet network. SPB-M Ethernet services use this configuration to encapsulate and tunnel data through the PBB network. The following main components of the OmniSwitch implementation of SPB-M provide this type of functionality:

- **ISIS-SPB**—A version of the Intermediate to Intermediate System (IS-IS) link state protocol that supports SPB TLV extensions. SPB-M uses ISIS-SPB to build sets of symmetric shortest path trees (SPTs) between any SPB switch.
- **Provider Backbone Bridge (PBB) IEEE 802.1ah**— Defines a MAC-in-MAC data encapsulation path for PBB networks that is supported by SPB-M.
- **Provider Backbone Bridge Network (PBBN)**—A network comprised of Backbone Edge Bridges (BEBs) and Backbone Core Bridges (BCB) that is used to interconnect Provider Bridge Networks (PBN) with other networks.
- **Backbone Edge Bridge (BEB)**—A SPB switch positioned at the edge of the PBB network that learns and encapsulates (adds an 802.1ah backbone header to) customer frames for transport across the backbone network. The BEB interconnects the customer network space with PBB network space.
- **Backbone Core Bridge (BCB)**—A SPB node that resides inside the PBB network core. The BCB employs the same BVLAN on two or more network ports. This BVLAN does not terminate on the switch itself; traffic ingressing on an SPB network port is switched to other SPB network ports. As a result, the BCB does not have to learn any of the customer MAC addresses. It mainly serves as a transit bridge for the PBB network.
- **SPB-M Service**—An OmniSwitch Service Manager service configured on the BEBs. Each service maps to a service instance identifier (I-SID) which is bound to a backbone VLAN. One backbone VLAN can accommodate multiple I-SIDs.
- **Backbone VLAN (BVLAN)**—A VLAN that serves as a transport VLAN for the SPB-M service instances and to connect SPB bridges together through SPT sets. Unlike standard VLANs, BVLANs do not learn source MAC addresses or flood unknown destination or multicast frames. Instead, BVLANs only forward on the basis of the forwarding database (FDB) as populated by the ISIS-SPB protocol.

The following illustration shows how SPB-M uses the above components to tunnel customer traffic through a Provider Backbone Bridge Network:





**Figure 1: SPB-M Network Components**

In this network,

- The BEBs are SPB-M capable (ISIS-SPB configured and enabled) and form a shortest path bridging network that also includes the SPB-M capable Backbone Core Bridges (BCBs).
- Each bridge calculates a shortest path tree (SPT) for each BVLAN with itself as the root of each tree.
- SPB Ethernet service instances identified by I-SIDs are created on each BEB. Each I-SID is associated with a BVLAN ID. The BVLAN is configured on each bridge (BEB and BCB) in the backbone network. However, the I-SID itself and the I-SID association with the BVLAN is only configured on each BEB that will service customer traffic.
- A Service Access Point (SAP) is configured on each BEB to identify the access port on which customer traffic will enter the PBBN, the SPB service instance that will tunnel the traffic through the network, and the type of customer traffic to forward (for example, only specific CVLAN IDs, untagged traffic only, or all tagged traffic). Basically, the SAP binds access ports and the specified customer traffic received on those ports to the service.
- Layer 2 traffic from the connected edge networks enters the BEBs through access ports. The SAP configuration on the receiving access port is applied to classify which frames are mapped to which services, if any.
- Classified traffic is then encapsulated into 802.1ah frames by the BEB before the frames are transmitted through the backbone network.
- The 802.1ah encapsulated frames are forwarded on the shortest path through the entire PBBN to reach the intended destination BEB. The BCBs switch traffic based on the destination backbone MAC address (BMAC)—bridge MAC address of the BEB—provided in the 802.1ah header and do not process any I-SID information in the frame.

## SPB-M Shortest Path Trees

The shortest path between two points is a straight line. Shortest Path Bridging (SPB) implements frame forwarding on the shortest path between any two bridges in an Ethernet network. The shortest path trees (SPTs) calculated by SPB provide the shortest and most efficient path to and from the intended destination. SPTs are formed along the direct, straight-line links between switches to make up an overall path through the topology that provides a robust, efficient direction for network traffic to travel.

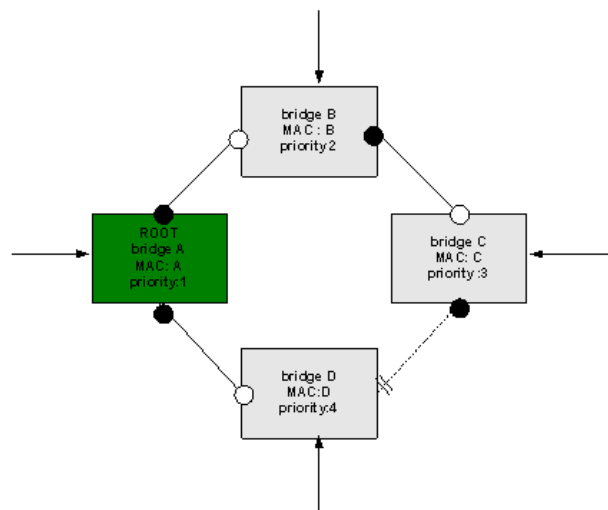
The SPB-M network topology consists of two layers:

- **The backbone infrastructure (control plane) layer.** ISIS-SPB builds the backbone layer by defining loop-free, shortest path trees (SPTs) through the backbone network.
- **The services (data plane) layer.** The service layer is based on the Provider Backbone Bridging (PBB) framework as defined in the IEEE 802.1ah standard. SPB-M supports the 802.1ah MAC-in-MAC method for data encapsulation. SPB-M services transport the encapsulated traffic over the ISIS-SPB infrastructure. (See “[SPB Services](#)” on page 3-12 for more information).

This section contains an example of ISIS-SPB operations in a small SPB-M network. In addition to describing how shortest path trees are created in the BVLAN domain, the flow of unicast and multicast traffic through the network, this example also shows the benefits of using SPB over Spanning Tree for VLAN traffic distribution.

## Spanning Tree

The following diagram shows an example Provider Backbone Bridge (PBB) network with a single backbone VLAN using the Spanning Tree protocol for network loop protection with the same path cost on all links:



**Figure 2: Spanning Tree Topology**

In this example, Bridge A is the Root bridge. As a result, customer traffic entering Bridge A would always use the shortest path to reach every other bridge in the network. However, traffic entering Bridge D that is destined for Bridge C must traverse the path through Bridge A to reach Bridge C, even though Bridge D is directly connected to Bridge C. Clearly the path from Bridge D to Bridge C is not the shortest path in this case.

## ISIS-SPB

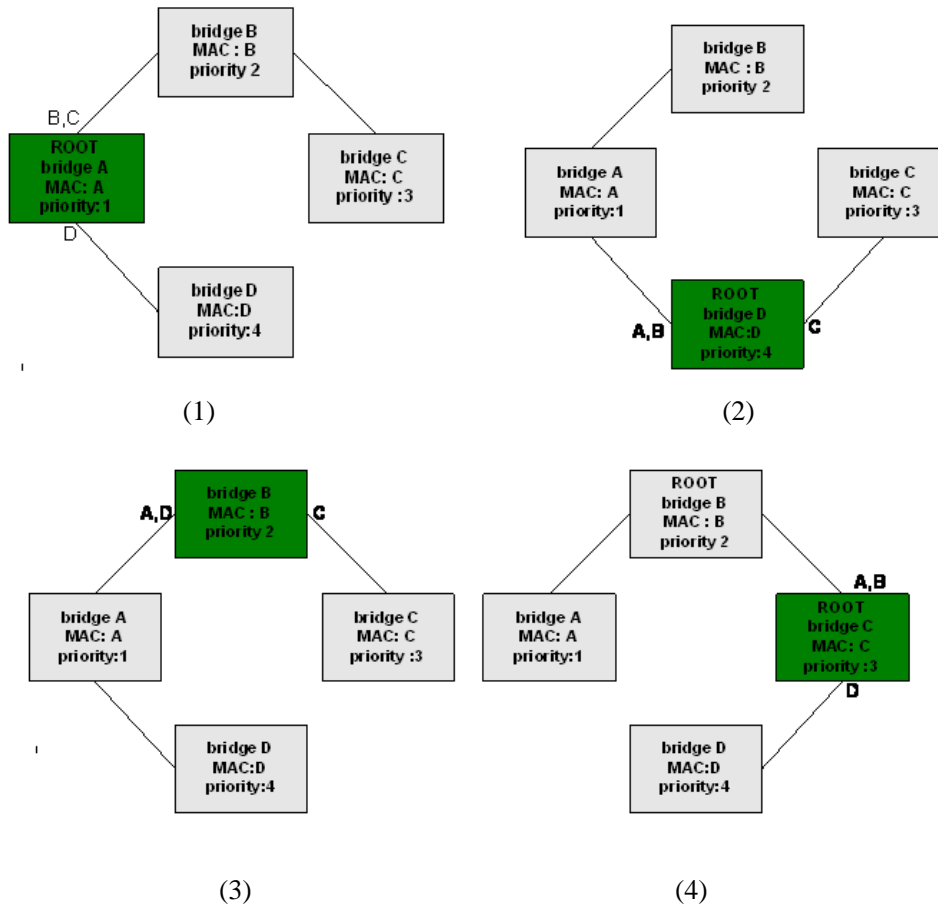
The IEEE 802.1aq standard for SPB specifies the use of the IS-IS link state protocol instead of Spanning Tree to form sets of shortest path trees through the network. When SPB is used, each bridge is the Root for all traffic entering that bridge. As a result, each bridge can provide the shortest path to every other bridge in the network.

The bridging methodology needed to allow each bridge to serve as its own root bridge is enforced through the use of SPB BVLANS. This type of VLAN does not learn customer MAC addresses or flood unknown unicast and multicast traffic. In addition, network loops are mitigated through strict ingress checks based on the source MAC address of frames received on the BVLAN (frames received from an unexpected source are discarded).

SPB-M uses an extended version of the IS-IS protocol that supports SPB (ISIS-SPB) to calculate the SPB-M network topology. In addition, the learning and propagation of source MAC addresses is handled through the ISIS-SPB control plane, instead of through the data plane.

When calculating the SPB-M network topology, ISIS-SPB must meet Layer 2 requirements to create congruent and symmetric paths. To do this, SPB-M supports 16 predefined Equal Cost Tree (ECT) algorithms to break ties when two or more equal cost paths to the same destination are discovered. The same ECT algorithm is configured for the same BVLAN ID on each SPB switch in the network to ensure congruent, symmetric paths for the service traffic bound to that BVLAN.

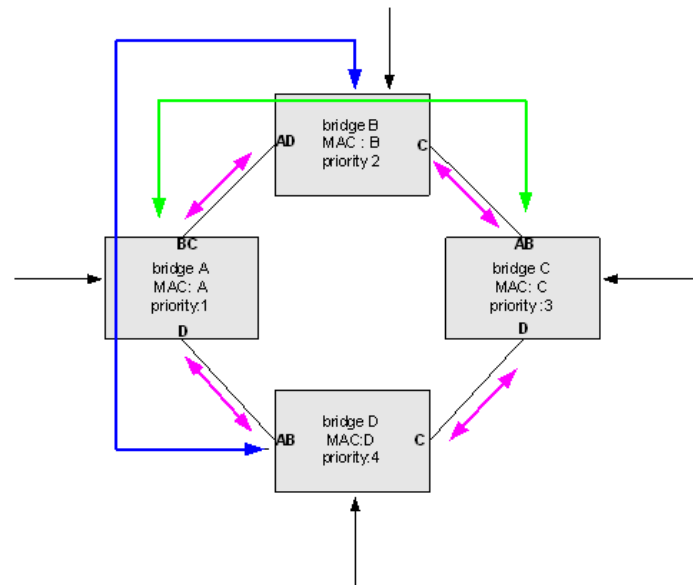
Basically, to create a unicast tree, SPB-M simply computes the shortest path from every bridge with each bridge serving as the Root (as shown below) and populates the Layer 2 forwarding database (FDB) on the SPB bridges with MAC addresses.



**Figure 3: ISIS-SPB Shortest Path Calculations**

The ISIS-SPB unicast trees shown in Figure 3 were built as follows:

- 1** Bridge A calculates the shortest path tree to Bridge B and then programs its FDB with MAC address B on the link, as shown in 4(1).
- 2** Bridge A will then calculate shortest paths to Bridge C and Bridge D and programs the MAC addresses according to the path computed.
- 3** All other bridges follow the same procedure (note that the actual computation is much more optimized and the description here is only for illustration purposes).
- 4** The following traffic pattern for this example network is the result of the ISIS-SPB SPT calculations:



**Figure 4: ISIS-SPB Topology**

As shown in Figure 4, all the backbone MAC (BMAC) addresses are learned by the switches when ISIS-SPB converges. The path taken by each unicast flow (for example ABC, CBA) are reverse path congruent and travel the shortest path through the network.

In the ISIS-SPB topology (Figure 4), the link between Bridge D and Bridge C carries traffic, whereas in the Spanning Tree topology (Figure 3), this link is blocked. Although these examples are based on traffic distribution for a single BVLAN, the ability to make all links in the topology available at all times is especially advantageous in highly redundant, meshed networks.

Although the link between Bridge D and Bridge C is used in the ISIS-SPB topology, traffic flow is relatively low in comparison to the other links. To make better use of this link, a second BVLAN could be created and assigned a different ECT algorithm to trigger ISIS-SPB calculations of a separate set of SPTs for the second BVLAN. This is similar to creating a new Multiple Spanning Tree (MST) instance in a Spanning Tree topology to create a different tree and assigning a new VLAN to that instance.

Each ECT algorithm uses a different calculation to break ties when paths between SPB bridges are equal cost. Another method to influence the SPT calculation is to modify the bridge priority for the switch or change the link cost metric for the SPB interface connection between two switches.

## Multicast Traffic

SPB-M supports two methods for replicating and forwarding multicast traffic (or unknown destination traffic) received from customer equipment: head-end replication and tandem replication.

- **Head-end replication.** Multicast traffic is replicated once for each receiver, encapsulated with the BMAC address, and then sent as a unicast packet to each destination. This method is more suited for networks where there is a low demand for multicast traffic.
- **Tandem replication.** Multicast traffic is replicated only where there is a fork in the SPT and each branch has at least one receiver. Each multicast source bridge in the SPB-M network is the root for a multicast distribution tree (MDT). A MDT is created per-source per-BVLAN and it is pruned according to whether the SPB node is on the shortest path of a multicast transmitter and receiver. For those

MDTs that cross a given Backbone Edge Bridge (BCB), that BCB needs to generate a multicast forwarding table for each such MDT.

Multicast traffic originating from a bridge is encapsulated with a special multicast BMAC DA that identifies the source of the traffic and then forwarded on the tree. Participating bridges that receive the packet will then know the source of the traffic and will use the multicast forwarding information for that source to switch the packet to the appropriate destination.

## SPB Services

The SPB-M network topology consists of two layers:

- **The backbone infrastructure (control plane) layer.** ISIS-SPB builds the backbone layer by defining loop-free, shortest path trees (SPTs) through the backbone network (see [“SPB-M Shortest Path Trees” on page 3-8](#) for more information).
- **The services (data plane) layer.** The service layer is based on the Provider Backbone Bridging (PBB) framework as defined in the IEEE 802.1ah standard. SPB-M supports the 802.1ah MAC-in-MAC method for data encapsulation. SPB-M services transport the encapsulated traffic over the ISIS-SPB infrastructure.

The SPB service layer framework is comprised of the following components:

- **Backbone Edge Bridge (BEB).** An OmniSwitch is considered a BEB if the switch is SPB capable and at least one service access point (SAP) and one SPB interface is configured on the switch. The BEB marks the boundary between the customer network and the PBB network (PBBN).
- **Backbone Core Bridge (BCB).** An OmniSwitch is considered a BCB if the switch is SPB capable and no SAPs are configured but at least one SPB interface is configured on the switch to forward encapsulated SPB-M network traffic. Note that the requirement for configuring a BCB is based on whether or not the network topology includes a transit bridge.
- **Service Instance Identifier (I-SID).** Configured only on a BEB, this component identifies a backbone service instance that will tunnel the encapsulated data traffic through the PBBN between BEBs. The I-SID is bound to a BVLAN ID and a Service Manager SPB service ID when the service is created.
- **Access Port.** A port or link aggregate configured as an SPB access port. This type of port is configured on the BEBs and defines the point at which traffic from other provider networks or directly from customer networks enters the PBBN. The access port is also associated with a Layer 2 profile that specifies how to process protocol control frames received on the port
- **Service Access Point (SAP)**—A SAP is a logical service entity (also referred to as a virtual port) that is configured on a BEB to bind an access port to an SPB service ID and specify the type of customer traffic ((untagged, single-tagged, double-tagged, or all) to encapsulate and tunnel through the PBBN.
- **SPB Interface (Network Port)**—A port or link aggregate configured as an SPB interface that resides on either a BEB or a BCB and connects to the backbone network. Network ports carry customer traffic encapsulated in 802.1ah frames and are associated with all BVLANs on the switch. Customer traffic ingressing on a network port is switched to another network port (on BCBs) or to an access port (on BEBs).

Once the ISIS-SPB infrastructure and the SPB service-based architecture is defined, the following service components are dynamically created by the OmniSwitch. No user-configuration is required.

- **Service Distribution Point (SDP)**—A SDP provides a logical point at which customer traffic is directed from one BEB to another BEB. SDPs are used to set up distributed services, which consist of

at least one SAP on a local node, one SAP on a remote node, and an SDP binding the service on both nodes.

- **Mesh SDP**—A mesh SDP represents the binding of a SPB service instance to an SDP. The SDP then distributes the service connectivity to other BEBs through the ISIS-SPB shortest path trees.

## Sample SPB-M Network Topology

The following diagram provides a sample SPB-M network topology that shows how the SPB-M service and ISIS-SPB backbone layers work together to basically extend (or virtualize) customer traffic across a Provider Backbone Bridge Network (PBBN):

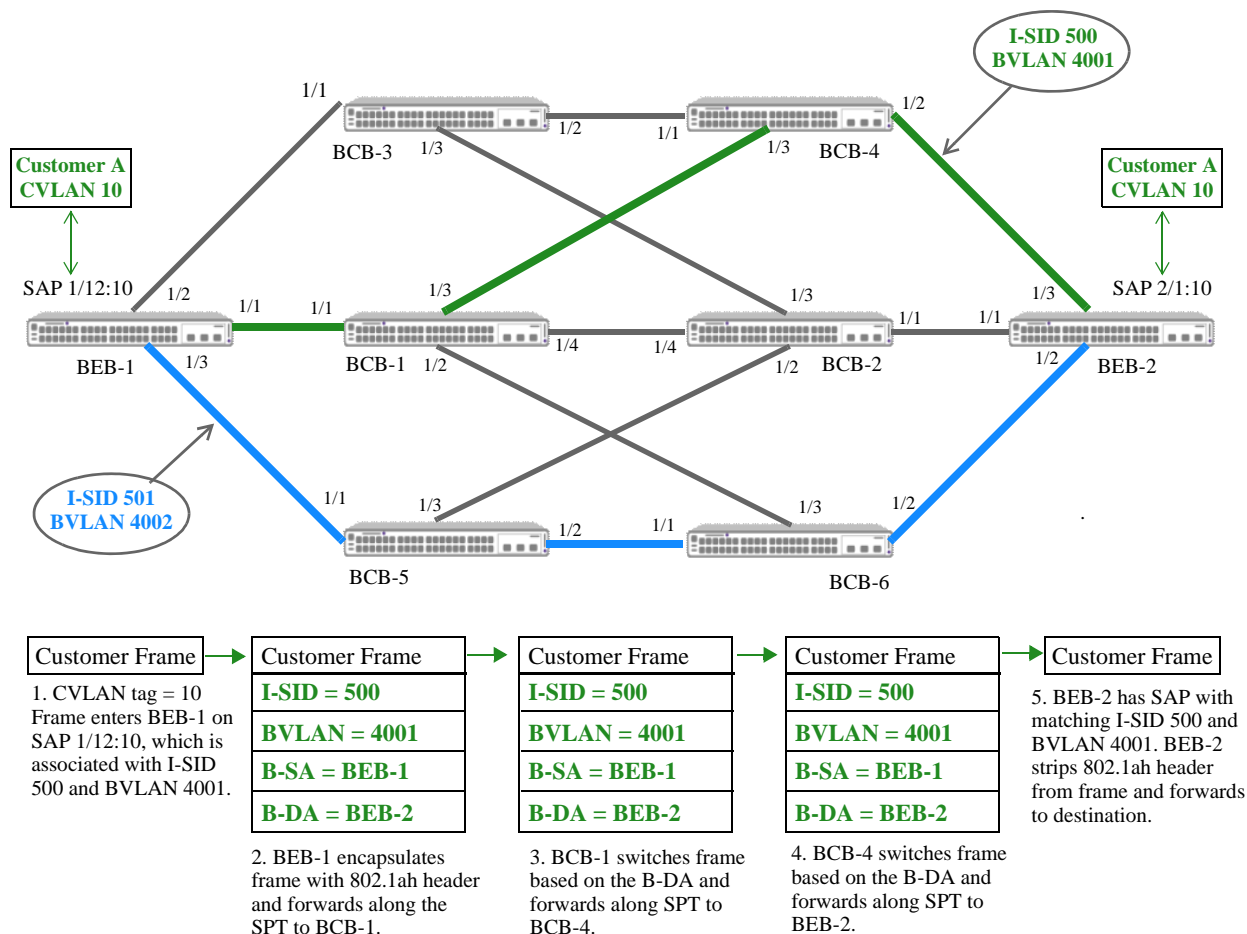


Figure 5: Sample SPB-M Network

In this sample SPB-M topology:

- The packet flow for Customer A frames tagged with VLAN 10 is shown as a typical example. These frames are mapped to an SPB service that represents a binding of I-SID 500 to BVLAN 4001. The path for this binding is shown in green.
- An additional path, shown in blue, is for another SPB service that represents a binding of I-SID 501 to BVLAN 4002. This provides an example of how adding an additional BVLAN and service configuration to the topology can provide an alternate service path for other traffic from the same customer or traffic from a completely different customer.

- SPB BVLAN 4001 and 4002 are created and assigned to ECT ID 1 and 2, respectively, on every switch (BEBs and BCBs) in the topology. These BVLANs serve as the transport entity on which ISIS-SPB builds the shortest path trees and SPB services tunnel data.
- The switch ports connecting each SPB switch with the next-hop SPB switch are configured as SPB interface ports. This type of port is used to forward ISIS-SPB control packets and serves as a network port for tunneling encapsulated traffic through SPB services.
- The service access points (SAPs) created on BEB-1 and BEB-2 determine which frames from Customer A are accepted on the SAP port, where they are then encapsulated and mapped to the associated service. Other SAPs exist on these switches for the other service path.
- When a frame tagged with VLAN 10 ingresses on port 1/12, the frame is encapsulated in an 802.1ah header. The header specifies the BMAC for BEB-1 as the B-SA, the BMAC for BEB-2 as the B-DA, the SAP I-SID (500), and the SAP BVLAN (4001).
- All other frames ingressing on SAP 1/12:10 that are not tagged with VLAN 10 are dropped, unless there are other SAPs configured for that port that will classify those frames.
- The encapsulated frame is then forwarded along the BVLAN 4001 shortest path tree (SPT) to BEB-2, where the 802.1ah header is stripped off and the frame is forwarded to the appropriate destination port.
- The entire process for encapsulating and tunneling customer frames is the same for frames ingressing on port 2/1 of BEB-2 destined for BEB-1.

## How it Works

- There is one instance of ISIS-SPB supported in the backbone topology. This instance is activated once the BVLANs and SPB interfaces are created and the administrative status of ISIS-SPB is enabled for each switch.
- When ISIS-SPB is administratively enabled on each switch, all the configured SPB interfaces start to advertise Hello packets to discover and establish adjacencies with other SPB switches.
- Once adjacencies are established, link state packets (LSPs) are generated with SPB-specific TLVs and shortest path trees from each switch to all other switches are calculated.
- Each SPB switch learns the backbone MAC (BMAC) address and associated BVLAN IDs of every SPB switch in the network and stores that information in a local forwarding database. The BMAC address is the bridge MAC address of the switch and is advertised by ISIS-SPB as the System ID.
- ISIS-SPB then informs Service Manager of the reachability of the BMAC/BVLAN combinations. This information is used to automatically create a service distribution point (SDP) between the same BVLAN on each BEB.
- When ISIS-SPB receives advertisement of a service instance identifier (I-SID) from a remote BEB that matches an I-SID created on the local switch, the SDP (BMAC/BVLAN) of the remote BEB is bound to the I-SID. The binding of a service to an SDP is referred to as a mesh SDP.
- Basically, a SDP is a dynamically created logical entity that distributes service connectivity to other BEBs through the ISIS-SPB shortest path trees. When customer frames are then classified into a specific SAP, the frames are encapsulated and tunneled through the mesh SDP (service/SDP bind) associated with that SAP.



## IP over SPB-M

The OmniSwitch implementation of SPB-M provides L2 VPN capability that bridges L2 customer LAN segments. Customer edge (CE) devices form peers and exchange routing information, as well as perform the necessary IP forwarding. Then the SPB-M BEBs bridge the already routed IP traffic across the SPB-M backbone.

In addition to L2 VPN, the OmniSwitch also provides an IP over SPB-M capability that consolidates the routing functionality of CE devices into the BEB devices. The Virtual Routing and Forwarding (VRF) instances on different BEBs are tied together via backbone I-SIDs across the same SPB-M backbone that is used to support Layer 2 VPNs.

The OmniSwitch IP over SPB-M solution supports two methods for combining L3 routing and L2 SPB-M in the same switch: VPN-Lite and L3 VPN.

### VPN-Lite

The VPN-Lite method provides a gateway between a regular SPB-M service and a router within the same OmniSwitch chassis. This solution provides a specific advantage in that it allows a single box to represent two tiers in a typical fat-tree network, which is popular in datacenter deployments.

In addition, a VPN-Lite configuration can act purely as a L3 VPN when configured correctly. In this mode, existing routing protocols can form adjacencies across the SPB-M PBB network. To keep it purely a L3 VPN, the administrator makes sure that no SPB-M SAPs that can inject bridged flows are allowed to attach to the I-SID designated for the specific VPN.

The VPN-Lite approach uses the SPB-M network in the same way a VLAN is used for transporting L3 frames. Each BEB or host can inject frames into the I-SID as needed, and BEBs can decide to bridge or route those frames based on their inner and outer destination MAC address.

### L3 VPN

When the L3 VPN method is implemented, the OmniSwitch acts as an access or edge router to multiple VRFs and connects these VRFs across an SPB-M PBB network. Each VPN is identified by a local VRF instance on each BEB and globally in the backbone by an I-SID in the PBB header. ISIS-SPB will import and export routes from the local routing protocols running inside their respective VRFs. In essence, ISIS-SPB is creating tunnels between BEBs through which routed frames are sent to reach their target networks.

The OmniSwitch L3 VPN solution is based on the IETF drafts *IP/IPVPN services with IEEE 802.1aq SPB(B) networks* and uses IS-IS TLVs to exchange routes between the BEBs that host the same VPN services. This approach also gives an administrator the ability to build VPNs and extend them over an SPB-M core.

### IP over SPB-M Loopback

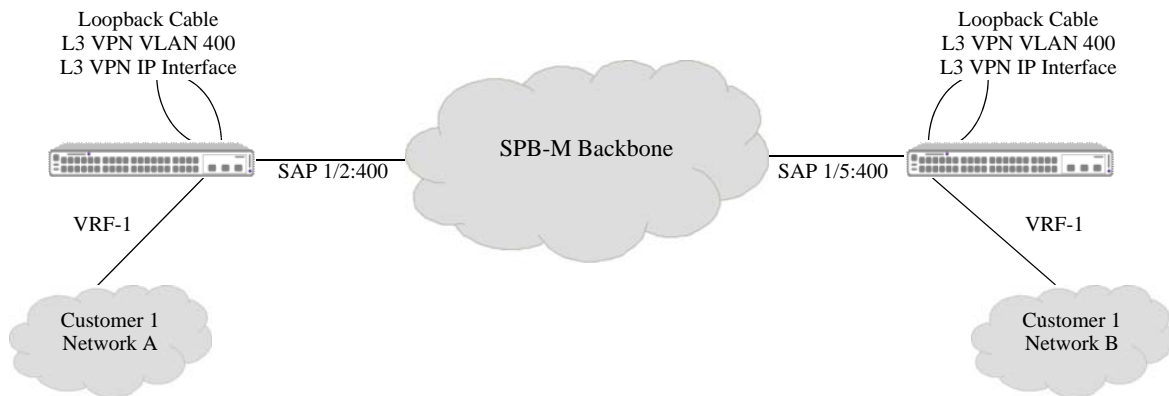
Both the VPN-Lite and L3 VPN solutions for routing IP over an SPB-M backbone network require a physical loopback port configuration on the BEB. A regular switch port or a static link aggregate can serve as a loopback port. In addition, multiple loopback port pairs are allowed and can be shared between different VRFs.

The loopback configuration consists of one port tagged with an IP interface VLAN that belongs to a single VRF instance connected to another port that is assigned to an SPB SAP, to which the VLAN ID associated with the other loopback port is assigned.

- The loopback port assigned to the IP VLAN is referred to as the L3 VPN router port.

- The loopback port assigned to the SPB SAP is referred to as the L3 VPN access port.
- The VLAN associated with both loopback ports is referred to as the L3 VPN VLAN.
- The IP interface assigned to the VLAN is referred to as the L3 VPN IP interface.

The loopback cable connects a VRF to an SPB SAP and can carry traffic from different VRFs tagged with different L3 VPN VLANs. The following diagram shows a logical depiction of the IP over SPB loopback configuration:



**Figure 6: IP over SPB Loopback**

## How it Works

This section describes the VPN-Lite and L3 VPN control and data plane operations in an IP over SPB network configuration. Although both approaches require the L3 VPN loopback configuration, they differ in how routing protocol control packets are exchanged and processed to support IP over SPB.

### VPN-Lite Control Plane Operations

When routing protocols or static routes are running on the L3 VPN IP interface of the loopback configuration, the interface can exchange IP routes with other L3 VPN IP interfaces that are running the same routing protocols and are associated with the same I-SID. By exchanging routes with other loopback IP interfaces, VRFs on different BEBs can learn remote networks from each other.

In this scenario, the routing protocol control packets sent from the L3 VPN IP interfaces travel through the L3 VPN router port, enter the L3 VPN access port, are distributed into different services by SAPs and are carried on SDPs into the SPB-M backbone. The control packets received from the SPB-M backbone travel from SDPs to the VRF following the same process but in reverse.

### L3 VPN Control Plane Operations

The ISIS-SPB support of the IPVPN TLV and IPv4 sub-TLV provides a different method for exchanging L3 routes between VRFs. Instead of running routing protocols on the L3 VPN IP interfaces, IP routes are imported into ISIS-SPB from VRFs. ISIS-SPB then carries these routes in the TLVs through the SPB-M cloud to other SPB-M BEBs. When ISIS-SPB receives IPVPN TLVs from the cloud, ISIS-SPB will export the routes to the appropriate VRFs.

The L3 VPN approach implements the importing and exporting of routes between ISIS-SPB and VRF instances and the transport of these routes using the supported TLVs. The administrator does not configure routing protocols on the L3 VPN IP interface. Implementing the L3 VPN approach requires careful consideration to avoid routing loops.

## VPN-Lite and L3 VPN Data Plane Operations

Data is moved in the same manner for both VPN-Lite and L3 VPN traffic, and the existing data plane forwarding mechanisms for SPB and IP are used without modification:

- A L3 VPN IP interface serves as an IP gateway to access remote networks. The network administrator has to ensure the IP subnet reachability of the L3 VPN addresses on the same SPB-M I-SID.
- L3 VPN IP interfaces use dynamic ARP to learn the MAC addresses of other L3 VPN IP interfaces and provide next-hop forwarding information to the switch.
- IP data plane packets travel the same path as VPN-Lite control packets (see [“VPN-Lite Control Plane Operations” on page 3-16](#) for more information).
- Data in the SPB-M cloud is encapsulated into the Provider Backbone Bridge (PBB) format (see [“SPB Services” on page 3-12](#) for more information).

For more information and configuration examples, see [“Configuring IP over SPB” on page 3-44](#) and [“IP over SPB Configuration Examples” on page 3-46](#).

# Interaction With Other Features

This section contains important information about Shortest Path Bridging MAC (SPB-M) interaction with other OmniSwitch features. Refer to the specific chapter for each feature to get more detailed information about how to configure and use the feature.

## Backbone VLANs (VLAN Manager)

VLAN Manager CLI commands are used to create a SPB backbone VLAN (BVLAN). Although a BVLAN is created in a similar manner as a standard VLAN, BVLANs differ from standard VLANs as follows:

- No Spanning Tree control—the Spanning Tree protocol is automatically disabled on each BVLAN and all ports associated with each BVLAN will remain in a forwarding state. However, Spanning Tree can remain operational on other types of VLANs.
- No source MAC address learning—normal hardware learning is disabled on BVLANs. Instead, the forwarding database (FDB) is populated by the ISIS-SPB protocol.
- There is no flooding of unknown destination or multicast frames.
- Ingress filtering based on the source MAC address—frames received on ports that do not have an incoming source MAC address pre-programmed by ISIS-SPB are discarded.
- IP interfaces are not supported on BVLANs.

## Dynamic Service Access Points (UNP)

The Universal Network Profile (UNP) feature supports two types of profiles: VLAN and service. The service profile type triggers the dynamic creation of a service access point (SAP) when traffic received on a UNP access port is classified and assigned to that profile. If the service that the SAP is associated with does not exist, the service is also dynamically created.

The main advantage of UNP service profiles is to allow incoming traffic to trigger dynamic SAP creation, thus reducing the amount of manual configuration required. In addition, no other protocols are required on the switch or host device to support this functionality.

## Hardware

All SPB functionality is supported on the OmniSwitch 6900 and OmniSwitch 10K switches. However there is one exception: the OS10K\_XNI\_U32 module does not support the configuration of access ports and service access point (SAPs). As a result, the level of SPB support on an OmniSwitch 10K with an OS10K\_XNI\_U32 module is as follows:

- The switch can serve as a Backbone Core Bridge (BCB) if ports on the OS10K\_XNI\_U32 are only configured as SPB interfaces (network ports).
- The switch can serve as a Backbone Edge Bridge (BEB) in a mixed chassis only if access ports and SAPs are configured on other module types and no SPB at all is configured on the OS10K\_XNI\_U32.

## Link Aggregation

- Both static and dynamic link aggregates are configurable as SPB-M service access ports and as SPB-M network interfaces.
- Note that a link aggregate must consist of all access ports or all network ports. SPB-M functionality is not supported on link aggregates that consist of a mixture of SPB-M ports and standard switch ports.

## OAM

- OAM support per the IEEE 802.1ah standard for Provider Backbone Bridging (PBB) is applied at the customer VLAN (CVLAN) and the backbone VLAN (BVLAN) level. Support at the service instance (I-SID) level is not available.
- The OmniSwitch proprietary Layer 2 ping and traceroute features are available to troubleshoot CVLAN and BVLAN domains, including an I-SID check.

## Quality of Service (QoS)

- The priority assignment of a user frame is determined at an access point. A Service Access Point (SAP) on an SPB access port can be configured to be trusted or un-trusted. If a SAP is configured to be trusted, then internal priority for ingress traffic on that SAP is derived from tagged or NULL tagged ingress packet priority or from default port priority if ingress packet is untagged. If a SAP is untrusted then internal priority can be user configured.
- QoS performs the following actions on ports configured as access ports:
  - > Access ports are automatically trusted and the default classification is set to 802.1p.
  - > The trust status and classification are not user-configurable on access ports.
  - > All QoS CLI configuration is blocked on access ports. This includes physical ports and ports that are members of a link aggregate.
  - > Untagged L2 control packets (such as BPDU, GVRP, AMAP) are always tunneled (if enabled) through the SPB domain with the default EXP bits set at 7, so that they can arrive at the destination at the highest priority of 7. Trusted/untrusted SAPs configured on access ports will not affect the priority assignment for Layer 2 control packets.
- QoS priority (802.1p) is applied as follows to trusted and untrusted SAPs:

SAP Configuration	Allowed Configuration	
Tagged (VLAN 1–4094)	Trusted	Tagged traffic priority derived from tags.
	Untrusted	Tagged traffic priority configured by user.
QinQ (outer VLAN 1–4094)	Trusted	Tagged traffic priority derived from outer tags.
	Untrusted	Tagged traffic priority configured by user.
Wild Card	Trusted	Tagged traffic priority derived from tags. Untagged traffic Port default (PRI 0).
	Untrusted	Tagged/ traffic priority configured by user
Untagged	Trusted	Untagged Traffic Port default (PRI 0)
	Untrusted	Priority configured by user.

- > By default, a SAP is trusted with best effort priority (0)

- A SAP can be dynamically changed to trusted/untrusted without admin down the sap
- A SAP priority may only be set when a SAP is untrusted.
- When a SAP is changed from untrusted to trusted, any previously assigned priority is reset with best effort priority (0).
- A trusted SAP that defines a double-tagged encapsulation (QinQ) will use the outer VLAN tag to determine the priority of the frame.
- Priority handling at the edge and core components of an SPB-M topology:
  - On a ingress Backbone Edge Bridge (BEB), a frame is classified to a SAP. The internal priority is determined based on the QoS settings of the SAP (for example, trusted vs. untrusted, default priority). This internal priority is mapped to the backbone VLAN (BVLAN) tag of the tunnel encapsulation.
  - The Backbone Core Bridge (BCB) acts as a Layer 2 device that switches the frame across ingress to egress ports in the BVLAN domain. The BVLAN tag is used to determine the internal priority queue on the egress port where the frame is enqueued.
  - On a egress BEB, the internal priority is determined from the BVLAN tag. The frame is de-encapsulated and enqueued to the egress queue(s) of the access port(s) based on this internal priority.

## VRF

IP over SPB uses Virtual Routing Forwarding (VRF) instances to exchange routes with I-SIDs. This accomplished via the Global Route Manager. VRF routes are exported to the GRM and imported into I-SIDs; I-SID routes are exported to the GRM and imported into VRFs.

- A binding is created between a VRF and the I-SIDs to identify which I-SIDs will export routes to GRM for the specified VRF (see [“Configuring IP over SPB” on page 3-44](#) for more information).
- The **vrf import** command has an optional **isid** parameter that notifies GRM to import the routes from the specified I-SID into the requesting VRF.

# Quick Steps for Configuring SPB-M

This section provides a quick tutorial for configuring the SPB-M network backbone (control plane) and the service encapsulation path (data plane). The Command Line Interface (CLI) commands provided in this section are used to configure the [“Sample SPB-M Network Topology”](#) on page 3-13.

## Quick Steps for Configuring the SPB-M Backbone

The following quick steps are used on each switch in the SPB-M backbone that will participate in the [“Sample SPB-M Network Topology”](#) on page 3-13. This includes both edge and transit (core) switches.

- 1 Use the **system name** command to assign a unique system name to each SPB switch in the domain.

```
-> system name BEB-1
-> system name BEB-2
-> system name BCB-1
-> system name BCB-2
-> system name BCB-3
-> system name BCB-4
-> system name BCB-5
-> system name BCB-6
```

- 2 Use the **spb bvlan** command to create BVLANS 4001 and 4002 on each switch (edge and core switches) that will participate in the SPB-M topology.

```
-> spb bvlan 4001
-> spb bvlan 4002
```

- 3 Use the **spb isis bvlan ect-id** command to change the equal cost tree (ECT) algorithm ID for the specified BVLAN, as necessary, to make sure that the same ECT ID is assigned to the same BVLAN ID on each switch in the SPB-M topology.

```
-> spb isis bvlan 4001 ect-id 1
-> spb isis bvlan 4002 ect-id 2
```

- 4 Use the **spb isis control-bvlan** command to designate one of the BVLANS on each SPB switch as the control BVLAN for the SPB instance. The control BVLAN is used to exchange ISIS-SPB control packets with neighboring SPB switches.

```
-> spb isis control-bvlan 4001
```

- 5 Use the **spb isis interface** command to configure a port or link aggregate as a SPB interface. This type of interface sends PDUs to detect neighboring SPB switches and form adjacencies and also serves as a network port that is used to carry encapsulated service traffic through the SPB-M backbone network.

```
-> spb isis interface port 1/1-3
-> spb isis interface port 1/1-4
```

- 6 Use the **spb isis admin-state** command to enable the SPB instance for the switch. Enabling ISIS-SPB on the switch triggers the transmission of hello packets from the SPB interfaces, which starts the process of defining the SPB infrastructure and calculating the shortest path trees (SPTs) through the topology.

```
-> spb isis admin-state enable
```

## Quick Steps for Configuring SPB Services

The following quick steps use the OmniSwitch Service Manager commands to configure the logical entities that comprise the SPB services in the [“Sample SPB-M Network Topology” on page 3-13](#).

**1** Use the **service access** command to configure a port or link aggregate on which customer traffic is received as a SPB service access port.

```
-> service access port 1/1
-> service access port 2/1
```

**2** Use the **service spb** command to create a SPB service and associate that service with a backbone service instance identifier (I-SID) and BVLAN.

```
-> service spb 1 isid 500 bvlan 4001 admin-state enable
-> service spb 2 isid 501 bvlan 4002 admin-state enable
```

**3** Use the **service spb sap** command to create a service access point (SAP) by associating a SPB service with SAP ID. A SAP ID is comprised of a port or link aggregate and an encapsulation value that identifies the customer traffic to associate with the service.

```
-> service spb 1 sap port 1/12:10 admin-state enable
-> service spb 1 sap port 2/1:10 admin-state enable
-> service spb 2 sap port 1/12:0 admin-state enable
-> service spb 2 sap port 1/12:all admin-state enable
```

In this example, SAPs 1/12:10 and 2/1:10 map traffic ingressing on these SAPs that has an outer tag (customer VLAN tag) equal to 10 to the service associated with the SAP. In this case, SPB service 1 (ISID=500, BVLAN=4001). SAPs 1/12:all and 2/1:all map all tagged and untagged traffic ingressing on these SAPs to the service associated with the SAP.

## Sample Command Configuration

This section provides the sequence of commands used on each switch to configure the [“Sample SPB-M Network Topology” on page 3-13](#). Note that the SPB-M backbone is configured on every switch first, then the SPB-M service architecture is configured second. Following this order of configuration is highly recommended to ensure proper switch participation in ISIS-SPB adjacencies and shortest path tree calculations.

### SPB-M Backbone Commands

The **system name** and Shortest Path Bridging (**spb**) commands are used to configure the SPB-M backbone infrastructure for the sample topology, as shown:

BEB-1	BEB-2	BCB-1
-> system name BEB-1	-> system name BEB-2	-> system name BCB-1
-> spb bvlan 4001	-> spb bvlan 4001	-> spb bvlan 4001
-> spb bvlan 4002	-> spb bvlan 4002	-> spb bvlan 4002
-> spb isis bvlan 4001 ect-id 1	-> spb isis bvlan 4001 ect-id 1	-> spb isis bvlan 4001 ect-id 1
-> spb isis bvlan 4002 ect-id 2	-> spb isis bvlan 4002 ect-id 2	-> spb isis bvlan 4002 ect-id 2
-> spb isis control-bvlan 4001	-> spb isis control-bvlan 4001	-> spb isis control-bvlan 4001
-> spb interface port 1/1-3	-> spb interface port 1/1-3	-> spb interface port 1/1-4
-> spb isis admin-state enable	-> spb isis admin-state enable	-> spb isis admin-state enable



**BCB-2**

```
-> system name BCB-2
-> spb bvlan 4001
-> spb bvlan 4002
-> spb isis bvlan 4001 ect-id 1
-> spb isis bvlan 4002 ect-id 2
-> spb isis control-bvlan 4001
-> spb interface port 1/1-4
-> spb isis admin-state enable
```

**BCB-3**

```
-> system name BCB-3
-> spb bvlan 4001
-> spb bvlan 4002
-> spb isis bvlan 4001 ect-id 1
-> spb isis bvlan 4002 ect-id 2
-> spb isis control-bvlan 4001
-> spb interface port 1/1-3
-> spb isis admin-state enable
```

**BCB-4**

```
-> system name BCB-4
-> spb bvlan 4001
-> spb bvlan 4002
-> spb isis bvlan 4001 ect-id 1
-> spb isis bvlan 4002 ect-id 2
-> spb isis control-bvlan 4001
-> spb interface port 1/1-3
-> spb isis admin-state enable
```

**BCB-5**

```
-> system name BCB-5
-> spb bvlan 4001
-> spb bvlan 4002
-> spb isis bvlan 4001 ect-id 1
-> spb isis bvlan 4002 ect-id 2
-> spb isis control-bvlan 4001
-> spb interface port 1/1-3
-> spb isis admin-state enable
```

**BCB-6**

```
-> system name BCB-6
-> spb bvlan 4001
-> spb bvlan 4002
-> spb isis bvlan 4001 ect-id 1
-> spb isis bvlan 4002 ect-id 2
-> spb isis control-bvlan 4001
-> spb interface port 1/1-3
-> spb isis admin-state enable
```

## SPB-M Service Commands

The Service Manager (**service**) commands are used to build the SPB-M services architecture for the sample topology, as shown. Note that services are only configured on designated BEB switches.

**BEB-1**

```
-> service access port 1/12
-> service spb 1 isid 500 bvlan 4001
-> service spb 2 isid 501 bvlan 4002
-> service spb 1 sap port 1/12:10 admin-state enable
-> service spb 2 sap port 1/12:0 admin-state enable
-> service spb 2 sap port 1/12:all admin-state enable
```

**BEB-2**

```
-> service access port 1/12
-> service spb 1 isid 500 bvlan 4001
-> service spb 2 isid 501 bvlan 4002
-> service spb 1 sap 1/12:10 admin-state enable
-> service spb 2 sap 1/12:0 admin-state enable
-> service spb 2 sap 1/12:all admin-state enable
```

# Configuring SPB-M

Configuring the SPB-M backbone and service layers requires several steps. These steps are outlined here and further described throughout this section. For a brief tutorial on configuring SPB-M, see [“Quick Steps for Configuring SPB-M”](#) on page 3-21.

## Configure the SPB-M Backbone (ISIS-SPB)

Only switches that are SPB capable can participate in the SPB-M network topology. The following configuration steps are required to make an OmniSwitch a SPB-capable node:

- 1 Create a BVLAN.** The BVLAN provides the foundation of the SPB-M infrastructure. A BVLAN is associated with an equal cost tree (ECT) algorithm ID and a SPB service instance ID that is used to carry customer traffic through the backbone network. See [“Backbone VLANs”](#) on page 3-25.
- 2 Configure SPB interfaces.** A SPB interface is associated with each BVLAN that is configured on the switch. At the ISIS-SPB level, this type of interface sends and received ISIS Hello packets and link state PDU (LSP) to discover adjacent SPB switches and calculate the shortest path trees through the SPB-M network topology. At the services level, the SPB interfaces serve as network ports that are used to carry encapsulated customer traffic through the network. See [“Configuring SPB Interfaces”](#) on page 3-28.
- 3 Configure global ISIS-SPB parameters.** In addition to enabling/disabling the ISIS-SPB instance for the switch, global configuration includes settings such as a system name for the switch, global bridge parameters, and various wait time intervals. When ISIS-SPB is enabled for the switch, default settings for these global bridge parameters and wait time intervals are active. It is only necessary to change these values if the default settings are not sufficient. See [“Configuring Global ISIS-SPB Parameters”](#) on page 3-29.

For more information about SPB-M commands, see [Chapter 7, “Shortest Path Bridging Commands,”](#) in the *OmniSwitch CLI Reference Guide*.

## Configure SPB-M Services

The OmniSwitch Service Manager application is used to configure the services layer of the SPB-M network topology. A service is defined by a specific set of logical entities that are configured only on the backbone edge bridges (BEBs) of the network. the following configuration steps are required to define the service-based architecture for a SPB-M network:

- 1 Create a SPB-M service.** A Service Manager service ID is associated with a BVLAN, a backbone service instance identifier (I-SID), and a service access point (SAP) to identify the customer traffic that the service will tunnel through the provider network. See [“Creating a SPB Service”](#) on page 3-35.
- 2 Configure access (customer-facing) ports.** One or more access ports are associated with a service access point (SAP) to identify to the service which ports will receive customer traffic that the service will process for tunneling through the provider network. When an access port is associated with a SAP, the SAP parameter attributes are applied to traffic received on the access port. See [“Configuring Service Access Ports”](#) on page 3-39.
- 3 Define access port profile attributes.** A default Layer 2 profile is automatically assigned to an access port at the time the port is configured as an access port. This profile determines how control frames received on the port are processed. It is only necessary to configure a Layer 2 profile if the default attribute values are not sufficient. See [“Configuring Layer 2 Profiles for Access Ports”](#) on page 3-39.

**4 Configure a SPB service access point (SAP).** A SAP binds a SPB service to an access (customer-facing) port and defines which customer traffic to tunnel through the service. Each SAP is associated to one service name, but a single service can have multiple SAPs to which it is associated. See [“Verifying the SPB Service Configuration” on page 3-37.](#)

For more information about Service Manager commands, see [Chapter 50, “Service Manager Commands,”](#) in the *OmniSwitch CLI Reference Guide*.

## SPB Configuration Guidelines

Configuring a SPB-M network topology involves setting up two layers of functionality: the ISIS-SPB backbone infrastructure and the Provider Backbone Bridge (802.1ah) services layer for MAC-in-MAC encapsulation. Review the guidelines in this section before attempting to configure the various components of the SPB-M infrastructure and services.

### ISIS-SPB

This implementation of the ISIS-SPB protocol supports only a single topology with a multi-topology identifier (MT-ID) of zero.

- The ISIS-SPB protocol instance is independent of IP IS-IS, or other network layer protocol identifiers (NLPIDs) riding in the same IS-IS implementation. However, ISIS-SPB and IP IS-IS can coexist on the same switch.
- ISIS-SPB interfaces, link state packet databases (LSPDB), and forwarding information are all created and maintained within the single ISIS-SPB instance.
- IS-IS Level 1 point-to-point adjacencies are supported; Level 2 is not supported at this time.
- SPB interfaces are associated with a link metric cost that is configurable, thus providing the ability to change the logical topology created by the ISIS-SPB instance. However, if different metric values are configured on each side of a link, ISIS-SPB will choose the higher-valued one as the metric to use for both sides. This is necessary to enforce the symmetry of SPT calculations in both directions across the link.
- Enabling SPB for the switch automatically triggers the transmission of Hello packets from the SPB interfaces, thus starting the process of discovery and forming adjacencies to build shortest path trees.

### Backbone VLANs

- The BVLAN configuration must be the same on each SPB switch within the PBB network. For example, if BVLAN 10 with an ECT ID of 1 is configured on one switch, then BVLAN 10 with an ECT ID of 1 must exist on all other SPB bridges in the network to ensure proper calculation of the ISIS-SPB shortest path trees through the backbone.
- If more than one BVLAN is needed, configure each BVLAN with a different ECT algorithm ID. For example, if two BVLANs (BVLAN 4001 and BVLAN 4002) are needed for a specific SPB-M topology, then create BVLAN 4001 with ECT ID 1 and BVLAN 4002 with ECT ID 2 on each switch that is going to participate in the topology.

---

**Note.** When adding another BVLAN to an existing SPB-M topology instance, create the new BVLAN and its associated ECT ID on every switch first, then configure the SPB service association for the BVLAN. Creating SPB services before the BVLAN configuration is complete on all switches can cause problems with forming adjacencies or may even cause an SPB switch to drop existing adjacencies.

---

- In most cases one BVLAN is sufficient for virtualizing traffic through the network backbone. However, configuring more than one BVLAN provides alternate routes for tunneling customer traffic. This can also provide a form of load balancing by distributing traffic over different BVLAN segments.
- All encapsulated traffic within the BVLAN domain is unicast with a resolved source and destination BMAC addresses. Frames received on BVLAN ports that do not have an incoming source MAC address pre-programmed by ISIS-SPB are discarded.

## Configuring BVLANS

The SPB-M backbone VLAN (BVLAN) provides the foundation on which ISIS-SPB shortest path trees are built and SPB-M services tunnel encapsulated customer data through the Provider Backbone Bridge network (PBBN). Configuring a BVLAN on a switch is also the first step in setting up the ISIS-SPB infrastructure and in making an OmniSwitch an SPB-capable node.

---

**Note.** The BVLAN configuration must be the same on each OmniSwitch that is going to participate in the SPB-M network topology. So if BVLAN 4001 is created on one switch, then BVLAN 4001 must be created on all other switches in the SPB-M network.

---

To create a BVLAN, use the **spb bvlan** command with the optional **name** parameter. For example:

```
-> spb bvlan 4001 name spb-4001
```

If the **name** parameter is not specified with this command, the VLAN ID is used for the name by default. For example, the following command creates BVLAN 4001 with “VLAN 4001” as the name:

```
-> spb bvlan 4001
```

To remove a BVLAN, use the **no** form of the **spb bvlan** command. For example:

```
-> no spb bvlan 4001
```

## Assigning the Equal Cost Tree ID

ISIS-SPB calculations may result in multiple paths of equal costs. The Equal Cost Tree (ECT) ID specifies a tie-breaking algorithm that is used when ISIS-SPB is calculating a set of shortest path trees from one switch to all other switches in the SPB domain. When a BVLAN is created, an ECT ID is automatically assigned to the BVLAN. If it is the first BVLAN created on the switch, ECT ID 1 is assigned, otherwise the next available ID number is used.

Each BVLAN created must be duplicated on all other participating switches in the SPB-M network and must use the same ECT ID number for that BVLAN on each switch. A BVLAN created on one switch may not be automatically assigned the same ECT ID on another switch. As a result, it may be necessary to modify the ECT ID number using the **spb isis bvlan ect-id** command. For example:

```
-> spb isis bvlan 4002 ect-id 2
```

---

**Note.** When adding another BVLAN to an existing SPB-M topology instance, create the new BVLAN and its associated ECT ID on every switch first, then configure the SPB service association for the BVLAN. Creating SPB services before the BVLAN configuration is complete on all switches can cause problems with forming adjacencies or may even cause an SPB switch to drop existing adjacencies.

---

## Configuring the Control BVLAN

One of the BVLANs configured on each switch serves as the control BVLAN for the ISIS-SPB instance. The control BVLAN exchanges ISIS-SPB control packets with neighboring SPB switches on behalf of all BVLANs configured on the local switch. The control packets are tagged with the control BVLAN ID.

By default, the first BVLAN created (or if there is only one), is the control BVLAN. To designate a different BVLAN as the control BVLAN, use the `spb isis control-bvlan` command. For example:

```
-> spb isis control-bvlan 4002
```

A control BVLAN also carries regular encapsulated SPB domain traffic in addition to ISIS-SPB control packets. In other words, a VLAN can serve as both a regular BVLAN and a control BVLAN at the same time.

## Configuring the Tandem Multicast Mode

The tandem multicast mode (\*,G) or (S,G) of a BVLAN is applied only to SPB services associated with the BVLAN that are using tandem replication for multicast traffic. When a BVLAN is created, the (S,G) tandem multicast mode is applied by default.

To change the tandem multicast mode for a BVLAN, use the `spb isis bvlan tandem-multicast-mode` command and specify either `gmode` (\*,G) or `sgmode` (S,G). For example:

```
-> spb isis bvlan 4001 tandem-multicast-mode sgmode
-> spb isis bvlan 4002 tandem-multicast-mode gmode
```

## Verifying the BVLAN Configuration

To view the BVLAN configuration for the switch, use the `show spb isis bvlans` command. For example:

```
-> show spb isis bvlans
```

```
SPB ISIS BVLANS:
```

BVLAN	ECT-algorithm	In Use	Services mapped	Num ISIDS	Tandem Multicast	Root Bridge (Name : MAC Address)
4001	00-80-c2-01	YES	YES	52	SGMODE	
4002	00-80-c2-02	YES	YES	51	SGMODE	
4003	00-80-c2-03	YES	YES	52	SGMODE	
4004	00-80-c2-04	YES	YES	51	SGMODE	

```
BVLANS : 4
```

The BVLAN is a special type of VLAN that is created and maintained by VLAN Manager. As a result, it also appears in the VLAN Manager **show** command displays. For example, in the following **show vlan** output display, VLANs 4001 through 4004 are included and “spb” appears in the “type” column:

```
-> show vlan
  vlan  type  admin  oper  ip    mtu   name
-----+-----+-----+-----+-----+-----+-----
    1   std    Dis    Dis   Dis   1500  VLAN 1
 1000   std    Ena    Ena   Ena   1500  VLAN 1000
 4001   spb    Ena    Ena   Dis   1524  VLAN 4001
 4002   spb    Ena    Ena   Dis   1524  VLAN 4002
 4003   spb    Ena    Ena   Dis   1524  VLAN 4003
 4004   spb    Ena    Ena   Dis   1524  VLAN 4004
 4094   mcm    Ena    Dis   Dis   9198  MCM IPC
```

To view configuration information for an individual BVLAN, use the **show vlan** command and specify the BVLAN ID. For example:

```
-> show vlan 4001
Name                : VLAN 4001,
Type                : Backbone vlan,
Administrative State : enabled,
Operational State   : disabled,
IP Router Port      : disabled,
IP MTU              : 1524
```

## Configuring SPB Interfaces

A port or link aggregate is configurable as a SPB interface. Each switch in the SPB-M topology should have at least one SPB interface configured. The SPB interface serves more than one purpose:

- Advertises IS-IS Hello packets to discover SPB neighbors and establish adjacencies.
- After adjacencies are established, exchanges link state packets (LSPs) with SPB neighbors to build a local LSP database (LSPDB). A switch’s adjacencies are reflected in the contents of its link state packets. This relationship between adjacencies and link state allows the protocol to detect downed routers in a timely fashion.
- Serves as a network port by forwarding encapsulated SPB service traffic on backbone VLANs (BVLANS) through the SPB-M Provider Backbone Bridge (PBB) network.

To configure a port or link aggregate as SPB interface, use the **spb isis interface** command. For example:

```
-> spb isis interface port 1/10
-> spb isis interface linkagg 5
```

When a port is converted to a SPB interface, the interface is automatically assigned to all existing BVLANS. There is one ISIS-SPB instance per switch, and each BVLAN and SPB interface are associated with that instance. However, it is also possible to tag SPB interfaces to carry traffic for standard VLANs.

The **spb isis interface** command is also used to optionally configure the following parameter values:

- **admin-state**—Administratively enables or disables the SPB interface. By default, the interface is enabled when the SPB interface is created.
- **hello-interval**—Specifies the amount of time, in seconds, to wait between each transmission of a hello packet from this interface. By default, the hello time interval is set to nine seconds.

- **hello-multiplier**—Specifies an integer value that is multiplied by the hello interval time to determine the amount of time, in seconds, a receiving bridge holds onto the hello packets transmitted from this interface. By default, the hello multiplier is set to three.
- **metric**—An integer value that specifies the link cost to reach the destination backbone MAC (BMAC). By default, the link cost is set to ten. Changing the link metric value provides a method for changing the logical topology as calculated by ISIS-SPB.

The following command examples change the default hello and metric values for the SPB interface:

```
-> spb isis interface port 4/7 hello-interval 60
-> spb isis interface linkagg 3 hello-multiplier 10
-> spb isis interface port 2/1 metric 100
-> spb isis interface linkagg 5 hello-interval 20 hello-multiplier 5 metric 200
```

## Verifying the SPB Interface Configuration

To view the SPB interface configuration for the switch, use the **show spb isis interface** command. For example:

```
-> show spb isis interface
SPB ISIS Interfaces:
```

Interface	Level	CircID	Oper state	Admin state	Link Metric	Hello Intvl	Hello Mult
1/1	L1	1	UP	UP	10	9	3
1/2	L1	2	UP	UP	10	9	3
1/3	L1	3	UP	UP	10	9	3
1/4	L1	4	DOWN	UP	10	9	3
1/5	L1	5	DOWN	UP	10	9	3
1/6	L1	6	DOWN	UP	22	9	3
1/7	L1	7	DOWN	UP	10	9	3

Interfaces : 7

## Configuring Global ISIS-SPB Parameters

This section describes the global configuration for the ISIS-SPB instance, which includes the following:

- “Configuring the System Name” on page 3-30.
- “Configuring the SPB Bridge Priority” on page 3-30.
- “Configuring the ISIS-SPB Area Address” on page 3-30.
- “Configuring the Shortest Path Source ID” on page 3-31.
- “Configuring the Control MAC Address” on page 3-31.
- “Configuring the Shortest Path First Wait Time” on page 3-31.
- “Configuring the Link State Packet Wait Time” on page 3-32.
- “Configuring the Overload State” on page 3-33.
- “Configuring Redundant Switches for Graceful Restart” on page 3-34.
- “Enabling/Disabling ISIS-SPB” on page 3-34.

To verify the global configuration parameter values for the switch, use the **show spb isis info** command. For example:

```
-> show spb isis info
SPB ISIS Bridge Info:
  System Id           = e8e7.3233.1831,
  System Hostname     = BEB-1,
  SPSourceID          = 03-18-31,
  SPBM System Mode    = auto,
  BridgePriority       = 32768 (0x8000),
  MT ID               = 0,
  Control BVLAN       = 4001,
  Area Address         = 0.0.0,
  Level Capability     = L1,
  Admin State         = UP,
  LSDB Overload       = Disabled,
  Last Enabled        = Thu Aug  2 22:43:19 2012,
  Last SPF             = Fri Aug  3 18:15:51 2012,
  SPF Wait            = Max: 1000 ms, Initial: 100 ms, Second: 300 ms,
  LSP Lifetime        = 1200,
  LSP Wait            = Max: 1000 ms, Initial: 0 ms, Second: 300 ms,
  Graceful Restart    = Disabled,
  GR helper-mode      = Disabled,
  # of L1 LSPs        = 8
  Control Address     = 01:80:C2:00:00:14 (AllL1)
```

## Configuring the System Name

Configuring a system name is required on each switch that is going to participate in the SPB-M topology. To configure a system name for the switch, use the **system name** command. For example:

```
-> system name BEB-1
```

ISIS-SPB advertises the system name to identify the switch to other SPB peer switches.

## Configuring the SPB Bridge Priority

A bridge is ranked within the SPB topology by its bridge ID (an eight byte hex number). The bridge priority value makes up the upper two bytes of the eight-byte SPB bridge ID. The lower six bytes of the Bridge ID contain the system ID, which is the dedicated bridge MAC address of the SPB bridge.

The bridge priority is used in shortest path tree calculations. The lower the priority value, the higher the priority. Setting a different bridge priority value on different SPB bridges will override the system ID significance during the shortest path tree (SPT) calculation.

By default, all SPB switches are assigned a priority value of 32768. To change the bridge priority value for a switch, use the **spb isis bridge-priority** command. For example:

```
-> spb isis bridge-priority 25590
```

## Configuring the ISIS-SPB Area Address

By default, the IS-IS area address for the ISIS-SPB instance is set to 0.0.0, which is typically sufficient for this implementation of SPB-M. Both ISIS-SPB and ISIS-IP instances may coexist on the same switch as long as they don't use the same area address.

If changing the area address is necessary, use the **spb isis area-address** command. For example:

```
-> spb isis area-address 1.1.1
```



---

**Note.** Each switch that is going to participate in the SPB topology must use the same area address and must use an address that is different from the ISIS-IP area address.

---

## Configuring the Shortest Path Source ID

The shortest path (SP) source ID, identifies the source of multicast frames and is relevant only in multi-cast tandem replication mode. By default, the last three least significant bytes of the system ID (local bridge MAC address) is used for the source ID value.

To change the source ID value, use the **spb isis source-id** command. For example:

```
-> spb isis source-id 07-0b-d3
```

To set the source ID back to the default value, use the **spb isis source-id** command with the **auto** parameter. For example:

```
-> spb isis source-id auto
```

## Configuring the Control MAC Address

The control MAC address is the destination MAC address used for ISIS-SPB control packets. Changing this address can enhance interoperability between an SPB-capable OmniSwitch and other third-party SPB-capable devices.

By default, the control MAC address is set to 01:80:C2:00:00:14 (all Level 1 Intermediate Systems). The following parameters are used with the **spb isis control-address** command to change the control MAC address:

- **alll1**—All Level 1 Intermediate Systems (01:80:C2:00:00:14).
- **alll2**—All Level 2 Intermediate Systems (01:80:C2:00:00:15).
- **allis**—All Intermediate Systems (09:00:2B:00:00:05).

For example, the following command changes the default control MAC address from ALLL1 to ALLL2:

```
-> spb isis control-address alll2
```

## Configuring the Shortest Path First Wait Time

The **spb isis spf-wait** command is used to configure the time intervals between the first, second, and subsequent ISIS-SPB shortest path first (SPF) calculations.

Subsequent SPF calculations, if required, are generated at exponentially increasing intervals of the SPF second wait time interval until the maximum wait time interval value is reached. For example, if the second-wait interval value is set to 1000 milliseconds, then the next SPF calculation is triggered after 2000 milliseconds and the next SPF calculation after that is triggered at 4000 milliseconds, and so on, until the maximum-wait interval value is reached.

When the maximum interval value is reached, the SPF wait interval will stay at the maximum value until there are no more SPF calculations scheduled during that interval. After a full interval without any SPF calculations, the SPF wait interval will reset back to the initial wait time interval value.

The following **spb isis spf-wait** command parameters are used to configure the SPF timers:

- **max-wait**—The maximum number of milliseconds to wait between two consecutive SPF calculations. The default maximum wait time value is set to 1000 milliseconds. Specify a maximum value that is the same or greater than the second wait time value.
- **initial-wait**—The number of milliseconds to wait before triggering an initial SPF calculation after a topology change. The default initial wait time value is set to 100 milliseconds. Specify a value that is the same or less than the maximum wait time value.
- **second-wait**—The number of milliseconds to wait between the first and second SPF calculation. The default second wait time value is set to 300 milliseconds. Specify a value that is the same or less than the maximum wait time value.

For example, the following command changes the SPF wait time values for the local SPB instance:

```
-> spb isis spf-wait max-wait 2500 initial-wait 1000 second-wait 1500
```

To change one or more of the wait time values, it is only necessary to specify the parameter for the desired change. For example:

```
-> spb isis spf-wait max-wait 5000
-> spb isis spf-wait initial-wait 1000
-> spb isis spf-wait second-wait 2000
```

To set the wait time values back to the default settings, use the **spb isis spf-wait** command without specifying any of the parameters. For example:

```
-> spb isis spf-wait
```

## Configuring the Link State Packet Wait Time

The **spb isis lsp-wait** command is used to configure the time intervals between the first, second, and subsequently generated link state packets (LSPs).

Subsequent LSP, if required, are generated at exponentially increasing intervals of the LSP second wait time interval until the maximum value is reached. For example, if the second-wait interval value is set to 10 seconds, then the next LSP is generation is triggered after 20 seconds and the next LSP generated after that is triggered at 40 seconds, and so on, until the maximum wait time interval value is reached.

When the maximum interval value is reached, the LSP wait interval will stay at the maximum value until there are no more LSP generations during that interval. After a full interval without any LSP generations, the LSP wait interval will reset back to the initial wait time interval value.

The following **spb isis lsp-wait** command parameters are used to configure the SPF timers:

- **max-wait**—The maximum number of seconds to wait between two consecutively generated LSPs. The default maximum wait time value is set to 1000 milliseconds. Specify a maximum value that is the same or greater than the second wait time value.
- **initial-wait**—The number of seconds to wait before triggering an initial LSP generation after a topology change. The default initial wait time value is set to 0 milliseconds. Specify a value that is the same or less than the maximum wait time value.
- **second-wait**—The minimum number of seconds to wait between the first and second generated LSPs. The default second wait time value is set to 300 milliseconds. Specify a value that is the same or less than the maximum wait time value.

For example, the following command changes the LSP wait time values for the local SPB instance:

```
-> spb isis lsp-wait max-wait 2000 initial-wait 1000 second-wait 1500
```

To change one or more of the wait time values, it is only necessary to specify the parameter for the desired change. For example:

```
-> spb isis lsp-wait max-wait 5000
-> spb isis lsp-wait initial-wait 2500
-> spb isis lsp-wait second-wait 3000
```

To set the wait time values back to the default settings, use the **spb isis lsp-wait** command without specifying any of the parameters. For example:

```
-> spb isis lsp-wait
```

## Configuring the Overload State

This implementation of ISIS-SPB supports the overload state mechanism, which allows an instance of ISIS-SPB to inform its neighbors that the instance is nearing or exceeding its capabilities. When peers see that a switch is advertising in this state, they will select an alternate path around the overloaded switch.

The ISIS-SPB instance for a switch may dynamically trigger the overload state condition when the instance detects that it is nearing or has reached resource limits. However, it is also possible to manually trigger the overload state condition using the **spb isis overload** command. For example:

```
-> spb isis overload
```

Some advantages of manually triggering the overload state condition, even if the instance is nowhere near its resource limits, are as follows:

- The switch is designated as “leaf node” that should never carry transit traffic. Configuring the link metric value for the SPB interfaces on the switch and attached peers is another method for preventing the switch from receiving transit traffic, but enabling the overload state is a much quicker way to achieve the same results and requires less configuration.
- When there is a need to remove the switch from service (temporarily or permanently). In this scenario, network availability is increased because peer switches will detect the overload state of the switch and gracefully transition to alternate paths, while the “manually overloaded” switch continues to forward packets. Just simply shutting the switch down would cause more disruption to network traffic.

When the overload state is either dynamically or manually enabled for the switch, the overload bit is set in LSP 0 to indicate that this ISIS-SPB instance is not available to accept transit traffic. However, an ISIS-SPB switch operating in the overload state is still used only if there is no alternate path to reach the intended destination.

When the overload state is enabled, the switch will operate in this state for an infinite amount of time. To configure the switch to remain in the overload state for only a specific amount of time (in seconds), use the **spb isis overload** command with the optional **timeout** parameter. For example:

```
-> spb isis overload timeout 500
```

To disable the overload state, use the **no** form of the **spb isis overload** command. For example:

```
-> no spb isis overload
```

It is also possible to specify that the overload state is enabled for the switch after every system bootup. This is done using the **spb isis overload-on-boot** command, which also has an optional **timeout** parameter. For example:

```
-> spb isis overload-on-boot timeout 500
```

To disable the overload-on-boot option, use the **no** form of the **spb isis overload-on-boot** command. For example:

```
-> no spb isis overload-on-boot timeout 500
```

Note that the **no spb isis overload** command does not disable the overload-on-bootup option.

## Configuring Redundant Switches for Graceful Restart

By default, ISIS-SPB graceful restart is disabled. When graceful restart is enabled, the switch can either be a helper (which helps a neighbor router to restart) or a restarting router, or both. When graceful restart is enabled on the switch, the helper mode is automatically enabled by default.

To configure ISIS-SPB graceful restart support on an OmniSwitch, use the **spb isis graceful-restart** command. For example, to configure graceful restart on the router, enter:

```
-> spb isis graceful-restart
```

The helper mode can be disabled on the switch with the **spb isis graceful-restart helper** command. For example, to disable the helper support for neighboring switches, enter the following:

```
-> ip isis graceful-restart helper disable
```

To disable support for graceful restart, use the **no** form of the **spb isis graceful-restart** command. For example:

```
-> no spb isis graceful-restart
```

## Enabling/Disabling ISIS-SPB

By default ISIS-SPB is disabled on the switch. To enable ISIS-SPB, use the **spb isis admin-state** command with the **enable** option. For example:

```
-> spb isis admin-state enable
```

To disable the ISIS-SPB instance on the switch, enter the **spb isis admin-state** command with the **disable** option. When the ISIS-SPB status is disabled for the switch, the related configuration settings and statistics are retained.

```
-> spb isis admin-state disable
```

---

**Note.** Enabling ISIS-SPB on a switch starts the process of ISIS-SPB discovery, adjacency building, and shortest path tree calculations. Make sure that the SPB-M configuration is set up first, then enable ISIS-SPB on each switch that will participate in the SPB-M network.

---

## Creating a SPB Service

A SPB service is identified by a service ID number, which represents an association between a backbone service instance identifier (I-SID) and an existing BVLAN. Basically, creating a SPB service binds the backbone I-SID to a BVLAN ID. All traffic mapped to the specific I-SID is then encapsulated and forwarded on the associated BVLAN to the intended destination.

The `service spb` command is used to create a SPB service. For example, the following command creates SPB service 1 and binds I-SID 100 to BVLAN 4001:

```
-> service spb 1 isid 500 bvlan 4001
```

The BVLAN ID specified with the `service spb` command must already exist in the switch configuration. However, the I-SID number specified creates a new I-SID that is bound to the BVLAN for this service.

---

**Note.** When adding another BVLAN to an existing SPB-M topology instance, create the new BVLAN and its associated ECT ID on every switch first, then configure the SPB service association for the BVLAN. Creating SPB services before the BVLAN configuration is complete on all switches can cause problems with forming adjacencies or may even cause an SPB switch to drop existing adjacencies.

---

## Modifying Default SPB Service Parameters

The following SPB service parameter values are set by default at the time the service is created. If necessary, use the specified commands to change the default values.

Parameter Description	Command	Default
Service description.	<code>service spb description</code>	None
Administrative status for statistics collection.	<code>service spb stats</code>	Disabled
Multicast replication mode	<code>service spb multicast-mode</code>	head-end
VLAN translation	<code>service spb vlan-xlation</code>	Disabled
Administrative status of the service	<code>service spb admin-state</code>	Disabled

Refer to the *OmniSwitch CLI Reference Guide* for more information about the above parameters and related commands.

## Using VLAN Translation

VLAN translation refers to the egress translation of VLAN tags on service access points (SAPs). When enabled for a service, the VLAN tags for outgoing customer frames on SAPs associated with that service are processed according to the local SAP configuration (the SAP on which the frames will egress) and not according to the configuration of the SAP on which the frames were received.

- If the local SAP is configured for untagged traffic (`slot/port:0`), the egress traffic is always sent out as untagged.
- If the local SAP is configured for 802.1q-tagged traffic (`slot/port:ctag`), the egress traffic is single-tagged with the tag value specified by the `ctag` (customer VLAN tag) value.
- If the local SAP is configured for double-tagged traffic (`slot/port:outer_tag.:inner_tag`), the egress traffic is double-tagged with the tag values specified by the `outer_tag` and `inner_tag` values.

When VLAN translation is disabled, frames simply egress without any modification of the VLAN tags. In other words, the frames are transparently bridged without tag modification.

The following table shows the required translation (tag is added or replaced) that takes place when the egress SAP configuration is applied to the possible frame types (untagged, tagged, double-tagged). Note that in this table the terms “ITAG” and “OTAG” refer to inner tag and outer tag, respectively.

Egress SAP (action required based on SAP type)			
	Untagged SAP	Single Tagged SAP	Double-Tagged SAP
<b>Incoming Frame</b>	Remove OTAG Remove ITAG	Replace OTAG (Note: Replace = implicit add) Remove ITAG	Replace OTAG (Note: Replace = implicit add) Add/Replace ITAG
Untagged	No tags, so no action taken	Add the SAP OTAG	Add the SAP OTAG Add the SAP ITAG.
Single-tagged	Remove the OTAG	Replace the OTAG	Add ITAG Replace OTAG
Double-tagged	Remove the ITAG Remove the ITAG	Remove the ITAG Replace the OTAG	Replace ITAG Replace OTAG

VLAN translation is enabled at two different levels: the service level and access port level. To enable translation at the service level, use the **service spb vlan-xlation** command. For example:

```
-> service spb 1 vlan-xlation enable
```

To enable VLAN translation for all services, use the **all** parameter with the same command. For example:

```
-> service spb all vlan-xlation enable
```

To disable VLAN translation, use the **service spb vlan-xlation** command with the disable parameter. For example:

```
-> service spb 1 vlan-xlation disable
-> service spb all vlan-xlation disable
```

To enable VLAN translation at the port level, use the **service access vlan-xlation** command. For example:

```
-> service access port 1/11 vlan-xlation enable
```

See “[Configuring Service Access Ports](#)” on page 3-39 for more information.

## Enable the Service

By default, the SPB service is disabled when the service is created. Once the service is created and any optional service parameters are configured, use the **service spb admin-state** command with the **enable** option to enable the service. For example:

```
-> service spb 1 admin-state enable
```

To disable the service, enter the following command:

```
-> service spb 1 admin-state disable
```

## Deleting a SPB Service

Before deleting a service from the switch configuration, disable the administrative status of the service. Once this is done, use the **no** form of the **service spb** command to delete the service. For example:

```
-> no service spb 1
```

## Verifying the SPB Service Configuration

To view the SPB service configuration for the switch, use the **show spb isis services** command. For example:

```
-> show service spb
```

Legend: \* denotes a dynamic object

SPB Service Info

```
SystemId : 00e0.b1e7.0188, SrcId : 0x70188, SystemName : BEB-1
```

ServiceId	Adm	Oper	Stats	SAP Count	Bind Count	Isid	BVlan	MCast Mode	(T/R)
1	Up	Up	N	4	1	1000	4001	Headend	(0/0)
2	Up	Up	N	4	1	1001	4001	Headend	(0/0)
3	Up	Up	N	4	1	1002	4001	Headend	(0/0)
4	Up	Up	N	4	1	1003	4001	Headend	(0/0)
5	Up	Up	N	4	1	1004	4001	Headend	(0/0)
6	Up	Up	N	4	1	1005	4001	Headend	(0/0)
7	Up	Up	N	4	1	1006	4001	Headend	(0/0)
8	Up	Up	N	4	1	1007	4001	Headend	(0/0)
9	Up	Up	N	4	1	1008	4001	Headend	(0/0)
10	Up	Up	N	4	1	1009	4001	Headend	(0/0)

To view the configuration for an individual service, use the **show spb isis services** command and specify the SPB service ID. For example:

```
-> show service spb 1
```

SPB Service Detailed Info

```
Service Id       : 1,
ISID             : 1000,
Multicast-Mode  : Headend,
Admin Status    : Up,
Stats Status    : No,
Service Type    : SPB,
MTU             : 9194,
SAP Count       : 4,
Ingress Pkts   : 0,
Egress Pkts    : 0,
Mgmt Change    : 08/10/2012 13:14:43,
Description     : ,
BVlan          : 4001,
Tx/Rx Bits     : 0/0,
Oper Status    : Up,
Vlan Translation : No,
Allocation Type : Static,
Def Mesh VC Id : 1,
SDP Bind Count : 1,
Ingress Bytes  : 0,
Egress Bytes   : 0,
Status Change  : 08/10/2012 13:14:00
```

## Configuring Service Access Points (SAPs)

A SAP identifies the location where customer traffic enters the Provider Backbone Bridge Network (PBBN) edge, the type of customer traffic to service, parameters to apply to the traffic, and the service that will process the traffic for tunneling through the provider network.

Configuring a SAP requires several steps. These steps are outlined here and further described throughout this section:

- Configure customer-facing ports or link aggregates as service access ports.
- Configure Layer 2 profiles to determine how control packets are processed on access ports.
- Create a SAP by associating a SAP ID with a SPB service ID. A SAP ID is comprised of an access port and an encapsulation value, which is used to identify the type of customer traffic (untagged, single-tagged, or double-tagged) to map to the associated service.

### SAP Configuration Guidelines

Consider the following when configuring a SAP:

- A SAP is a unique local entity for any given device. The same SAP ID value can be used on other BEB switches.
- There are no SAPs configured by default; explicit configuration of a SAP is required.
- A SAP is administratively disabled at the time the SAP is created.
- When a SAP is deleted, all configuration parameters for the SAP are also deleted.
- A SAP is owned by and associated with the service that was specified at the time the SAP was created.
- If a port is administratively shutdown, all SAPs on that port become operationally out of service.
- Both fixed ports and link aggregates are configurable as access ports. Only access ports are associated with SAPs.
- Bridging functionality is not supported on access ports or link aggregates.
- Configuring multiple SAPs on an access port that map different VLAN tags to the same service can cause a MAC move when the same customer MAC (CMAC) ingresses the access port with different VLAN tags. For example, a CMAC has two flows tagged with VLAN 10 and VLAN 20 ingressing access port 1/1 and both are mapped to service 100.

```
-> service spb 100 sap port 1/1:10
-> service spb 100 sap port 1/1:20
```

To avoid the MAC move in this scenario, use one of the following alternative SAP configurations.

Configure the SAPs with different services:

```
-> service spb 100 sap port 1/1:10
-> service spb 200 sap port 1/1:20
```

Configure a default SAP to classify both flows into the same service:

```
-> service spb 100 sap port 1/1:all
```

See [“Creating the Service Access Point” on page 3-41](#) for more information.



## Configuring Service Access Ports

Each SAP is comprised of an access port or link aggregate and an encapsulation type value. Access ports are customer-facing ports that reside on a provider edge router. Traffic received on these ports is classified for one or more SAPs and forwarded onto the intended destination by the associated SPB service.

To configure a port or link aggregate as an access port, use the [service access](#) command. For example, the following command configures port 1/2 and link aggregate 5 as access ports:

```
-> service access port 1/2
-> service access linkagg 5
```

To revert an access port back to a regular switch port or link aggregate, use the no form of the service access command. For example:

```
-> no service access port 1/2
-> no service access linkagg 5
```

## VLAN Translation on Access Ports

VLAN translation refers to the egress translation of VLAN tags on service access points (SAPs). For more information about VLAN translation is applied, see [“Using VLAN Translation” on page 3-35](#).

By default, VLAN translation is disabled on access ports. Enabling VLAN translation on an access port implicitly enables translation for all SAPs associated with that port. However, translation must also be enabled for the services associated with these SAPs. This ensures that all SAPs associated with a service will apply VLAN translation.

To enable VLAN translation on an access port, use the [service access vlan-xlation](#) command with the **enable** option. For example:

```
-> service access port 1/3 vlan-xlation enable
-> service access linkagg 10 vlan-xlation enable
```

To disable VLAN translation on an access port, use the [service access vlan-xlation](#) command with the **disable** option. For example:

```
-> service access port 1/3 vlan-xlation disable
-> service access linkagg 10 vlan-xlation disable
```

## Configuring Layer 2 Profiles for Access Ports

A Layer 2 profile determines how control frames ingressing on an access port are processed. When a port is configured as an access port, a default Layer 2 profile (**def-access-profile**) is applied to the port with the following default values for processing control frames:

Protocol	Default
STP	tunnel
802.1x	drop
802.1ab	drop
802.3ad	peer
GVRP	tunnel
MVRP	tunnel

Protocol	Default
AMAP	discard
CISCO PDU	discard
CISCO VLAN	discard
CISCO uplink	discard

If the default profile values are not sufficient, use the **service l2profile** command with the **tunnel**, **discard**, and **peer** options to create a new profile. For example, the following command creates a profile named “DropL2”:

```
-> service l2profile DropL2 stp discard gvrp discard 802.1ab discard
```

Consider the following when configuring Layer 2 profiles:

- Not all of the control protocols are currently supported with the **peer**, **tunnel**, and **discard** parameters. Use the following table to determine the parameter combinations that are supported:

Protocol	Reserved MAC	peer	discard	tunnel
STP	01-80-C2-00-00-00	no	yes	yes
802.1x	01-80-C2-00-00-03	no	yes	yes
802.1ab	01-80-C2-00-00-0E	yes	yes	yes
802.3ad	01-80-C2-00-00-02	yes	no	no
GVRP	01-80-C2-00-00-21	no	yes	yes
MVRP	—	no	yes	yes
AMAP	00-20-DA-00-70-04	yes	yes	no
CISCO PDU	01-00-0C-CC-CC-CD	no	yes	yes
CISCO VLAN	01-00-0C-CC-CD-CE	no	yes	yes
CISCO uplink	01-00-0C-CC-CD-CF	no	yes	yes

- When a profile is created, the new profile inherits the default profile settings for processing control frames. The default settings are applied with the new profile unless they are explicitly changed. For example, the profile “DropL2” was configured to discard STP, GVRP, and 802.1ab frames. No other protocol settings were changed, so the default settings still apply for the other protocols.
- Remove any profile associations with access ports before attempting to modify or delete the profile.

To delete a Layer 2 profile, use the **no** form of the **service l2profile** command. For example, the following command deletes the “DropL2” profile:

```
-> no service l2profile DropL2
```

Use the **show service l2profile** command to view a list of profiles that are already configured for the switch. This command also displays the attribute values for each profile.

## Assigning Layer 2 Profiles to Access Ports

After a Layer 2 profile is created, it is then necessary to assign the profile to an access port or link aggregate. When this is done, the current profile associated with the port is replaced with the new profile.

The `service access l2profile` command is used to assign a new profile to an access port. For example, the following command assigns the “DropL2” profile to access port 1/4:

```
-> service access port 1/4 l2profile DropL2
-> service access linkagg 5 l2profile DropL2
```

To change the profile associated with the access port back to the default profile (`def-access-profile`), use the `default` option with the `service access l2profile` command. For example:

```
-> service port 1/4 l2profile default
-> service access linkagg 5 l2profile DropL2
```

Use the `show service access` command to display profile associations for access ports.

## Verifying the Access Port Configuration

To view the access port configuration for the switch, use the `show service access` command. For example:

```
-> show service access
Port      Link  SAP      SAP      Vlan
Id        Status Type     Count    Xlation L2Profile
-----+-----+-----+-----+-----+-----
1/11      Up    Manual   100      N       def-access-profile
1/12      Down  Manual   100      N       def-access-profile
1/13      Down  Manual   100      N       def-access-profile
1/14      Down  Manual   100      N       def-access-profile
1/15      Down  Dynamic  0        Y       def-access-profile
1/16      Up    Dynamic  1        Y       def-access-profile
1/17      Down  Dynamic  0        Y       def-access-profile
1/18      Down  Dynamic  0        Y       def-access-profile
```

```
Total Access Ports: 8
```

## Creating the Service Access Point

Each SPB service is bound to at least one Service Access Point (SAP). A SAP identifies the point at which customer traffic enters the Provider Backbone Bridge Network (PBBN). Creating a SAP on a SPB switch designates that switch as a Backbone Edge Bridge (BEB) in the PBBN. An SPB switch that does not have a SAP but does have at least one BVLAN and an SPB interface is designated as Backbone Core Bridge (BCB) in the PBBN.

Once the SPB topology is determined and switches that will serve as BEBs are identified, a SAP is created on each BEB. A SAP is created by associating a SAP ID with a SPB service. A SAP ID is comprised of a customer-facing port (referred to as an access port) and an encapsulation value that is used to identify the type of customer traffic (untagged, single-tagged, or double-tagged) to map to the associated service.

The `service spb sap` command is used to configure a SAP. This command specifies the SPB service ID number and the SAP ID (slot/port:encapsulation). The following parameter values are used with this command to specify the encapsulation value:

SAP Encapsulation Value	Customer Traffic Serviced
<b>0 (null)</b>	All untagged packets; tagged packets are dropped.
<b>all</b>	All tagged and untagged packets not already classified into another SAP*
<i>qtag</i>	Only traffic 802.1q-tagged with the specified VLAN ID.
<i>outer_qtag.innnger_qtag</i>	Only traffic double-tagged (QinQ) with the specified outer and inner VLAN IDs.

\*Note that the **:all** (wildcard) parameter is also configurable as the inner tag value for double-tagged frames (for example, “10:all” specifies double-tagged packets with an outer tag equal to 10 and an inner tag with any value).

The following **service spb sap** command example creates a SAP that will direct customer traffic ingress-ing on access port 1/4 that is tagged with VLAN ID 50 to service 100:

```
-> service spb 100 sap 1/4:50 description "BEB1 to SPB100 CVLAN 50"
```

In the above example, the 1/4:50 designation is referred to as the SAP ID or the encapsulation ID. This means that if no other SAPs are configured for port 1/4, then any traffic ingressing on that port is dropped if the traffic is not tagged with VLAN 50.

It is possible to configure more than one SAP for the same access port, which provides a method for segregating incoming traffic into multiple services. For example, the following SAP configuration for port 2/3 sends incoming traffic to three different services based on the VLAN tags of the frames received:

```
-> service spb 2000 sap port 2/3:all
-> service spb 200 sap port 2/3:100
-> service spb 1000 sap port 2/3:100.200
```

In this example,

- Frames double-tagged with 100 (outer tag) and 200 (inner tag) are sent on service 1000.
- Frames single-tagged with VLAN 100 are sent on service 200.
- All other frames (those that are not single-tagged with 100 or double-tagged with 100 and 200) are sent on service 2000.

The following SAP ID classification precedence is applied when there are multiple SAPs for one access port:

- 1** Double-tagged (Outer VLAN + Inner VLAN) - Highest
- 2** Double-tagged (Outer VLAN + all)
- 3** Single-tagged (VLAN)
- 4** Single-tagged (wildcard)
- 5** Untagged - Lowest.

## Modifying Default SAP Parameters

The following parameter values are set by default at the time the SAP is created. If necessary, use the specified commands to change the default values.

Parameter Description	Command	Default
SAP description.	<code>service spb sap description</code>	None
SAP trust mode	<code>service spb sap trusted</code>	Trusted
Administrative status for the SAP	<code>service spb sap admin-state</code>	Disabled
Administrative status for statistics collection.	<code>service spb sap stats</code>	Disabled

Refer to the *OmniSwitch CLI Reference Guide* for more information about the command parameters.

## Configuring the SAP Trust Mode

The `service spb sap trusted` command is used to configure the trust mode for a SAP. A trusted SAP can accept 802.1p values in incoming packets; an untrusted SAP will set any 802.1p values to zero in incoming packets, unless an 802.1p value is configured with this command.

Note that untagged Layer 2 control packets (for example, BPDU, GVRP, and AMAP) are always tunneled (if enabled) through the Provider Backbone Bridge (PBB) network with the default EXP bits set to 7, so that they can arrive at the destination bridge at the highest COS queue of 7. As a result, trusted and untrusted SAPs configured on the access ports will not affect the Layer 2 control packets ingressing on the access ports.

By default, a SAP is trusted with the priority set to best effort (zero). Use the **no** form of the `service spb sap trusted` command with the **priority** option to change the SAP mode to untrusted. For example:

```
-> service spb 100 sap 1/4:50 no trusted priority 7
```

When a SAP is trusted, the priority value contained in tagged customer packets is used; untagged packets are assigned the default priority value (zero). When a SAP is untrusted, the priority value configured for the SAP is assigned to both tagged and untagged customer packets.

## Enabling/Disabling the SAP

By default, a SAP is disabled at the time the SAP is created. To enable the SAP administrative status, use the `service spb sap admin-state` command. For example:

```
-> service spb 100 sap port 1/4:50 admin-state enable
-> service spb 200 sap linkagg 5:all admin-state enable
```

To disable the SAP, enter the following command:

```
-> service spb 100 sap port 1/4:50 admin-state disable
-> service spb 200 sap linkagg 5:all admin-state disable
```

## Deleting the SAP

When a SAP is administratively disabled, the SAP configuration is not removed from the switch. To delete a SAP from the switch configuration, use the **no** form of the `service spb sap` command. For example:

```
-> service spb 100 no sap port 1/4:50
-> service spb 200 no sap linkagg 5:all
```

## Verifying the SAP Configuration

A SAP is a type of virtual port that is associated with a SPB service. To determine the SAP configuration for a specific service, use the **show service spb ports** command to view the virtual ports associated with a specific service. For example:

```
-> show service spb 1 ports
Legend: * denotes a dynamic object
SPB Service Info
  Admin : Up, Oper  : Up, Stats      : N, Mtu      : 9194,  VlanXlation : N,
  ISID  : 1000,     BVlan : 4001,   MCast-Mode : Headend, Tx/Rx    : 0/0

Identifier      Adm  Oper  Stats  Sdp SystemId:BVlan  Intf  Sap Description /
-----+-----+-----+-----+-----+-----+-----+-----
sap:1/11:1000   Up   Up    N      Y:x                1/11  -
sap:1/12:1000   Up   Down  N      Y:x                1/12  -
sap:1/13:1000   Up   Down  N      Y:x                1/13  -
sap:1/14:1000   Up   Down  N      Y:x                1/14  -
sdp:32776:1*    Up   Up    Y      e8e7.3233.1831:4001 1/1    BEB-1
```

Total Ports: 5

To then view configuration information for a specific SAP, use the **show service spb sap** command. For example:

```
-> show service spb 1 sap port 1/11:1000
SAP Detailed Info
  SAP Id          : 1/11:1000,      Description      : ,
  Admin Status    : Up,            Oper Status     : Up,
  Stats Status    : No,            Vlan Translation : No,
  Service Type    : SPB,           Allocation Type  : Static,
  Trusted         : Yes,           Priority         : 0,
  Ingress Pkts   : 0,              Ingress Bytes   : 0,
  Egress Pkts    : 0,              Egress Bytes    : 0,
  Mgmt Change     : 08/07/2012 23:39:29, Status Change   : 08/10/2012 15:13:08
```

## Configuring IP over SPB

As described in section “[IP over SPB-M](#)” on page 3-15, there are two approaches for routing L3 traffic over a L2 SPB-M backbone network: VPN-Lite and L3 VPN. The following general steps are used to configure each one of these solutions:

**1 Configure the L3 VPN loopback for both VPN-Lite and L3 VPN scenarios.** Identify the BEBs that will participate in routing L3 traffic through the SPB-M core. On each of these BEBs, configure the required loopback port configuration. For example:

```
-> vlan 200
-> vlan 200 members 1/1 tagged
-> spb bvlan 500
-> service access port 1/2
-> service spb 1000 isid 1000 bvlan 500 admin-state enable
-> service spb 1000 sap port 1/2:200
-> vrf-1
-> vrf-1 ip interface L3vpn1 vlan 200 address 10.1.1.1/24
```

In this example, VLAN 200, port 1/1, access port 1/2, and IP interface “L3vpn1” will serve as the L3 VPN components required for the loopback configuration. A physical cable will connect port 1/1 to access port 1/2. The SPB BVLAN and I-SID are required to create the SAP for access port 1/2.

**2 Configure VPN-Lite routing protocols.** If using the VPN-Lite approach, configure the routing protocols on the “L3vpn1” IP interface created in Step 1. Otherwise, skip Step 2 and go to Step 3. For example:

```
-> vrf-1 ip static-route 20.0.0.0/24 gateway 10.1.1.2
```

The VPN-Lite approach only requires configuring the physical L3 VPN loopback and configuring routing protocols on the L3 VPN IP interface to exchange routes. The remaining steps in this section are used to configure the L3 VPN approach, which uses ISIS-SPB to exchange routes.

**3 Configure VRF-ISID bindings for the L3 VPN.** A VRF-ISID binding identifies the loopback port configuration that will do L3 forwarding on the VRF side of the loopback and SPB bridging on the SAP side of the loopback. VRF import and export commands are used to exchange routes between the VRF and I-SID specified in the binding configuration. For example:

```
-> spb ipvpn bind vrf-1 isid 1000 gateway 10.1.1.1 all-routes
-> vrf-1 ip export all-routes
-> vrf-1 ip import isid 1000 all-routes
```

In this example, “vrf-1” is bound to SPB I-SID 1000 and gateway 10.1.1.1 identifies the loopback IP interface. All routes in “vrf-1” are exported to the Global Route Manager (GRM), which then exports the routes to I-SID 1000. The last command in this sequence sets up the import of I-SID 1000 routes from the GRM into “vrf-1”. The **all-routes** parameter specifies that no route-map filtering is applied to exported or imported routes; all routes are allowed.

**4 Optionally configure route redistribution between VRFs and/or I-SIDs.** Route redistribution is configurable between a VRF and I-SID or between two I-SIDs (inter-I-SID route leaking). For example:

```
-> spb ipvpn redistrib source-isid 2000 destination-isid 1000 all-routes
-> spb ipvpn redistrib source-vrf-2 destination-isid 2000 all-routes
```

## Verifying L3 VPN Configuration and Routes

VRFs are bound to I-SIDs to identify a VRF mapping to a specific SPB service instance for the purposes of exchanging routes between the VRF and I-SID via the switch GRM. To verify the VRF mapping configuration on the local switch, use the **show spb ipvpn bind** command. For example:

```
-> show spb ipvpn bind
Legend: * indicates bind entry is active
SPB IPVPN Bind Table:
-----+-----+-----+-----
      VRF              ISID              Gateway              Route-Map
-----+-----+-----+-----
* vrf-1                1000                1.1.1.2
* vrf-2                2000                2.2.2.2
```

Total Bind Entries: 2

In addition to exchanging routes between VRFs and I-SIDs, it is also possible to configure redistribution of routes between two I-SIDs or between a VRF and an I-SID. To verify the redistribution configuration for L3 VPN routes, use the **show spb ipvpn redistrib** command. For example:

```
-> show spb ipvpn redistrib
Legend: * indicates redistrib entry is active
SPB IPVPN Redistrib ISID Table:
      Source-ISID              Destination-ISID              Route-Map
```

```

-----+-----+-----
* 4001                4003
* 4003                4001

```

Total Redist ISID Entries: 2

To display the L3 VPN route table, use the [show spb ipvpn route-table](#). For example:

```
-> show spb ipvpn route-table
```

Legend: \* indicates IPVPN route has matching locally configured ISID

SPB IPVPN Route Table:

	ISID	Destination	Gateway	Source Bridge (Name : BMAC)	Metric
*	4001	1.1.1.0/24	1.1.1.1	L2-DUT1 : 00:e0:b1:db:c3:65	1
*	4001	1.1.1.0/24	1.1.1.2	L2-DEV1 : e8:e7:32:00:23:f9	1
*	4001	2.2.2.0/24	1.1.1.2	L2-DEV1 : e8:e7:32:00:23:f9	1
*	4001	10.10.10.0/24	1.1.1.1	L2-DUT1 : 00:e0:b1:db:c3:65	1
*	4001	15.1.1.0/24	1.1.1.1	L2-DUT1 : 00:e0:b1:db:c3:65	1
*	4003	1.1.1.0/24	2.2.2.2	L2-DEV1 : e8:e7:32:00:23:f9	1
*	4003	2.2.2.0/24	2.2.2.1	L2-DUT2 : 00:e0:b1:dd:99:db	1
*	4003	2.2.2.0/24	2.2.2.2	L2-DEV1 : e8:e7:32:00:23:f9	1
*	4003	10.10.10.0/24	2.2.2.2	L2-DEV1 : e8:e7:32:00:23:f9	1
*	4003	15.1.1.0/24	2.2.2.2	L2-DEV1 : e8:e7:32:00:23:f9	1

## IP over SPB Configuration Examples

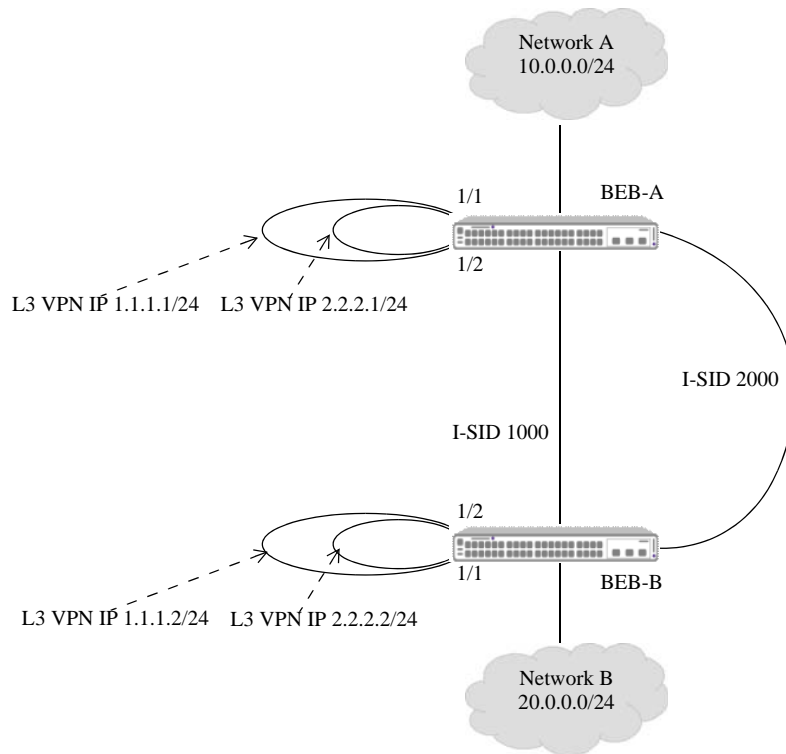
This section contains diagrams and CLI command examples for configuring the following IP over SPB scenarios:

- [“Multiple I-SIDs” on page 3-47.](#)
- [“I-SID Routing in One VRF” on page 3-49.](#)
- [“I-SID Routing in Two VRFs” on page 3-51.](#)



## Multiple I-SIDs

In this sample IP over SPB topology, Network A is able to communicate with Network B across two I-SIDs. This scenario could be expanded to connect multiple customer sites together to form a VPN cloud.



**Figure 7: Multiple I-SIDs**

The following CLI command examples are used to configure the sample IP over SPB topology shown in “[Figure 7: Multiple I-SIDs](#)” on page 3-47. In this topology, port 1/1 is the L3 VPN router port, port 1/2 is the L3 VPN access port, and VLAN 200 and VLAN 400 are the L3 VPN VLANs.

Common for BEB-A and BEB-B:

```
-> service access port 1/2
-> vlan 200
-> vlan 200 members port 1/1 tagged
-> vlan 400
-> vlan 400 members port 1/1 tagged
-> service spb 1000 isid 1000 bvlan 40 admin-state enable
-> service spb 1000 sap port 1/2:200 admin-state enable
-> service spb 2000 isid 2000 bvlan 41 admin-state enable
-> service spb 2000 sap port 1/2:400 admin-state enable
-> vrf-1
```

BEB-A:

```
-> vrf-1 ip interface l3vpn1 vlan 200 address 1.1.1.1/24
-> vrf-1 ip interface l3vpn2 vlan 400 address 2.2.2.1/24
```

BEB-B:

```
-> vrf-1 ip interface l3vpn1 vlan 200 address 1.1.1.2/24
-> vrf-1 ip interface l3vpn2 vlan 400 address 2.2.2.2/24
```

### VPN-Lite

The following commands are used only in a VPN-Lite configuration:

**BEB-A:**

```
-> vrf-1 ip static-route 20.0.0.0/24 gateway 1.1.1.2
-> vrf-1 ip static-route 20.0.0.0/24 gateway 2.2.2.2
```

**BEB-B:**

```
-> vrf-1 ip static-route 10.0.0.0/24 gateway 1.1.1.1
-> vrf-1 ip static-route 10.0.0.0/24 gateway 2.2.2.1
```

**L3 VPN**

The following commands are used only in a L3 VPN configuration:

**BEB-A:**

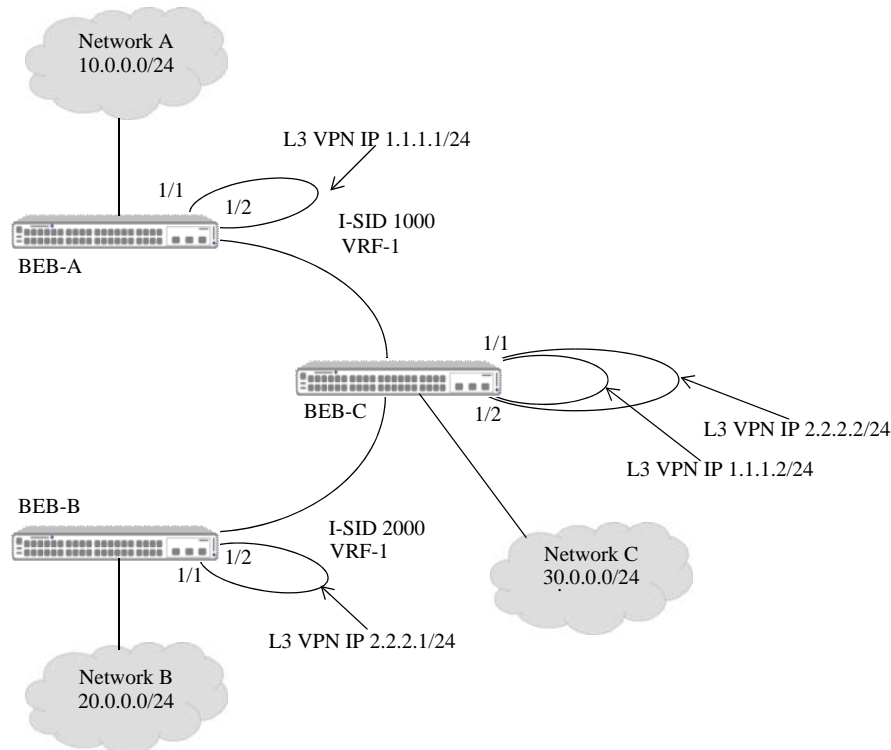
```
-> vrf-1 ip export all-routes
-> spb ipvpn bind vrf-1 isid 1000 gateway 1.1.1.1 all-routes
-> spb ipvpn bind vrf-1 isid 2000 gateway 2.2.2.1 all-routes
-> vrf-1 ip import isid 1000 all-routes
-> vrf-1 ip import isid 2000 all-routes
```

**BEB-B:**

```
-> vrf-1 ip export all-routes
-> spb ipvpn bind vrf-1 isid 1000 gateway 1.1.1.2 all-routes
-> spb ipvpn bind vrf-1 isid 2000 gateway 2.2.2.2 all-routes
-> vrf-1 ip import isid 1000 all-routes
-> vrf-1 ip import isid 2000 all-routes
```

## I-SID Routing in One VRF

In this sample IP over SPB configuration, Networks A and B can communicate with each other within the same VRF but on different I-SIDs due to the routing (redistribution) between I-SID 1000 and 2000 on BEB-C. In addition, Network C is also able to communicate with Networks A and B.



**Figure 8: Inter-ISID Routing Example (One VRF)**

The following CLI command examples are used to configure the sample IP over SPB topology shown in “[Figure 8: Inter-ISID Routing Example \(One VRF\)](#)” on page 3-49. In this topology, Network A binds to I-SID 1000 in VRF-1, Network B binds to I-SID 2000 in VRF-1, and Network C binds to both I-SIDs in VRF-1.

### BEB-A:

```
-> service access port 1/2
-> vlan 200
-> vlan 200 members port 1/1 tagged
-> service spb 1000 isid 1000 bvlan 40 admin-state enable
-> service spb 1000 sap port 1/2:200 admin-state enable
-> vrf-1
-> vrf-1 ip interface l3vpn1 vlan 200 address 1.1.1.1/24
-> vrf-1 ip export all-routes
-> spb ipvpn bind vrf-1 isid 1000 gateway 1.1.1.1 all-routes
-> vrf-1 ip import isid 1000 all-routes
```

### BEB-B

```
-> service access port 1/2
-> vlan 400
-> vlan 400 members port 1/1 tagged
-> service spb 2000 isid 2000 bvlan 41 admin-state enable
-> service spb 2000 sap port 1/2:400 admin-state enable
```

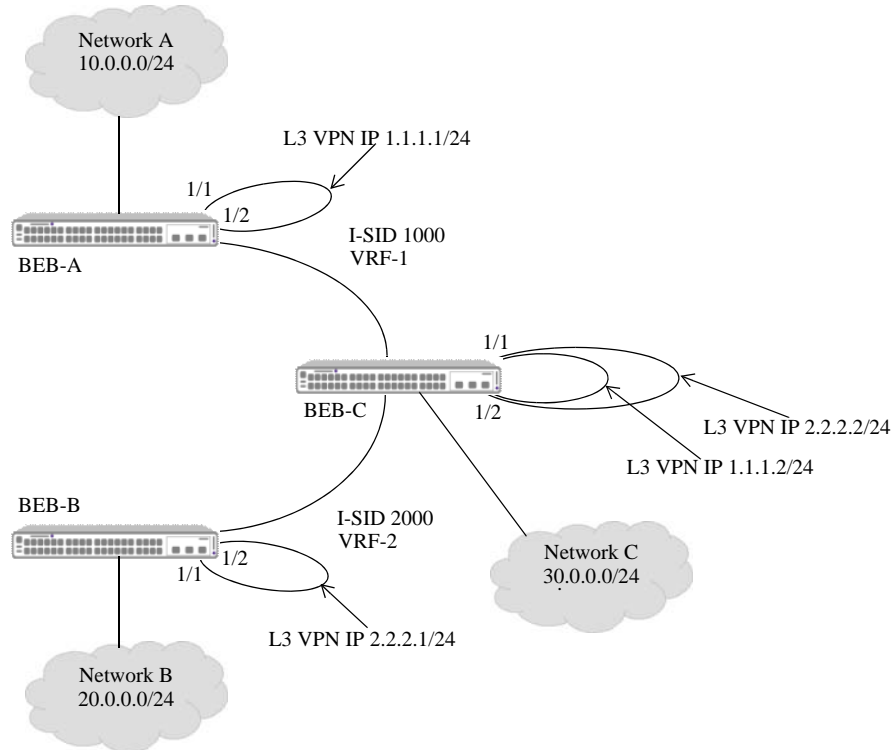
```
-> vrf-1
-> vrf-1 ip interface l3vpn2 vlan 400 address 2.2.2.1/24
-> vrf-1 ip export all-routes
-> spb ipvpn bind vrf-1 isid 2000 gateway 2.2.2.1 all-routes
-> vrf-1 ip import isid 2000 all-routes
```

**BEB-C:**

```
-> service access port 1/2
-> vlan 200
-> vlan 200 members port 1/1 tagged
-> vlan 400
-> vlan 400 members port 1/1 tagged
-> service spb 1000 isid 1000 bvlan 40 admin-state enable
-> service spb 1000 sap port 1/2:200 admin-state enable
-> service spb 2000 isid 2000 bvlan 41 admin-state enable
-> service spb 2000 sap port 1/2:400 admin-state enable
-> vrf-1
-> vrf-1 ip interface l3vpn1 vlan 200 address 1.1.1.2/24
-> vrf-1 ip interface l3vpn2 vlan 400 address 2.2.2.2/24
-> spb ipvpn bind vrf-1 isid 1000 gateway 1.1.1.2 all-routes
-> spb ipvpn bind vrf-1 isid 2000 gateway 2.2.2.2 all-routes
-> spb ipvpn redist source-isid 1000 destination-isid 2000 all-routes
-> spb ipvpn redist source-isid 2000 destination-isid 1000 all-routes
-> vrf-1 ip import all-routes
-> vrf-1 ip export all-routes
```

## I-SID Routing in Two VRFs

In this sample IP over SPB configuration, Networks A and B can communicate with each other between two different VRFs on different I-SIDs due to the routing (redistribution) between I-SID 1000 and 2000 on BEB-C. In addition, Network C is also able to communicate with Networks A and B.



**Figure 9: Inter-ISID Routing Example (Two VRF)**

The following CLI command examples are used to configure the sample IP over SPB topology shown in “[Figure 9: Inter-ISID Routing Example \(Two VRF\)](#)” on page 3-51. In this topology, Network A binds to I-SID 1000 in VRF-1, Network B binds to I-SID 2000 in VRF-2, and Network C binds to both I-SIDs in VRF-1.

### BEB-A:

```
-> service access port 1/2
-> vlan 200
-> vlan 200 members port 1/1 tagged
-> service spb 1000 isid 1000 bvlan 40 admin-state enable
-> service spb 1000 sap port 1/2:200 admin-state enable
-> vrf-1
-> vrf-1 ip interface l3vpn1 vlan 200 address 1.1.1.1/24
-> vrf-1 ip export all-routes
-> spb ipvpn bind vrf-1 isid 1000 gateway 1.1.1.1 all-routes
-> vrf-1 ip import isid 1000 all-routes
```

### BEB-B

```
-> service access port 1/2
-> vlan 400
-> vlan 400 members port 1/1 tagged
-> service spb 2000 isid 2000 bvlan 41 admin-state enable
-> service spb 2000 sap port 1/2:400 admin-state enable
```

```
-> vrf-2
-> vrf-2 ip interface l3vpn2 vlan 400 address 2.2.2.1/24
-> vrf-2 ip export all-routes
-> spb ipvpn bind vrf-1 isid 2000 gateway 2.2.2.1 all-routes
-> vrf-2 ip import isid 2000 all-routes
```

**BEB-C:**

```
-> service access port 1/2
-> vlan 200
-> vlan 200 members port 1/1 tagged
-> vlan 400
-> vlan 400 members port 1/1 tagged
-> service spb 1000 isid 1000 bvlan 40 admin-state enable
-> service spb 1000 sap port 1/2:200 admin-state enable
-> service spb 2000 isid 2000 bvlan 41 admin-state enable
-> service spb 2000 sap port 1/2:400 admin-state enable
-> vrf-1
-> vrf-1 ip interface l3vpn1 vlan 200 address 1.1.1.2/24
-> vrf-1 ip interface l3vpn2 vlan 400 address 2.2.2.2/24
-> spb ipvpn bind vrf-1 isid 1000 gateway 1.1.1.2 all-routes
-> spb ipvpn bind vrf-2 isid 2000 gateway 2.2.2.2 all-routes
-> spb ipvpn redist source-isid 1000 destination-isid 2000 all-routes
-> spb ipvpn redist source-isid 2000 destination-isid 1000 all-routes
-> spb ipvpn redist source-vrf vrf-1 destination-isid 2000 all-routes
-> vrf-1 ip import isid 1000 all-routes
-> vrf-1 ip import isid 2000 all-routes
-> vrf-1 ip export all-routes
-> vrf-2 ip import vrf-1 all-routes
-> vrf-2 ip import isid 1000 all-routes
-> vrf-2 ip import isid 2000 all-routes
```

# Verifying the SPB Backbone and Services

Displaying the SPB-M configuration is helpful to verify the actual configuration on each SPB switch in the topology and to troubleshoot ISIS-SPB backbone and SPB service connectivity.

## Verifying the ISIS-SPB Backbone Configuration

To display information about the ISIS-SPB infrastructure (backbone), use the **show** commands listed in this section.

<b>show spb isis info</b>	Displays the global status and configuration for the ISIS-SPB instance on the switch.
<b>show spb isis bvlan</b>	Displays the backbone VLAN (BVLAN) configuration for the switch.
<b>show spb isis interface</b>	Displays the SPB interface (network port) configuration for the switch.
<b>show spb isis adjacency</b>	Displays information about the ISIS-SPB adjacencies created for the SPB switch.
<b>show spb isis database</b>	Displays ISIS-SPB topology information maintained in the link state database (LSDB).
<b>show spb isis nodes</b>	Displays the discovered node-level parameter values for all of the ISIS-SPB switches participating in the topology.
<b>show spb isis unicast-table</b>	Displays the unicast forwarding information for the BVLAN topology.
<b>show spb isis services</b>	Displays a network-wide view of existing services to help verify that SPB services are correctly advertised and learned by ISIS-SPB
<b>show spb isis spf</b>	Displays the shortest path first (SPF) information to all known SPB switches for a specific BVLAN.
<b>show spb isis multicast-table</b>	Displays the multicast forwarding entries for services.
<b>show spb isis multicast-sources</b>	Displays all the known multicast sources across the SPB domain and BVLANs.
<b>show spb isis multicast-sources-spf</b>	Displays the shortest path first (SPF) readability for a known multicast source bridge for a specific BVLAN.
<b>show spb isis ingress-mac-filter</b>	Displays the ingress MAC filter for multicast traffic for a given BVLAN operating in the (*,G) mode.

## Verifying the SPB Service Configuration

To display information about the Service Manager configuration for SPB service connectivity, use the **show** commands listed in this section

<b>show service access</b>	Displays the service access (customer-facing) port configuration.
<b>show service l2profile</b>	Displays the Layer 2 profile definitions. These profiles are applied to service access ports to determine how Layer 2 control protocol frames are processed on these ports.
<b>show service</b>	Displays the service configuration.
<b>show service spb ports</b>	Displays all the virtual ports (SAPs, SDPs) that are associated with an SPB service.

---

<b>show service spb sap</b>	Displays the configuration information for the specified SAP ID associated with the specified service.
<b>show service sdp</b>	Displays the dynamic Service Distribution Point (SDP) configuration.
<b>show service mesh-sdp</b>	Displays the dynamic SDP-to-service binding configuration.
<b>show service spb debug-info</b>	Displays debug information for the virtual ports associated with the SPB service.
<b>show service spb counters</b>	Displays the traffic statistics for the specified SPB service and associated virtual ports.
<b>clear service spb counters</b>	Clears the traffic statistics for the specified SPB service and associated virtual ports.

For more information about the resulting displays from these commands, see the *OmniSwitch CLI Reference Guide*.



# 4 Configuring FIP Snooping

The OmniSwitch implementation of Fibre Channel over Ethernet (FCoE) Initiation Protocol (FIP) snooping supports the FCoE technology used to tunnel Fibre Channel frames encapsulated within Ethernet MAC frames. When FIP snooping functionality is configured and enabled, the OmniSwitch serves as an FCoE transit switch. In this role, the OmniSwitch implementation of Data Center Bridging (DCB) is also used to provide the lossless Ethernet network required to support FCoE.

This implementation of FIP Snooping ensures the security of the FCoE network and maintains a virtual point-to-point network connection between FCoE Nodes (ENodes) and FCoE Forwarder (FCF) devices. An FCoE OmniSwitch is placed between ENodes (FCoE-capable servers) and an FCF to extend the reach of the FCoE network without extending the physical Fibre Channel (FC) connections.

## In This Chapter

This chapter describes FCoE and FIP in general and how FIP snooping VLAN and port configurations are applied to the switch. It provides information about configuring FIP snooping through the Command Line Interface (CLI). CLI commands are used in the configuration examples; for more details about the syntax of commands, see the *OmniSwitch CLI Reference Guide*.

The following topics and procedures are included in this chapter:

- [“FIP Snooping Specifications” on page 4-2.](#)
- [“FIP Snooping Defaults” on page 4-3.](#)
- [“Terms and Definitions” on page 4-4.](#)
- [“FIP Snooping Overview” on page 4-5.](#)
- [“Interaction with Other Features” on page 4-12.](#)
- [“Configuring FIP Snooping” on page 4-15.](#)
- [“Configuration Example” on page 4-23.](#)
- [“Verifying the FIP Snooping Configuration” on page 4-25.](#)

# FIP Snooping Specifications

The FIP Snooping and FCoE functionality described in this chapter is supported on the OmniSwitch 10K and OmniSwitch 6900, unless otherwise stated in the following specifications table or specifically noted within any other section of this chapter. Note that any maximum limits provided in this table are subject to available system resources.

Platforms Supported	OmniSwitch 10K, 6900
OmniSwitch Software License	Data Center
INCITS Standards Supported	<ul style="list-style-type: none"><li>• T11 Fibre Channel Backbone - 5 (FC-BB-5) Rev 2.00 June 4, 2009</li><li>• FC-BB-5 Annex C: Increasing FC-BB_E Robustness Using Access Control Lists</li><li>• T11 Switch Fabric - 5 (FC-SW-5) Rev 8.5 June 3, 2009</li></ul>
Maximum number of FIP Snooping Sessions	128
Required port types	10G or faster Ethernet with DCB profile and DCBx enabled with PFC/ETS active (ports and link aggregates)

## FIP Snooping Defaults

By default, FIP snooping functionality is disabled on the switch. When FIP snooping is enabled, the following global default values are applied:

Parameter Description	Command	Default
FCoE Initialization Protocol (FIP) addressing mode.	<b>fcoe address-mode</b>	FPMA (fabric-provided MAC address)
The 802.1p priority value assigned to FCoE traffic.	<b>fcoe priority</b>	3
Whether or not only FCoE traffic marked with the FCoE priority value is allowed.	<b>fcoe priority-protection</b>	Disabled
Whether or not FCoE traffic marked with a different priority is dropped or remarked (only applied when priority protection is enabled).	<b>fcoe priority-protection action</b>	Dropped
Trap threshold value to indicate that a percentage of filtering resources has been used.	<b>fcoe filtering-resource trap-threshold</b>	80%
The amount of time the switch waits for keep alive messages from ENodes and FCFs before clearing a FIP session.	<b>fcoe house-keeping-time-period</b>	300 seconds

## Terms and Definitions

The OmniSwitch implementation of FIP snooping and DCB allows the switch to operate as an FCoE transit switch in an FCoE network. The following terms and definitions provided in this section describe FCoE network components discussed in this chapter.

- **CNA (Converged Network Adapter):** A network interface card (NIC) that combines the functionality of a FC host bus adapter (HBA) and a lossless Ethernet network interface card (NIC). A CNA is used to connect servers to a FC storage area network (SAN) and an Ethernet network.
- **DCB (Data Center Bridging):** A set of standards that define protocols to extend the capabilities of Ethernet to support the convergence of data and storage in the data center. DCB is required to provide the lossless Ethernet transport required to support FCoE (see [Chapter 2, “Configuring Data Center Bridging,”](#) for more information).
- **ENode (FCoE Node):** The FCoE functionality of a CNA that instantiates virtual FC interfaces (VN\_Ports) to connect to virtual FC interfaces (VF\_Ports) on an FCF switch. The virtual links established between the ENode and FCF allows FCoE servers attached to an Ethernet network to access storage devices in the SAN.
- **Fibre Channel (FC):** A high-speed network technology primarily used for storage networking in the data center. FC is used to pool disk storage for all hosts in the network, which helps to provide a centralized storage solution that is easier to manage and more reliable.
- **FCoE (Fibre Channel over Ethernet):** A standard (T11 FC-BB-5) that defines a virtual FC link topology and encapsulation for transporting native FC frames over an Ethernet network. FCoE frames are identified by Ethertype 0x8906 to differentiate them from other traffic on the same network.
- **FCF (FCoE Forwarder):** A switch capable of transmitting native FC frames and FCoE frames. The FCF sits at the edge of a FC SAN to translate and forward traffic between an FCoE network and the FC SAN. In an OmniSwitch FCoE network configuration, a third-party FCF is required.
- **FCoE Initialization Protocol (FIP):** An Ethernet Layer 2 control protocol that establishes and maintains virtual FC links between pairs of ENodes and FCFs over the FCoE network. FIP control frames do not carry any payload and are identified by Ethertype 0x8914 to differentiate them from FCoE data frames and other Ethernet traffic on the same network.
- **FIP Snooping:** A security mechanism implemented on FCoE transit switches. FIP snooping inspects FIP frames received on transit switch ports and then dynamically builds ACLs based on the data in those frames. The ACLs filter FIP frames received on FCoE ports associated with FCoE VLANs and will only allow traffic from FCoE sources that successfully logged into the FCoE fabric.
- **FCoE Transit Switch:** A Layer 2 DCB switch capable of transporting encapsulated FCoE frames. May also use FIP snooping to secure FCoE traffic.
- **FCoE VLAN:** A type of VLAN that is dedicated to carrying only FCoE and FIP traffic. ENodes use this type of VLAN to transmit and receive FCoE and FIP traffic and establish virtual links to the FCF. Only FCoE ports are tagged with this VLAN and all other non-FCoE traffic should be segregated into other VLAN types.
- **VF\_Port (Virtual Fabric Port):** The virtual FCoE counterpart to a F\_Port in a native FC network. An FCF switch instantiates VF\_Ports to connect with ENode VN\_Ports to establish a virtual point-to-point link between the ENode and FCF.
- **VN\_Port (Virtual Node Port):** The virtual FCoE counterpart to an N\_port in a native FC network. ENodes instantiate VN\_Ports to connect with FCF VF\_Ports to establish a virtual point-to-point link between the ENode and FCF.

# FIP Snooping Overview

FCoE transit switches are used to extend the reach of an FCoE network without having to use additional, more expensive Fibre Channel (FC) equipment that is usually required to extend the native FC network. The OmniSwitch can operate as an FCoE transit switch within a multi-hop FCoE network.

To provide the necessary FCoE transit switch functionality, the OmniSwitch supports FCoE Initialization Protocol (FIP) snooping and Data Center Bridging (DCB) protocols. Both FIP snooping and DCB are required to provide a secure, lossless Ethernet fabric for FCoE traffic. A transit switch is basically a Layer 2 DCB switch that bridges encapsulated FCoE traffic over the Ethernet fabric between FCoE end devices.

FCoE is a standard (T11 FC-BB-5) that builds a topology of virtual FC links on top of a physical Ethernet network. To transport native FC traffic across this topology, FCoE encapsulates native FC frames into Ethernet MAC frames, which are then forwarded through the Ethernet network in the same manner as any other Ethernet frame. As a result, FCoE supports the convergence of FC and Ethernet traffic on the same physical Ethernet network.

Encapsulated FC frames are forwarded along the Ethernet transport between two types of FCoE endpoints: an FCoE Node (ENode) and an FCoE Forwarder (FCF). An ENode refers to the FCoE capability of a converged network adapter (CNA). An FCF is a switch that is capable of transmitting FCoE frames onto the FCoE network and native FC frames onto the FC storage area network (SAN). Both ENodes and FCFs handle the encapsulation and de-encapsulation of FC frames.

It is important to remember that an FCoE transit switch forwards native FC frames that are already encapsulated into Ethernet frames. The transit switch does *not* encapsulate or de-encapsulate native FC frames.

- The ENode and FCF devices in the FCoE fabric are responsible for encapsulating native FC frames into Ethernet and then forwarding those frames onto the FCoE transit switch path.
- FCoE encapsulated frames received by an ENode or FCF are then de-encapsulated and the native FC frames are forwarded onto the intended FC network destination.

The encapsulation of native FC frames into Ethernet allows the use of transit switches between ENodes and FCFs to extend the reach of the FCoE network. An Ethernet switch can participate in an FCoE network as an FCoE Forwarder (FCF) switch or as an FCoE transit switch.

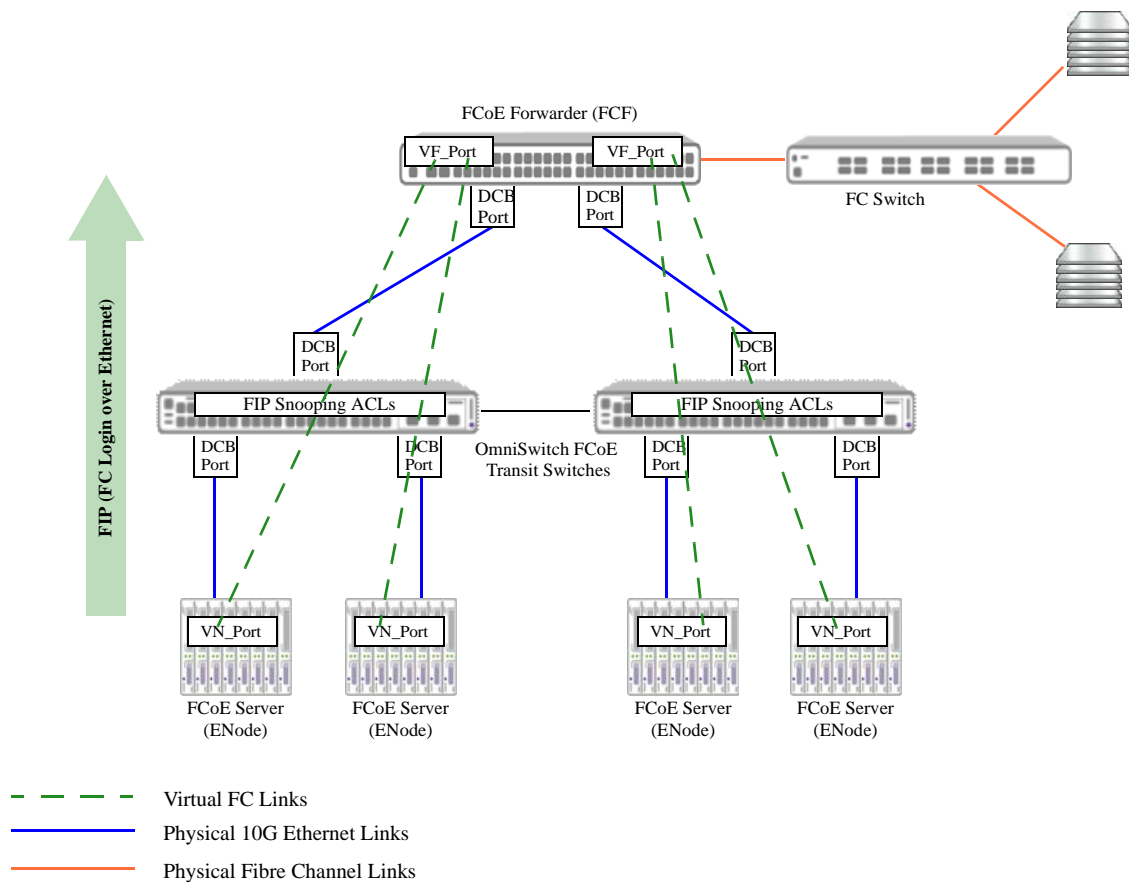
- An FCF is a dual-stack switch with both native FC and FCoE capability. The FCF encapsulates native FC frames into Ethernet frames and then forwards those frames over the FCoE network. When the FCF receives encapsulated frames, the FCF strips the Ethernet encapsulation and then forwards the native FC frame on the FC network.
- An FCoE transit switch is an access switch that transports encapsulated FCoE frames between ENodes and FCFs. A transit switch does not encapsulate or de-encapsulate native FC frames; the switch acts as a pass-through switch in an FCoE topology.

---

**Note.** The OmniSwitch can participate in an FCoE network as an FCoE transit switch or as an FCoE Forwarder. See [Chapter 5, “Configuring an FCoE/FC Gateway,”](#) for more information about using the OmniSwitch as an FCoE Forwarder.

---

As an FCoE transit switch, the OmniSwitch will sit in between ENodes and an FCF as part of a multi-hop FCoE network, as shown in Figure 1 on [page 4-6](#). In this role, the OmniSwitch is a pass-through switch that is transparent to the ENode and FCF endpoints.



**Figure 1: OmniSwitch Transit Switch Connecting FCoE Endpoints**

The virtual FC links between ENode VN\_Ports and FCF VF\_Ports are discovered, created, and maintained by the FCoE Initialization Protocol (FIP). Based on FIP requests and responses traversing these links, the transit OmniSwitch will dynamically create ACLs to filter the FIP traffic received on the switch FCoE ports connected to FCoE devices. These ACLs perform the FIP snooping function that protects the FCoE network from unauthorized access and data transmission through the transit switch.

## FCoE Initialization Protocol

Extending the reach of an FC network using FCoE requires FCoE to provide the same point-to-point connectivity and security that is required within a native FC network. While FCoE encapsulation handles the forwarding of native FC frames over 10G (or higher) Ethernet, the FCoE Initialization Protocol (FIP) is needed to provide the point-to-point connectivity and base FC device login process over Ethernet.

FIP is a control protocol that establishes and maintains virtual FC links between ENode VN\_Ports and FCF VF\_Ports in the FCoE network. These virtual links can traverse a multi-hop FCoE transit switch path but appear as direct point-to-point connections to the FCF. The underlying FCoE Ethernet link path is invisible to the FCF. FIP allows ENodes the ability to discover FCFs and complete the FC login process over the FC virtual links.

The remainder of this section describes the major functions FIP performs to support the FC login process over FCoE.

## FIP VLAN Discovery

ENodes send out FIP VLAN discovery requests to find FCoE VLANs on which the ENode can transmit and receive other FIP protocol traffic and FCoE encapsulated frames.

- The ENode sends FIP VLAN discovery requests untagged to the ALL-FCF-MACs multicast address. Because the request packets are sent untagged, the VLAN discovery process takes place on the default VLAN. On the OmniSwitch, this is the default VLAN assigned to the FCoE ports.
- FCFs also listen to the ALL-FCF-MACs multicast address. Those FCFs associated with the same default VLAN as the ENode will respond to the request with a list of available FCoE VLANs for ENode VN\_Port login.
- The ENode selects one of the FCoE VLANs received in the FCF notifications and then continues the FIP process on that VLAN.

FIP discovery requests are the only FIP packets carried on the default VLAN. All other FIP functions and FCoE run on the discovered FCoE VLANs.

## FIP FCF Discovery

Once an ENode has selected an FCoE VLAN, the next step is to discover FCFs that belong to the same VLAN and then select an FCF that is available for logins. There are two ways FCF discovery can occur:

- An FCF initiates discovery by periodically sending multicast FIP FCF discovery advertisements on each configured FCoE VLAN. These advertisements are sent to the ALL-ENODE-MACs and are used by the FCFs to notify ENodes that the FCF has VF\_Ports available for establishing virtual links with ENode VN\_Ports.
- When a new ENode comes online, the ENode sends a unicast FIP FCF discovery solicitation message on the FCoE VLAN to which the ENode belongs. The solicitation message is sent to the ALL-FCF-MACs multicast address to request a unicast advertisement. An FCF that receives the solicitation can send a unicast FIP FCF discovery advertisement back to the ENode that initiated the request.

Once an ENode has gathered all of the FIP FCF discovery advertisements, the ENode selects an FCF to contact for fabric login and virtual link establishment.

## FIP Login

When an ENode selects an FCF, the ENode sends a fabric login (FLOGI) request to the FCF. The FCF acknowledges the request and, depending on the MAC addressing mode used, the FCF sends the ENode a locally unique MAC address to use for FCoE encapsulation.

There are two MAC addressing modes that ENodes and FCFs can use to provide a locally unique MAC address for FCoE operations: fabric-provided MAC address (FPMA) mode or server-provided MAC address (SPMA) mode. The OmniSwitch supports both FPMA and SPMA modes.

- When the FPMA mode is in use by the ENode, the MAC address assigned to the ENode VN\_Port is a 48-bit address, which consists of the 24-bit FCoE mapped address prefix (FC-MAP) combined with a 24-bit FC identifier (FCID). The FC-MAP is a configurable value on OmniSwitch FCoE VLANs. In this mode, the ENode can have different VN\_Ports, each with their own unique MAC address.
- When the SPMA mode is in use by the ENode, the server provides the MAC address to the FCF, which then determines if the SPMA is an address approved for FCoE access.

The ENode and the FCF must use the same addressing mode to establish virtual links between the ENode and FCF. Note that the ENode continues to use the globally unique ENode MAC address (CNA MAC) for FIP control operations; the FPMA or SPMA is used for FCoE frame transactions.

A successful ENode fabric login establishes a new virtual link between the ENode VN\_Port and FCF VF\_Port. FCoE servers attached to an Ethernet network can access storage devices in the FC SAN by exchanging FCoE frames (encapsulated FC payloads) over the virtual link.

## OmniSwitch FIP Snooping ACLs

FIP snooping is a security mechanism used in multi-hop FCoE networks to ensure that only traffic from ENodes that have successfully logged into the FC fabric passes through the FCoE ports and VLANs on the transit switch and reaches the FCF. In other words, securing the dedicated FC virtual point-to-point link that is established between an ENode and FCF through the FIP login process to prevent unauthorized access to the FC network.

Based on Annex C (*Increasing FC-BB\_E Robustness Using Access Control Lists*) of the T11 FC-BB-5 standard, the OmniSwitch implementation of FIP Snooping filters FIP traffic on FCoE ports using QoS ACLs. The OmniSwitch QoS feature inspects FIP packets and dynamically configures FIP snooping ACL entries based on the contents of those packets. The ACLs are then applied to determine whether frames are forwarded or discarded on OmniSwitch FCoE ports.

- The FCoE ACLs take precedence over other QoS policies.
- If a frame matches multiple FCoE ACL entries programmed in the hardware, the first matching ACL entry in the hardware is applied to the frame.

When FIP snooping is globally enabled for the transit OmniSwitch, the switch will deny traffic from any ENode on any FCoE VLAN until the ENode successfully logs on to the FC fabric.

## FCoE Port Roles

The dynamic ACL configuration applied to an OmniSwitch FCoE port is determined by the role the port is assigned within the FCoE network. A port is assigned one of the following roles when the port is configured as an FCoE port:

- **Edge port**—a direct link to an ENode host (ENode-facing port).
- **ENode-only port**—a link between FCoE transit switches that carries traffic from ENode to FCF but not the reverse.
- **FCF-only port**—a link between FCoE transit switches that carries traffic from FCF to ENode but not the reverse.
- **Mixed port**—a link between FCoE transit switches that carries traffic in both directions (from FCF to ENode or from ENode to FCF).
- **Trusted port**—the FCoE port is trusted; traffic on this port is not filtered by FIP ACLs. This port role type is typically assigned to the switch FCoE port that connects to an Ethernet port on the FCF.

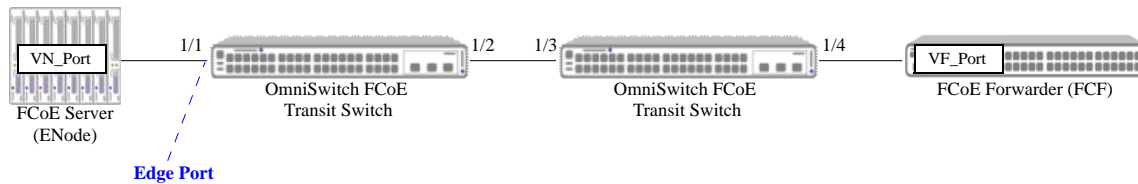
The port role assigned depends on where the port resides within the FCoE transit switch path and the source of the traffic (ENode, FCF, or both) the port will receive. Although FCoE ports can pass bidirectional FCoE traffic, the dynamic ACL configuration is based on the ingress traffic.

The remaining subsections provide an example of port role locations within the network and the ACL functionality applied for the given port role.



## Edge Port

The following diagram shows an example of where an FCoE edge port is placed in the FCoE network topology:



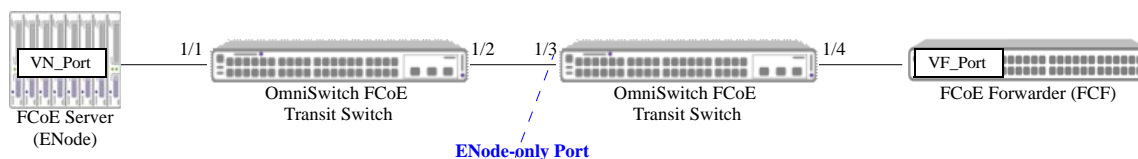
**Figure 2: FCoE Port Role-Edge**

ACL entries are applied on an ENode-facing edge port to:

- Enable transmission of FCoE frames from ENodes to FCF.
- Ensure that only FIP frames are sent to the FCF.
- Prevent transmission of FCoE frames from ENodes prior to fabric login (FLOGI).
- Prevent transmission of FCoE frames from one ENode to another ENode.
- Ensure that after fabric login (FLOGI) the ENode only uses FCoE source MAC addresses assigned by the FCF.
- Ensure that after FLOGI the assigned source MAC address is only used for FCoE traffic.
- Ensure that after the FLOGI that FCoE frames are only addressed to the FCF.
- Ensure that ACL entries do not interfere with other switch ACLs.
- Ensure that the ACL entries do not restrict non-FCoE traffic (traffic that does not match the FCoE Ethertype and does not use an FCoE source MAC address).

## ENode-only Port

The following diagram shows an example of where an FCoE ENode-only port (port that carries ingress traffic from the ENode) is placed in the FCoE network topology:



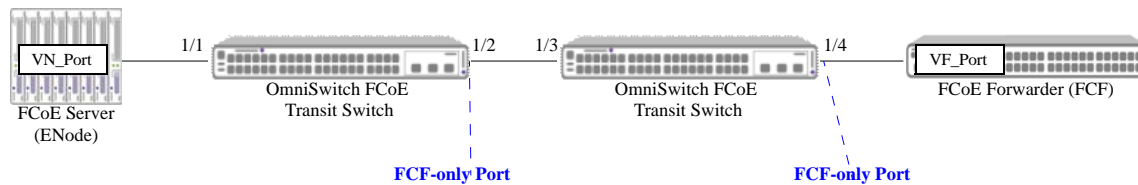
**Figure 3: FCoE Port Role-ENode**

ACL entries are applied on an ENode-only port to:

- Verify that all FIP frames are addressed to the FCF.
- Verify that all FCoE frames are addressed to the FCF.
- Verify that all FCoE frames are only sourced by ENodes (if FPMA is in use).
- Prevent FCoE traffic from flowing between FCF (optional).

## FCF-only Port

The following diagram shows an example of where an FCoE FCF-only port (port that carries ingress traffic from the FCF) is placed in the FCoE network topology:



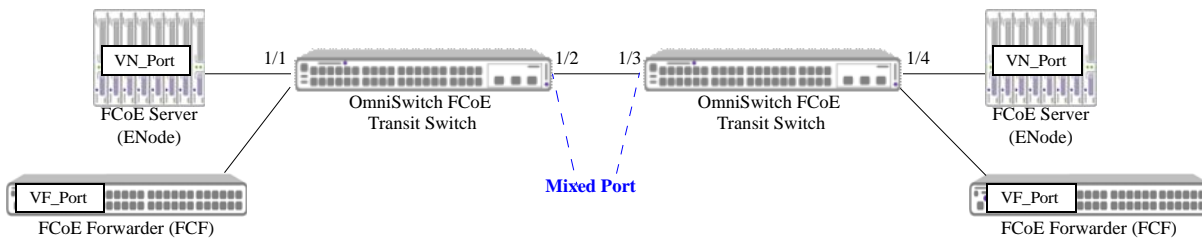
**Figure 4: FCoE Port Role–FCF**

ACL entries are applied on an FCF-only port to:

- Verify that all FIP frames are sourced from an FCF and are not destined to another FCF.
- Verify that all FCoE frames are sourced from an FCF.
- Verify that all FCoE frames are destined to an ENode (FPMA only).
- Prevent the forwarding of FCoE frames to an FCF (optional).

## Mixed Port

The following diagram shows an example of where an FCoE mixed port (port that carries FCF-to-ENode traffic and ENode-to-FCF traffic) is placed in the FCoE network topology:



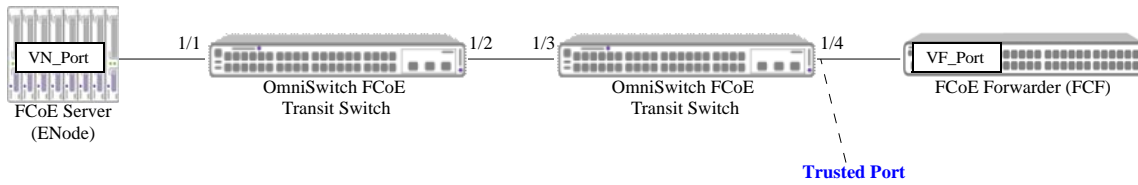
**Figure 5: FCoE Port Role–Mixed**

ACL entries are applied on mixed port to:

- Verify that all FIP frames are sourced from or destined to an FCF.
- Verify that all FCoE frames are sourced from an FCF and are sent to an ENode or FCF, or verify that all FCoE frames are sourced from an ENode and are sent to an FCF (FPMA only).
- Verify that all FCoE frames are sourced from or sent to an FCF.

### Trusted Port

The following diagram shows an example of where an FCoE trusted port is placed in the FCoE network topology:



**Figure 6: FCoE Port Role–Trusted**

No ACLs are applied when the port is configured as an FCoE trusted port.

# Interaction with Other Features

This section contains important information about how other OmniSwitch features interact with the FCoE and FIP Snooping features. Refer to the specific chapter for each feature to get more detailed information about how to configure and use the feature.

## Data Center Bridging (DCB)

FCoE requires an underlying lossless Ethernet network. The OmniSwitch supports the following Data Center Bridging (DCB) protocols that extend Ethernet capabilities to support the convergence of storage and data in virtualized networks:

- **Priority-Based Flow Control (PFC)**—Based on the IEEE 802.1Qbb standard, PFC pauses traffic based on congestion priority instead of blocking the entire link when congestion occurs. Allows lossless and lossy traffic with different priorities on the same physical port.
- **Enhanced Transmission Selection (ETS)**—Based on the IEEE 802.1Qaz standard, ETS provides a common framework for dynamic bandwidth management. ETS groups related traffic into priority groups (Traffic Classes) to which bandwidth guarantees and scheduling are applied.
- **Data Center Bridging Exchange (DCBX)**—Based on the IEEE 802.1Qaz standard, DCBX uses the Link Layer Discovery Protocol (LLDP) to exchange and negotiate PFC and ETS information between two directly connected peer switches. Enabled by default, DCBX is responsible for auto-negotiation and auto-configuration of link parameters for DCB functions.

The OmniSwitch implementation of DCB provides enhanced QoS congestion and bandwidth allocation to support multiple traffic types on the same Ethernet link. The supported DCB protocols use the Class of Service (CoS) 802.1p markings found in the frame header to define traffic groups. These markings are the result of QoS classification that occurs prior to and separately from the application of DCB functionality. DCB does not replace existing QoS classification and enqueueing mechanisms.

On the OmniSwitch, embedded port profiles are used to apply the supported PFC, ETS, and DCBx configuration to traffic flows. This approach is similar to how QSet profiles are used to apply the QoS configuration for bandwidth management and egress port queue scheduling. However, DCB and QSet profiles are mutually exclusive on the same port.

By default, when the OmniSwitch Data Center software license is installed, DCB profile 8 (DCP 8) is applied to all switch ports. In addition, DCBx is enabled on each switch port. However, DCP 8 applies lossy priority flow control to ingress traffic flows. However, the FCoE transit switch path requires a lossless Ethernet transport. As a result, it is necessary to apply to each participating FCoE port one of the other pre-defined profiles or create a custom profile that guarantees lossless priority flow control for the 802.1p value that is used to mark FCoE traffic.

For more information about DCB profiles, see [Chapter 2, “Configuring Data Center Bridging,”](#) of the *OmniSwitch AOS Release 7 Data Center Switching Guide*.

For more information about CoS 802.1p priority bit classification and marking, see [“How Traffic is Classified and Marked” on page 25-6 in Chapter 25, “Configuring QoS,”](#) of the *OmniSwitch AOS Release 7 Network Configuration Guide*.

## 802.1AB Link Layer Discovery Protocol

DCBX uses the IEEE 802.1AB Link Layer Discover Protocol (LLDP) to discover and exchange Ethernet capabilities between two link peers (such as between an ENode and an edge FCoE switch). There are specific LLDP Type-Length-Values (TLVs) that DCBX defines for exchanging PFC and ETS attribute values.

In addition to the DCBX TLVs, the OmniSwitch implementation of LLDP also supports the Application Priority TLV, which is used to advertise the designated FCoE priority to a peer device. Activating the Application Priority TLV and specifying the FCoE priority value advertised through this informational TLV is configured using LLDP CLI commands. Make sure that the priority value advertised through the Application Priority TLV matches the priority value designated for FCoE traffic classification and priority flow control.

## Loop Avoidance

Configuring MSTP or ERP for loop avoidance is recommended. When using MSTP assign each FCoE VLAN its own MST ID. This will improve performance in a topology where each VSAN assigned to a VLAN uses different FCFs.

## Universal Network Profile (UNP)

- Configuring FCoE on a UNP port is allowed. UNP can dynamically assign an FCoE port to a default VLAN, but manually tagging the port with the FCoE VLAN is still required.
- An FCoE VLAN cannot be assigned to a UNP.
- Enabling UNP on an FCoE port that is tagged with non-FCoE VLANs is not allowed. Enabling UNP on an FCoE port is allowed only on FCoE ports that are tagged with FCoE VLANs.
- Changes to a UNP port triggers a MAC flush of all MAC addresses learned on that port, including FIP and FCoE MAC addresses if the UNP port is also an FCoE port.
- When UNP is enabled or disabled on an FCoE port, all MAC addresses are flushed, including FIP and FCoE MAC addresses.

For more information about UNP, see [Chapter 27, “Configuring Universal Network Profiles.”](#)

## QoS

The QoS application is responsible for the dynamic configuration of the ACLs that FIP snooping uses to secure the traffic flowing through an OmniSwitch FCoE transit switch. As a result FIP snooping ACLs and other QoS functions share the same system resources. This means that the number of FIP snooping sessions allowed is subject to the availability of the shared resources.

The following limitations apply when FCoE is configured on a switch port:

- FCoE port configuration overrides global QoS settings.
- If there are no QoS system resources available for FIP snooping, an error is given indicating that resources were not allocated.
- To achieve priority protection of the FCoE traffic, if any of the non-FCoE traffic with the FCoE priority is received on the FCoE port, the traffic is remarked to a priority lower than that of the default

FCoE priority and the configured FCoE priority. Other lossless traffic will have its own priority and should not clash with the FCoE priority.

## Multiple VLAN Registration Protocol (MVRP)

- The FCoE transit switch path between ENodes and the FCF requires configuration of the same FCoE VLAN on each switch. To configure FCoE VLANs dynamically through the core network that connects FCF and ENodes, enable MVRP on all FCoE ports. However, on edge switches that are connected to an ENode or FCF, statically configuring the FCoE VLANs and tagging them with the appropriate FCoE ports is still required.
- If an MVRP join message is received on a FCOE VLAN, no VPA is created whether or not the receiving port is an FCoE port. However, as part of an MVRP advertisement, the same VLAN will be transmitted so that it can be learned on core switches.
- Manually configuring the FCoE VLAN end-to-end is recommended, but if there are switches that are not directly connected to ENodes or an FCF and they are secure, then MVRP may help reduce FCoE FIP snooping configuration steps.

# Configuring FIP Snooping

Configuring OmniSwitch FIP snooping in an FCoE multi-hop network requires the following general steps:

**1 Configure the DCB lossless Ethernet transport.** The following steps are required on each transit OmniSwitch to ensure lossless transport of FCoE traffic through the multi-hop FCoE network.

**a** Select a pre-defined DCB profile or create a custom profile that defines lossless priority flow control for a specific CoS priority value. By default, FCoE CoS is set to 3.

**b** Determine the end-to-end lossless data path between ENodes and the FCF and apply the profile identified in Step 1a to each port that will forward FCoE traffic. For example, the following command assigns DCB profile 11 to ports 8/1 and link aggregate 10.

```
-> qos qsi port 8/1 qsp dcb dcp-11
-> qos qsi linkagg 10 qsp dcb dcp-11
```

In addition, make sure that the DCBx protocol is enabled on each participating port. DCBx is enabled by default on all switch ports, but the following command is used to enable DCBx, if necessary:

```
-> qos qsi port 8/1 dcb dcbx admin-state enable
-> qos qsi linkagg 10 dcb dcbx admin-state enable
```

**c** Make sure that the traffic requiring lossless transmission is marked with the same CoS priority value to which the DCB profile will apply lossless priority flow control. The switch will only pause frames sent with the same priority value as the lossless priority value specified in the DCB profile.

**d** Make sure that all participating FCoE ports are configured as QoS trusted ports with the default classification set to 802.1p. For example:

```
-> qos port 8/1 trusted default classification 802.1p
-> qos linkagg 10 trusted default classification 802.1p
```

**e** Configure 802.1ab LLDP on FCoE ports to send the application-priority TLV. This is done to provide the peer device with the priority value set for FCoE traffic. For example:

```
-> lldp port 8/1 tlv application enable
-> lldp port 8/1 tlv application priority 3
```

**2 Configure FCoE VLANs.** Create the same FCoE VLAN on each participating OmniSwitch. For example:

```
-> fcoe vlan 100
```

**3 Configure FCoE ports.** Configure each port that will carry FCoE/FIP traffic as an FCoE port, which includes specifying the FCoE role (Edge, ENode-only, FCF-only, Mixed, Trusted) for the port. For example, port 8/1 connects to an ENode and link aggregate 10 will forward ingress traffic from the FCF.

```
-> fcoe port 8/1 role edge
-> fcoe linkagg 10 role fcf-only
```

**4 Assign FCoE ports to FCoE VLANs.** Tag each FCoE port with the same FCoE VLAN on each participating OmniSwitch. For example:

```
-> vlan 100 members port 8/1 tagged
-> vlan 100 members linkagg 10 tagged
```

**5 Enable FIP snooping.** Administratively enable FIP snooping on each participating OmniSwitch. For example:

```
-> fcoe fip-snooping admin-state enable
```

Additional FIP snooping global, VLAN, and port parameters are configurable to fine tune FIP snooping on the OmniSwitch. These parameters have default values that apply when FIP snooping is enabled for the switch. For more information about configuring FIP snooping parameters, see the following sections:

- [“Configuration Guidelines” on page 4-16.](#)
- [“Configuring Lossless DCB for FCoE” on page 4-17.](#)
- [“Configuring Global FCoE Parameters” on page 4-19.](#)
- [“Configuring FCoE VLANs” on page 4-21.](#)
- [“Configuring FCoE Ports” on page 4-22.](#)
- [“Configuration Example” on page 4-23.](#)

For more information about DCB profiles, see [Chapter 2, “Configuring Data Center Bridging,”](#) of the *OmniSwitch AOS Release 7 Data Center Switching Guide*.

For more information about CoS 802.1p priority bit classification and marking, see [“How Traffic is Classified and Marked” on page 25-6](#) in [Chapter 25, “Configuring QoS,”](#) of the *OmniSwitch AOS Release 7 Network Configuration Guide*.

## Configuration Guidelines

Review the guidelines in this section before attempting to configure an OmniSwitch FIP snooping transit switch path between ENodes and FCFs.

### FCoE VLANs

- Manual configuration of the FCoE VLAN is required along the transit switch path and on the FCF. However, the ENode may invoke FIP VLAN discovery to discover the FCoE VLANs within the transit path. If not, manual configuration of the FCoE VLANs may also be required on the appropriate ENodes.
- The ENode, transit switches, and the FCF must all use the same addressing mode (FPMA or SPMA) to establish virtual links between the ENode and FCF. On the OmniSwitch, the addressing mode is a configurable parameter for the FCoE VLAN. Make sure each OmniSwitch FCoE VLAN in the transit switch path is configured with the same dress mode used by the ENode and FCF.
- FCoE VLANs must be configured with an MTU size of at least 2500 bytes to accommodate FCoE frames, which are larger than the standard Ethernet frame size. Make sure the MTU size is configured end-to-end to prevent fragmentation of FCoE frames.
- Configuring an FCoE VLAN as a default VLAN for a port or link aggregate is not allowed. In addition, configuring default VLAN 1 as an FCoE VLAN is not allowed.
- The following features are not supported on FCoE VLANs:
  - > IGMP Snooping
  - > IP interface
  - > HA VLANs



- > SVLAN and CVLAN
- > Shortest Path Bridging (SPB)
- > MCLAG
- > UDP Relay, DHCP Snooping
- > Universal Network Profile (UNP)

## FCoE Ports

- FCoE is only supported on 10G or faster ports that are associated with an FCoE lossless DCB profile. In addition, DCBX must be enabled on the port with both PFC and ETS in an active state (either forced or negotiated via DCBX). The DCB configuration is done separately using QoS port and profile commands.
- FCoE ports must be manually assigned to a default VLAN and then tagged with the FCoE VLAN that will forward the FCoE and FIP frames on that port.
- The default VLAN for the FCoE ports will carry FIP VLAN discovery requests through FCoE network. As a result, it is necessary to configure the same default VLAN for FCoE ports on each switch. All other FIP traffic and FCoE frames are carried on the FCoE VLAN.

The following features are not supported on FCoE ports:

- Learned Port Security (LPS)
- Port Mirroring and Remote Port Mirroring
- Shortest Path Bridging (SPB)
- Edge Virtual Bridging (EVB).

## FCoE Priority

Make sure to use the same FCoE priority value when configuring the following:

- The FCoE priority value used to classify FCoE traffic (QoS class of service markings).
- The lossless priority value defined in the DCB profile applied to participating FCoE ports.
- The FCoE priority value defined in the LLDP Application Priority TLV that is advertised on FCoE edge ports connected to ENodes.
- The FCoE priority that is protected when FCoE priority protection is enabled for the switch. Priority protection is disabled by default.

## Configuring Lossless DCB for FCoE

The OmniSwitch Data Center Bridging (DCB) configuration is applied using DCB profiles. By default, DCB profile 8 (DCP 8) is applied to all switch ports and link aggregates. However, DCP 8 does not apply lossless priority flow control; all traffic classes are set to lossy. The FCoE transit switch path requires a lossless Ethernet transport. As a result, it is necessary to apply a profile that defines lossless flow control for the desired FCoE traffic priority on all of the following participating ports:

- FCoE edge ports that connect to ENodes (servers).
- FCF only ports that connect to FCoE Forwarders (FCFs).
- All port connections between OmniSwitch FCoE transit switches.

There are 11 predefined DCB profiles (DCP 1–11) available, and some of these profiles provide lossless traffic classes for one or more priorities. However, these profiles may not provide lossless traffic classes for the designated FCoE traffic priority or may provide lossless traffic classes for all priorities (DCP 7, 9, and 11). In this case, creating and applying a custom DCB profile, as follows, is recommended:

- 1 Create the new custom profile based on DCP 9 or 11. For example:

```
-> qos qsp dcb dcp-12 import qsp dcb dcp-11
```

- 2 Configure all traffic classes in DCP 12 to lossy, except for the priority designated for lossless FCoE traffic. In this example, priority 3 is the FCoE priority so its traffic class is not changed to lossy:

```
-> qos qsp dcb dcp-12 tc 0 pfc flow-type nLL
-> qos qsp dcb dcp-12 tc 1 pfc flow-type nLL
-> qos qsp dcb dcp-12 tc 2 pfc flow-type nLL
-> qos qsp dcb dcp-12 tc 4 pfc flow-type nLL
-> qos qsp dcb dcp-12 tc 5 pfc flow-type nLL
-> qos qsp dcb dcp-12 tc 6 pfc flow-type nLL
-> qos qsp dcb dcp-12 tc 7 pfc flow-type nLL
```

- 3 Apply custom profile DCP 12 to all participating FCoE ports. For example:

```
-> qos qsi port 1/5/16 qsp dcb dcp-12
-> qos qsi linkagg 5 qsp dcb dcp-12
```

---

**Note.** If FCoE priority protection is enabled for the switch, make sure the DCB profile used on participating FCoE ports ensures lossless traffic classes for the protected priority values (see [“Enabling/Disabling FCoE Priority Protection”](#) on page 4-19).

---

For more information about DCB profiles, including predefined profile definitions, see [Chapter 2, “Configuring Data Center Bridging,”](#) of the *OmniSwitch AOS Release 7 Data Center Switching Guide*.

## Configure the LLDP Application Priority TLV

DCBX uses the IEEE 802.1AB Link Layer Discover Protocol (LLDP) to discover and exchange Ethernet capabilities between two link peers (such as between an ENode and an edge FCoE switch). There are specific LLDP Type-Length-Values (TLVs) that DCBX defines for exchanging PFC and ETS attribute values.

In addition to the DCBX TLVs, the OmniSwitch implementation of LLDP also supports the Application Priority TLV. In an FCoE network, this TLV is used to advertise the designated FCoE priority to a server (ENode) directly connected to an FCoE edge port.

Activating the Application Priority TLV and specifying the FCoE priority value to advertise is configured using [lldp tlv application](#) command with the **fcoe priority** option. For example:

```
-> lldp port 1/5/16 tlv application fcoe priority 3
-> lldp port 2/4/16 tlv application fcoe priority 3
```

Make sure the FCoE priority value advertised with the Application Priority TLV is the same priority value that is designated for FCoE traffic in the network configuration.

## Configuring Global FCoE Parameters

This section describes the global FIP snooping configuration, which includes enabling and disabling FIP snooping, changing the FIP addressing mode, configuring the FCoE traffic priority, setting the filtering resource threshold value, and configuring the housekeeping time period.

### Enabling/Disabling FIP Snooping

By default FIP snooping is disabled on the switch. To enable FIP snooping, use the `fcoe fip-snooping` command. For example:

```
-> fcoe fip-snooping admin-state enable
```

When FIP snooping is enabled, FCoE traffic is dropped on all switch VLANs and ports that are *not* configured as FCoE VLANs and ports.

To disable FIP snooping, enter the `fcoe fip-snooping` command with `disable` option:

```
-> fcoe fip-snooping admin-state disable
```

### Changing the FIP Addressing Mode

The FIP addressing mode determines if the MAC address assigned to an ENode is a fabric-provided MAC address (FPMA) assigned by the FCF or a server-provided MAC address (SPMA) assigned by the ENode server but verified by the FCF. The MAC address is assigned during the FIP login process and is used to identify the ENode for FCoE frame transactions and FC virtual link termination.

Consider the following when changing the FIP addressing mode:

- The FIP addressing mode configured for the OmniSwitch must match the mode used by the ENode and FCF, otherwise the FIP login process and virtual link establishment will fail.
- Disabling FIP snooping on the switch is required before the addressing mode can be changed.

By default, the addressing mode is set to FPMA. To change the FIP addressing mode, use the `fcoe address-mode` command. For example:

```
-> fcoe fip-snooping disable
-> fcoe address-mode spma
-> fcoe fip-snooping enable
```

### Enabling/Disabling FCoE Priority Protection

---

**Note.** By default, FCoE priority protection is disabled on the switch. When enabled, any non-FCoE traffic *received on any switch port (not just FCoE ports)* that matches the protected FCoE priority value is either dropped or remarked, based on the priority protection action configured for the switch.

---

The default priority value that is protected is set to 3, which is a common 802.1p value assigned to FCoE traffic. However, this value is not protected until priority protection is enabled for the switch using the `fcoe priority-protection` command. For example:

```
-> fcoe priority-protection enable
```

## Changing the Priority Protection Action

By default, non-FCoE traffic that matches the protected priority value is dropped. However, using the **fcoe priority-protection action** command, it is possible to mark the traffic with a different priority value instead of dropping the traffic. For example, the following command will mark non-FCoE traffic that matches the protected priority value (set to 3 for this example) to priority 7:

```
-> fcoe priority-protection action remark 7
```

## Changing the FCoE Protected Priority Value

The FCoE priority value specifies an 802.1p value for FCoE traffic that is protected when FCoE priority protection is enabled for the switch. When non-FCoE traffic received on *any* switch port is marked with a priority value that matches the FCoE priority *and* priority protection is enabled, the non-FCoE traffic is dropped or re-marked with a different priority value.

By default, the 802.1p priority value for FCoE traffic is set to 3. To change the FCoE priority value, use the **fcoe priority** command. For example:

```
-> fcoe priority 7
```

In addition, it is also possible to configure two priority values for FCoE traffic. For example, the following command specifies priority 2 and 7:

```
-> fcoe priority 2 7
```

Each time a priority value is configured, the existing priority value is overwritten.

If two priority values are configured but there is a need to change only one of the values, both priority values must be specified with the **fcoe priority** command. For example, if the current priority is set to 2 and 5, to change priority 2 to 3, specify both 3 and 5 as the priority values.

```
-> fcoe priority 3 5
```

Make sure that the FCoE protected priority value is the same priority value that is designated for FCoE traffic in the network configuration.

## Configuring the Filtering Resource Threshold

The filtering resource threshold specifies a percentage of switch resources used by FCoE ACLs that when reached, will generate an SNMP trap. By default, this threshold is set to 80%. To change the threshold value, use the **fcoe filtering-resource trap-threshold** command. For example:

```
-> fcoe filtering-resource trap-threshold 50
```

To disable trap generation in this case, set the threshold value to zero. For example:

```
-> fcoe filtering-resource trap-threshold 0
```

## Configuring the Keepalive Waiting Time

ENodes and FCFs will send keepalive messages to verify the reachability of the ENode or FCF. These messages will pass through a FIP snooping OmniSwitch that is part of a multi-hop FCoE network. The OmniSwitch will wait a configurable amount of time for the next keepalive message. If no message is received before that time is reached, the switch will clear the FCoE session information.

By default, the amount of time the switch will wait for keepalive messages is set to 300 seconds. To change this amount of time, use the **fcoe house-keeping-time-period** command. For example:

```
-> fcoe house-keeping-time-period 120
```

To disable the house keeping timer, set the timer value to zero. For example:

```
-> fcoe house-keeping-time-period 0
```

## Configuring FCoE VLANs

The OmniSwitch implementation of FIP snooping is deployed using FCoE VLANs. Configuring an FCoE VLAN on each transit OmniSwitch between ENodes and FCF is required. Ports or link aggregates (FCoE interfaces) tagged with an FCoE VLAN will carry the FIP control frames and FCoE encapsulated frames through the Ethernet network.

To configure an FCoE VLAN on the OmniSwitch, use the **fcoe vlan** command. For example:

```
-> fcoe vlan 500
```

When creating an FCoE VLAN, specify a VLAN ID that does not exist in the switch configuration, or specify the ID of a dynamically created MVRP VLAN.

By default, an FCoE VLAN is administratively enabled when the VLAN is created. To disable the FCoE VLAN, use the **fcoe vlan** command with the **disable** option:

```
-> fcoe vlan 500 disable
```

To enable a disabled FCoE VLAN, use the **fcoe vlan** command with the **enable** option:

```
-> fcoe vlan 500 enable
```

When an FCoE VLAN is created, it is possible to assign an ASCII text name to the VLAN using the **fcoe vlan** command with the **name** option:

```
-> fcoe vlan 500 name fcoe-vlan1
```

If a name is not assigned to an FCoE VLAN when the VLAN is created, the VLAN ID is used by default. For example, when FCoE VLAN 500 is created without a name, then “VLAN-500” is used for the name.

## Configuring the FCF MAC Address

Virtual FC links that traverse the lossless Ethernet network through transit switches send FCoE frames to and from the FCF MAC address. This address is learned through the FIP FCF discovery process, but it is possible to statically configure an FCF MAC address for an FCoE VLAN.

By default, the learned FCF MAC address is used. To statically configure an FCF MAC for an FCoE VLAN, use the **fcoe fcf mac** command and specify an existing FCoE VLAN ID. For example:

```
-> fcoe fcf 30:10:94:01:00:00 vlan 100
```

To remove a statically configured FCF MAC address from an FCoE VLAN, use the **no** form of the **fcoe fcf mac** command. For example:

```
-> no fcoe fcf 30:10:94:01:00:00 vlan 100
```

## Configuring the FC-MAP Prefix

When the FIP addressing mode is set to FPMA, the FCF assigns a 48-bit address to ENode VN\_Ports. This address consists of the 24-bit FCoE mapped address prefix (FC-MAP) combined with a 24-bit FC identifier (FCID). The FC-MAP is a configurable value on FCoE VLANs.

The configured FC-MAP value assigned to an FCoE VLAN must match the FC-MAP value used by the FCF device. FCF advertisement packets that contain a different FC-MAP are not processed by the switch.

By default, the FC-MAP is set to 0E:FC:00. To statically configure an FC-MAP for an FCoE VLAN, use **fcoe fc-map** command and specify an existing FCoE VLAN ID. For example:

```
-> fcoe fc-map 0E:FC:04 vlan 30
```

To remove a statically configured FC-MAP address from an FCoE VLAN, use the **no** form of the **fcoe fc-map** command. For example:

```
-> no fcoe fc-map 0E:FC:04 vlan 30
```

## Configuring FCoE Ports

FCoE ports will carry the FCoE data frames (encapsulated FC frames) and the FIP control frames through the physical OmniSwitch transit switch path between ENodes and FCFs. In addition, dynamic ACLs generated by OmniSwitch QoS are applied to FCoE ports to monitor and secure FIP communications.

The **fcoe role** command is used to designate a switch port or link aggregate as an FCoE port and define the role the port will play in the FCoE transit switch path. The type of role assigned determines the ACL entries the switch generates to apply to the FCoE port (see [“FCoE Port Roles” on page 4-8](#)). For example, the following commands configure a port as an FCoE edge port and a link aggregate as an FCF only port:

```
-> fcoe port 1/8/1 role edge
-> fcoe linkagg 10 role fcf-only
```

In these examples, port 1/8/1 is configured as an FCoE edge port because the port will connect the switch to an FCoE-capable server (CNA with FCoE capability). Link aggregate 10 is configured as an FCoE FCF only port because the aggregate will process ingress traffic from an FCF that is destined for an ENode.

To change the role of an FCoE port, first remove the FCoE configuration from the port, then configure FCoE and the new role again for that same port. For example:

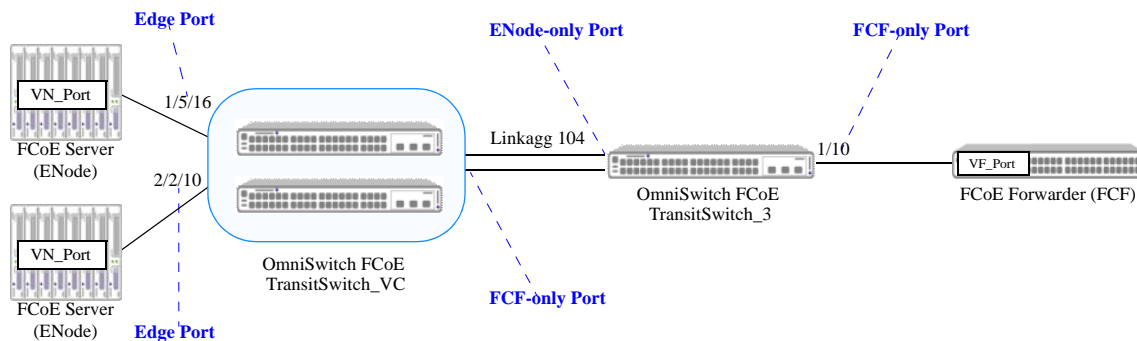
```
-> no fcoe port 1/8/1 role edge
-> fcoe port 1/8/1 enode-only
```

## Configuration Example

FIP snooping provides a mechanism for securing FCoE traffic to facilitate lossless storage convergence. The OmniSwitch supports FIP snooping through the use of dynamic ACLs that ensure that traffic only flows between valid ENodes and FCFs.

The FIP snooping capability along with the support of DCB protocols (PFC, ETS, and DCBx) allow the OmniSwitch to participate in a multi-hop FCoE network as an FCoE transit switch. When FIP snooping is enabled, the switch will inspect FIP frames received on FCoE ports that are assigned to FCoE VLANs. OmniSwitch QoS will then dynamically create the necessary ACLs based on the information in the inspected frames.

The following diagram shows a sample FCoE topology in which FIP snooping OmniSwitches are operating as FCoE transit switches to secure and forward FCoE traffic between FCoE endpoints (ENodes and FCF switch). To the ENodes and FCF, the underlying Ethernet links through the transit switch path appear as a direct point-to-point connection that is expected in a native FC network.



**Figure 7: Sample FCoE Topology**

The following switch configuration information provides an example of how the transit OmniSwitches are configured in the sample FCoE topology.

- ENodes are connected to ports 1/5/16 and 2/2/10 on TransitSwitch\_VC; both ports are configured as FCoE edge ports because they directly connect to a server ENode (FCoE-capable CNA). For example:

```
TransitSwitch_VC-> fcoe port 1/5/16 role edge
TransitSwitch_VC-> fcoe port 2/2/10 role edge
```

- The FCF is connected to port 1/10 on TransitSwitch\_3; port 1/10 is configured as an FCF-only port. For example:

```
TransitSwitch_3-> fcoe port 1/10 role fcf-only
```

- Linkagg 104 ports are spread across switch modules on both the master and slave chassis of TransitSwitch\_VC; linkagg 104 is configured as an FCF-only port on TransitSwitch\_VC and as an ENode-only port on TransitSwitch\_3. For example:

```
TransitSwitch_VC-> fcoe linkagg 104 role fcf-only
TransitSwitch_3-> fcoe linkagg 104 role enode-only
```

- FCoE VLAN 46 is configured across the transit switch path (manually on each OmniSwitch and, if necessary, manually on the ENodes and FCF). For example:

```
TransitSwitch_VC-> fcoe vlan 46
TransitSwitch_3-> fcoe vlan 46
```

- All FCoE ports and the link aggregate ID are assigned (untagged) to default VLAN 200. The default VLAN carries the ENode VLAN discovery messages; all other FCoE traffic is forwarded on FCoE VLAN 46. For example:

```
TransitSwitch_VC-> vlan 200
TransitSwitch_VC-> vlan 200 members port 1/5/16 untagged
TransitSwitch_VC-> vlan 200 members port 2/1/10 untagged
TransitSwitch_VC-> vlan 200 members linkagg 104 untagged

TransitSwitch_3-> vlan 200
TransitSwitch_3-> vlan 200 members port 1/10 untagged
TransitSwitch_3-> vlan 200 members linkagg 104 untagged
```

- All FCoE ports and the link aggregate ID are tagged with FCoE VLAN 46. For example:

```
TransitSwitch_VC-> vlan 46 members port 1/5/16 tagged
TransitSwitch_VC-> vlan 46 members port 2/1/10 tagged
TransitSwitch_VC-> vlan 46 members linkagg 104 tagged

TransitSwitch_3-> vlan 46 members port 1/10 tagged
TransitSwitch_3-> vlan 46 members linkagg 104 tagged
```

- All FCoE ports are QoS trusted with the default classification set to 802.1p. For example:

```
TransitSwitch_VC-> qos port 1/5/16 trusted default classification 802.1p
TransitSwitch_VC-> qos port 2/1/10 trusted default classification 802.1p
TransitSwitch_VC-> qos linkagg 104 trusted default classification 802.1p

TransitSwitch_3-> qos port 1/10 trusted default classification 802.1p
TransitSwitch_3-> qos linkagg 104 trusted default classification 802.1p
```

- Pre-defined DCB profile 9 (DCP-9) is assigned to all FCoE ports; DCP-9 configures lossless priority flow control for all CoS priorities. For example:

```
TransitSwitch_VC-> qos qsi port 1/5/16 qsp dcb dcp-9
TransitSwitch_VC-> qos qsi port 2/1/10 qsp dcb dcp-9
TransitSwitch_VC-> qos qsi linkagg 104 qsp dcb dcp-9

TransitSwitch_3-> qos qsi port 1/10 qsp dcb dcp-9
TransitSwitch_3-> qos qsi linkagg 104 qsp dcb dcp-9
```

In the sample FCoE topology, FCoE traffic is marked with priority 3.

- The LLDP Application Priority TLV is configured for FCoE priority 3 on FCoE ports 1/5/16, 2/2/10, and 1/10 to allow DCBx (enabled by default) to negotiate Ethernet capabilities and advertise the FCoE priority to the ENodes and FCF. For example:

```
TransitSwitch_VC-> lldp port 1/5/16 tlv application enable
TransitSwitch_VC-> lldp port 1/1/16 tlv application fcoe priority 3
TransitSwitch_VC-> lldp port 2/1/10 tlv application enable
TransitSwitch_VC-> lldp port 2/1/10 tlv application fcoe priority 3

TransitSwitch_3-> lldp port 1/10 tlv application enable
TransitSwitch_3-> lldp port 1/10 tlv application fcoe priority 3
```

- The FIP snooping functionality is globally enabled for all FCoE transit OmniSwitches. When FIP snooping is enabled, FCoE traffic is dropped on all switch VLANs and ports that are *not* configured as FCoE VLANs and ports. For example:

```
TransitSwitch_VC-> fcoe fip-snooping admin-state enable
TransitSwitch_3-> fcoe fip-snooping admin-state enable
```



## Verifying the FIP Snooping Configuration

Displaying the FIP Snooping configuration is helpful to verify the actual configuration and monitor FIP snooping sessions on each OmniSwitch that will operate as an FCoE transit switch. To display information about the FIP snooping configuration, use the **show** commands listed in this section.

<b>show fcoe</b>	Displays the global FCoE FIP snooping status and configuration for the switch.
<b>show fcoe ports</b>	Displays the FCoE port configuration, including port roles, for the switch. The port role determines the FIP snooping ACL configuration for the port.
<b>show fcoe sessions</b>	Displays the FIP sessions associated with the local switch.
<b>show fcoe enode</b>	Displays ENode information for the FIP sessions associated with the local switch.
<b>show fcoe fcf</b>	Displays FCF information for the FIP sessions associated with the local switch.
<b>show fcoe fc-map</b>	Displays the Fibre Channel Mapped Address Prefix (FC-MAP) configuration.
<b>show fcoe statistics</b>	Displays ENode and FCF generated session statistics. Use the <b>clear fcoe statistics</b> command resets the statistics information.



# 5 Configuring an FCoE/FC Gateway

The Alcatel-Lucent OmniSwitch provides Fibre Channel over Ethernet (FCoE) convergence solutions that facilitate the expansion of a Fibre Channel (FC) storage area network (SAN) across an existing Ethernet infrastructure, without having to purchase or manage additional, costly FC equipment. FCoE convergence features supported include the following:

- **FCoE transit switch**—The OmniSwitch supports the FCoE technology used to tunnel FC frames encapsulated within Ethernet MAC frames. To provide the necessary FCoE transit switch functionality, the OmniSwitch supports FCoE Initialization Protocol (FIP) snooping and Data Center Bridging (DCB) protocols for lossless Ethernet. A transit switch is basically a Layer 2 DCB switch that bridges encapsulated FCoE traffic over the Ethernet fabric between FCoE end devices.
- **FCoE/FC gateway switch**—The OmniSwitch serves as an FCoE forwarder to connect FCoE nodes to FC switches, connect FC nodes to an FCoE forwarder, and connect native FC fabrics across an FCoE network. To provide the necessary FCoE/FC gateway functionality, the OmniSwitch supports the following operational modes:
  - N\_Port proxy operation to aggregate FCoE Node (ENode) logins over a single OmniSwitch FC port that is connected to an FC switch.
  - F\_Port proxy operation to connect FC nodes to an FCoE forwarder or another gateway switch through an FCoE network or on the same gateway switch.
  - E\_Port proxy operation to provide a transparent point-to-point FC link between native E\_Ports. This allows inter-switch link (ISL) tunneling between FC fabrics over an FCoE network.

An OmniSwitch FCoE transit switch can connect to an OmniSwitch FCoE/FC gateway to access the necessary gateway services needed to transport FCoE traffic to or from the FC SAN. An OmniSwitch FCoE/FC gateway runs FIP snooping on the 10G Ethernet FCoE ports that connect to an FCoE network. On the same switch, FC ports connect to native FC switches or nodes. Traffic is transmitted between the FCoE network and the FC SAN through the gateway switch.

## In This Chapter

This chapter describes the OmniSwitch FCoE/FC gateway functionality in general and how the components of this functionality are applied to the switch. It provides information about configuring FCoE/FC gateway components through the Command Line Interface (CLI). CLI commands are used in the configuration examples; for more details about the syntax of commands, see the *OmniSwitch CLI Reference Guide*.

The following topics and procedures are included in this chapter:

- “FCoE/FC gateway Specifications” on page 5-4.
- “FCoE/FC Gateway Overview” on page 5-5.
- “Interaction with Other Features” on page 5-19.
- “FCoE/FC Gateway Configuration Guidelines” on page 5-21
- “Configuring an N\_Port Proxy Operation” on page 5-23.
- “Configuring an F\_Port Proxy Operation” on page 5-25.
- “Configuring an E\_Port Proxy Operation” on page 5-26.
- “FCoE/FC Gateway Configuration Examples” on page 5-28.
- “Verifying the FCoE/FC Gateway Configuration” on page 5-33.

The acronyms and abbreviations used in this chapter are defined here:

<b>CNA</b>	Converged Network Adapter
<b>CVL</b>	Clear Virtual Link
<b>E2E</b>	E_Port-to-E_Port Tunnel
<b>ELP</b>	Exchange Link Parameters
<b>ENode</b>	FCoE Node
<b>E_Port</b>	Expansion Port
<b>F_Port</b>	Fabric Port
<b>FC</b>	Fibre Channel
<b>FCF</b>	FCoE Forwarder
<b>FDISC</b>	Fabric Discovery
<b>FCID</b>	Fabric Port ID (same as N_Port ID).
<b>FCoE</b>	Fibre Channel over Ethernet
<b>FIP</b>	FCoE Initialization Protocol
<b>FLOGI</b>	Fabric Login
<b>FLOGO</b>	Fabric Logout
<b>HBA</b>	Host Bus Adapter
<b>ISL</b>	Inter-switch Link
<b>NPIV</b>	N_Port ID Virtualization
<b>N_Port</b>	Node Port

<b>NP_Port</b>	Proxy Node Port
<b>TE_Port</b>	Tunnel Expansion Port
<b>VE_Port</b>	Virtual E_Port
<b>VF_Port</b>	Virtual F_Port
<b>VN_Port</b>	Virtual N_Port
<b>VSAN</b>	Virtual Storage Area Network
<b>WWNN</b>	World Wide Node Name
<b>WWPN</b>	World Wide Port Name

For more information about the OmniSwitch FIP snooping implementation, see [Chapter 4, “Configuring FIP Snooping.”](#)

For more information about OmniSwitch support for lossless Ethernet, see [Chapter 2, “Configuring Data Center Bridging.”](#)

# FCoE/FC gateway Specifications

The FCoE/FC gateway functionality described in this chapter is supported on the OmniSwitch 6900, unless otherwise stated in the following specifications table or specifically noted within any other section of this chapter. Note that any maximum limits provided in this table are subject to available system resources.

Platforms Supported	OmniSwitch 6900
OmniSwitch Software License	Data Center
INCITS Standards Supported	<ul style="list-style-type: none"> <li>• FC-PI-4 Fibre Channel T11/08-138v1</li> <li>• FC-PI-5 Fibre Channel T11 2118-D/Rev 6.10</li> <li>• FC-BB-5 Backbone 5 T11/1871-D</li> <li>• FC-BB-6 Backbone 6 T11/2159-D (CNA switching only)</li> </ul>
Fibre Channel functionality supported	<ul style="list-style-type: none"> <li>• FCoE transit bridge <ul style="list-style-type: none"> <li>&gt; FCoE tunneling of encapsulated FC frames</li> <li>&gt; FCoE initialization protocol (FIP) snooping</li> </ul> </li> <li>• FCoE/FC gateway switch <ul style="list-style-type: none"> <li>&gt; N_Port proxy (NPIV)</li> <li>&gt; F_Port proxy (Reverse-NPIV)</li> <li>&gt; E_Port proxy (E2E-tunnel)</li> </ul> </li> </ul>
Supported port types	<ul style="list-style-type: none"> <li>• Fibre Channel for FCoE/FC gateway—OS-XNI-U12E module with SFP-FC-SR transceiver</li> <li>• Ethernet for FCoE/FIP snooping—10G or faster with DCB profile, DCBx enabled with PFC/ETS active (ports and link aggregates)</li> </ul>
OmniSwitch 64-bit World Wide Node Name (WWNN)	10:00:xx:xx:xx:xx:xx:xx (where xx = next available increment of the switch base MAC address)
OmniSwitch 64-bit World Wide Port Name (WWPN) for each Fibre Channel port	10:00:xx:xx:xx:xx:xx:xx (where xx = port MAC address)
VSAN-FC port associations	Multiple FC port assignments per VSAN allowed. Only one VSAN assignment per FC port allowed.
VSAN-FCoE VLAN mapping	One-to-one
VSAN scalability per switch	Based on the number of FC ports (for example, if switch has 12 FC ports, then 12 VSANs; one for each FC port). Note that an FC port configured as an E2E tunnel endpoint does not use up a VSAN assignment.
Maximum number of VSANs per network	4094
E2E tunnel scalability	One tunnel termination per FC port up to the number of available FC ports on the switch or virtual chassis.
Maximum frame size supported	2180
Load Balancing	NP_Port load balancing only: <ul style="list-style-type: none"> <li>• Dynamic</li> <li>• Dynamic-reorder</li> <li>• ENode-based</li> <li>• Static</li> </ul>

# FCoE/FC Gateway Overview

The OmniSwitch implementation of FCoE/FC gateway functionality allows the switch to transparently connect FCoE and FC nodes with an FC SAN across an FCoE (lossless Ethernet) network. To provide this type of connectivity, an OmniSwitch FCoE/FC gateway supports the following three modes of operation that are used to converge FC over Ethernet and FC-to-FC over Ethernet:

- **N\_Port Proxy mode**—allows ENodes in an FCoE network and FC switches in an FC SAN to communicate with each other. To an ENode the OmniSwitch gateway emulates an FCoE forwarder; to an FC switch the OmniSwitch gateway emulates an N\_Port ID Virtualization (NPIV) host. See [“Using the N\\_Port Proxy Mode” on page 5-7](#) and [“Configuring an N\\_Port Proxy Operation” on page 5-23](#) for more information.
- **F\_Port Proxy mode**—allows FC nodes to connect with FC switches and FCFs across an FCoE network. The OmniSwitch gateway forwards login requests from an FC node (N\_Port on a server or storage with an HBA) across Ethernet via an FCoE VLAN to an NPIV node or FCF. See [“Using the F\\_Port Proxy Mode” on page 5-12](#) and [“Configuring an F\\_Port Proxy Operation” on page 5-25](#) for more information.
- **E\_Port Proxy mode**—allows FC switches to set up inter-switch link trunking between FC fabrics over an FCoE network. The OmniSwitch gateway provides an E\_Port to E\_Port (E2E) tunneling function that emulates a point-to-point FC link between E\_Ports on native FC switches. See [“Using the E\\_Port Proxy Mode” on page 5-15](#) and [“Configuring an E\\_Port Proxy Operation” on page 5-26](#) for more information.

The OmniSwitch FCoE/FC gateway sits at the entry point of an FC fabric, which is required to handle the login process for ENodes and FC nodes accessing the fabric through the gateway switch.

## OmniSwitch FCoE/FC Gateway Fabric

The OmniSwitch FCoE/FC gateway fabric identifies the ports on which FC traffic is converted to FCoE traffic and FCoE traffic is converted to FC traffic. The fabric itself is defined through the association of the FC ports to an OmniSwitch VSAN and the association of FCoE ports to an FCoE VLAN. The VSAN is then mapped to the FCoE VLAN to define the traffic path for converted FC or FCoE frames.

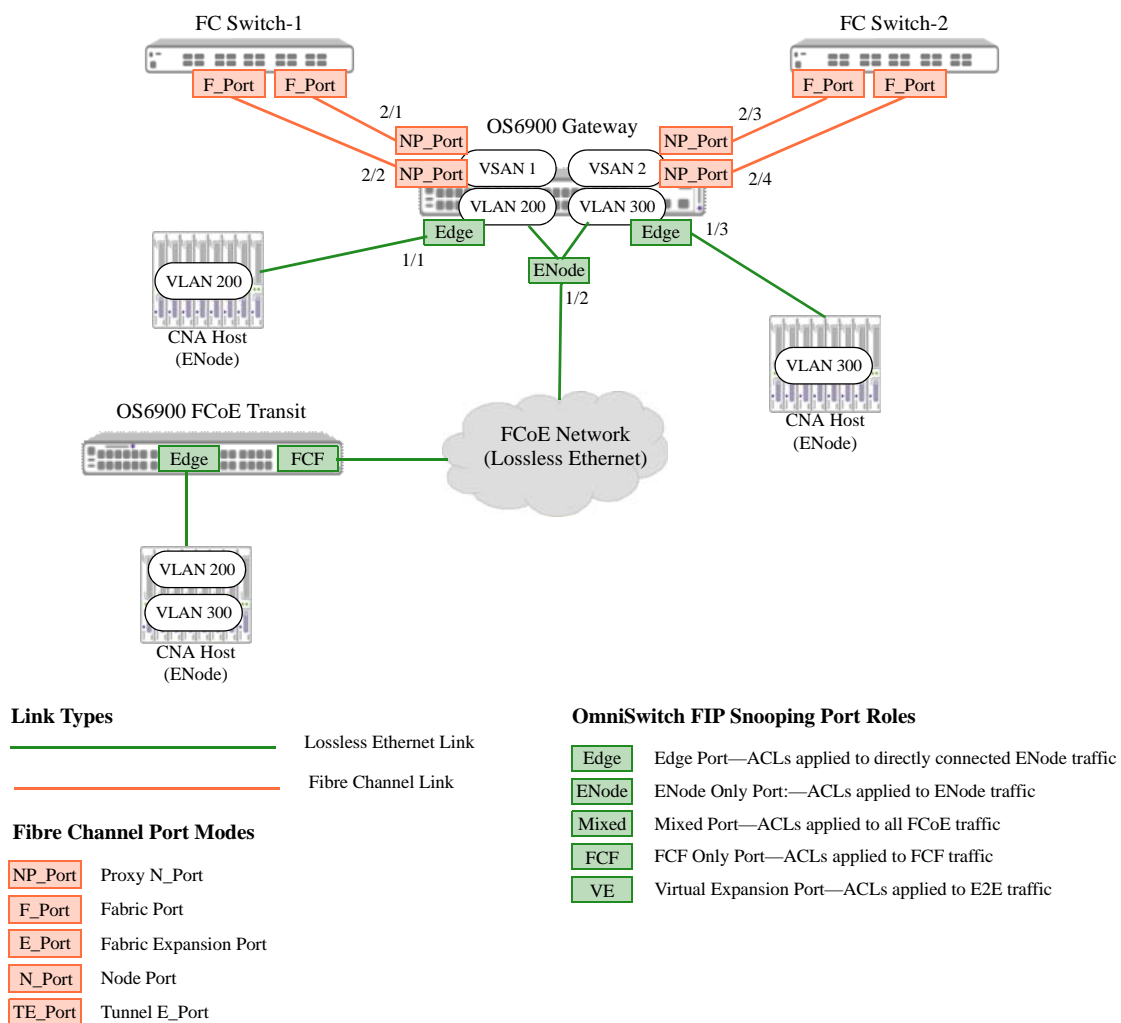
- **VSAN**—carries native FC traffic to and from FC SAN devices accessing the gateway through the OmniSwitch FC ports assigned to the VSAN. Initially all FC ports are assigned to a default VSAN (VSAN 1). Additional VSANs are configurable to define additional gateway fabrics, similar to how VLANs are used to define additional broadcast domains.
- **FCoE VLAN**—carries FCoE traffic to and from FCoE network devices accessing the gateway through OmniSwitch FCoE ports associated with the FCoE VLAN.
- **FCoE ports**—OmniSwitch 10G Ethernet ports configured with a lossless Data Center Bridging profile and an FCoE port role for FIP snooping. Each interface is assigned to an FCoE VLAN.
- **FC ports**—OmniSwitch Fibre Channel ports configured to operate in N\_Port proxy mode or F\_Port proxy mode. Each FC interface is assigned to a single VSAN, but it is possible to assign multiple FC ports to the same VSAN.
- **VSAN-to-FCoE VLAN mapping**—an association between one OmniSwitch VSAN and one FCoE VLAN.
  - Native FC traffic received on FC ports associated with the mapped VSAN is converted to FCoE traffic (encapsulated FC frames) and forwarded on the FCoE ports associated with the FCoE VLAN.

- FCoE traffic received on FCoE ports associated with the mapped FCoE VLAN is converted to native FC traffic.

**Note.** The OmniSwitch FCoE/FC gateway fabric is only used for N\_Port proxy and F\_Port proxy operations. The E\_Port proxy mode does not use the FCoE gateway fabric to establish an E2E tunnel path.

## How it Works

Each VSAN-to-FCoE VLAN mapping and associated ports represents an independent fabric on the OmniSwitch gateway. Devices only communicate with other devices connected through the same fabric on the gateway switch. For example, the following illustration shows an OmniSwitch FCoE/FC gateway switch with two separate fabrics: one fabric flows through VSAN 1 and the second fabric flows through VSAN 2.



**Figure 1: OmniSwitch FCoE/FC Gateway Fabric**

In this example:

- VSAN 1 is associated with FC ports 2/1 and 2/2 and is mapped to FCoE VLAN 200, which is tagged with FCoE ports 1/1 and 1/2.



- VSAN 2 is associated with FC ports 2/3 and 2/4 and is mapped to FCoE VLAN 300, which is tagged with FCoE ports 1/2 and 1/3.
- VSAN 1 FC ports are connected to FC Switch 1; VSAN 2 FC ports are connected to FC Switch 2.
- The OS6900 gateway switch sends out two discovery advertisement messages: one message for VSAN 1 and one message for VSAN 2. *Each active VSAN represents a separate FCoE forwarder.*
- ENode requests with VLAN ID 200 are sent to FC Switch X and ENode requests with VLAN 300 are sent to FC Switch Y.
- Communication between VSANs (fabrics) is not allowed. As a result, VLAN 200 traffic cannot reach FC Switch Y and VLAN 300 traffic cannot reach FC Switch X.

## Using the N\_Port Proxy Mode

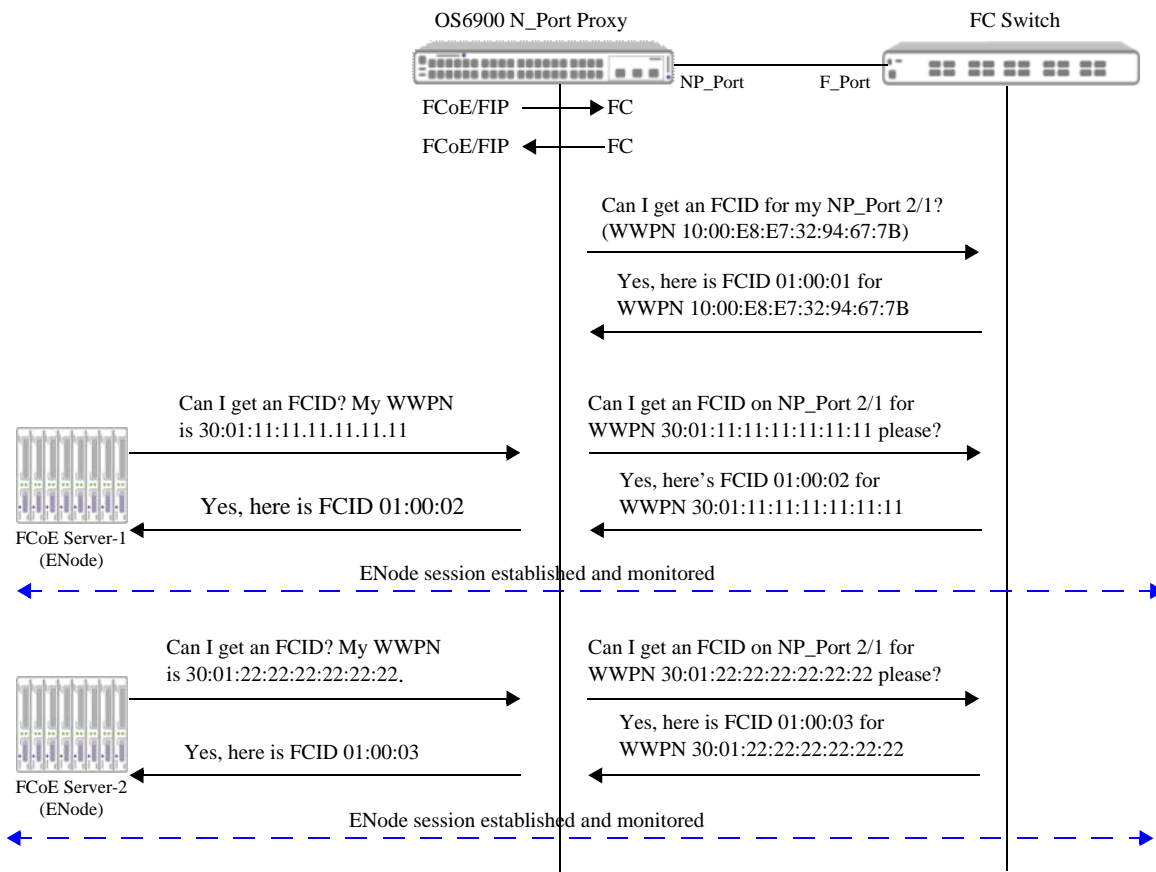
The OmniSwitch FCoE/FC gateway utilizes N\_Port virtualization functionality to allow the switch to serve as an N\_Port proxy for FCoE nodes (ENodes) accessing an FC SAN over an FCoE network.

- Fabric log in requests and log in accept messages are relayed between ENodes and FC switches through the gateway switch.
- An FC switch connected to the gateway switch is presented with a proxy N\_Port (NP\_Port) that can request multiple N\_Port IDs on the same physical port. To the FC switch, the OmniSwitch gateway is just another NPIV-enabled host. As a result, the FC switch should support NPIV.
- An ENode accessing the FC switch fabric through the gateway switch is presented with a VF\_Port upon successful fabric login.

As an N\_Port proxy, the OmniSwitch aggregates FCoE node (ENode) fabric login requests and subsequent N\_Port sessions through one physical OmniSwitch FC port. Each session is still identified by the WWPN and FCID assigned to the ENode even though the session is part of an aggregate flow.

The FC switch that provides login services through the OmniSwitch NP\_Port treats each ENode session as a separate virtual N\_Port connection. This helps to conserve on the number of physical ports required to handle ENode traffic, while making it possible to still track and service VMs, users, or applications from one or more ENodes on an individual basis.

The following diagram provides a high-level overview of the NPIV functionality the FCoE/FC Gateway provides when the switch is operating in the N\_Port proxy mode:



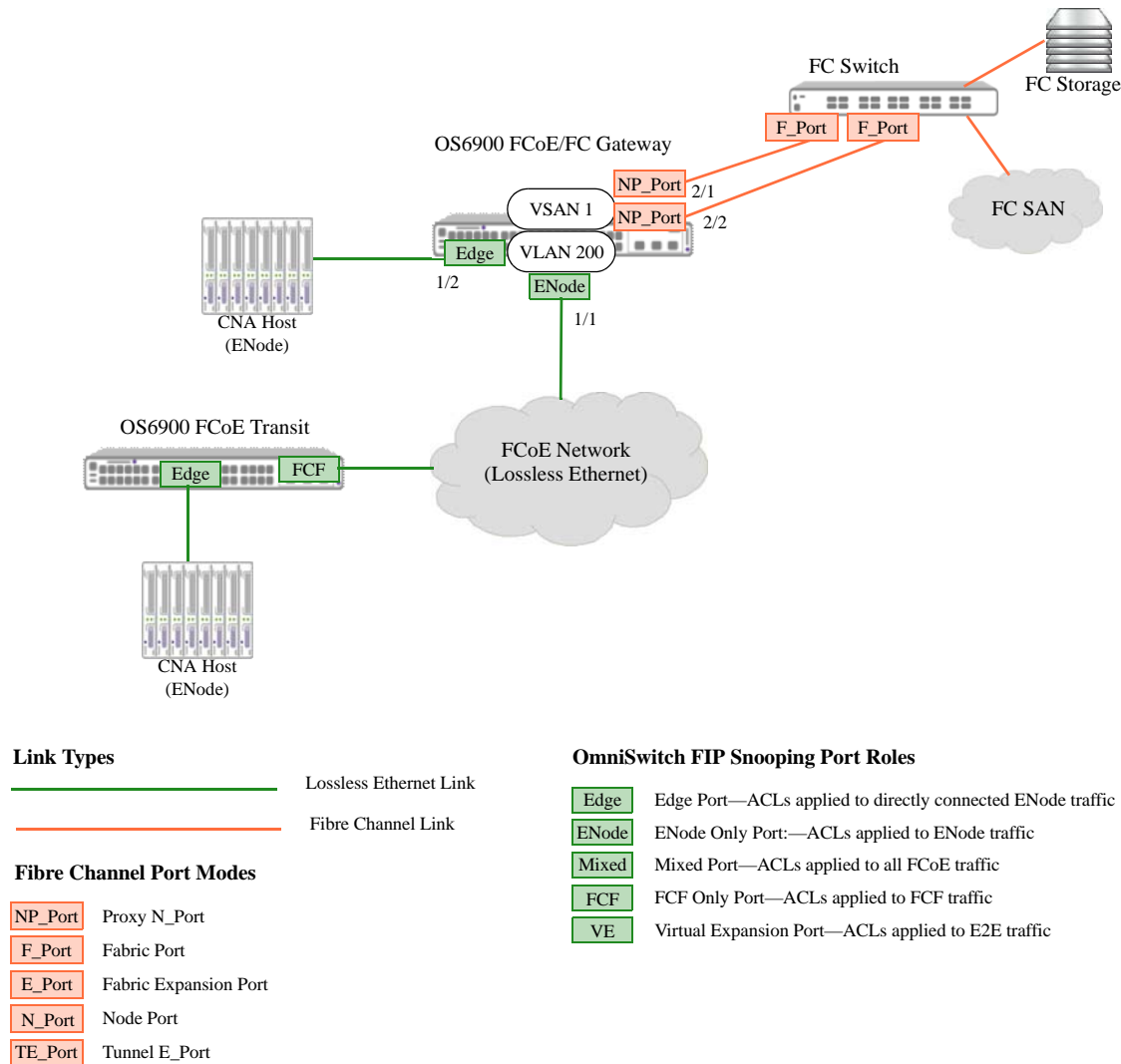
**Figure 2: OmniSwitch N\_Port Proxy Overview**

In this diagram,

- 1 The OS6900 port connecting to the FC switch is an FC port configured to operate as an NP\_Port.
- 2 When the NP\_Port comes up, the port attempts a fabric login through the F\_Port on the FC switch.
- 3 Upon a successful login, the NP\_Port is assigned a Fibre Channel ID (also called an N\_Port ID).
- 4 The NP\_Port can then request additional N\_Port IDs for ENodes over the same physical port

The N\_Port proxy mode is activated on the switch when an FCoE VLAN is mapped to a VSAN, and at least one FC interface assigned to the mapped VSAN is configured as a proxy node port (NP\_Port).

The following diagram shows an example network topology in which an OmniSwitch FCoE/FC gateway is operating as an N\_Port proxy to allow ENodes to log into an FC SAN:



**Figure 3: OmniSwitch N\_Port Proxy (NPIV) Example**

In this example configuration,

- The OmniSwitch FCoE/FC gateway fabric consists of VSAN 1 mapped to FCoE VLAN 200, NP\_Ports 2/1 and 2/2 assigned to VSAN 1, and FCoE ports 1/1 and 1/2 tagged with FCoE VLAN 200.
- FCoE VLAN 200 is a VLAN dedicated to carrying FCoE and FIP traffic between ENodes and an FCF through an FCoE network. The OS6900 transit switch is part of the FCoE network on which FCoE VLAN 200 is also configured.
- Port 1/1 is configured as an FCoE ENode-only port; and port 1/2 is configured as an FCoE edge port.
  - If the load balancing method for the switch is set to dynamic or dynamic re-order, ACLs are needed irrespective of port roles. An ACL entry is created for sessions established over FCoE ports configured as edge, ENode-only, mixed, and trusted ports.

- If the load balancing method for the switch is set to ENode-based or FCoE ports are statically mapped to FC ports, then an ACL entry is created for sessions established over edge ports only (similar to FIP Snooping).
- FCoE traffic received from ENode devices (CNA hosts) on FCoE ports 1/1 and 1/2 is converted to FC traffic and sent out on FC port 2/1 or port 2/2 to the FC switch.
- FC traffic received from the FC switch on NP\_Ports 2/1 and 2/2 is converted to FCoE traffic and sent out on FCoE port 1/1 or 1/2.

See [“Configuring an N\\_Port Proxy Operation” on page 5-23](#) for more information about configuring the N\_Port proxy mode on an OmniSwitch FCoE/FC gateway.

## How it Works

When an OmniSwitch FC port is configured to operate as an NP\_Port, the OmniSwitch gateway automatically initiates a fabric login for the NP\_Port. If the fabric login is successful, the FC switch assigns a unique Fibre Channel ID (FCID) to the NP\_Port. At this point, the N\_Port proxy mode is active on the NP\_Port.

Once the N\_Port proxy mode is active and there is a VSAN-to-FCoE VLAN mapping, the OmniSwitch gateway can provide services to ENodes by participating in FCoE VLAN discovery and FCF discovery by sending discovery advertisement messages.

- When an FCoE VLAN discovery message is received from an ENode, the OmniSwitch gateway will respond back with the FCoE VLAN ID (for example, FCoE VLAN 200 as shown in the diagram on [page 5-9](#)).
- When an FCF discovery solicitation message from an ENode is received, the OmniSwitch gateway will respond back with a unicast discovery advertisement message.

The OmniSwitch FCoE/FC gateway forwards multicast discovery messages initiated by an ENode within the designated FCoE VLAN. If this VLAN also provides access to FCFs or other NPIV gateways, the ENode may receive replies from multiple FCFs or gateways, including the OmniSwitch. The FCF replies contain a FIP priority, a verified maximum FCoE frame size and an availability bit (A-bit) setting. Based on this information, the ENode selects one of the FCFs for FC fabric login (FLOGI).

- If the OmniSwitch gateway is not selected, then the switch will serve as an FCoE transit switch (FIP snooping bridge) and forward the ENode traffic. No N\_Port proxy functionality is provided for the ENode.
- If the ENode does select the OmniSwitch gateway, the ENode sends a unicast FIP FLOGI request to the OmniSwitch.

In the example shown on [page 5-9](#), the OmniSwitch FCoE/FC gateway is the selected FCoE forwarder for both ENodes. When the OmniSwitch receives a FIP FLOGI request with VLAN 200 from an ENode, the OmniSwitch creates an FC fabric discovery (FDISC) message and transmits the message on one of the NP\_Ports in VSAN 1.

After the FDISC message is sent, the OmniSwitch gateway may receive an FC Link Service Accept (LS\_ACC) or Link Service Reject (LS\_RJT) reply from the FC switch connected to the NP\_Port on which the FDISC message was sent.

- If an FC LS\_ACC message is received, the OmniSwitch gateway will generate a unique VN\_Port MAC address (combination of FC\_MAP + FCID), convert the FC LS\_ACC message into a FIP LS\_ACC message containing the VN\_Port MAC address and then forward the LS\_ACC to the ENode.

- If an FC LS\_RJT message is received from the FC switch, the OmniSwitch gateway will simply convert the message to a FIP LS\_RJT message and then forward the message to the ENode.

When the ENode successfully logs into the fabric, the OmniSwitch gateway will dynamically install ACLs (FIP snooping) on the FCoE port where the FLOGI request from the ENode was received. In addition, the OmniSwitch gateway will monitor the established login connections as follows:

- The OmniSwitch will send out periodic multicast discovery advertisements to the ALL-ENODE-MAC to notify ENodes of its availability for new ENodes and its reachability for ENodes with sessions already established through the gateway.
- The OmniSwitch listens for ENode and VN\_Port keep-alive messages. If keep-alive messages are not received, the switch will clear all the corresponding sessions.
- There are two types of logout events that will cause the switch to clear all corresponding sessions:
  - An explicit logout event triggered by an ENode, FC switch, or OmniSwitch gateway.
  - An implicit logout event triggered by a link down or missed keep-alive messages.

## N\_Port Proxy Load Balancing

When more than one FC port is assigned to the same VSAN, the switch will automatically apply a load balancing algorithm to determine on which NP\_Port the FC FDISC packet is forwarded to an FC switch. If one of the FC ports goes down, then all sessions on that port are torn down and logged in again on one of the other available FC ports in the same VSAN.

There are four load balancing methods supported: dynamic (the default), dynamic reorder, dynamic ENode based, and static. All three of the load balancing algorithms are dynamic, in that the process used to distribute ENode sessions automatically determines which NP\_Ports to use. However, it is also possible to manually configure a static association between an FCoE port and an NP\_Port.

The active load balancing algorithm is globally configured and applied on all OmniSwitch NP\_Ports; load balancing is not applied on OmniSwitch F\_Ports. In addition, the port selection is made based on the incoming ENode FLOGI requests, not on the outgoing FDISC messages for those same requests.

This section provides an overview of each load balancing method, including static FCoE port assignment.

### Dynamic Load Balancing

By default, the dynamic load balancing algorithm is used. When this method is applied, the switch selects the NP\_Port with the least number of logins. If all the NP\_Ports in the VSAN have the same number of logins, then a round-robin selection method is used.

The dynamic load balancing method is only applied to new login requests received; none of the sessions already active on the NP\_Ports are disturbed. For example, if there is one NP\_Port with active sessions and another NP\_Port is added to the same VSAN, the existing sessions on the first NP\_Port are left alone. New requests are forwarded on the new NP\_Port because initially it has the least number of logins.

### Dynamic Reorder Load Balancing

When the dynamic reorder method is selected, load balancing is applied to all ENode login sessions (not just the new ones) to ensure an even distribution of session across all NP\_Ports in the same VSAN. This means that the switch might have to tear down some of the existing sessions and re-establish those sessions on a different NP\_Port.

For example, if an NP\_Port has two active sessions and another NP\_Port is added to the same VSAN, the switch explicitly logs out one of the ENode sessions and sends a CVL to the ENode that was logged out.

When the ENode receives the CVL, will send another FLOGI request that the switch will now redirect onto the added NP\_Port. The other ENode session continues on the first port.

## ENode-based Load Balancing

When the ENode-based load balancing method is selected, each NP\_Port in the VSAN will send a multi-cast discovery advertisement to all ENodes in the mapped FCoE VLAN. The FCF MAC address in the discovery advertisement is the MAC address of the FC port. The ENode then decides which NP\_Port to use for the login.

Similar to the dynamic load balancing method, the ENode-based method does not disturb existing ENode sessions.

## Static FCoE Port Mapping

Static mapping of an FCoE port to an NP\_Port ensures that all ENode sessions on the FCoE port are specifically directed to the mapped NP\_Port. No other NP\_Port is eligible to receive sessions from this FCoE port. For example, if FCoE port 1/1 is mapped to NP\_Port 2/1, then all ENode FLOGI requests are automatically sent to NP\_Port 2/1. There is no consideration for the number of sessions already on that port, or on any other NP\_Port in the VSAN.

---

**Note.** Use caution when configuring this method. A static assignment of an FCoE port to an NP\_Port overrides dynamic load balancing.

---

For more information about configuring the four load balancing methods, see [“Configuring N\\_Port Proxy Load Balancing” on page 5-24](#).

## Using the F\_Port Proxy Mode

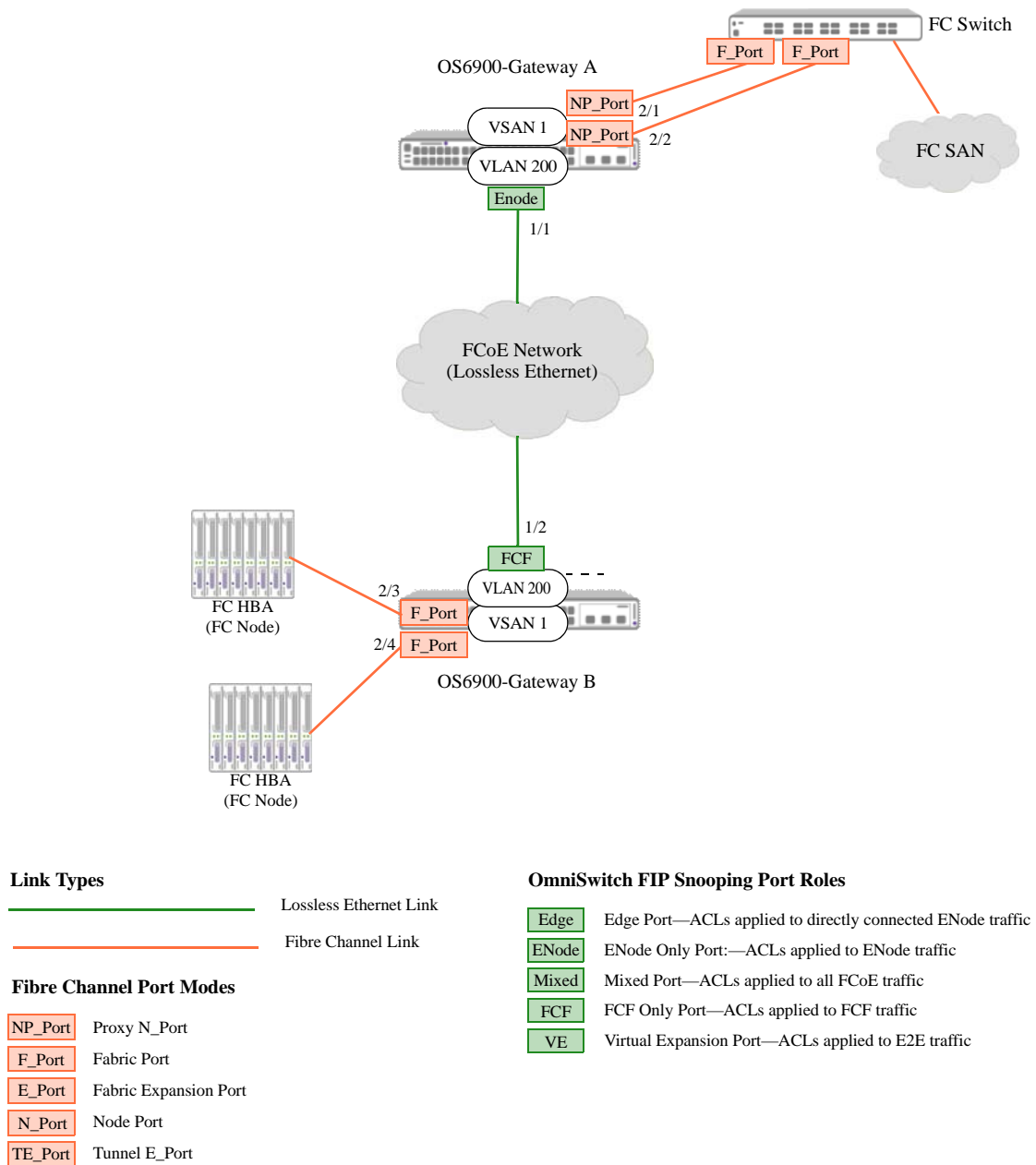
The F\_Port proxy functionality allows the OmniSwitch gateway to provide an F\_Port fabric to FC nodes (HBA N\_Ports). On the FC-facing side of the gateway switch, FC nodes are presented with F\_Ports. On the FCoE-facing side of the gateway switch, FCoE forwarders or other NPIV proxy gateways are presented with VN\_Ports.

Utilizing the F\_Port proxy functionality, an OmniSwitch gateway can perform the following tasks on behalf of FC nodes connected to the gateway in an FCoE network:

- Discovers FCoE forwarders and NPIV proxy gateways on the FCoE network and selects one of these devices to receive converted FLOGI requests from the attached FC nodes.
- Converts FC node FLOGI requests to FIP FLOGI requests and forwards them to the selected FCoE forwarder or NPIV proxy gateway through the FCoE network.
- Converts FIP login accept or reject responses received from the FCoE forwarder to FC responses that are then forwarded to the originating FC node.
- Sends periodic keep-alive messages to the FCoE forwarder to maintain virtual connections between the attached FC nodes and the FC SAN switch.

The reverse NPIV proxy functionality is activated on the switch when FIP snooping is configured and enabled, an FCoE VLAN is mapped to a VSAN, and at least one FC interface assigned to the mapped VSAN is configured as a fabric port (F\_Port).

The following diagram shows an example network topology in which an OmniSwitch FCoE/FC gateway is operating as an F\_Port proxy to allow FC nodes to log into an FC SAN over an FCoE network:



**Figure 4: OmniSwitch F\_Port Proxy (Reverse-NPIV) Example**

In this example configuration,

- The OS6900-Gateway A switch is operating as an N\_Port proxy. The OS6900-Gateway B switch is operating as an F\_Port proxy switch.
- The FCoE/FC gateway fabric connects the two gateway switches through the FCoE (lossless Ethernet) network, allowing the FC nodes to access the FC switch.
- The FCoE/FC gateway fabric is configured as follows on each OmniSwitch gateway:
  - VSAN 1 is mapped to FCoE VLAN 200.
  - FC ports are assigned to VSAN 1.

- FCoE ports are assigned to VLAN 200.
- FCoE VLAN 200 is a VLAN dedicated to carrying FCoE and FIP traffic between the two gateways.
- On the Gateway B switch, FC ports 2/3 and 2/4 are configured to operate in the fabric port (F\_Port) mode and are also assigned to VSAN 1. These ports are connected to FC nodes. The Gateway B switch will provide the connected FC nodes with access to FC devices across the FCoE network.
- On the Gateway A switch, NP\_Ports 2/1 and 2/2 are assigned to VSAN 1. These ports will provide N\_Port proxy access to the FC switch.
- FCoE ports 1/1 and 1/2 are tagged with FCoE VLAN 200.
  - Port 1/1 is configured as an FCoE ENode-only port, which means that dynamic ACLs are applied to ENode traffic received on that port from the R-NPIV proxy switch.
  - Port 1/2 is configured as an FCoE FCF-only port, which means that dynamic ACLs are applied to traffic received directly from the NPIV proxy switch.
- FC traffic received from the FC nodes on F\_Ports 2/3 and 2/4 is converted to FCoE traffic and sent out FCoE port 1/2 to the Gateway A switch.
- FCoE traffic received from the Gateway B switch on FCoE port 1/1 is converted to FC traffic and sent out on NP\_Ports 2/1 or 2/2 to the FC switch.

See [“Configuring an F\\_Port Proxy Operation” on page 5-25](#) for more information about configuring the F\_Port proxy mode on an OmniSwitch FCoE/FC gateway.

## How it Works

This subsection describes how the OmniSwitch gateway provides F\_Port proxy services that connect FC node devices with FC switches across an FCoE network. The process described is based on the example shown in [“Figure 4: OmniSwitch F\\_Port Proxy \(Reverse-NPIV\) Example” on page 5-13](#).

- When the FC ports 2/3 and 2/4 are configured to operate as F\_Ports and are up and running, the F\_Port proxy functionality is implicitly enabled on those ports. One of the main functions of the F\_Port proxy switch is to find FCoE forwarders and/or NPIV gateways on behalf of the FC nodes connected to the F\_Ports.
- The FCoE/FIP snooping functionality on the OmniSwitch gateway allows the switch to participate in the FIP FCF discovery process to locate FCoE forwarders and NPIV gateway switches in the FCoE network. In the example, the F\_Port proxy switch listens on FCoE VLAN 200 for FIP FCF discovery advertisements and finds the OS6900 N\_Port proxy gateway.
- The F\_Port proxy switch uses FCoE VLAN 200 for FIP FCF discovery because this VLAN is mapped to VSAN 1 as part of the FCoE/FC gateway fabric. Therefore, the F\_Port proxy switch did not have to participate in the FIP VLAN discovery process to find VLAN 200 on which to transport FCoE traffic.
- If the F\_Port proxy switch found more than one compatible FCF or gateway, the switch maintains a list of these devices by continuing to monitor discovery advertisements. If the selected FCF or gateway switch goes down, another device is selected from the list and all sessions are torn down and instantiated again with the new FCF or gateway.
- When the FC nodes connected to F\_Ports 2/3 and 2/4 attempt to log in to the FC fabric by initiating FC FLOGI requests, the F\_Port proxy switch converts the requests to FIP FLOGI then forwards the FIP FLOGI requests on FCoE VLAN 200 to the N\_Port proxy switch.
- If an FC node sends FDISC messages from an NPIV HBA, these messages are converted in the same manner as the FLOGI messages are converted for transport to and from the FC fabric.



- The N\_Port proxy switch converts the FIP FLOGIs received to FC FLOGIs and relays the FC FLOGIs on VSAN 1 to the FC switch. The FC switch then processes the log in request and responds with an FC accept (LS\_ACC) or reject (LS\_RJT) message.
- When the N\_Port proxy switch receives a response from the FC switch, the gateway converts the response to a FIP response and forwards the response on FCoE VLAN 200 to the F\_Port proxy switch. The F\_Port proxy switch then converts the response back to an FC frame and forwards the response on VSAN 1 to the appropriate FC node.
- If the FC node login was accepted and a session established, the F\_Port proxy switch maintains the virtual connection by sending periodic keep-alive messages to the N\_Port proxy switch on behalf of the FC nodes. This is because an HBA on an FC node does not maintain keep-alive messages.
- If the F\_Port proxy switch determines the N\_Port proxy switch is no longer reachable, virtual connections established with that switch are torn down. The FC nodes can then initiate login requests again and the F\_Port proxy will select another FCoE forwarder or NPIV gateway to establish new virtual connections.

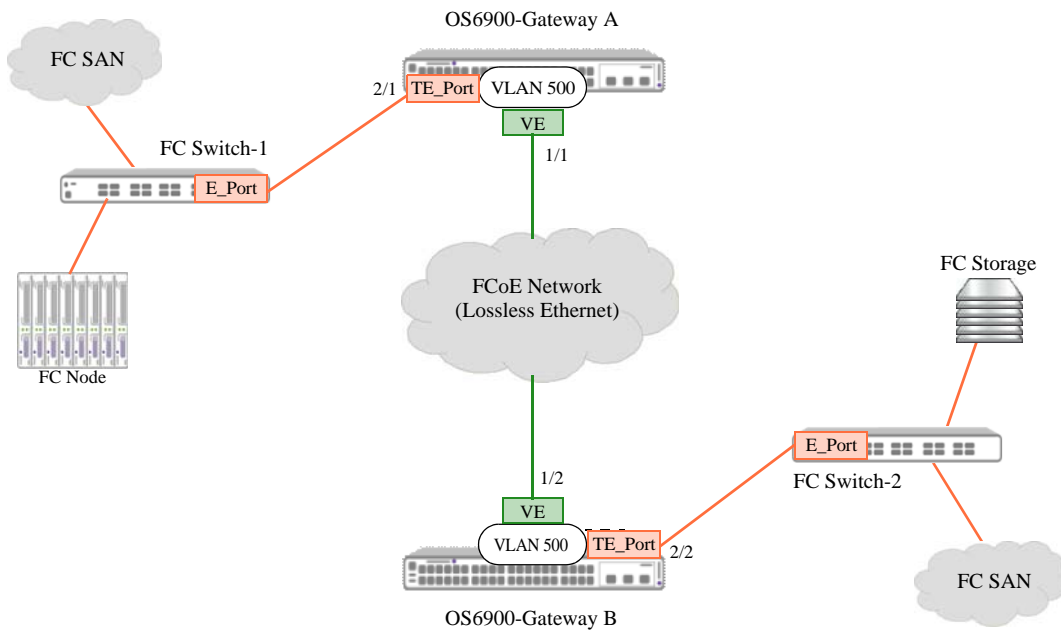
## Using the E\_Port Proxy Mode

E\_Port-to-E\_Port (E2E) tunneling provides a method for transparently connecting native FC switches over an FCoE network. The OmniSwitch FCoE/FC gateway serves as an E\_Port proxy through the configuration of tunnel expansion ports (TE\_Ports) and virtual expansion ports (VE\_Ports) to provide connectivity between E\_Ports on FC switches over an FCoE network.

There are two ways to establish an E\_Port proxy operation to provide an E2E tunnel between two FC switch E\_Ports:






- **VE\_Port tunneling between two switches**—the tunnel is established through an FCoE network between a VE\_Port on one switch and a VE\_Port on another switch. Each VE port is part of a tunnel endpoint that is mapped to an OmniSwitch TE\_Port via an FCoE VLAN (both port types are associated with the same FCoE VLAN). The TE\_Port connects directly to an FC switch E\_Port.
- **TE\_Port tunneling on the same switch**—an internal tunnel is established between two OmniSwitch TE\_Ports on the same switch (or in a virtual chassis configuration). All tunnel traffic is switched across the internal tunnel, instead of over an FCoE network.

The following diagram shows an example of an OmniSwitch VE\_Port tunnel configuration:






**Link Types**

	Lossless Ethernet Link
	Fibre Channel Link

**Fibre Channel Port Modes**

	Proxy N_Port
	Fabric Port
	Fabric Expansion Port
	Node Port
	Tunnel E_Port

**OmniSwitch FIP Snooping Port Roles**

	Edge Port—ACLs applied to directly connected ENode traffic
	ENode Only Port:—ACLs applied to ENode traffic
	Mixed Port—ACLs applied to all FCoE traffic
	FCF Only Port—ACLs applied to FCF traffic
	Virtual Expansion Port—ACLs applied to E2E traffic

**Figure 5: OmniSwitch E\_Port Proxy (E2E Tunnel) Example**

In this example configuration,

- An E2E tunnel connects the two FC SANs through the FCoE network.
- FCoE VLAN 500 is configured on both switches and identifies the tunnel path through the FCoE network.
- OS6900 10G Ethernet ports 1/1 and 1/2 are configured as FCoE VE\_Ports.
- OS6900 FC ports 2/1 and 2/2 are configured as tunnel expansion ports (TE\_Ports) and connect to an E\_Port on an FC switch.
- The E2E tunnel path is defined by associating TE\_Port 2/1 and VE\_Port 1/1 with FCoE VLAN 500 on Gateway A and associating TE\_Port 2/2 and VE\_Port 1/2 with FCoE VLAN 500 on Gateway B.
- Traffic from each FC SAN is tunneled to and from the OS6900 FC TE\_Ports through the FCoE VE\_Ports. Each FC switch thinks it is directly connected to the other FC switch; the underlying tunnel through the FCoE network is transparent.

See [“Configuring an E\\_Port Proxy Operation” on page 5-26](#) for more information about configuring the E\_Port proxy mode on an OmniSwitch FCoE/FC gateway.

## How it Works

In an FC SAN, the FC switches provide connectivity for nodes accessing devices in the SAN. FC switches are connected to each other through ports designated as expansion ports (E\_Ports). This type of port allows two FC switches to set up an inter-switch link (ISL) between each other.

In an FCoE network virtual expansion ports (VE\_Ports) provide connectivity between FCoE forwarders or gateways, similar to E\_Port connectivity between native FC switches. This type of connectivity provides the ability to extend E\_Port connections across an FCoE network by tunneling E\_Port traffic between VE\_Ports.

On the OmniSwitch, FCoE ports that will tunnel E\_Port traffic through the FCoE network, are configured as VE\_Ports. An OmniSwitch FC port that will connect directly to an E\_Port on an FC switch is configured as a tunnel expansion port (TE\_Port). These two port types and a common FCoE VLAN are used to define the E2E tunnel endpoints.

As shown in the Figure 5 example [page 5-16](#), two OmniSwitch FCoE/FC gateways are configured with a single VE\_Port and FCoE VLAN 500. In addition, both switches have an FC TE\_Port connected directly to an FC switch. On each OmniSwitch, an E2E tunnel endpoint is defined by the association of the VE\_Port and the TE\_Port to the FCoE VLAN 500.

Defining the E2E tunnel endpoints does not establish a tunnel session. Once the required tunnel components are configured and active, successful exchange of exchange link parameters (ELP) between a TE\_Port and a VE\_Port will establish the tunnel session.

- VE\_Ports participate in the FIP FCF discovery process. In the example, the VE\_Ports discover the other OS6900 gateway switch on FCoE VLAN 500.
- The OS6900 gateway will convert ELP received on the TE\_Port to FIP ELP and relay the FIP ELP to the selected FCF.
- The receiving VE\_Port will validate the FIP ELP request, remove the Ethernet header, and send the FC ELP request to the corresponding TE\_Port.
- If the ELP exchange is successful, the tunnel session is instantiated on the VE\_Ports.

Once the tunnel session is established, the VE\_Ports will maintain the link by sending periodic FIP discovery messages. At this point, all ISL data traffic is allowed to pass through the tunnel.

## FIP Packet Processing

The following table shows how the OmniSwitch FCoE/FC gateway switch identifies FIP snooping traffic (FCoE packets) received on FCoE ports to determine which gateway function will process the traffic:

Packet Type	Verifies ...	Assigns Packet to ...
MDS (multicast discovery solicitation)	FCF (F) bit setting	E_Port proxy if the F bit is set to one.  N_Port proxy if the F bit is set to zero.
UDS (unicast discovery solicitation)	Unicast destination MAC	F_Port proxy if destination MAC is an FC F_Port.  E_Port proxy if destination MAC is an FC TE_Port.
MDA (multicast discovery advertisement)	Multicast destination MAC	E_Port proxy if the destination MAC is ALL-FCF-MAC.  F_Port proxy if the destination MAC is ALL-ENODE-MAC.
CVL (clear virtual link)	Unicast destination MAC	F_Port proxy if the destination MAC is an FC F_Port.  E_Port proxy if the destination MAC is an FC TE_Port.
FIP FLOGI / FIP FDISC	Packet type	N_Port proxy
FIP FLOGO (fabric logout)	Packet type	N_Port proxy
FLOGI/FIDSC LS_ACC FLOGI/FDISC LS_RJCT  FLOGO LS_ACC FLOGO LS_RJCT	Packet type	F_Port proxy
VLAN Discovery	F-bit setting	E_Port proxy if the F-bit is set to one.  N_Port proxy if the F-bit is set to zero.
FIP ELP  SW_ACC SW_RJCT	Packet type	E_Port proxy
Keep-alive	Packet type	N_Port proxy

# Interaction with Other Features

This section contains important information about how other OmniSwitch features interact with the FCoE/FC gateway features. Refer to the specific chapter for each feature to get more detailed information about how to configure and use the feature.

## FIP Snooping

The OmniSwitch FCoE/FC gateway functionality is a superset of the OmniSwitch FIP snooping implementation. FIP snooping ensures the security of the FCoE traffic entering the gateway switch. The gateway switch converts FCoE encapsulated frames to FC frames for forwarding onto an FC SAN; the gateway also converts FC frames to FCoE frames for forwarding onto an FCoE network.

Configuring the following FIP snooping components is *required* to support and define the FCoE/FC gateway fabric on the OmniSwitch:

- **FCoE VLAN**—A type of VLAN that is dedicated to carrying only FCoE and FIP traffic. ENodes use this type of VLAN to transmit and receive FCoE and FIP traffic and establish virtual links to an FC SAN through the NPIV gateway.

Manual configuration of the FCoE VLAN is required along the FCoE network path and on the gateway switch. However, the ENode may invoke FIP VLAN discovery to discover the FCoE VLANs in the network. If not, manual configuration of the FCoE VLANs may also be required on the appropriate ENodes

If there is no FCoE/FC gateway functionality mapped to an FCoE VLAN, then only FIP snooping is applied to traffic forwarded on the VLAN even if the VLAN is configured on an FCoE/FC gateway switch.

- **FCoE Port Roles**—Determines the types of ACLs that are dynamically generated and applied to FCoE traffic that ingresses on the FCoE port. For the FCoE/FC gateway E2E tunneling feature, the virtual expansion port (VE\_Port) role is used to allow exchange link parameters (ELP) to travel between FC switches through an FCoE network. The mixed port role is used when the FCoE port will carry a mixture of traffic (for example, E2E tunnel, ENode, FCF only traffic) all on the same port.

Having a good understanding of FCoE and FIP snooping is highly recommended when attempting to configure FCoE/FC gateway features. For more information, see [Chapter 4, “Configuring FIP Snooping.”](#)

## Data Center Bridging (DCB)

FCoE requires an underlying lossless Ethernet network. The OmniSwitch supports the following Data Center Bridging (DCB) protocols that extend Ethernet capabilities to support the convergence of storage and data in virtualized networks:

- **Priority-Based Flow Control (PFC)**—Based on the IEEE 802.1Qbb standard, PFC pauses traffic based on congestion priority instead of blocking the entire link when congestion occurs. Allows lossless and lossy traffic with different priorities on the same physical port.
- **Enhanced Transmission Selection (ETS)**—Based on the IEEE 802.1Qaz standard, ETS provides a common framework for dynamic bandwidth management. ETS groups related traffic into priority groups (Traffic Classes) to which bandwidth guarantees and scheduling are applied.
- **Data Center Bridging Exchange (DCBX)**—Based on the IEEE 802.1Qaz standard, DCBX uses the Link Layer Discovery Protocol (LLDP) to exchange and negotiate PFC and ETS information between

two directly connected peer switches. Enabled by default, DCBX is responsible for auto-negotiation and auto-configuration of link parameters for DCB functions.

---

**Note.** A custom DCB profile for each of the FC ports is required to ensure an end-to-end lossless path through the FCoE/FC gateway. See [“Configuring FC Ports” on page 5-22](#) for more information.

---

For more information about DCB see [Chapter 2, “Configuring Data Center Bridging,”](#)

## Hardware

This implementation of OmniSwitch FCoE/FC gateway functionality requires support for both 10G Ethernet and Fibre Channel ports. The OmniSwitch 6900 provides both with the addition of the OS-XNI-U12E module with SFP-FC-SR transceiver. Fibre Channel support is not available on the OmniSwitch 10K.

The Fibre Channel ports are not available for use until they are configured to operate in a specific Fibre Channel mode (NP\_Port, F\_Port, or TE\_Port). Once configured, each port is assigned a 64-bit world wide port name (WWPN) that is comprised of “10:00” plus the MAC address of the port. For example, 10:00:e8:e7:32:94:67:7d.

The World-wide Node Name (WWNN) for the OmniSwitch 6900 is comprised of “10:00” plus the next available increment of the switch base MAC address. For example, if the switch MAC address is e8:e7:32:6c:4a:39, then the WWNN for the switch is 10:00: e8:e7:32:6c:4a:3a.

# FCoE/FC Gateway Configuration Guidelines

OmniSwitch FCoE/FC gateway operations are not automatically activated for the switch; there is no single command to enable or disable gateway functionality. Instead, the configuration of the following software components enables one or more of the supported gateway operations:

- **FIP snooping**—FCoE/FC gateway functionality requires an active FIP snooping configuration. FIP Snooping ensures the security of an FCoE network.
- **Virtual Storage Area Network (VSAN)**—an FC port is assigned to a VSAN to create an NP\_Port or F\_Port connection to an FC switch or node in that VSAN. Not required for E2E Tunnel configuration.
- **VSAN-to-FCoE VLAN mapping**—identifies the FCoE/FC gateway fabric for the ENode or FC node login process via the FCoE VLAN. Not required for E\_Port proxy (E2E Tunnel) configuration. See [“OmniSwitch FCoE/FC Gateway Fabric” on page 5-5](#) for more information.
- **FCoE port role**—an FCoE port serves as an E2E tunnel endpoint in an FCoE network only when the port is configured as a virtual E\_Port (VE\_Port). The role of other FCoE ports is configured based on the FIP snooping configuration for the gateway switch.
- **FC port mode**—the operational mode of the FC port determines the type of gateway functionality provided on that port. There are three modes supported: N\_Port proxy, F\_Port proxy, and E\_Port proxy. See [“Configuring FC Ports” on page 5-22](#) for more information.

Consider the following general guidelines when configuring an OmniSwitch FCoE/FC gateway:

- FCoE/FC gateway functionality is only supported on an OmniSwitch 6900 equipped with an OS-XNI-U12E module with SFP-FC-SR transceiver. This module provides the physical Fibre Channel interfaces that connect the OmniSwitch to native FC switches or nodes.
- FIP snooping is a required component of the OmniSwitch FCoE/FC gateway functionality. An FCoE VLAN and one or more FCoE ports are used to configure the FCoE/FC gateway fabric. For more information about how to configure these items, see [Chapter 4, “Configuring FIP Snooping.”](#)
- The FCoE/FC gateway fabric defines the ports on which FC traffic is converted to FCoE traffic and FCoE traffic is converted to FC traffic. The fabric itself is defined through the association of the FC ports to an OmniSwitch VSAN and the association of FCoE ports to an FCoE VLAN. The FCoE VLAN is then mapped to the OmniSwitch VSAN to define the traffic path for converted FC or FCoE frames.
  - The FCoE/FC gateway fabric applies only to N\_Port and F\_Port proxy modes; the E\_Port mode defines the traffic path via an E2E tunnel, which does not require a VSAN association.
  - The VSAN-to-FCoE VLAN association is a one-to-one mapping only.
  - An OmniSwitch FC port is assigned to only one VSAN at a time; however, it is possible to assign multiple FC ports to the same VSAN.
- An FCoE VLAN is a required component for an N\_Port proxy operation, an F\_Port Proxy operation, and a VE\_Port E2E tunnel (not required for TE\_Port tunnels). It is possible to use the same FCoE VLAN in all cases, but only one VE\_Port tunnel per VLAN is allowed.
- An OmniSwitch VSAN only applies to the local switch configuration as a means to identify a local gateway fabric. There is no correlation between an OmniSwitch VSAN and VSANs created within a native FC SAN.
- A default VSAN 1 is provided on the switch and all FC ports are automatically assigned to that VSAN. Configuring additional VSANs is allowed to define additional gateway fabrics.

- An OmniSwitch VSAN is a required component for an N\_Port proxy and F\_Port proxy configuration, but a VSAN is not required to configure an E\_Port proxy tunnel endpoint.

## Configuring FC Ports

Configuring OS-XNI-U12E module ports with an SFP-FC-SR transceiver to ensure lossless gateway connectivity with FC SAN devices requires the following configuration tasks:

- 1 Assigning a Data Center Bridging (DCB) profile to each FC port.
- 2 Setting the SFP-FC-SR transceiver port type to Fibre Channel.
- 3 Configuring the Fibre Channel port mode to determine the FCoE/FC gateway operation provided on the port.

Note that the operational mode is configured on a per-port basis. This allows the OmniSwitch to provide multiple N\_Port proxy, F\_Port proxy, and E\_Port proxy operations on the same switch.

### Assigning a DCB Profile for FC Ports

Use the following steps to assign a DCB profile to each FC port on the switch:

- 1 Create the DCB profile using the `qos qsp dcb import` command with the **802.3x-pause** option. For example, the following command creates custom profile DCB 20, based on predefined profile 8:

```
-> qos qsp dcb 20 import qsp dcb 8 802.3x-pause
```

- 2 After creating the DCB profile, set the Priority Flow Control (PFC) willing bit to “no” on each FC port. For example:

```
-> qos qsi port 2/6 dcb dcbx pfc willing no
```

- 3 Disable PFC TLVs on each FC port. For example:

```
-> qos qsi port 2/6 dcb dcbx pfc tlv disable
```

- 4 Apply the custom DCB profile to each FC port. For example:

```
-> qos qsi port 2/6 qsp dcb 20
```

See [Chapter 2, “Configuring Data Center Bridging,”](#) for more information.

### Configuring the FC Port Type and Mode

By default, the port type for all OS-XNI-U12E module ports is set to Ethernet, even for ports equipped with the SFP-FC-SR transceiver. In addition, there is no FC operational mode set for each port. To enable FCoE/FC gateway functionality on these ports, use the **fibre-channel port mode** command. For example:

```
-> fibre-channel port 2/1 mode np
```

The **fibre-channel port mode** command sets the port type to Fibre Channel and configures the FC operational mode for the port. In the example, port 2/1 is configured as a Fibre Channel port that will provide N\_Port proxy operations for the gateway switch.



# Configuring an N\_Port Proxy Operation

The NPIV capabilities of the OmniSwitch gateway allows the switch to serve as an N\_Port proxy for ENode devices attempting to access FC switches over an FCoE network. The N\_Port proxy functionality is operational when the following components are configured:

- FIP snooping is configured and enabled for the switch.
- An FCoE VLAN is mapped to a VSAN.
- At least one FC port that is assigned to the mapped VSAN is configured to operate as a proxy node port (NP\_Port).
- At least one FCoE port is tagged with the FCoE VLAN.

The **fibre-channel port mode** command is used to configure an FC port to operate in the N\_Port proxy mode. For example, the following command configures FC port 2/1 to operate as an NP\_Port:

```
-> fibre-channel port 2/1 mode np
```

NP\_Port 2/1 will connect to a native FC switch to provide N\_Port login services for ENodes accessing the OmniSwitch FCoE/FC gateway through an FCoE network. To identify which ENode traffic is serviced on NP\_Port 2/1, complete the following steps:

**1** Create an OmniSwitch VSAN and map the VSAN to the FCoE VLAN on which ENode traffic is forwarded. In the following example, the **fibre-channel vsan** command is used to create VSAN 10 and the **fcoe vsan-map** command is used to map VSAN 10 to FCoE VLAN 500:

```
-> fibre-channel vsan 10  
-> fcoe vsan-map vsan 10 vlan 500
```

**2** Assign the NP\_Port to the VSAN that is mapped to the FCoE VLAN using the **fibre-channel vsan members** command. For example, the following command assigns NP\_Port 2/1 to VSAN 10:

```
-> fibre-channel vsan 10 members 2/1
```

**3** Configure the appropriate FCoE port role for the switch ports that will receive the FCoE traffic using the **fcoe role** command. For example, the following command configures port 1/1 and 1/2 as FCoE mixed ports:

```
-> fcoe port 1/1-2 role mixed
```

**4** Tag the FCoE ports configured in Step 3 with FCoE VLAN 500. For example:

```
-> vlan 500 members port 1/1-2 tagged
```

To display the N\_Port proxy configuration on the OmniSwitch:

- Use the **show fibre-channel vsan** command and the **show fibre-channel vsan members** command to display the VSAN configuration and the FC ports associated with the configured VSANs.
- Use the **show fibre-channel port** command to display the operational mode and active state for each FC port. An NP\_Port should be in an up state to trigger the N\_Port proxy mode service.
- Use the **show fcoe vsan-map** command to determine which FCoE VLANs are mapped to which OmniSwitch VSANs.

For more information about the N\_Port proxy feature, see [“Using the N\\_Port Proxy Mode” on page 5-7](#).

## Configuring N\_Port Proxy Load Balancing

When two NP\_Ports are assigned to the same VSAN, the OmniSwitch will load balance ENode FLOGI requests on the two NP\_Ports. By default, the dynamic load balancing algorithm is used. When this method is applied, the switch selects the NP\_Port with the least number of logins. If all the NP\_Ports in the VSAN have the same number of logins, then a round-robin selection method is used.

There are two other methods supported: dynamic re-ordering and ENode-based selection. To change the load balancing method, use the **fibre-channel npiv-proxy load-balance** command. For example,

```
-> fibre-channel npiv-proxy load-balance dynamic-reorder
-> fibre-channel npiv-proxy load-balance enode-based
```

The load balancing method is applied globally to all switch NP\_Ports. To change the load balancing method applied back to dynamic, use the **default** parameter with the **fibre-channel npiv-proxy load-balance** command. For example,

```
-> fibre-channel npiv-proxy load-balance default
```

To display the load balancing information for the OmniSwitch:

- Use the **show fibre-channel** command to determine the active load balancing method for the switch.
- Use the **show fibre-channel npiv-proxy load-balance** command with the **session-count** option to display the number of sessions on each FC port.

For more information, see [“N\\_Port Proxy Load Balancing”](#) on page 5-11.

## Statically Assigning FCoE Ports to NP Ports

To override the active load balancing method, it is possible to configure a static mapping between an FCoE port and an NP\_Port. When this is done, the switch will only process ENode FLOGI requests received on the specified FCoE port by sending the related FDISC messages only on the specified NP\_Port. No load balancing is applied to the statically mapped NP\_Port.

To configure this type of static port mapping, use the **fibre-channel npiv-proxy load-balance static** command. For example:

```
-> fibre-channel npiv-proxy load-balance static port 1/1 fc-port 2/1
```

In this example, FCoE port 1/1 is mapped to FC port 2/1, which is configured to operate as an NP\_Port. All ENode login requests received on port 1/1 are processed and directly forwarded to NP\_Port 2/1.

To remove a static port mapping, use the **no** form of the **fibre-channel npiv-proxy load-balance static** command. For example,

```
-> no fibre-channel npiv-proxy load-balance static port 1/1 fc-port 2/1
```

Use the **show fibre-channel npiv-proxy load-balance** command with the **static** option to display the static FCoE port mapping configuration for the switch.

For more information, see [“Statically Assigning FCoE Ports to NP Ports”](#) on page 5-24.

# Configuring an F\_Port Proxy Operation

The OmniSwitch F\_Port proxy functionality allows the switch to provide an F\_Port fabric for FC nodes attempting to access FC switches over an FCoE network. The F\_Port proxy functionality is operational when the following components are configured:

- FIP snooping is configured and enabled for the switch.
- An FCoE VLAN is mapped to a VSAN.
- At least one FC port that is assigned to the mapped VSAN is configured to operate as a fabric port (F\_Port).
- At least one FCoE port (configured as an FCF only, mixed, or trusted FCoE port) tagged with the FCoE VLAN.

The **fibre-channel port mode** command is used to configure an FC port to operate in the F\_Port proxy mode. For example, the following command configures FC port 2/2 to operate as an F\_Port proxy:

```
-> fibre-channel port 2/2 mode f
```

To identify which FC node traffic is serviced on F\_Port 2/2, complete the following steps:

**1** Create an OmniSwitch VSAN and map the VSAN to the FCoE VLAN on which converted FC node traffic will be forwarded through the FCoE network. In the following example, the **fibre-channel vsan** command is used to create VSAN 10 and the **fcoe vsan-map** command is used to map VSAN 10 to FCoE VLAN 500:

```
-> fibre-channel vsan 10
-> fcoe vsan-map vsan 10 vlan 500
```

**2** Assign the F\_Port to the VSAN that is mapped to the FCoE VLAN using the **fibre-channel vsan members** command. For example, the following command assigns F\_Port 2/2 to VSAN 10:

```
-> fibre-channel vsan 10 members 2/2
```

**3** Configure at least one port as an FCF only, mixed, or trusted FCoE port using the **fcoe role** command. For example, the following command configures port 1/1 as an FCoE mixed port:

```
-> fcoe port 1/1 role mixed
```

**4** Tag the FCoE ports that will forward the converted FC traffic to FCoE VLAN 500. Make sure that at least one of the FCoE ports tagged is configured as an FCF only, mixed, or trusted FCoE port. In the following example, the **vlan members tagged** command is used to tag FCoE ports 1/1 and 1/2 to FCoE VLAN 500:

```
-> vlan 500 members port 1/1-2 tagged
```

To display the F\_Port proxy configuration on the OmniSwitch:

- Use the **show fibre-channel vsan** command and the **show fibre-channel vsan members** command to display the VSAN configuration and the FC ports associated with the configured VSANs.
- Use the **show fibre-channel port** command to display the operational mode for each FC port.
- Use the **show fcoe vsan-map** command to determine which FCoE VLANs are mapped to which OmniSwitch VSANs.

For more information about F\_Port proxy functionality, see [“Using the F\\_Port Proxy Mode” on page 5-12](#).

## Configuring an E\_Port Proxy Operation

The OmniSwitch gateway provides an E\_Port to E\_Port (E2E) tunneling function that emulates a point-to-point FC link between E\_Ports on native FC switches. This E2E tunneling capability allows the OmniSwitch to serve as an E\_Port proxy through which FC switches can set up inter-switch links between FC fabrics over an FCoE network.

The E\_Port proxy functionality is activated when FIP snooping is configured and enabled and a tunnel endpoint is defined on each OmniSwitch gateway that will provide this functionality. The tunnel endpoints determine if the tunnel is a VE\_Port tunnel that is established through an FCoE network or an internal tunnel that is established between two TE\_Ports on the same switch or virtual chassis.

In both cases, configuring an OmniSwitch FC port as a tunnel expansion (TE\_Port) is required. An FC TE\_Port connects directly to an E\_Port on an FC switch. The **fibre-channel port mode** command is used to configure an FC port to operate as a TE\_Port. For example, the following command configures FC port 2/3 to operate as a TE\_Port:

```
-> fibre-channel port 2/3 mode te
```

### Creating a VE\_Port Tunnel

To create an E2E tunnel that will traverse an FCoE network, configure a tunnel endpoint on each gateway switch that will define the tunnel path. A VE\_Port tunnel endpoint consists of an FC TE\_Port and an FCoE VE\_Port associated with the same FCoE VLAN. The FCoE VLAN is used to tunnel traffic from the TE\_Port through the FCoE network via the FCoE VE ports.

The **fcoe role** command is used to configure an FCoE port (10G Ethernet DCB port) as a VE\_Port. For example, the following command designates port 1/1 as a VE\_Port:

```
-> fcoe port 1/1 role ve
```

The FCoE port role is used by FIP snooping to define ACLs that are applied to the VE\_Port to secure tunnel traffic through the FCoE network.

After the FCoE port is configured as a VE\_Port, the port is then tagged with the designated FCoE VLAN for the tunnel. For example, the following **vlan members tagged** command tags FCoE port 1/1 with FCoE VLAN 500.

```
-> vlan 500 members port 1/1 tagged
```

Once the VE\_Port is tagged with the tunnel FCoE VLAN, the FC TE\_Port is then assigned to the same FCoE VLAN using the **fcoe e-tunnel** command. This command also assigns a tunnel ID for this endpoint. For example, the following command assigns TE\_Port 2/3 to FCoE VLAN 500 (the same VLAN to which VE\_Port 1/1 was assigned):

```
-> fcoe e-tunnel 10 fc-port1 2/3 vlan 500
```

The E2E tunnel endpoint is now configured. When the other endpoint is configured with a different tunnel ID and all associated ports are up and running, the tunnel session is established between the two endpoints via exchange link parameters (ELP). The ELP are transmitted through the FCoE network between the FC switches connected to the OmniSwitch TE\_Ports associated with the same tunnel ID.

## Creating a TE\_Port Tunnel

To create an E2E tunnel that will directly connect two FC fabrics through the same switch or a virtual chassis configuration, use the **fcoe e-tunnel** command to create the tunnel endpoint with two TE\_Ports. For example, the following command creates a tunnel between TE\_Port 2/4 and TE\_Port 2/5 on the same switch:

```
-> fcoe e-tunnel 20 fc-port1 2/4 fc-port2 2/5
```

In this example, the **fc-port2** parameter is used instead of the **vlan** parameter. This defines the tunnel as an internal TE\_Port tunnel, instead of a VE\_Port tunnel. A TE\_Port tunnel does not traverse the FCoE network, so there is no need to associate the TE\_Port with an FCoE VLAN. Each TE\_Port is the endpoint for this type of tunnel.

To display the E2E tunnel configuration for the switch, use the **show fcoe e-tunnel** command.

For more information about the E2E tunneling feature, see [“Using the E\\_Port Proxy Mode” on page 5-15](#).

# FCoE/FC Gateway Configuration Examples

## Example 1: Multiple Gateway Operations on the Same Fabric

The following diagram shows a sample network topology in which two OmniSwitches are operating as FCoE/FC gateway switches to secure and forward the following traffic flows:

- FCoE traffic between FCoE endpoints (ENode and FC switch)
- FC traffic between FC endpoints (FC node and FC switch)
- Inter-switch link trunking between E2E tunnel endpoints (FC switch to FC switch).

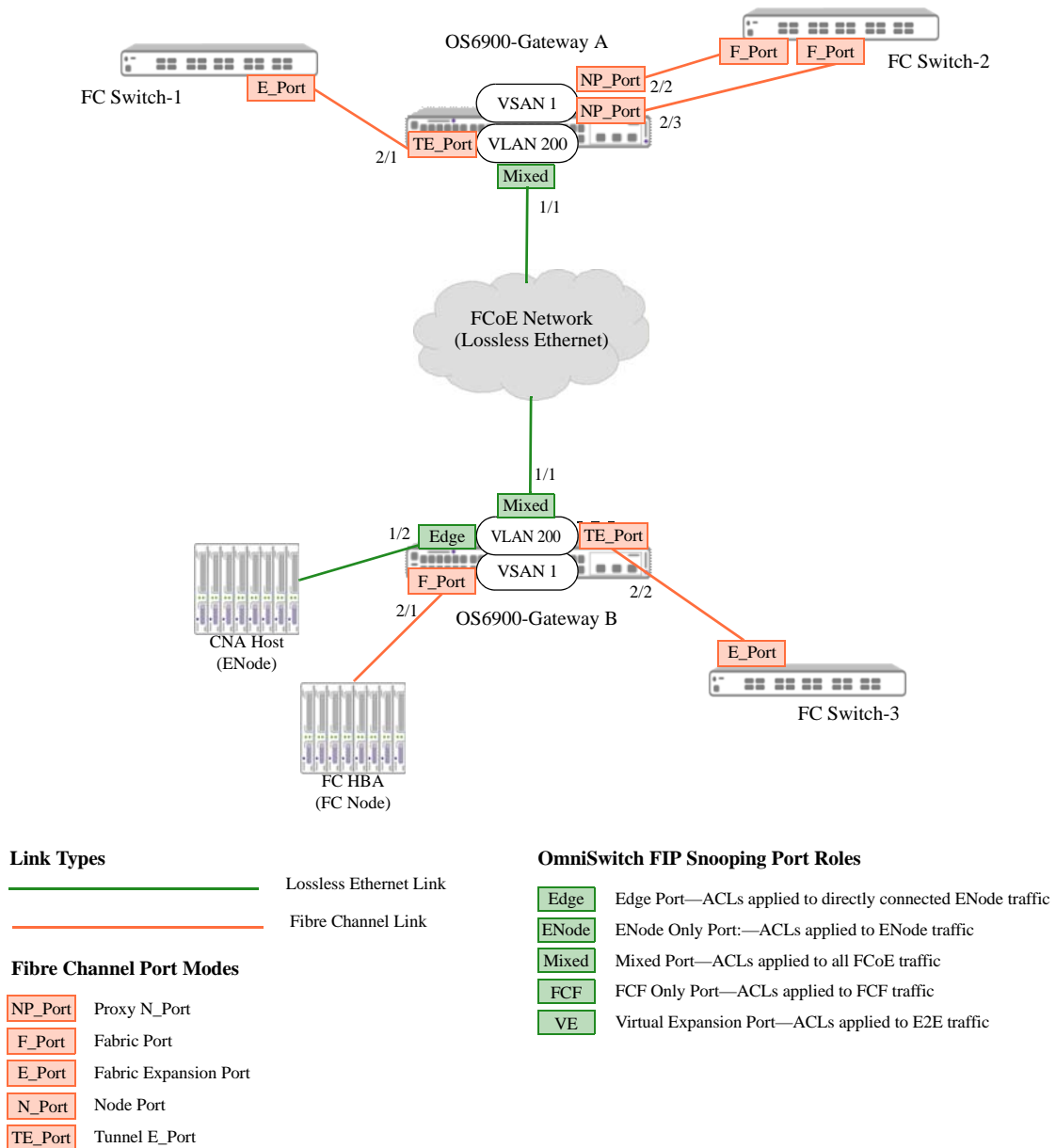


Figure 6: Sample FCoE/FC Gateway Topology

To the ENodes, FC nodes, and FC switch in this sample topology, the underlying Ethernet links through the FCoE network path appear as a direct point-to-point connection that is expected in a native FC SAN.

- The FIP snooping, E\_Port proxy, and E\_Port proxy operations are enabled on the OS6900-Gateway A.
- The FIP snooping, F\_Port proxy, and E\_Port proxy operations are enabled on the OS6900-Gateway B.
- Port 1/1 on Gateway A receives all traffic (FIP snooping, F\_Port, E\_Port) from Gateway B through FCoE VLAN 200. As a result, FCoE port 1/1 is configured with the FCoE port role set to mixed so that FIP snooping is applied to all traffic types.
- Port 1/1 on Gateway B also receives all traffic (FIP snooping, NP\_Port, E\_Port) from Gateway A through FCoE VLAN 200. As a result, FCoE port 1/1 is configured with the FCoE port role set to mixed so that FIP snooping is applied to all traffic types.
- FCoE VLAN 200 is configured through the FCoE network (manually on each OS6900 gateway switch and, if necessary, manually on the ENodes).

The following configuration information provides an example of how the FCoE/FC gateway functionality is configured for each OmniSwitch in the sample FCoE/FC gateway topology.

### OS6900-Gateway A

FIP snooping is a required component for setting up an OmniSwitch FCoE/FC gateway. The following sample commands configure FIP snooping and lossless Ethernet on OS6900-Gateway A:

```
-> fcoe vlan 200
-> fcoe port 1/1 role mixed
-> vlan 200 members port 1/1 tagged
-> qos port 1/1 trusted default classification 802.1p
-> qos qsp dcb FCOE-1 import qsp dcb 7 fcoe
-> qos qsi port 1/1 qsp dcb FCOE-1
-> lldp port 1/1 tlv application enable
-> lldp port 1/1 tlv application fcoe priority 3
-> fcoe fip-snooping admin-state enable
```

By default, VSAN 1 already exists in the switch configuration and all OmniSwitch FC ports are assigned to VSAN 1. As a result, the only VSAN configuration required for this example is to map VSAN 1 to FCoE VLAN 200. For informational purposes, the commands to create the VSAN and assign the NP\_Ports are included in the following sample commands used to configure N\_Port proxy functionality on OS6900-Gateway A:

```
-> fibre-channel vsan 1
-> fcoe vsan-map vsan 1 vlan 200
-> fiber-channel fc-port 2/2 mode np
-> fiber-channel fc-port 2/3 mode np
-> fibre-channel vsan 1 members 2/2-3
```

The following sample commands configure an E\_Port proxy (E2E tunnel) endpoint on OS6900-Gateway A. Note that a VSAN mapping to an FCoE VLAN is not required for E2E tunnel traffic. Instead, an FC port configured to operate as a tunnel E\_Port is directly associated with FCoE VLAN 200 to carry tunnel traffic.

```
-> fiber-channel fc-port 2/1 mode te
-> fcoe e-tunnel 1
-> fcoe e-tunnel 1 fc-port1 2/1 vlan 200
```

## OS6900-Gateway B

The following sample commands configure FIP snooping and lossless Ethernet on the OS6900-Gateway B switch:

```
-> fcoe vlan 200
-> fcoe port 1/1 role mixed
-> fcoe port 1/2 role edge
-> vlan 200 members port 1/1-2 tagged
-> qos port 1/1 trusted default classification 802.1p
-> qos qsp dcb FCOE-1 import qsp dcb 7 fcoe
-> qos qsi port 1/1-2 qsp dcb FCOE-1
-> lldp port 1/1-2 tlv application enable
-> lldp port 1/1-2 tlv application fcoe priority 3
-> fcoe fip-snooping admin-state enable
```

The following sample commands configure F\_Port proxy functionality on the OS6900-Gateway B switch:

```
-> fiber-channel vsan 1
-> fcoe vsan-map vsan 1 vlan 200
-> fiber-channel fc-port 2/1 mode f
-> fiber-channel vsan 1 members port 2/1
```

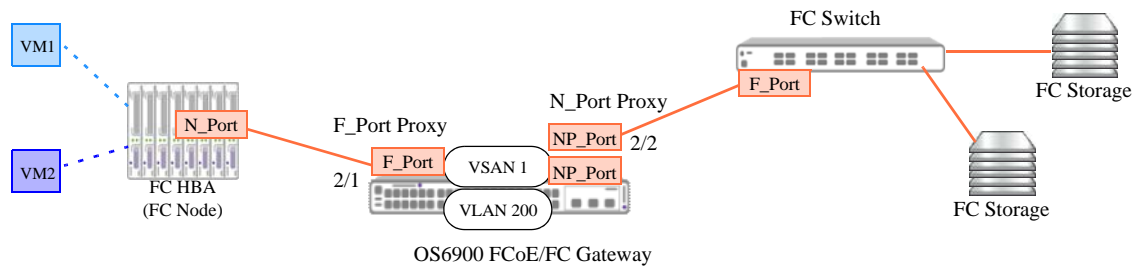
The following sample commands configure an E\_Port proxy (E2E tunnel) endpoint on the OS6900-Gateway B switch. Note that the tunnel ID used in this sample is the same as the tunnel ID used on the OS6900-Gateway A switch. This identifies both endpoints as members of the same tunnel.

```
-> fibre-channel fc-port 2/2 mode te
-> fcoe e-tunnel 1
-> fcoe e-tunnel 1 fc-port1 2/2 vlan 200
```



## Example 2: OmniSwitch Gateway with NPIV Host






The following diagram shows a sample network topology in which an OmniSwitch FCoE/FC gateway provides N\_Port proxy and F\_Port proxy services on the same switch.








### Link Types

	Lossless Ethernet Link
	Fibre Channel Link

### Fibre Channel Port Modes

	Proxy N_Port
	Fabric Port
	Fabric Expansion Port
	Node Port
	Tunnel E_Port

### OmniSwitch FIP Snooping Port Roles

	Edge Port—ACLs applied to directly connected ENode traffic
	ENode Only Port—ACLs applied to ENode traffic
	Mixed Port—ACLs applied to all FCoE traffic
	FCF Only Port—ACLs applied to FCF traffic
	Virtual Expansion Port—ACLs applied to E2E traffic

**Figure 7: OmniSwitch FCoE/FC Gateway with NPIV Host**

In this example, the OmniSwitch gateway emulates both an FC switch and an FCoE forwarder to allow the FC NPIV server and associated VMs to connect and access the targeted FC storage in the FC SAN.

- The F\_Port proxy operation is enabled on port 2/1, which connects directly to an FC HBA port on the server. This allows the FC HBA to communicate over Ethernet (OmniSwitch gateway) with the FC switch fabric.
- The N\_Port proxy operation is enabled on port 2/2, which connects directly to an F\_Port on the native FC switch. This allows the OmniSwitch to also serve as an FCoE forwarder for the FC HBA sessions.
- Port 2/1 and port 2/2 are assigned to VSAN 1, which is mapped to FCoE VLAN 200.
- The OmniSwitch gateway presents an F\_Port to an N\_Port on the HBA and an NP\_Port to the F\_Port on the FC switch.
- NPIV is enabled on the FC HBA port to allow the server to request a separate FCID (also called an N\_Port ID) for VM1 and VM2. This is similar to how the N\_Port proxy operation uses NPIV to allow the OmniSwitch to request multiple FCIDs on the same physical port.
- Once the physical FC HBA port logs into the FC switch fabric through the OmniSwitch gateway, the server then requests additional logins for each of the VMs.
- In this scenario, the OmniSwitch gateway does not have to track FC-to-FC session reachability (does not need keep-alive messages to track the sessions). However, the OmniSwitch does maintain session information for the three login sessions (one for the HBA port and one for each VM).
- Note that there is no Ethernet FCoE port configuration needed in this example. The FCoE VLAN handles the convergence of the FC traffic for ports associated with VSAN 1. So FC over Ethernet is achieved on the same gateway switch to allow a native FC node to communicate with a native FC switch fabric.

The following **show fibre-channel sessions** command output provides a sample of how the FC HBA sessions might appear on the OmniSwitch gateway (the device represented in each session is highlighted with the same color used to represent the device in the example diagram on [page 5-31](#)):

```
-> show fibre-channel sessions
Total FIBRE-CHANNEL Sessions      : 6,
Total NPIV Sessions                : 3,
Total R-NPIV Sessions              : 3,
Total E-TUNNEL Sessions            : 0
```

Port	Mode	VSAN	T-ID	WWPN	FCID	Status	Login Type
2/1	F	1	--	10:00:E8:E7:32:94:67:7A	01:04:01	SUCCESS	FLOGI
2/1	F	1	--	10:00:E8:E7:32:94:67:7A	01:04:03	SUCCESS	FDISC
2/1	F	1	--	10:00:E8:E7:32:94:67:7A	01:04:04	SUCCESS	FDISC
2/6	NP	1	--	10:00:E8:E7:32:94:67:7D	01:04:01	SUCCESS	FLOGI
2/6	NP	1	--	10:00:E8:E7:32:94:67:7D	01:04:03	SUCCESS	FDISC
2/6	NP	1	--	10:00:E8:E7:32:94:67:7D	01:04:04	SUCCESS	FDISC

In this example,

- The WWPNs are those of the OmniSwitch FC ports (F\_Port 2/1 and NP\_Port 2/6) on which the sessions were established.
- The “FCID” column displays a unique FCID for each device per each session type on each port. Since there are two VMs on the FC node server, there are three sessions on each OmniSwitch port: one for the physical FC HBA port and one for each VM.
- The “Login” column displays “FLOGI” or “FDISC”. The FLOGI session represents the initial fabric login of the physical FC HBA port. Each FDISC session represents the fabric login for each VM. When using NPIV, the NP\_Port sends FDISC requests on behalf of devices accessing the fabric on the NP\_Port. The FLOGI request is for the initial fabric login of the physical NP\_Port.
- Sessions are shown for the same three FCIDs on each port. This is because the port 2/1 sessions represent the F\_Port sessions directly from the FC HBA and the port 2/6 sessions represent the N\_Port proxy sessions requested by the OmniSwitch FCoE forwarder for the same FCIDs.
- To the FC HBA and the two VMs, the connection through the OmniSwitch gateway to the FC switch fabric is transparent. Because the HBA is also using NPIV, each VM session is uniquely identified and, therefore, separately tracked.

# Verifying the FCoE/FC Gateway Configuration

Displaying the OmniSwitch FIP snooping and FCoE/FC gateway configuration is helpful to verify the actual configuration and monitor FIP snooping, N\_Port proxy, F\_Port proxy, and E\_Port proxy sessions.

To display information about the FCoE/FC gateway configuration, use the **show** commands listed in the following table:

<b>show fibre-channel vsan</b>	Displays the local OmniSwitch VSAN configuration. VSANs identify the OmniSwitch FC ports as members of a specific FCoE/FC gateway fabric.
<b>show fibre-channel vsan members</b>	Displays the FC ports associated with each VSAN.
<b>show fibre-channel port</b>	Displays the OmniSwitch FC port configuration, including port operating modes, for the switch. The port mode determines the FCoE/FC gateway functionality enabled on the port.
<b>show fcoe vsan-map</b>	Displays the VSAN mapping configuration for the switch. VSANs are mapped to FCoE VLANs as part of the local gateway fabric.
<b>show fibre-channel sessions</b>	Displays the N_Port proxy, F_Port proxy, and E_Port proxy sessions associated with the local switch. Use the <b>clear fibre-channel sessions</b> command to clear sessions on all FC ports.
<b>show fibre-channel node</b>	Displays a list of FC nodes connected to OmniSwitch FC ports.
<b>show fcoe e-tunnel</b>	Displays the E2E tunnel configuration for the switch.
<b>show fibre-channel</b>	Displays the global FCoE/FC gateway parameter values for the switch.
<b>show fibre-channel statistics</b>	Displays FC port statistics. Use the <b>clear fibre-channel statistics</b> command to reset the statistics information.

To display information about the FIP snooping configuration, use the **show** commands listed in the following table:

<b>show fcoe</b>	Displays the global FCoE FIP snooping status and configuration for the switch.
<b>show fcoe ports</b>	Displays the FCoE port configuration, including port roles, for the switch. The port role determines the FIP snooping ACL configuration for the port.
<b>show fcoe sessions</b>	Displays the FIP sessions associated with the local switch.
<b>show fcoe enode</b>	Displays ENode information for the FIP sessions associated with the local switch.
<b>show fcoe fcf</b>	Displays FCF information for the FIP sessions associated with the local switch.
<b>show fcoe fc-map</b>	Displays the Fibre Channel Mapped Address Prefix (FC-MAP) configuration.
<b>show fcoe discovery-advertisement</b>	Displays the FIP discovery advertisement message parameter values.
<b>show fcoe statistics</b>	Displays ENode and FCF generated session statistics. Use the <b>clear fcoe statistics</b> command resets the statistics information.



# 6 Virtual Machine Classification

Server virtualization in the data center is happening now. The OmniSwitch implements two features to help automate the discovery and movement of virtual machines (VMs):

- **Virtual Network Profile (vNP)**—A vNP is basically a Universal Network Profile (UNP) that will classify virtual machines in the same manner as any other device connected to a UNP port. Once a virtual machine (or any other such device) is assigned to a vNP, the virtual machine traffic is bound to a VLAN or Shortest Path Bridging (SPB) service as defined in the profile. In addition, any QoS policies associated with the profile are also applied to the VM traffic.
- **Edge Virtual Bridging (EVB)**—based on the IEEE 802.1Qbg standard, EVB defines architecture to standardized connections between hosts with virtual machines and switches. The OmniSwitch implementation provides the ability for the switch to interact with EVB servers.
  - EVB runs between an EVB bridge (OmniSwitch running EVB) and an EVB station (server running EVB) to discover VMs running on the station or detect VM movement and associate each VM with an appropriate VLAN bridging instance.
  - The Link Layer Delivery Protocol (LLDP) is used by the EVB bridge to discover the EVB station and establish links between the station and bridge.

## In This Chapter

This chapter describes the functionality provided with the vNP and EVB features and how these features are used to classify virtual machines into the OmniSwitch bridging or SPB service domain. It provides information about configuring vNP and EVB through the Command Line Interface (CLI). CLI commands are used in the configuration examples; for more details about the syntax of commands, see the *OmniSwitch CLI Reference Guide*.

The following topics and procedures are included in this chapter:

- [“UNP \(vNP\) Specifications” on page 6-2.](#)
- [“EVB Specifications” on page 6-2.](#)
- [“Server Virtualization Overview” on page 6-3.](#)
- [“UNP Overview” on page 6-4.](#)
- [“Using EVB” on page 6-8.](#)
- [“Tracking Virtual Machines” on page 6-13.](#)

For more information about UNP, see [Chapter 27, “Configuring Universal Network Profiles,”](#) in the *OmniSwitch AOS Release 7 Network Configuration Guide*.

## UNP (vNP) Specifications

Platforms supported	OmniSwitch 10K, 6900
Number of UNPs per switch	4K (includes static and dynamic profiles).
Number of UNP users per switch	2K
Authentication type	MAC-based authentication
Profile type	VLAN or Shortest Path Bridging (SPB)
UNP port type	Bridge (VLAN-based classification) or access (service-based classification)
UNP classification rules	MAC address, MAC-range, IP address, and VLAN tag
Number of QoS policy lists per switch	32 (includes the default list)
Number of QoS policy lists per UNP	1

## EVB Specifications

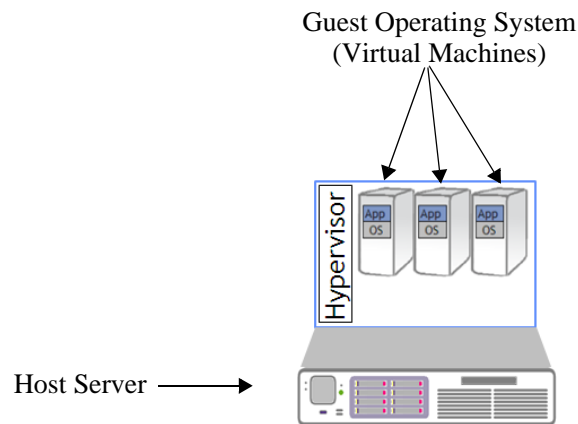
Platforms Supported	OmniSwitch 10K, 6900
OmniSwitch Software License	Data Center
IEEE Standards Supported	P802.1Qbg Standard Draft, Revision D2.2. February 18, 2012— <i>Virtual Bridged Local Area Networks—Amendment 21: Edge Virtual Bridging</i>
EVB mode	Bridging (virtual machines request the required CVLAN ID tag)
Edge Relay (ER) support	Single ER per switch port. The ER can operate as a Virtual Ethernet Port Aggregator (VEPA) or as a Virtual Ethernet Bridge (VEB).

# Server Virtualization Overview

Data center virtualization, as well as the virtualization of many other types of networks, facilitates the sharing and management of network resources. The virtual network is in essence a single, logical entity that is not restricted by physical boundaries. This is similar in concept to the use of virtual local area networks (VLANs) and virtual private networks (VPNs).

A key factor in the transition of the traditional data center network is the widespread adoption of server virtualization. A virtualized server is comprised of the following main components (see Figure 1):

- Guest virtual machine—the software representation of a self-contained operating system and application software instance that runs on a host server.
- A host server—the underlying physical hardware that provides computing resources for guest virtual machines.
- Hypervisor—a software program that provisions and manages the guest virtual machines for the host server. This program enables multiple guest virtual machines to run in isolation, side-by-side on the same physical server.



**Figure 1: Virtualized Server**

Server virtualization provides the following benefits:

- **Consolidation of server resources.** The ability to run multiple virtual machines on one server reduces the number of servers required to support data center applications. Using virtual machines maximizes physical server resources; the server is no longer tied to only one application or a single task.
- **Power savings.** The need for fewer servers means less power usage. In addition, existing servers can remain powered down until they are needed for virtual machine support. It is not necessary to have an underutilized server up and running all the time.
- **Hardware independence.** Virtual machines operate independently from the physical hardware on which they reside. This means that the configuration of virtual machine components (for example, RAM, CPU, network interface) can differ from the underlying components of the physical server. In addition, each virtual machine on the same physical server can run a different operating system.
- **Virtual machine mobility.** A virtual machine (VM) is basically an encapsulation of virtual hardware resources with its own operating system and applications. These resources operate independently from the underlying physical server hardware. As a result, virtual machines can easily move from one location to another without having to make underlying resource changes or disrupt live applications.

# Classifying Virtual Machines

To facilitate the discovery and movement of virtual machines, the OmniSwitch provides the Universal Network Profile (UNP) feature and supports the IEEE P802.1Qbg Edge Virtual Bridging (EVB) protocol.

## UNP Overview

This section provides an overview of UNP functionality, which is not restricted to supporting only data center solutions. For more information about UNP, see [Chapter 27, “Configuring Universal Network Profiles,”](#) in the *OmniSwitch AOS Release 7 Network Configuration Guide*.

The UNP feature provides the ability to define and apply network access control to specific types of devices by grouping such devices according to specific matching profile criteria. This allows network administrators to create virtual network profiles (vNPs) *and* user network profiles from a unified framework of operation and administration.

Basically, UNP functionality is used to define profile-based VLANs or SPB service access points (SAPs) to which network devices are assigned. The profile can allow, deny, or require actions by users or machines on the network. Because membership to a VLAN or service is based on UNP profile criteria, devices assigned to the VLAN or service are not tied to a specific port or switch. This flexibility allows device mobility within the network while maintaining network security.

Implementing UNP functionality to create virtual network profiles is particularly useful in an Alcatel-Lucent data center switching architecture. A vNP can define information such as access control rights, a VLAN or SPB service assignment, along with expected quality of service (QoS) levels for data center machines. This type of information can help virtual machines to securely bind to a VLAN bridged domain or an SPB service domain.

## Profile Types

The OmniSwitch supports two separate traffic domains: VLAN and service. Traffic is classified into each domain based on the UNP profile type.

- **VLAN profile.** This type of profile creates a VLAN-port association (VPA) for device traffic that is classified into the profile. The VPA represents an association between the UNP port on which the device traffic was received and the VLAN ID specified by the profile. In other words, once classified into a specific profile, device traffic is tagged to forward on the UNP VLAN.
- **Service profile.** The OmniSwitch supports Shortest Path Bridging (SPB) services, which are based on the Provider Backbone Bridge Network (PBBN) architecture. This type of profile creates an association between device traffic that is classified into the profile and a SPB service access point (SAP).

## Edge Port Types

Virtual machines bound to either a VLAN domain or a service domain can run on hosts with and without Edge Virtual Bridging (EVB) enabled. This results in three types of data center edge ports, all of which are provided through the OmniSwitch UNP and EVB features:

- UNP-enabled bridge port
- UNP-enabled service access port
- EVB-enabled bridge port

EVB service ports are not supported. See the [“Using EVB” on page 6-8](#) for more information.



UNP also provides the ability to group UNP ports into a single logical customer domain. Once a UNP port is assigned to a specific customer domain ID, only profile classification rules associated with the same domain are applied to that port.

Using customer domains to segment network traffic ties a profile, such as a vNP, to specific UNP ports. For example, UNP ports carrying traffic for Customer A are grouped into domain 2, and profiles tailored for Customer A are also assigned to domain 2. As a result, domain 2 profiles are applied only to domain 2 UNP ports.

## Classification Rules

A vNP can also define specific classification rules that are used to assign virtual machines to a profile. The following classification rules (shown in the order of precedence) are used to classify traffic received on UNP bridge and access ports:

- 1 MAC address + VLAN tag
- 2 MAC address
- 3 MAC address range + VLAN tag
- 4 MAC address range
- 5 IP address + VLAN tag
- 6 IP address
- 7 VLAN tag

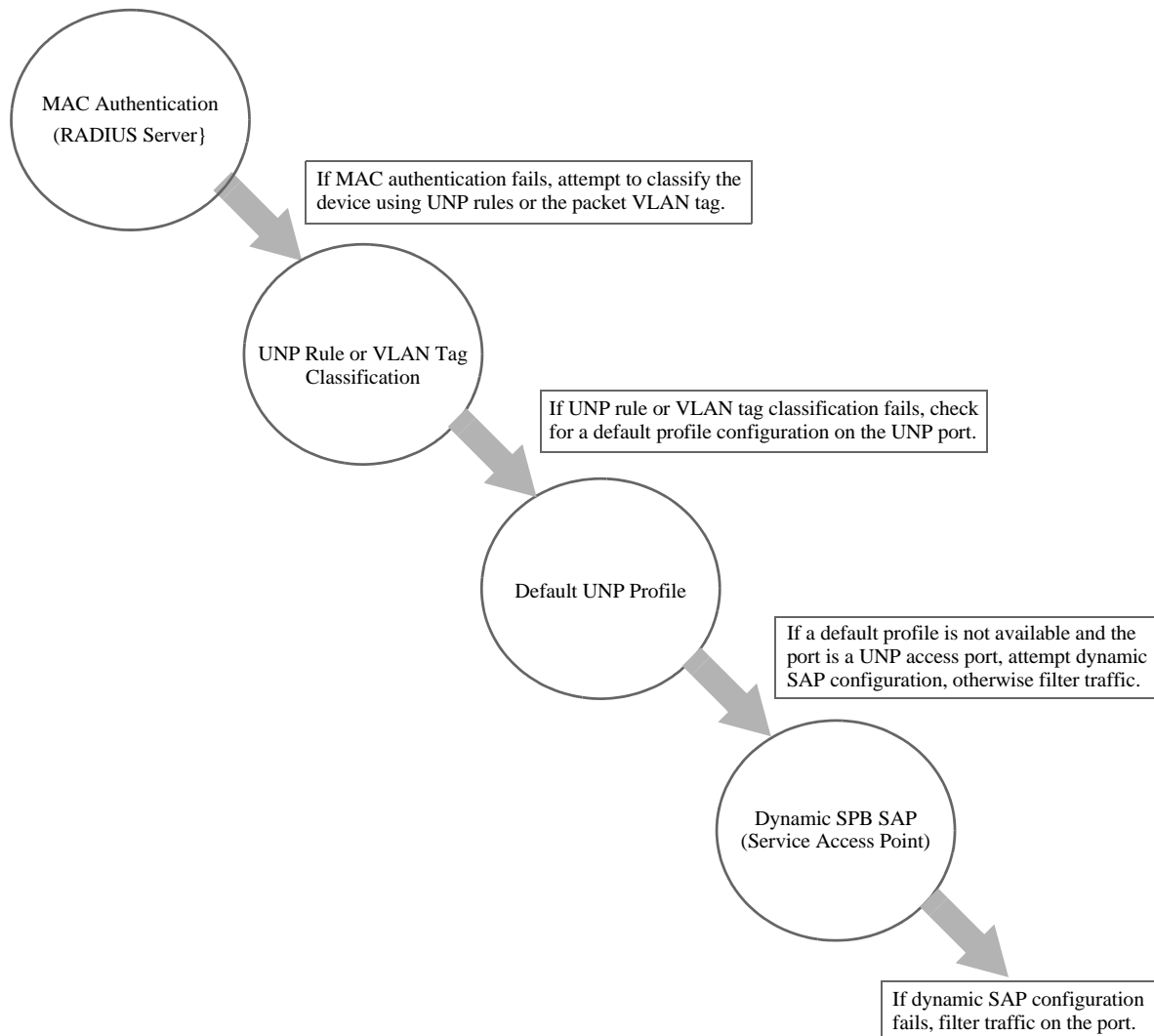
In addition to classification rules, UNP provides the ability to trust the VLAN tag of incoming traffic. This means that virtual machines can be provisioned into the VLAN tag pre-assigned by the virtualized server hypervisor. Each UNP port can then be assigned a default UNP to classify untagged traffic when all classification mechanisms either fail or are not available.

## How it Works

Dynamic assignment of devices using UNPs is achieved through port-based functionality that provides the ability to authenticate and classify device traffic. Authentication verifies the device identity and provides a UNP name. In the event authentication is not available or is unsuccessful, classification rules associated with the UNPs, as well as the UNP port configuration attributes, are applied to the traffic to determine the UNP assignment.

There is no global switch setting to invoke UNP (vNP) functionality. Instead, switch ports are configured as a UNP bridge port or a UNP access port and profiles are defined to determine the dynamic VLAN or SPB service assignment for devices connected through the UNP ports.

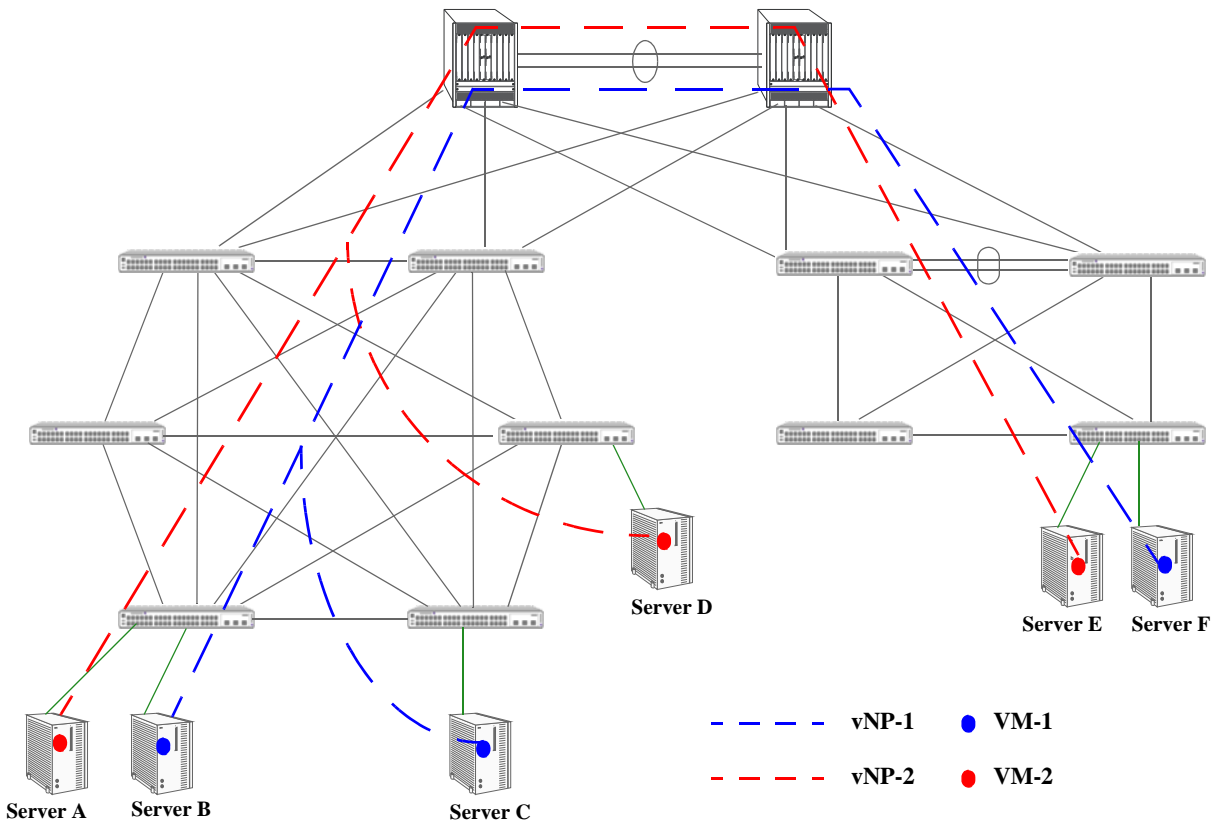
When UNP is enabled on a switch port or link aggregate, the following device classification process is triggered when the port receives traffic:



**Figure 1: UNP Classification Overview**

## Virtual Network Profile Example

The following simplified illustration shows an example of vNP functionality in a virtualized network:



**Figure 2: Virtual Network Profile–VM Mobility**

In this example,

- On each switch that is connected to a server, virtual network profiles vNP-1 and vNP-2 are configured to classify and forward traffic for virtual machines VM-1 and VM-2.
- The virtual network profiles specify either a VLAN ID or service ID that will carry virtual machine traffic through the data center mesh to other servers (either within the same mesh or through the cloud to another mesh) and if any QoS policies are applied to virtual machine traffic.
- Each port connected to a server is enabled with UNP functionality.
- When the MAC address for VM-1 and for VM-2 is discovered on a UNP port, the switch applies the profile configuration to each MAC address to classify each virtual machine into a VLAN or SPB service. As a result,
  - VM-1 is assigned to the vNP-1 profile. The profile attributes then determine the VLAN or service assignment and if any QoS policies are applied to VM-1 traffic.
  - VM-2 is assigned to the vNP-2 profile. The profile attributes then determine the VLAN or service assignment and if any QoS policies are applied to VM-2 traffic.
- If VM-1 or VM-2 is moved to another server, as shown in this sample network, VM-1 will get assigned to vNP-1 and VM-2 will get assigned to vNP-2, thus the network access control configuration for each virtual machine is preserved whenever it moves to another server.

## Using EVB

In the data center environment, data center servers run specialized hypervisor software to provision and manage multiple virtual machines (VMs) within the host server. These VMs can be dynamically created, deleted, or even moved to other host servers in the network.

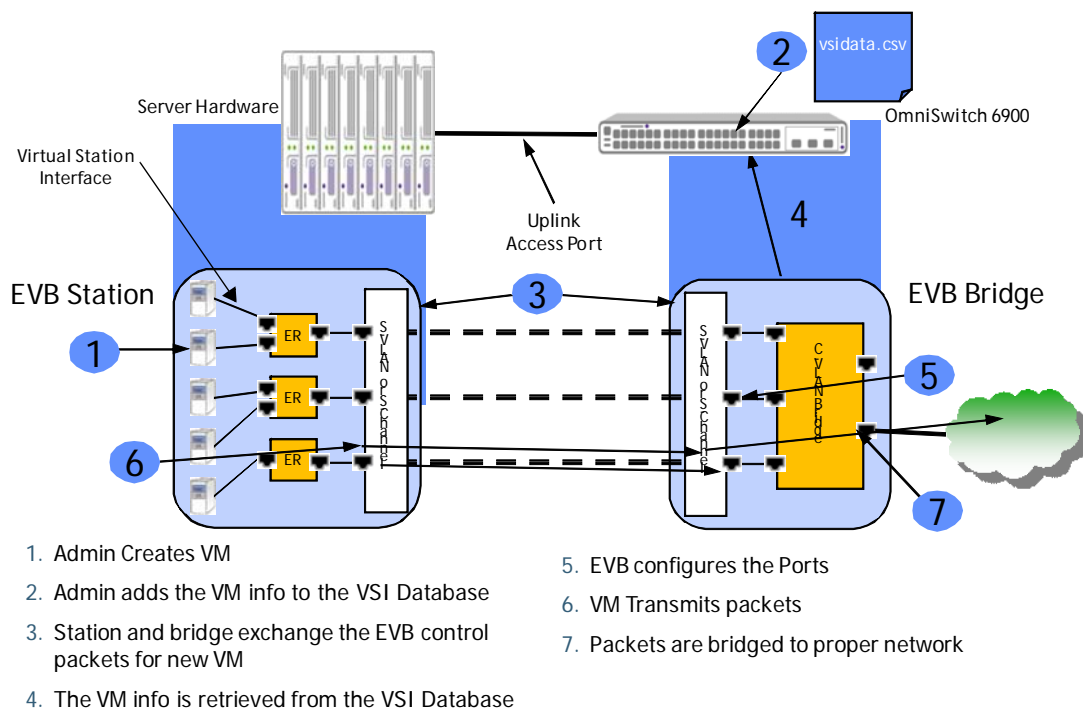
Each VM may require different network connections and services. Typically, similar types of VMs are grouped together to form a VM network to control the unicast and multicast domains and to make a connection to storage area networks (SAN) or wide area networks (WAN).

The OmniSwitch implementation of Edge Virtual Bridging (EVB) helps automate the discovery of virtual machines and connects them to the appropriate network domain. EVB runs between a station (data center server) and a bridge (an OmniSwitch).

EVB defines logical components of the station (server): Virtual Station Interface (VSI), Edge Relay (ER), and Uplink Relay Port. The OmniSwitch supports a single ER per switch port. The ER can operate as a Virtual Ethernet Port Aggregator (VEPA) or Virtual Ethernet Bridge (VEB).

- VEPA (Virtual Ethernet Port Aggregator)—the hypervisor treats the NIC as a single interface connected to each VM and all outgoing traffic is sent through the NIC to an external switch (the Edge Relay function per IEEE 802.1Qbg in the host server).
- VEB (Virtual Ethernet Bridge)—the hypervisor creates a virtual Ethernet switch inside the host that can bridge data between VMs and send data out to the network as needed.

When a switch port is configured as an EVB bridging or service access port, then the EVB protocol is used between the station and bridge to discover VMs. A virtual switch interface (VSI) database file on the local switch is used to verify the discovered VM information. If there is a match, then EVB dynamically creates the necessary VLAN or SAP to carry the VM traffic.



**Figure 3: EVB Operational Overview**

EVB ports rely on the following EVB protocols to exchange key VM VSI information:

- S-Channel Discovery and Configuration Protocol (CDCP)—Used to set up S-Channels, which are needed when a station has two or more ERs.
- EVB TLVs—Used to advertise and negotiate the EVB capabilities of a station or bridge.
- VSI Discovery and Configuration Protocol (VDP)—Assigns VSI instances to C-VLANs. Assigning multiple VLANs and MAC addresses to the same VSI instance is allowed.
- Edge Control Protocol (ECP)—Provides reliable delivery of VDP DUs between the station and the bridge.

If EVB protocol fails, then the VM connection will fail and no packets will be sent to the switch.

## VSI Database

The OmniSwitch VSI database is managed and maintained by the UNP application. The EVB uses information collected during the VSI Discovery Protocol (VDP) exchange between the EVB OmniSwitch and the EVB station (server) to send a VSI database query to the UNP application. UNP compares the VSI database entries to the values in the EVB query. If there is a match, UNP provides the matching VSI entry values to EVB. In this manner, EVB confirms the VSI information discovered and gets the C-VLAN and/or priority information that is then used to classify the VM on the OmniSwitch.

In essence, the VSI database serves as a VM reservation list on the OmniSwitch. The EVB exchange between the switch and the station is done to confirm the reservation and obtain classification information for the VM. See [“Creating the VSI Database File” on page 6-11](#) for more information.

## EVB Configuration Guidelines

Only the EVB VLAN bridging mode is supported at this time. In this mode,

- The VM requests specific C-VLAN(s). Note that the VM must provide the requested C-VLANs when it sends EVB protocol packets to the switch.
- The switch dynamically creates the corresponding VLANs and propagates those VLANs to the other switches in the network using the MVRP protocol.
- After the VM is associated to the bridge using the EVB protocol, then the VM can send data traffic tagged with the requested C-VLAN.
- EVB does not support the use of GroupID and S-channel in the VLAN bridging mode.

UNP and EVB are mutually exclusive on the same port; only UNP or EVB functionality is allowed on a specific port or link aggregate at any given time.

- If a port is configured as an EVB bridging port, then the EVB protocol is used between the station and bridge to discover VMs.
- If a port is configured as a UNP bridging or access port, then the regular UNP classification methods are used to assign the VM traffic to a dynamic VLAN or dynamic SAP.
- The UNP method is mainly used for stations that are not using EVB.
- The EVB method is only for interaction with EVB stations.

## Configuring EVB

Configuring EVB functionality on the OmniSwitch involves the following steps:

- 1** Create the VSI database. The **vsidata.csv** file is a user-configured and maintained database that contains the virtual switch interface (VSI) information for VMs. This file must be in the **.csv** format and contain specific identifying information that is used to classify VMs. See [“Creating the VSI Database File” on page 6-11](#) for more information.
- 2** Upload the VSI database file to the switch. The **vsidata.csv** file must reside in the **/flash/vsidata/** directory on the local switch. If this directory does not exist, then manually create it on the switch making sure to use “vsidata” as the directory name. Upload the vsidata.csv file into the /flash/vsidata/ directory.
- 3** Load the VSI database file into switch memory. Once the vsidata.csv file is created and uploaded to the /flash/vsidata/ directory, use the **unp reload vsi-type-database** command to load the file contents into switch memory. For example:

```
-> unp reload vsi-type-database
```

Note that any time a new VSI file is uploaded or the existing file is changed, a file reload into switch memory is required to apply the updated file.

- 4** Configure a switch port or link aggregate as an EVB port. There are two methods for configuring EVB ports: dynamic or manual configuration of EVB on the port.

The dynamic option is a global option that applies to all switch ports. When enabled, any port that receives S-Channel Discovery and Configuration Protocol (CDCP) or EVB TLV (type, length, value) control packets, is automatically configured as an EVB port. The manual option applies EVB directly to a specific port.

To enable dynamic configuration of a port as an EVB bridge or access port, use the **evb port auto enable type** command with the **vlan-bridging** or **service-access** option. For example:

```
-> evb port auto enable type vlan-bridging
-> evb port auto enable type service-access
```

To manually configure a port as an EVB bridge or access port, use the **evb port type** command with the **vlan-bridging** or **service-access** option. For example:

```
-> evb port 1/1 type vlan-bridging
-> evb port 1/2 type service-access
```

- 5** *Optional.* Modify the following EVB parameter values that are set by default at the time the EVB session is created. If necessary, use the specified commands to change the default values.

Parameter Description	Command	Default
Use local EVB TLV settings instead of exchanging local and remote EVB TLV settings to determine the operational configuration.	<b>evb evb-lldp manual</b>	Disabled
The amount of time to wait for an EVB Edge Control Protocol (ECP) acknowledgement (ACK).	<b>evb ecp default-ack-timer</b>	14 (approximately 163.84 milliseconds)

Parameter Description	Command	Default
The maximum number of times to retransmit ECP packets when no ACK is received. Retransmit is triggered after the ACK timer expires.	<b>evb ecp default-max-retry</b>	3
The VSI Discovery Protocol (VDP) resource timeout	<b>evb vdp default-resource-wait-delay</b>	20 (approximately 10.48576 seconds)
The VDP keep alive timeout	<b>evb vdp default-keep-alive-timeout</b>	20 (approximately 10.48576 seconds)

## Creating the VSI Database File

The network administrator manually defines a VSI database entry in the **vsidata.csv** file for each VSI instance and then copies that file to the **/flash/vsidata** directory on each OmniSwitch bridge that is connected to EVB stations. Each database entry consists of the following values:

- VSI Manager ID
- VSI Type
- VSI Type Version
- VSI Instance ID
- Return Entry Type
- VLAN
- Priority
- MAC
- Group ID

When creating the **vsidat.csv** file and VSI entries, use the following required guidelines:

- The first row of the VSI database file must contain the following column names in the following order:
- VSI Manager ID, VSI Instance ID and MAC address should be hexadecimal only. All other fields should be decimal.
- Duplicate entries are not allowed.
- Number of return entries are up to sixteen.
- Wild cards are allowed on VSI Instance ID and VSI Type fields only
- For VSI Manager ID and VSI Instance ID, the delimiter should be space. For MAC, the delimiter should be colon. For all other entries, the delimiter should be comma.
- The data file name must be **vsidata.csv** and located in **/flash/vsidata/** directory.

The following shows the contents and layout of a sample **vsidata.csv** file:

	A	B	C	D	E	F	G	H	I
1	VSI Manager ID	VSI Type	VSI Type Version	VSI Instance ID	Type	VLAN	Priority	MAC	Group ID
2	20 02 00 00 00 00 00 00 00 00 00 00 00 00 02	1		C11120 DE 07 2B 11E185 A100 10 94 00 00 03	1	100	1		
3	30 02 00 00 00 00 00 00 00 00 00 00 00 00 02	12		C11120 DE 07 2B 11E185 A100 10 94 00 00 03	2	200	2	aabbccddeeff	
4	10 02 00 00 00 00 00 00 00 00 00 00 00 00 02	13		C11120 DE 07 2B 11E185 A100 10 94 00 00 03	1	300,500,600,700	14,5,7		4
5	11 02 00 00 00 00 00 00 00 00 00 00 00 00 02	14		C11120 DE 07 2B 11E185 A100 10 94 00 00 03	3	400	3		5,110,120,130
6	12 02 00 00 00 00 00 00 00 00 00 00 00 00 02	15		C11120 DE 07 2B 11E185 A100 10 94 00 00 03	1	100,200,300	1,2,3		
7	13 02 00 00 00 00 00 00 00 00 00 00 00 00 02	16		C11120 DE 07 2B 11E185 A100 10 94 00 00 03	2	19,20,21	30,40,50	aa:cc:bb:ee:dd:ff	
8	14 02 00 00 00 00 00 00 00 00 00 00 00 00 02	11		C11120 DE 07 2B 11E185 A100 10 94 00 00 03	4	11,12,13,14,15,16,17,18	1,2,3,4,5,6,7,8	22:33:44:55:66:77	101.102.103
9	14 02 00 00 00 00 00 00 00 00 00 00 00 00 01	*		0 12 23 34 45 56 67 78 90 01 12 23 34 45 56 67 78	1	199	2		
10	14 02 00 00 00 00 00 00 00 00 00 00 00 00 02	*		9 *	1	211,222	4,5		
11	14 02 00 00 00 00 00 00 00 00 00 00 00 00 02	*		7 C11120 DE 07 2B 11E185 A100 10 94 00 00 03	1	777,111,222	3,4,5		
12	20 02 00 00 00 00 00 00 00 00 00 00 00 00 02	20		C11120 DE 07 2B 11E185 A100 10 94 00 00 03	1	100	1		
13	30 02 00 00 00 00 00 00 00 00 00 00 00 00 02	21		C11120 DE 07 2B 11E185 A100 10 94 00 00 03	2	200	2	aabbccddeeff	
14	10 02 00 00 00 00 00 00 00 00 00 00 00 00 02	22		C11120 DE 07 2B 11E185 A100 10 94 00 00 03	1	300,500,600,700	14,5,7		4
15	11 02 00 00 00 00 00 00 00 00 00 00 00 00 02	23		C11120 DE 07 2B 11E185 A100 10 94 00 00 03	3	400	3		5,110,120,130
16	12 02 00 00 00 00 00 00 00 00 00 00 00 00 02	24		C11120 DE 07 2B 11E185 A100 10 94 00 00 03	1	100,200,300	1,2,3		
17	13 02 00 00 00 00 00 00 00 00 00 00 00 00 02	25		C11120 DE 07 2B 11E185 A100 10 94 00 00 03	2	19,20,21	30,40,50	aa:cc:bb:ee:dd:ff	
18	14 02 00 00 00 00 00 00 00 00 00 00 00 00 02	26		C11120 DE 07 2B 11E185 A100 10 94 00 00 03	4	11,12,13,14,15,16,17,18	1,2,3,4,5,6,7,8	22:33:44:55:66:77	101.102.103
19	20 02 00 00 00 00 00 00 00 00 00 00 00 00 02	27		C11120 DE 07 2B 11E185 A100 10 94 00 00 03	1	100	1		
20	30 02 00 00 00 00 00 00 00 00 00 00 00 00 02	28		C11120 DE 07 2B 11E185 A100 10 94 00 00 03	2	200	2	aabbccddeeff	
21	10 02 00 00 00 00 00 00 00 00 00 00 00 00 02	29		C11120 DE 07 2B 11E185 A100 10 94 00 00 03	1	300,500,600,700	14,5,7		4
22	11 02 00 00 00 00 00 00 00 00 00 00 00 00 02	30		C11120 DE 07 2B 11E185 A100 10 94 00 00 03	3	400	3		5,110,120,130
23	12 02 00 00 00 00 00 00 00 00 00 00 00 00 02	31		C11120 DE 07 2B 11E185 A100 10 94 00 00 03	1	100,200,300	1,2,3		
24	13 02 00 00 00 00 00 00 00 00 00 00 00 00 02	32		C11120 DE 07 2B 11E185 A100 10 94 00 00 03	2	19,20,21	30,40,50	aa:cc:bb:ee:dd:ff	

UNP parses the VSI manager ID with a three-field key (VSI Type; VSI Type Version; VSI Instance ID).

On an EVB bridged port, only the VLAN ID from the database is matched with the VLAN tag pre-assigned by the hypervisor with UNP trust-tag functionality. The VM is classified into the matching profile VLAN and associated QoS policy list.

- If UNP dynamic VLAN configuration is enabled, the switch will dynamically create the VLAN if necessary.
- If UNP dynamic profile configuration is enable, the switch will dynamically create the vNP profile if necessary.

On an EVB service access port, the VLAN ID and Group ID fields are used to determine the matching profile. UNP will request the service manager to dynamically create the SPB service on this switch if:

- The expected service does not exist;
- The system level dynamic-service-configuration is enabled; and
- The profile has the information required to create the service.

See the [Chapter 3, “Configuring Shortest Path Bridging,”](#) and [Chapter 27, “Configuring Universal Network Profiles,”](#) for more information about SPB services and vNP profiles.



## Tracking Virtual Machines

Virtual hypervisor managers can perform virtual machine (VM) movement without intervention. When this type of movement occurs, the hypervisor manager indicates to which hypervisor (virtual server) the virtual machine was moved. However, it is not easy to tell which OmniSwitch port in the data center mesh the virtual machine is now using to access the network.

Alcatel-Lucent network management tools provide management and visibility for virtual machines in the mesh. The suite of management tools includes:

- OmniVista 2500 Virtual Machine Management (VMM) for management of VM mobility and integration with standard hypervisors.
- OmniVista 2500 NMS for mesh provisioning and management.

OmniVista 2500 provides a single pane of management for VMs across the network, which includes VM visibility and their point of association to the network. This gives network administrators a dashboard to determine and track the following information:

- Virtual machine locations.
- Virtual machine types.
- The switch ports to which the virtual machines are connected.
- The duration of the connections.
- Which virtual network profile (vNP) the virtual machine is using.

VMM is an embedded application within OmniVista that consists of the following components:

- **vCenter integration:** VM discovery, VM polling and event listener service will listen to preference changes and event notifications.
- **vNP configuration:** Profiles of UNP configuration that specify the configuration associated with each VM VLAN. The profiles can be assigned to switches, as and when needed.
- **VLAN Notification:** Displays a list of VM instances where the network is missing some configuration to effectively handle VMs attached. This allows the administrator to be proactive in identifying and correcting any necessary network configuration.
- **OmniVista Locator:** Provides the ability to search for specific VMs using various search criteria.



# A Software License and Copyright Statements

This appendix contains Alcatel-Lucent and third-party software vendor license and copyright statements.

## Alcatel-Lucent License Agreement

### ALCATEL-LUCENT SOFTWARE LICENSE AGREEMENT

---

**IMPORTANT.** Please read the terms and conditions of this license agreement carefully before opening this package.

---

**By opening this package, you accept and agree to the terms of this license agreement. If you are not willing to be bound by the terms of this license agreement, do not open this package. Please promptly return the product and any materials in unopened form to the place where you obtained it for a full refund.**

1. **License Grant.** This is a license, not a sales agreement, between you (the “Licensee”) and Alcatel-Lucent. Alcatel-Lucent hereby grants to Licensee, and Licensee accepts, a non-exclusive license to use program media and computer software contained therein (the “Licensed Files”) and the accompanying user documentation (collectively the “Licensed Materials”), only as authorized in this License Agreement. Licensee, subject to the terms of this License Agreement, may use one copy of the Licensed Files on the Licensee’s system. Licensee agrees not to assign, sublicense, transfer, pledge, lease, rent, or share their rights under this License Agreement. Licensee may retain the program media for backup purposes with retention of the copyright and other proprietary notices. Except as authorized under this paragraph, no copies of the Licensed Materials or any portions thereof may be made by Licensee and Licensee shall not modify, decompile, disassemble, reverse engineer, or otherwise attempt to derive the Source Code. Licensee is also advised that Alcatel-Lucent products contain embedded software known as firmware which resides in silicon. Licensee may not copy the firmware or transfer the firmware to another medium.

2. **Alcatel-Lucent’s Rights.** Licensee acknowledges and agrees that the Licensed Materials are the sole property of Alcatel-Lucent and its licensors (herein “its licensors”), protected by U.S. copyright law, trademark law, and are licensed on a right to use basis. Licensee further acknowledges and agrees that all rights, title, and interest in and to the Licensed Materials are and shall remain with Alcatel-Lucent and its licensors and that no such right, license, or interest shall be asserted with respect to such copyrights and trademarks. This License Agreement does not convey to Licensee an interest in or to the Licensed Materials, but only a limited right to use revocable in accordance with the terms of this License Agreement.

3. **Confidentiality.** Alcatel-Lucent considers the Licensed Files to contain valuable trade secrets of Alcatel-Lucent, the unauthorized disclosure of which could cause irreparable harm to Alcatel-Lucent. Except as expressly set forth herein, Licensee agrees to use reasonable efforts not to disclose the Licensed Files to any third party and not to use the Licensed Files other than for the purpose authorized by this License Agreement. This confidentiality obligation shall continue after any termination of this License Agreement.

4. **Indemnity.** Licensee agrees to indemnify, defend and hold Alcatel-Lucent harmless from any claim, lawsuit, legal proceeding, settlement or judgment (including without limitation Alcatel-Lucent's reasonable United States and local attorneys' and expert witnesses' fees and costs) arising out of or in connection with the unauthorized copying, marketing, performance or distribution of the Licensed Files.

5. **Limited Warranty.** Alcatel-Lucent warrants, for Licensee's benefit alone, that the program media shall, for a period of ninety (90) days from the date of commencement of this License Agreement (referred to as the Warranty Period), be free from defects in material and workmanship. Alcatel-Lucent further warrants, for Licensee benefit alone, that during the Warranty Period the Licensed Files shall operate substantially in accordance with the functional specifications in the User Guide. If during the Warranty Period, a defect in the Licensed Files appears, Licensee may return the Licensed Files to Alcatel-Lucent for either replacement or, if so elected by Alcatel-Lucent, refund of amounts paid by Licensee under this License Agreement. EXCEPT FOR THE WARRANTIES SET FORTH ABOVE, THE LICENSED MATERIALS ARE LICENSED "AS IS" AND ALCATEL-LUCENT AND ITS LICENSORS DISCLAIM ANY AND ALL OTHER WARRANTIES, WHETHER EXPRESS OR IMPLIED, INCLUDING (WITHOUT LIMITATION) ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. SOME STATES DO NOT ALLOW THE EXCLUSION OF IMPLIED WARRANTIES SO THE ABOVE EXCLUSIONS MAY NOT APPLY TO LICENSEE. THIS WARRANTY GIVES THE LICENSEE SPECIFIC LEGAL RIGHTS. LICENSEE MAY ALSO HAVE OTHER RIGHTS WHICH VARY FROM STATE TO STATE.

6. **Limitation of Liability.** Alcatel-Lucent's cumulative liability to Licensee or any other party for any loss or damages resulting from any claims, demands, or actions arising out of or relating to this License Agreement shall not exceed the license fee paid to Alcatel-Lucent for the Licensed Materials. IN NO EVENT SHALL ALCATEL-LUCENT BE LIABLE FOR ANY INDIRECT, INCIDENTAL, CONSEQUENTIAL, SPECIAL, OR EXEMPLARY DAMAGES OR LOST PROFITS, EVEN IF ALCATEL-LUCENT HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. SOME STATES DO NOT ALLOW THE LIMITATION OR EXCLUSION OF LIABILITY FOR INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THE ABOVE LIMITATION OR EXCLUSION TO INCIDENTAL OR CONSEQUENTIAL DAMAGES MAY NOT APPLY TO LICENSEE.

7. **Export Control.** This product is subject to the jurisdiction of the United States. Licensee may not export or reexport the Licensed Files, without complying with all United States export laws and regulations, including but not limited to (i) obtaining prior authorization from the U.S. Department of Commerce if a validated export license is required, and (ii) obtaining "written assurances" from licensees, if required.

8. **Support and Maintenance.** Except as may be provided in a separate agreement between Alcatel-Lucent and Licensee, if any, Alcatel-Lucent is under no obligation to maintain or support the copies of the Licensed Files made and distributed hereunder and Alcatel-Lucent has no obligation to furnish Licensee with any further assistance, documentation or information of any nature or kind.

9. **Term.** This License Agreement is effective upon Licensee opening this package and shall continue until terminated. Licensee may terminate this License Agreement at any time by returning the Licensed Materials and all copies thereof and extracts therefrom to Alcatel-Lucent and certifying to Alcatel-Lucent in writing that all Licensed Materials and all copies thereof and extracts therefrom have been returned or erased by the memory of Licensee's computer or made non-readable. Alcatel-Lucent may terminate this License Agreement upon the breach by Licensee of any term hereof. Upon such termination by

Alcatel-Lucent, Licensee agrees to return to Alcatel-Lucent or destroy the Licensed Materials and all copies and portions thereof.

**10. Governing Law.** This License Agreement shall be construed and governed in accordance with the laws of the State of California.

**11. Severability.** Should any term of this License Agreement be declared void or unenforceable by any court of competent jurisdiction, such declaration shall have no effect on the remaining terms herein.

**12. No Waiver.** The failure of either party to enforce any rights granted hereunder or to take action against the other party in the event of any breach hereunder shall not be deemed a waiver by that party as to subsequent enforcement of rights or subsequent actions in the event of future breaches.

**13. Notes to United States Government Users.** Software and documentation are provided with restricted rights. Use, duplication or disclosure by the government is subject to (i) restrictions set forth in GSA ADP Schedule Contract with Alcatel-Lucent's reseller(s), or (ii) restrictions set forth in subparagraph (c) (1) and (2) of 48 CFR 52.227-19, as applicable.

**14. Third Party Materials.** Licensee is notified that the Licensed Files contain third party software and materials licensed to Alcatel-Lucent by certain third party licensors. Some third party licensors are third party beneficiaries to this License Agreement with full rights of enforcement. Please refer to the section entitled "[Third Party Licenses and Notices](#)" on page -4 for the third party license and notice terms.

## Third Party Licenses and Notices

Legal Notices applicable to any software distributed alone or in connection with the product to which this document pertains, are contained in files within the software itself located at: **/flash/foss**.

Also, if needed, we provide all FOSS (Free and Open Source Software) source code used into this release at the following URL: <https://service.esd.alcatel-lucent.com/portal/page/portal/EService/release>

# Index

## B

backbone VLAN 3-18, 3-26  
BVLAN *see* backbone VLAN

## D

Data Center Bridging 2-1, 4-12, 5-19  
    Interaction with Other Features 2-13, 4-12, 5-19  
    Profiles 2-16  
Data Center Bridging Exchange 2-1, 2-10  
DCB *see* Data Center Bridging  
DCBx *see* Data Center Bridging Exchange

## E

Edge Virtual Bridging 6-1  
Enhanced Transmission Selection 2-1, 2-7  
ETS *see* Enhanced Transmission Selection  
EVB *see* Edge Virtual Bridging

## F

FCoE Initialization Protocol 4-1  
FCoE *see* Fibre Channel over Ethernet  
FCoE/FC Gateway 5-1  
    E\_Port proxy 5-1  
    F\_Port proxy 5-1  
    N\_Port proxy 5-1  
Fibre Channel over Ethernet 4-1  
FIP snooping 4-1  
FIP *see* FCoE Initialization Protocol

## I

ISIS-SPB 3-9

## P

PBB *see* Provider Backbone Bridge  
PFC *see* Priority-based Flow Control  
Priority-based Flow Control 2-1, 2-5  
Provider Backbone Bridge  
    802.1AH 3-6  
    network 3-6  
    SPB services 3-12

## Q

qos command 4-19  
qos qsi qsp dcb command 2-23  
qos qsi qsp dcb dcbx admin-state command 2-25  
qos qsi qsp dcb dcbx ets command 2-25  
qos qsi qsp dcb dcbx pfc command 2-25

qos qsp dcb import command 2-23  
qos qsp dcb tc command 2-23

## S

service access l2profile command 3-41  
service access vlan-xlation command 3-36, 3-39  
service l2profile command 3-40  
service spb admin-state command 3-36  
service spb command 3-35  
service spb sap admin-state command 3-43  
service spb sap command 3-41  
service spb sap trusted command 3-43  
service spb vlan-xlation command 3-36  
Shortest Path Bridging 3-1, 3-6  
    benefits 3-1  
    bridge ID 3-30  
    bridge priority 3-30  
    BVLAN 3-26  
    ISIS-SPB 3-9  
    services 3-12  
    shortest path trees 3-8  
    SPT 3-8  
    system ID 3-30  
    topology example 3-13  
show qos qsi dcbx command 2-25  
show qos qsp dcb command 2-23  
spb isis admin-state command 3-34  
spb isis area-address command 3-30  
spb isis bridge-priority command 3-30  
spb isis lsp-wait command 3-32  
spb isis overload command 3-33  
spb isis overload-on-boot command 3-33  
spb isis source-id command 3-31  
spb isis spf-wait command 3-31  
SPB *see* Shortest Path Bridging

## U

Universal Network Profile  
    Profile Types 6-4  
    Virtual Network Profile 6-1

## V

Virtual Machines  
    Classifying 6-4  
    Tracking 6-13

