Network Configuration Example

Configuring Layer 2 Cloud Data Center Tenants

Published: 2014-09-19

Juniper Networks, Inc.
1194 North Mathilda Avenue
Sunnyvale, California 94089
USA
408-745-2000
www.juniper.net

Network Configuration Example Configuring Layer 2 Cloud Data Center Tenants
NCE0113

The information in this document is current as of the date on the title page.

YEAR 2000 NOTICE

Juniper Networks hardware and software products are Year 2000 compliant. Junos OS has no known time-related limitations through the year 2038. However, the NTP application is known to have some difficulty in the year 2036.

END USER LICENSE AGREEMENT

The Juniper Networks product that is the subject of this technical documentation consists of (or is intended for use with) Juniper Networks software. Use of such software is subject to the terms and conditions of the End User License Agreement ("EULA") posted at http://www.juniper.net/support/eula.html. By downloading, installing or using such software, you agree to the terms and conditions of that EULA.

Table of Contents

Cloud Data Center Overview

The cloud data center (CDC) focuses on providing data center connectivity from various external networks to services located within a multi-tenant data center. A CDC typically has more challenging business requirements than a more traditional data center that only services a single entity. Multi-tenancy requires high security, scale, and performance.

The advanced routing and switching features of Juniper Networks® MX Series 3D Universal Edge Routers enables the creation of a secure, scalable, and high performance data center. Depending on the size of the data center and how many tenants are required, various tiers of the data center network can be collapsed into a single tier.

This document outlines the problems and challenges that multi-tenant data center operators face, the various roles and requirements of each tier in the architecture, and an end-to-end solution architecture to solve these problems and meet crucial requirements.

Customer Use Case

Many customers are in the process of converting select points of presence (POPs) and remote sites to cloud data center (CDC) sites. One of the main business drivers behind these conversions is the demand for new revenue-generating services. Wireline service providers in particular are experiencing commoditization pressure. Where these service providers once generated healthy revenue and profits on Internet connectivity services such as Virtual Private Networks (VPNs), increased competition has now made these services considered the standard with pressure to shrink profit margins.

At the same time, smaller cloud service providers are generating revenues by backhauling traffic across the Internet through fast pipes and VPN services that they purchased from larger service providers. The larger service providers are offering inexpensive backhaul services to the smaller cloud service providers when they could be capitalizing on their infrastructure. In other words, the large service provider is in a position to offer these same services, on a broader scale and potentially reduced cost, to enable creation of new, in-demand, revenue-generating services.

Cloud and virtualization technologies are relatively new and have, until recently, been controlled by only a few providers who have kept their deployments secret. As these technologies are better understood, and new technology options are brought to market that facilitate the implementation of such cloud data centers, cloud service deployments have become more commonplace.

As all of these new technology trends continue to converge, wireline service providers must move rapidly to develop cloud and data center service offerings. This document is intended to empower service providers to become true cloud providers by showing how best practice configurations for the creation of the Layer 2 CDC can be implemented.

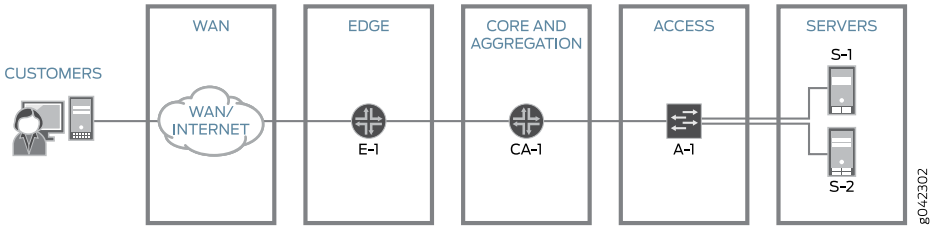
Solution: Technical Overview

The following sections provide a more in-depth technical overview of the cloud data center and its key functional components:

End-to-End Architecture

Cloud data center CDC networks contain four basic building blocks to provide transport between customers and servers. Each component of this solution is broken out into a discrete functional block as shown in Figure 1 on page 2.

Figure 1: Architectural Function Blocks



As shown in the diagram, the CDC solution provides customer access to virtual servers and is agnostic to the underlying WAN. Each customer is completely isolated from other customers in the data center.

Functional Description

Each component of the solution has a discrete set of responsibilities that must work together to create an end-to-end solution architecture. The major components in the CDC include the customers, WAN, edge router, core and aggregation router, network services, and access switch. Depending on the requirements and deployment scenario, some could choose to collapse several of these tiers into a single tier, trading complexity and segmentation for simplicity and cost savings.

The following sections provide more detail about these CDC components mentioned above:

Customers

The types of customers supported by a CDC deployment can vary, depending on the deployment scenario configured. The CDC is most commonly used by a business or corporate customer who purchases virtual servers from a service provider. Another common customer type is an end user accessing cloud-hosted content from a workstation through a Web browser.

WAN

The role of the WAN is to provide transport from the user to the data center. How the WAN looks depends on the requirements associated with the customer deployment scenario.

- Internet on page 3
- MPLS Backbone on page 3
- Private Peering on page 3

Internet

The Internet offers various forms of data center access. This access can be as simple as accessing a public IP address on the Internet, but can include more complex access options, for example, using tunneling protocols such as generic routing encapsulation (GRE) and IPsec to provide private IP addressing and security.

MPLS Backbone

Another common option to delivering customer access into the CDC is through the use of a customer's existing managed MPLS network. Again, the type of VPN used with this option depends on the customer's access requirements. However, the most common examples are to use Layer 2 or Layer 3 VPNs.

Private Peering

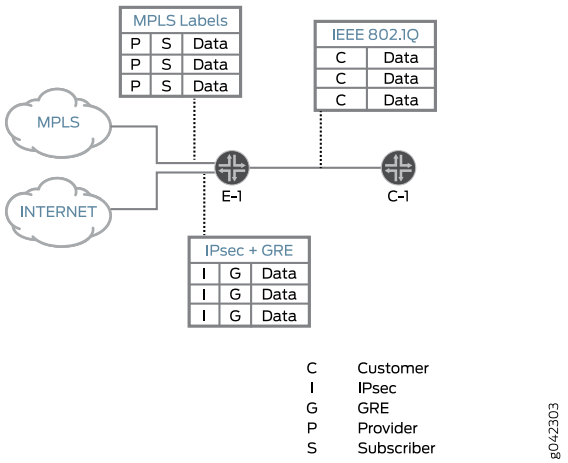
Customers with higher performance and security requirements might want to peer directly with the cloud provider. Private peering is dedicated to a single tenant where other tenant traffic does not impact the transport. This transport method also provides a trusted medium between the tenant and the service provider, typically eliminating the need for any transport encryption.

Edge Router

The edge router has the most requirements and responsibilities in a CDC solution. The primary responsibilities of the edge router are to aggregate all edge networks into the data center and to serve as the throttle point for all ingress and egress traffic for the data center.

In addition, the edge router has the unique responsibility of multiplexing and demultiplexing customer traffic from various edge networks prior to it entering the data center. Figure 2 on page 4 shows edge router E1 accepting various edge network traffic (for example, MPLS backbone and Internet transit traffic) and mapping it to a specific customer.

Figure 2: Edge Router Multiplexing and Demultiplexing



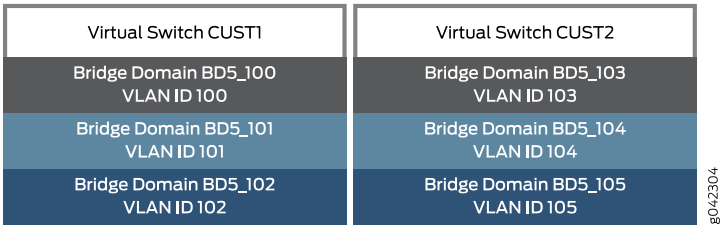
To ensure that customer traffic is isolated and secure, the edge router must have a consistent method of mapping customer traffic from the edge networks into the data center.

Environments containing multiple tenants place additional requirements on the edge routers, making it unlike a traditional core router in a single-tenant data center, campus network, or core network. Traditionally, the edge router is a pure Layer 3 device and does not participate in Layer 2. This means that supporting multiple tenants typically requires that service providers provide both Layer 2 and Layer 3 access to customers.

Both the edge and aggregation routers in a multi-tenant environment must support high-scale Layer 2. To support the high-scale requirements of a multi-tenant data center, the core device must use the concept of a virtual switch to transport customer traffic. IEEE 802.1Q scales up to 4,094 VLANs. There are deployment scenarios where customer access switches do not support more than 4094 VLANs. In these cases the edge and aggregation routers must support VLAN normalization and offer IEEE 802.1Q to such access switches. Deterministic VLAN normalization is required to maintain customer isolation in a multi-tenant environment. This will be detailed further in this document when taking a look at the example deployments.

The edge and the customer access routers are the throttle points for all bridged traffic. To support more than 4,094 VLANs, these routers must be able to support IEEE 802.1Q bridge domains and virtual switches as shown in Figure 3 on page 5 (only the first three VLANs are shown). These virtual switch instances constitute customer deployments on the edge routers while, on the core (and/or aggregation) routers, virtual switch instances are used in a less granular manner.

Figure 3: Bridge Domain and Virtual Switch Support



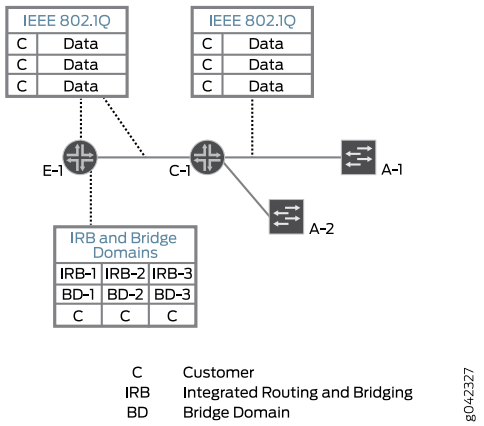
One of the drawbacks of using IEEE 802.1Q is that it requires a large number of logical interfaces. This level of scale is amplified for every network-to-network interface (NNI). For example, to support 32,000 VLANs, each NNI must support 32,000 logical interfaces. Each node in the network typically has at least threeNNIs to connect any upstream and inter-chassis links. However, this does not include downstream devices like access switches. To support 20 access switches and 3 NNIs, each core node must support 736,000 logical interfaces.

The alternative to supporting this number of interfaces is to configure a virtual switch, where each instance can support up to 4094 bridge domains or VLANs on the core device. To support 32,000 VLANs, a total of eight virtual switch instances with 4094 bridge domains each would be required on each customer access router with the links to the access top of racks (ToRs) and edge device being configured as trunk ports. This method greatly reduces the number of logical interfaces required, but at the expense of using more NNIs. In contrast, using virtual switches requires a network port for each routing instance. For example, configuring 32,000 tenants requires a total of eight network ports. However, to maintain deterministic performance and reduce latency, you can use one NNI per 4094 bridge domains. How the virtual switches and bridge domains are mapped at the core and edge will be discussed in more detail in the following sections.

Core Access

In the core of the cloud data center (CDC), additional demands are placed on the core and aggregation routers that make up this layer, unlike the core in a single tenant data center or core network. In this design core routers only provide Layer 2 functions, as they are only providing transport across networking infrastructure only. In the high multi-tenancy design of the CDC core, these devices need to have the capabilities to support Layer 2 functions to map customer traffic into the servers and services that reside in the access layer.

Figure 4: CDC Core and Aggregation Router Overview

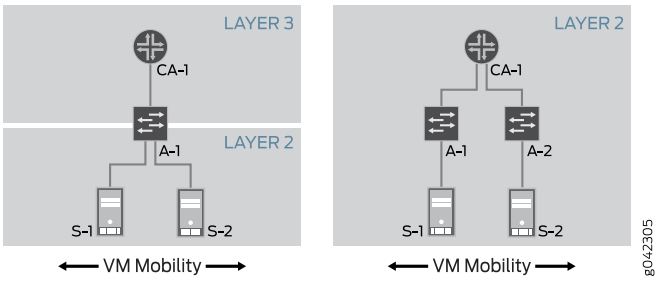


Access Switch

The access tier terminates all server and end-host connections. Depending on the virtual machine (VM) mobility requirements, the access switch can operate at only Layer 2 or both Layer 2 and Layer 3. If VM mobility is limited to a single access switch, then it is possible for the access switch to operate at both Layer 2 and Layer 3. However, if VM mobility must traverse multiple access switches, so must the broadcast domain. Figure 5 on page 6 illustrates how VM mobility impacts the access tier with regard to Layer 2 and Layer 3 boundaries. A1 shows the access switch, and S1 and S2 show individual servers.

NOTE: If the access switch peers in Layer 3 with the core and aggregation switch, you cannot deliver a Layer 2 VPN service to the customer.

Figure 5: VM Mobility Options



Because of the varying customer requirements, we always recommend that you deploy the access switches as Layer 2. This deployment enables the customer to freely change between a Layer 2 or Layer 3 hand-off. The access switch has the fewest scaling requirements, because the majority of the scaling is managed by the core and aggregation router.

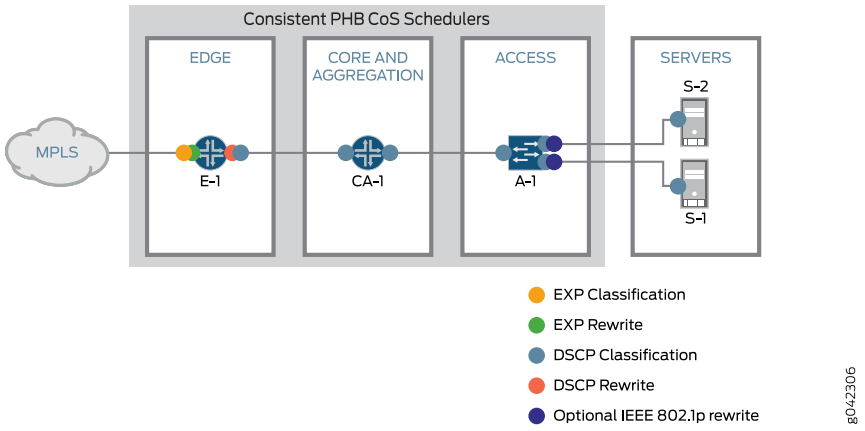
Using the recommended approach of configuring virtual switches and bridge domains, the access switch need only know how to handle IEEE 802.1Q Ethernet frames.

CoS Decision Points

In a multi-tenant data center, class of service (CoS) is subject to customer requirements. However, generally speaking, the network infrastructure must support two levels of service.

Figure 6 on page 7 shows the various CoS classifications and rewrites in a CDC topology. All tiers in the network should have a consistent per-hop behavior (PHB) with regard to scheduler configuration and classification. The edge router must be configured to preserve the classification between MPLS and core networks with MPLS textual conventions (TCs) and differentiated services code point (DSCP) rewriting. Other routers and switches in the topology must be configured for consistent scheduling and classification of DSCP packets.

Figure 6: CDC CoS Locations



The schedulers are divided into four forwarding classes: network-control, expedited-forwarding, assured forwarding, and best-effort. This number of forwarding classes provides enough granularity to mark two levels of business applications, voice, and video traffic.

Layer 2 Customer Configuration Requirements

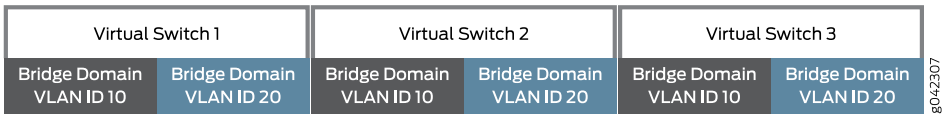
The Layer 2 customer configurations were designed to adhere to the following requirements:

- No Layer 3 was provided by the cloud provider.
- Customers each have their own VPLS instance.
- Customers might each have multiple bridge domains (VLANs).

- Customers must each be logically separated from any other customer and not be able to detect the existence of any other customer.
- The cloud provider must be able to flexibly assign ToR VLANs to a customer such that the customer is not tied to one ToR or VLAN domain and is not explicitly tied to one access or ToR VLAN ID.

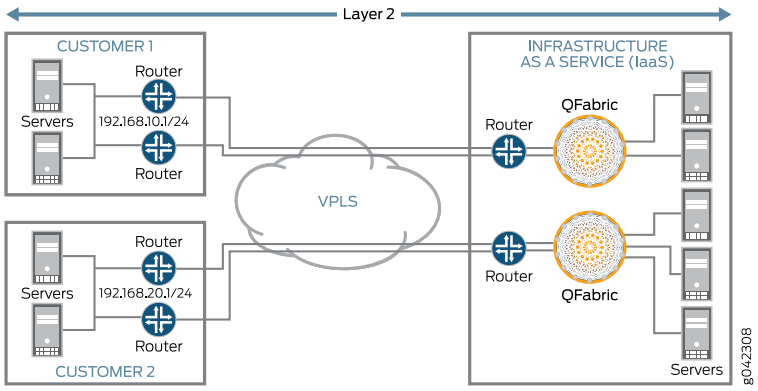
To implement these requirements, each customer configuration was defined at the edge as a separate virtual switch instance. This concept is shown in Figure 7 on page 8 where three Layer 2 customers are each assigned two bridge domains.

Figure 7: Layer 2 Customer Assignments to Bridge Domains



In addition, an integrated routing and bridging (IRB) instance was not configured for any customer bridge domains. Figure 8 on page 8 shows that all the Layer 3 addressing and accessing the VLANs is provided by either the remote PE (IRB instances in the bridge domains) or by using plain Layer 3 interfaces on the customer premise device.

Figure 8: Layer 3 Addressing and VLAN Access



For further reference, see Figure 9 on page 9. The figure shows the Layer 2 customer requirements in more detail.

Figure 9: Layer 2 Customer Requirements (Detailed View)

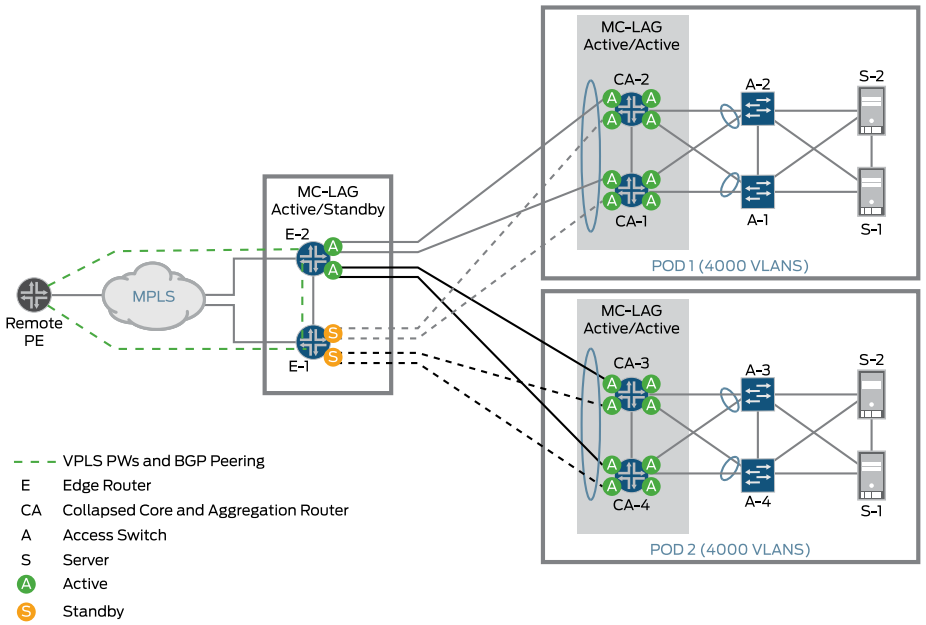
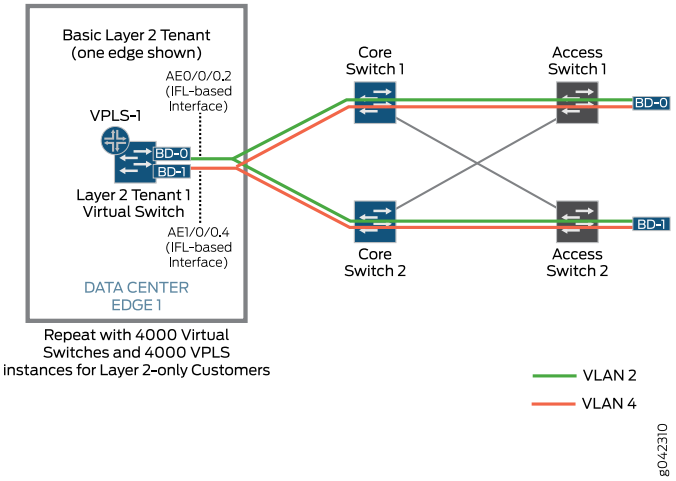


Figure 10 on page 10 shows an example in which a Layer 2 customer is assigned VLANs from a point of delivery (POD). There is more than one POD, and each POD contains 4,000 separate VLAN domains (none of which have any knowledge of the other). Multichassis link aggregation group (MC-LAG) active/active is configured from the access switches A-1 and A-2 to the collapsed core and aggregation routers, while MC-LAG at the edge is configured as active/standby, because MC-LAG active/active is not supported in combination with VPLS at the edge. To minimize the number of logical interfaces across the topology, trunk ports are used from the access switches and the core switches. On the edge, the trunks are logical interface-based so that the logical interface units from the MC-LAG instances (ae0) can each be assigned to a bridge domain. Both bridge domains are then assigned to a virtual switch and then to a VPLS instance. This design allows for VLAN normalization with the bridge domain.

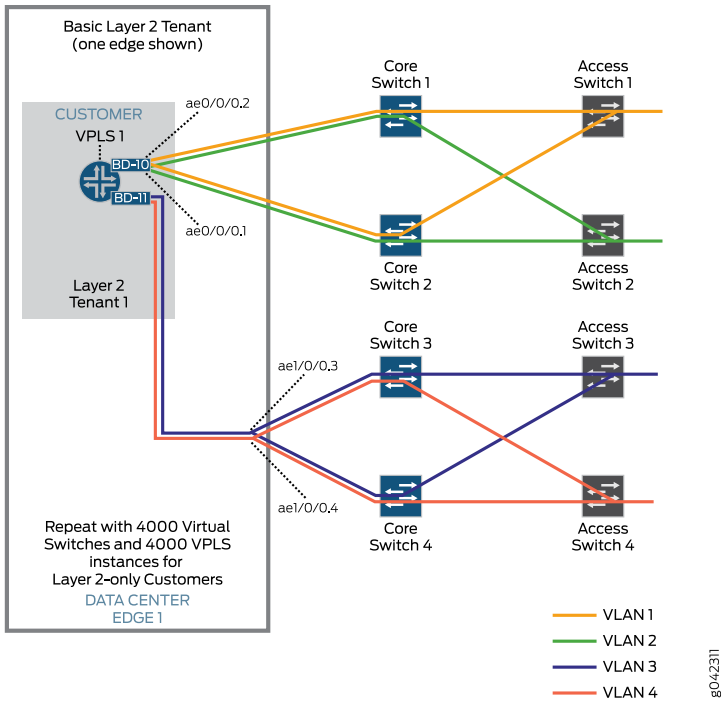
Figure 10: Layer 2 Customer Requirements (Simple View)



In addition, the use of virtual switches for VPLS configuration means that, even with multiple VLANs assigned, each customer requires only one VPLS instance in the WAN.

Figure 11 on page 10 shows this model extended further to provide the requirement for flexible access and core layer VLAN assignments to customers across PODs, while keeping the VLANs transparent to the customers due to their bridge domain VLAN IDs not changing.

Figure 11: Layer 2 Customer Requirements (Extended View)

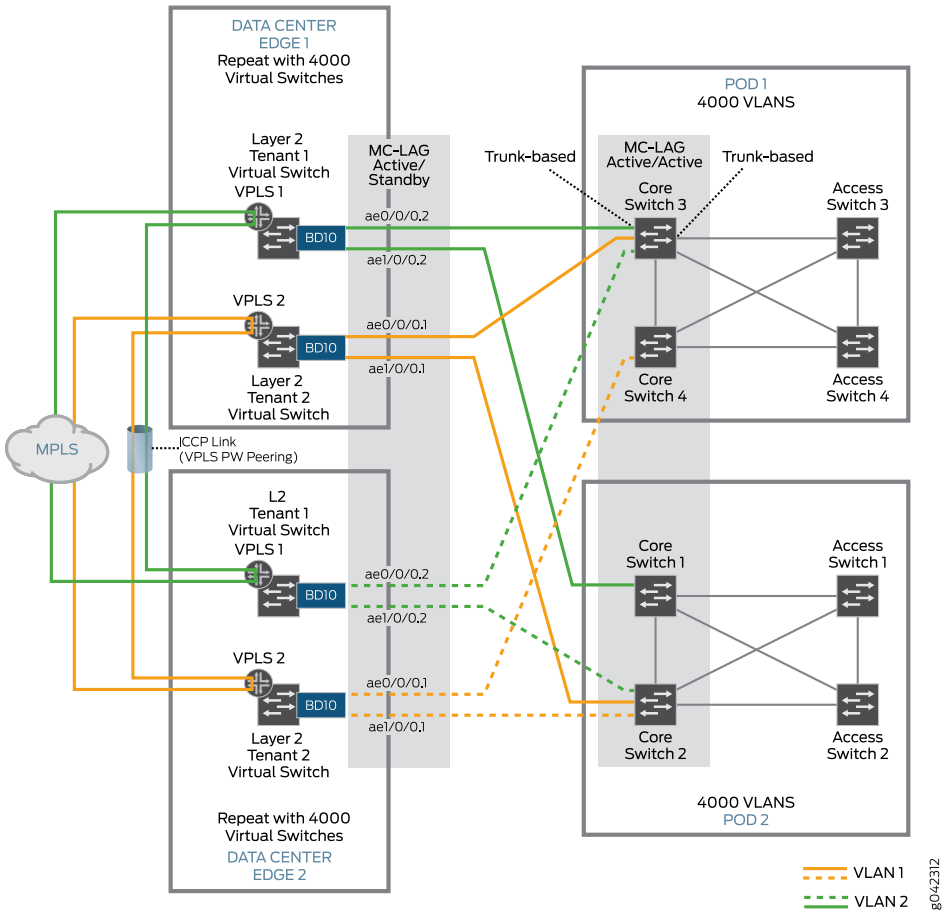


Even though the customer has only two VLANs assigned within the data center, the provider can assign as many VLANs as required for capacity planning and for resiliency planning because the customer is spread across VLAN PODs, which minimizes impacts to the customer during any planned or unplanned outages.

Figure 12 on page 12 shows the complete data center edge design, including the VPLS connectivity. In this topology, the Edge 1 data center VPLS routing instance for the customer is active, and on standby for Edge 2. The diagram also shows the MC-LAG active/active and active/standby configurations and how they provide resiliency throughout the data center.

It is important to note that there is a full mesh of VPLS pseudowires to the remote PE as well as between the data center edge routers to provide resiliency for East-West POD traffic for a customer when the active aggregated Ethernet interface in the MC-LAG active/standby fails on the edge router. In addition, "dummy" logical tunnel interfaces are configured on the edge routers for each VPLS instance to ensure that, in the event of a node failure at the edge, though they are not used for forwarding, the pseudowires are already operational and able to connect to the standby edge router. This configuration greatly reduces the failover learning time when the active VPLS forwarding device fails for a given vSwitch.

Figure 12: Complete Data Center Edge Design Topology



The following failure scenarios are based on Figure 12 on page 12:

- *Data Center Edge 1 interface ae0 failure* – Bridge domain BD-10 for both customers becomes active on device Data Center (DC) EDGE 2. When this occurs, however, a bridge domain becomes active on device Data Center EDGE 1 because interface ae1/0/0.1 is still up in that bridge domain (BD-10). This means that VPLS is active on both edge routers and is where the VPLS site ID's and the VPLS pseudowire peering between the edge routers is used to forward traffic between edge routers. Bridge domain BD-10 traffic from POD 1 enters device DC EDGE 2 and crosses the VPLS pseudowire to bridge domain BD-10 on device DC EDGE 1, enabling clients in VLAN-3 on POD 2 to communicate with clients in VLAN-1 on POD 1. Clients must continue to access the WAN over device DC EDGE 1.
- *Core Failure*- All MC-LAG active/active sessions use the remaining active router and the edge and access devices use the remaining link in the MC-LAG active/standby routing instance. No WAN core change occurs.
- Access switch failure - When testing for server (end host) resiliency, where it is dual-homed to both access switches in the POD, the access switch uses the remaining

link to the active access switch, and the core and/or aggregation router uses the remaining link in the MC-LAG active/active routing instance.

- *Aggregated Ethernet link failure*- All aggregated Ethernet instances are set with a minimum link of 1 to ensure that the aggregated Ethernet bundle remains active. The traffic impact is that it becomes MAC learning.

Table 1 on page 13 explains a few of the possible failure and action scenarios for Figure 12 on page 12.

Table 1: Network Failure Points and Resulting Actions

Failure Point	Action
Device Date Center EDGE 1 interface ae0	• Bridge domain BD-10 for both customers becomes active on device Data Center EDGE 2. • A bridge domain becomes active on device Data Center EDGE-1 and Edge-2 because interface ae1/0/0.2 and ae1/0/0.1 are still up on EDGE-1 even when AE0 fails on EDGE-1. • VPLS remains active on both edge routers (where the VPLS site IDs and the VPLS pseudowire peering between the edge routers is used to forward traffic between edge routers). • Bridge domain BD-10 traffic from POD 1 enters device Data Center EDGE 2 and crosses the VPLS pseudowire to bridge domain BD-10 on device Data Center EDGE 1, This action enables clients in VLAN 1on POD 2 to communicate with clients in VLAN 1 on POD 1. • Clients continue to access the WAN over device Data Center EDGE 1.
Core/Aggregation router	• All MC-LAG active/active sessions use the remaining active router; and the edge and access devices use the remaining link in the MC-LAG active/standby routing instance. • No WAN core change occurs.
Access switch (testing for server (end host) resiliency, where it is dual-homed to both access switches in the POD)	• The access switch uses the remaining link to the active access switch. • The core or aggregation device uses the remaining link in the MC-LAG active/active routing instance.
Aggregated Ethernet Link	• All aggregated Ethernet instances are set with a minimum link of 1 to ensure that the aggregated Ethernet bundle remains active. • The traffic becomes MAC learning.

Cloud Data Center Design

Design Requirements

The following sections describe the various design requirements for the cloud data center (CDC).

Layer 2 Connection to the Cloud Data Center

The service provider must offer Layer 2 connection abilities to the cloud environment to enable some applications to reach other cloud assets using Layer 2, to provide extension of the storage network from the customer environment to the cloud, and to be able to accept a customer's virtual machine.

Layer 3 Connection to the Cloud Data Center

The cloud provider must provide Layer 3 connectivity to the cloud to enable routed connectivity between the cloud and the customer network, as well as to provide Internet access to customer cloud assets (for example, a customer webserver running in the cloud).

Cloud Infrastructure Mapping of Layer 2 and Layer 3 Customer Addresses

The cloud provider must provide Layer 2 and Layer 3 connectivity. Two possible methods of providing Layer 2 and Layer 3 connectivity are as follows:

- Extend the customer Layer 2 addressing schema and translate the Layer 2 address (that is, VLAN) at the cloud boundary. Typically, each customer gets one VLAN ID assigned in the cloud. That VLAN ID must represent that customer on that data center site and, in some cases, across data center sites. The latter implies that the unique VLAN ID must be carried over.
- Extend the customer Layer 3 addressing schema and do not translate between the customer and the cloud connection. This must be implemented in a way that enables the use of overlapping IP addresses while traffic separation occurs between tenants at the VLAN level.

High Degree of Network Investment Leverage

The data center provider must run technologies that enable active/active and multipath forwarding on the LAN and on the WAN. No interface on the path of equal cost network paths should be left un-utilized or under-utilized.

Flexible and Collapsed Network Design

Network design has to be easy or flat. Within the data center, in terms of traffic and scaling, any point to any point connections should occur through shortest and consistent length paths without throttle points. Complicated network designs cause scaling, operational, and provisioning problems. The cloud provider must be able to connect the

WAN service and path instance of a customer to the compute and storage assets with maximum ease and flexibility.

The network paths of each tenant must be established across the topology with ease and should allow for easy, on-demand expansion and contraction. One of the primary reasons customers purchase cloud services is the ease in which the services can be rapidly built up or torn down as the need for the service dictates. This flexibility must be built into the network design and be supported by any technologies chosen.

High Degree of Scaling and Virtualization

The CDC must provide a high degree of scale, virtualization, and flexibility to ensure that the service is profitable. The cloud business case relies on a much lower compute and storage unit cost, and profitability is achieved only when the degree of scaling and virtualization is high. Economies of scale help offer profitable services. Without this scale, unit cost becomes too high, leaving no profit for the provider.

Compute and storage resources might run on servers to achieve a certain degree of utilization, while tenant virtual machines or storage devices can be moved from under-utilized resources to shut down other servers to save energy and cooling costs.

Separation of Multi-Tenant Traffic

Cloud providers must ensure that the traffic of each tenant is separated such that each tenant is unable to communicate with another tenant unless it occurs through the use of a firewall or other controlled mechanism. This separation alleviates cloud customer concerns surrounding security within the data center as well as between the cloud and the enterprise data center.

Building a High-Performance Data Center

The cloud operator's data center must be a high-performance data center that utilizes fast pipes to minimize delays between virtual machines and the storage environment. In addition, the data center must have a large number of network, server, and storage ports to satisfy any scale and profitability requirements.

Network paths must display predictable and consistent latency, and any network or cloud performance must not depend on the location of any cloud assets.

Virtual Machine Mobility

The CDC must allow for virtual machine (VM) mobility to optimize compute resources as described above within the date center, and, in some cases, between the customer and the provider.

VM mobility technology requires that Layer 2 paths be used within the data center. To satisfy this requirement, Layer 2 paths can reside directly on the wire or be configured in overlay mode.

VM mobility poses a new challenge to the network infrastructure. Previously, wireline networks were static and end stations did not move. With the introduction of the cloud, and due to high scale and high utilization requirements for profitability, VMs can now move within and between data center sites. This means that Layer 2 and Layer 3 network

paths, and in some cases firewall state information, can move in lock step with the VM. In many cases, perfect (time) alignment of VM and network path migrations are difficult to achieve. To ease this migration, network designs or technologies that enable the movement of Layer 2 and Layer 3 paths, extending them with ease, are preferred. Network path extensions, in many cases, include replication of network state information between old and new locations, avoiding the occurrence of traffic interruption and application failures following the movement of any VM.

VM mobility is often necessary to optimize resource utilization within the data center. This resource optimization during VM mobility must occur without the knowledge of the customer, making network path extension and state replication abilities critical.

Programmability of Network Elements

The cloud solution infrastructure must provide a programmable means of building network paths in order to provide consistent, repeatable, and simplified automation. The high costs associated with automation and the complexity of network management are two of the more difficult items for the cloud operator to address. The value-added abilities of a cloud network infrastructure should help alleviate some of this cost and complexity.

Juniper Networks Cloud Data Center Solutions

Table 2 on page 16 outlines how the Juniper Networks cloud data center addresses the above-mentioned design requirements.

Table 2: CDC Requirements and Technology Options

Requirements	CDC Technology Options
Connection to cloud with Layer 2	VPLS, MPLS, IP
Mapping of customer Layer 2 and Layer 3 addresses to cloud infrastructure	Layer 2: BD, vSwitch, VLAN translations
High degree of (network) investment leverage (active/active, Layer 2-Layer 3-MP)	MC-LAG active/active, ECMP
Fast recovery from (network) failures	MC-LAG active/active, MC-LAG active/standby
Easy / flat network design	vSwitch, BD
High degree of scaling and virtualization	24K VLANs, Many >4K VPN and VRF instances
Separation of multi-tenant traffic	vSwitch, VLAN
Building high-performance data center	High-density platforms, high-density line cards, 1GE/10GE/40GE/100GE interfaces
VM mobility	VM mobility in the DC and between DC sites, VMware VM, Azure VM

Example: Configuring a Simple Layer 2 Cloud Data Center Customer Deployment on a Juniper Networks MX Series Device

This example details the steps required on all elements in the end-to-end configuration of the simple Layer 2 customer deployment, including the VPLS configuration, core/access routers, edge routers, and class of service (CoS).

Table 3 on page 17 lists the various network nodes/devices, their roles in the network, and their configuration features.

Table 3: Nodes/Devices and Features

Network Node/Device Roles	Configuration Features
Remote PE Router	Interfaces: GE, XE Protocols: OSPF, OSPF3, IS-IS, BGP, RSVP, MPLS, BFD, VLAN Services: VPLS, BD High Availability: GRES, NSB, NSR
MPLS Provider (P) Routers	Interfaces: GE, XE Protocols: OSPF, OSPF3, IS-IS, BFD
Data Center Edge Routers	Interfaces: GE, XE, AE, MC-AE Protocols: OSPF, OSPF3, IS-IS, BFD, MPLS, RSVP, BGP, VLAN, ICCP, Layer 2-LEARNING, LACP Services: VPLS, VRF, BD, MC-LAG active/standby and active/active High Availability: GRES, NSB, NSR
Data Center Core Routers	Interfaces: GE, XE, AE, MC-AE Protocols: VLAN, ICCP, Layer 2-LEARNING, LACP Services: VIRTUAL-SWITCH, MC-LAG active/active High Availability: GRES, NSR, NSB
Top of Racks (ToRs)	Interfaces: GE, XE, AE Protocols: VLAN, LACP Services: BD (VLANs in EX Terminology)

This example is based on the following topologies. Figure 13 on page 18 shows the logical topology showing both Layer 2 and Layer 3 customers.

Figure 13: Detailed Logical View of Test Lab Topology

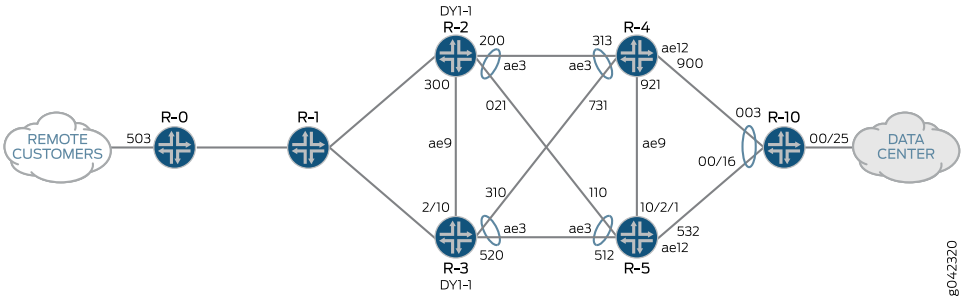
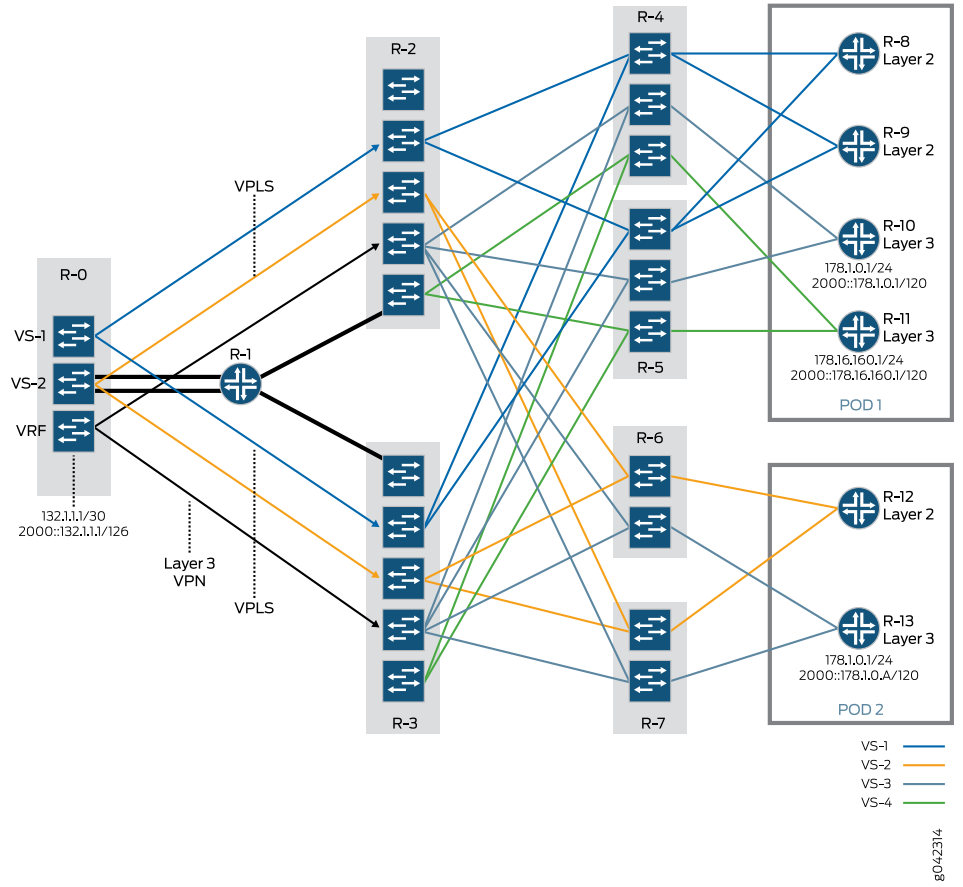


Figure 14 on page 18 shows the physical topology for the complete lab set up, including all edge and access switches for both Layer 2 and Layer 3 customers.

Figure 14: Detailed Physical View of Test Lab Topology



Device R-0 is the remote PE device where tenants can initiate a Layer 2 or Layer 3 connection to the data center. Device R-1 is the provider (P) router in the MPLS network. Devices R-2 and R-3 are the provider edge (PE) routers at the data center edge. Devices R-2 and R-3 form a node redundancy (to alleviate issues should one of the routers go down). In addition, both devices R-2 and R-3 each also contain two routing engines to provide routing engine redundancy.

Two PODs are designed into the network. Devices R-4 and R-5 form the node redundancy for POD 1 that contains devices R-8, R-9, R-10, and R-11 (all top of rack switches). In this deployment scenario, the Layer 2 network is designed from end-to-end (that is, WAN-to-LAN and LAN-to-WAN).

The following sections explain the Layer 2 configuration in more detail:

Requirements

Table 6 on page 76 lists the hardware used on each node/device in this example.

Table 4: Node/Device Hardware

Node/Device	Hardware
Remote Provider Edge Router (R-0)	Chassis: MX480 RE0: RE-S-1800x4 RE1: RE-S-1800x4 FPC0: MPCE Type 2 3D EQ FPC5: MPC 3D 16x 10GE
MPLS Provider Router (R-1)	Chassis: MX480 RE0: RE-S-2000 RE1: NONE FPC0: MPC 3D 16x 10GE FPC4: MPC 3D 16x 10GE FPC5: MPC Type 2 3D EQ

Table 4: Node/Device Hardware *(continued)*

Node/Device	Hardware
Data Center Edge Router (R-2)	Chassis: MX480 RE0: RE-S-1800x4 RE1: RE-S-1800x4 FPC0: MPC Type 2 3D EQ FPC1: MPC 3D 16x 10GE FPC2: MPC Type 2 3D EQ FPC3: MPC 3D 16x 10GE
Data Center Edge Router (R-3)	Chassis: MX960 RE0: RE-S-1800x4 RE1: RE-S-1800x4 FPC0: MPC 3D 16x 10GE FPC1: MPC Type 2 3D EQ FPC2: MPC 3D 16x 10GE FPC3: MPC 3D 16x 10GE FPC4: MPC 3D 16x 10GE FPC5: MPC Type 2 3D EQ
Data Center Core Router (R-4)	Chassis: MX960 RE0: RE-S-1800x4 RE1: RE-S-1800x4 FPC0: MPC 3D 16x 10GE FPC2: MPC 3D 16x 10GE FPC3: MPC 3D 16x 10GE FPC4: MPC Type 2 3D EQ FPC5: MPC Type 2 3D EQ FPC7: MPC Type 2 3D EQ FPC8: MPC 3D 16x 10GE FPC9: MPC Type 2 3D EQ

Table 4: Node/Device Hardware *(continued)*

Node/Device	Hardware
Data Center Core Router (R-5)	Chassis: MX960 RE0: RE-S-1800x4 RE1: RE-S-1800x4 FPC1: MPC Type 2 3D EQ FPC2: MPC 3D 16x 10GE FPC4: MPC Type 2 3D EQ FPC5: MPC Type 2 3D EQ FPC7: MPC Type 2 3D EQ FPC9: MPC Type 2 3D EQ FPC10: MPC Type 2 3D EQ FPC11: MPC 3D 16x 10GE
Data Center Core Router (R-6)	Chassis: MX480 RE0: RE-S-2000 RE1: NONE FPC0: DPCE 20x 1GE R EQ FPC3: MPC Type 2 3D EQ
Data Center Core Router (R-7)	Chassis: MX240 RE0: RE-S-2000 RE1: RE-S-2000 FPC1: DPCE 20x 1GE R EQ
Top-of-Racks (TORs) (R-8 through R-13)	Chassis: EX4500-40F RE0: EX4500-40F RE1: NONE FPC0: EX4500-40F

All MX Series devices in this example use Juniper Networks Junos® OS Release 12.3R4. Table 5 on page 22 lists the scaling values used in configuring each device.

Table 5: Node / Device Scaling Targets

Node/Device	Targeted Feature Scale Values
Remote Provider Edge Router	Interfaces: ~25K IFL Protocols: OSPF - 8, OSPF3 - 8, IS-IS -8, BGP - 2, RSVP -4 Sessions, MPLS LSP - 2 Ingress LSPs + 2 Egress LSPs, BFD -22, VLAN -(1-4094) X 8 Services: VPLS - 4002, VRF - 4K, BD - 8012
MPLS Provider Router	Interfaces: 42 IFL Protocols: OSPF - 24, OSPF3 - 24, IS-IS - 24 , BFD - 48, RSVP LSP -4 Transit LSP
Data Center Edge Router	Interfaces: ~48630 IFL (8K IRB), AE - 8, MC-AE - 8 Protocols: OSPF - 8, OSPF3 - 8, IS-IS - 8, BFD – 23 sessions, MPLS -3 Ingress LSPs + 3 Egress LSPs, RSVP -3 Sessions, BGP - 3, VLAN -(1-4094) X 8, ICCP -1 Session Services: VPLS - 4002, VRF -4K, BD - 20200, MC-LAG active/standby - 3
Data Center Core Router	Interfaces: ~75 , AE - 8, MC-AE active/active - 8 Protocols: VLAN - (1-4094) X 8, ICCP - 1 Services: VIRTUAL-SWITCH -4
Top-of-Racks (ToRs)	Interfaces: ~10 IFL Protocols: VLAN -(1-4094) Services: BD (VLANs in EX) - (1-4094)

Before you configure the Layer 2 cloud data center customer:

- Make sure to configure a loopback interface (lo0) on each routing device.

Overview

In this deployment scenario, each Layer 2 tenant has two VLANs assigned from a single point of delivery (POD). This configuration provides the procedures necessary to configure routing-instance virtual switches and bridge domains that reside within each virtual switch. This example also enables the VPLS protocol in the virtual switch to enable the configured VPLS instance.

This configuration assumes that the router baseline configuration already exists and that the IGP and BGP protocols are up and running.

NOTE: No IGP or BGP configuration detail is provided because no specific functionality is used in this example except for a standard MPLS and RSVP IGP implementation.

Topology

Figure 15 on page 23 shows the topology for simple Layer 2 tenant deployment as it applies to this configuration example.

Figure 15: Lab Test Topology for a Layer 2 Only Customer

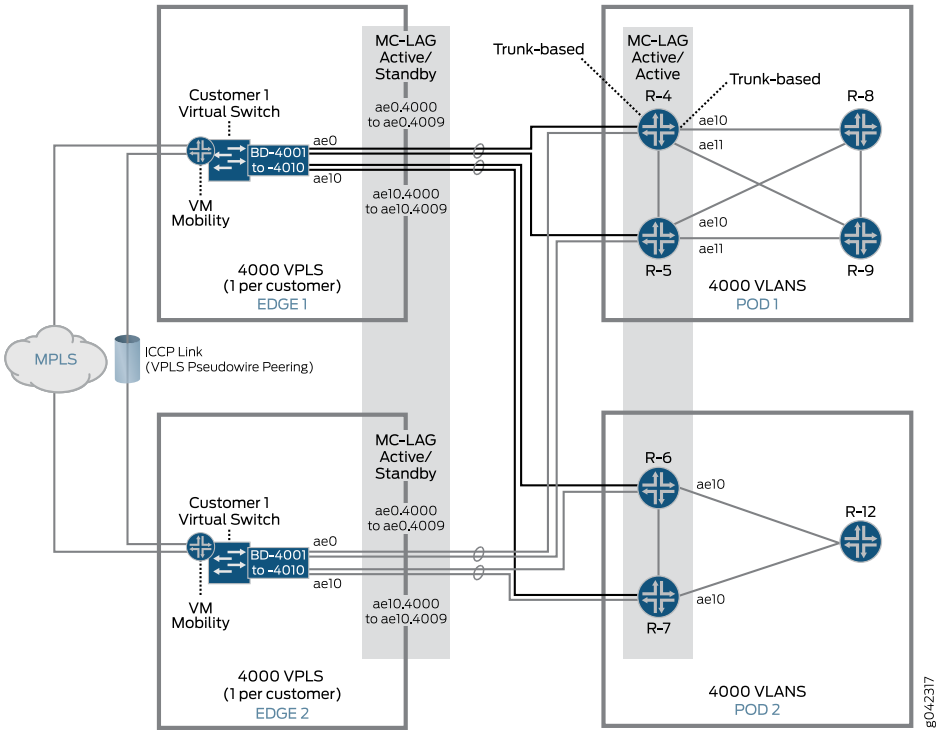
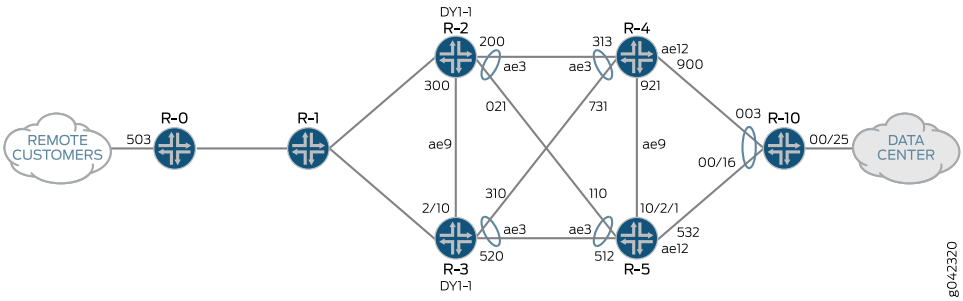


Figure 16 on page 23 shows the logical topology for a Layer 2 customer configured in this example.

Figure 16: Logical Test Topology for a Layer 2 Customer



Configuring Routing Engine Redundancy

Step-by-Step Procedure

Each router in the network must have chassis redundancy enabled for graceful switchover, enhanced IP to use chassis enhanced mode capabilities, and network optimization enabled to increase performance and system stability.

To enable chassis redundancy:

1. Access each router CLI.
2. Configure the master Routing Engine to switch over gracefully to a backup Routing Engine without interruption to packet forwarding.

   ```
   [edit]
   user@host# set chassis redundancy graceful-switchover
   ```

3. Set the router network services to enhanced Internet Protocol and use enhanced mode capabilities.

   NOTE: Only Trio MPCs and MS-DPCs are powered on in the chassis. Non-service DPCs do not work with enhanced network services mode options.

   ```
   [edit]
   user@host# set chassis network-services enhanced-ip
   ```

4. Enable network optimization.

   ```
   [edit]
   user@host# set chassis network-optimization enable-nexthop-optimization
   ```

Configuring Device Interfaces

The following sections define how to configure the router interfaces in this example:

Configuring Device R-0 Interfaces

Step-by-Step Procedure

To configure interfaces for Device R-0:

1. Access the CLI for Device R-0.
2. Configure interface xe-0/3/1 on Device R-0.

NOTE: Interface xe-0/3/1 is the customer edge (CE) interface on Device R-0.

```
[edit]
user@host# set interfaces xe-0/3/1 flexible-vlan-tagging
user@host# set interfaces xe-0/3/1 encapsulation flexible-ethernet-services
user@host# set interfaces xe-0/3/1 unit 0 encapsulation vlan-bridge
user@host# set interfaces xe-0/3/1 unit 0 vlan-id 1
user@host# set interfaces xe-0/3/1 unit 1 encapsulation vlan-bridge
user@host# set interfaces xe-0/3/1 unit 1 vlan-id 2
```

Configuring Device R-2 Interfaces

Step-by-Step Procedure

To configure interfaces for each device in the network:

1. Access the CLI for Device R-2.
2. Configure interface ae0.

NOTE: Interface ae0 is the interface on Device R-2 that faces Device R-4. Only VLANs 1 and 2 on interface ae0 are shown for this customer.

```
[edit]
user@host# set interfaces ae0 flexible-vlan-tagging
user@host# set interfaces ae0 encapsulation flexible-ethernet-services
user@host# set interfaces ae0 unit 0 encapsulation vlan-bridge
user@host# set interfaces ae0 unit 0 vlan-id 1
user@host# set interfaces ae0 unit 1 encapsulation vlan-bridge
user@host# set interfaces ae0 unit 1 vlan-id 2
```

3. Configure ae0 member links.

NOTE: The interface ae0 member links on Device R-2 are xe-3/1/3 and xe-3/2/2.

```
[edit]
user@host# set interfaces xe-3/1/3 gigether-options 802.3ad ae0
user@host# set interfaces xe-3/2/2 gigether-options 802.3ad ae0
```

4. Configure the ae9 member link.

NOTE: This is the Layer 3 ICCP link for MC-LAG.

```
[edit]
user@host# set interfaces xe-3/0/0 gigether-options 802.3ad ae9
```

Configuring Device R-3 Interfaces

Step-by-Step Procedure

To configure interfaces for each device in the network:

1. Access the CLI for Device R-3.
2. Configure interface ae0.

NOTE: Interface ae0 is the interface on Device R-3 that will connect to devices R-4 and R-5. Only VLANs 1 and 2 on interface ae0 are shown for this customer.

```
[edit]
user@host# set interfaces ae0 flexible-vlan-tagging
user@host# set interfaces ae0 encapsulation flexible-ethernet-services
user@host# set interfaces ae0 unit 0 encapsulation vlan-bridge
user@host# set interfaces ae0 unit 0 vlan-id 1
user@host# set interfaces ae0 unit 1 encapsulation vlan-bridge
user@host# set interfaces ae0 unit 1 vlan-id 2
```

3. Configure interface ae0 member links.

NOTE: The ae0 member links on R-3 are xe-0/2/2 and xe-4/0/1.

```
[edit]
user@host# set interfaces xe-0/2/2 gigether-options 802.3ad ae0
user@host# set interfaces xe-4/0/1 gigether-options 802.3ad ae0
```

4. Configure ae9 member links.

NOTE: The AE9 member links on R-3 are xe-0/2/2 and xe-3/0/0. These are the Layer 3 ICCP links for MC-LAG.

```
[edit]
user@host# set interfaces xe-0/2/2 gigether-options 802.3ad ae9
user@host# set interfaces xe-3/0/0 gigether-options 802.3ad ae9
```

Configuring Device R-4 Interfaces

Step-by-Step Procedure

To configure interfaces for each device in the network:

1. Access the CLI for Device R-4.
2. Configure interface xe-9/2/0.

NOTE: Interface xe-9/2/0 is connected to Device R-5 and is the inter-chassis link used for server-to-server communication within the POD and by MC-LAG active/active.

```
[edit]
user@host# set interfaces xe-9/2/0 flexible-vlan-tagging
user@host# set interfaces xe-9/2/0 encapsulation flexible-ethernet-services
user@host# set interfaces xe-9/2/0 unit 0 family bridge interface-mode trunk
user@host# set interfaces xe-9/2/0 unit 0 family bridge vlan-id-list 1-4094
```

3. Configure interface ae0.

NOTE: Interface ae0 is connected to Devices R-2 and R-3.

```
[edit]
user@host# set interfaces ae0 flexible-vlan-tagging
user@host# set interfaces ae0 unit 0 family bridge interface-mode trunk
user@host# set interfaces ae0 unit 0 family bridge vlan-id-list 1-4094
```

4. Configure ae0 member links.

```
[edit]
user@host# set interfaces xe-0/0/0 gigether-options 802.3ad ae0
user@host# set interfaces xe-4/3/0 gigether-options 802.3ad ae0
```

5. Configure interface ae10.

NOTE: Interface ae10 is connected to Device R-8.

```
[edit]
user@host# set interfaces ae10 flexible-vlan-tagging
user@host# set interfaces ae10 unit 0 family bridge interface-mode trunk
user@host# set interfaces ae10 unit 0 family bridge vlan-id-list 1-4094
```

6. Configure ae10 member links.

```
[edit]
user@host# set interfaces xe-8/0/2 gigether-options 802.3ad ae10
```

7. Configure interface ae11.

NOTE: Interface ae11 is connected to Device R-9.

```
[edit]
user@host# set interfaces ae11 flexible-vlan-tagging
user@host# set interfaces ae11 unit 0 family bridge interface-mode trunk
user@host# set interfaces ae11 unit 0 family bridge vlan-id-list 1-4094
```

8. Configure ae11 member links.

```
[edit]
user@host# set interfaces xe-2/2/2 gigether-options 802.3ad ae11
```

Configuring Device R-5 Interfaces

Step-by-Step Procedure

To configure interfaces for each device in the network:

1. Access the CLI for Device R-5.
2. Configure interface xe-10/2/0.

NOTE: Interface xe-10/2/0 is connected to Device R-4 and is the inter-chassis link used for server-to-server communication within the POD and by MC-LAG active/active.

```
[edit]
user@host# set interfaces xe-10/2/0 flexible-vlan-tagging
user@host# set interfaces xe-10/2/0 hold-time up 100
user@host# set interfaces xe-10/2/0 hold-time down 10000
user@host# set interfaces xe-10/2/0 encapsulation flexible-ethernet-services
user@host# set interfaces xe-10/2/0 unit 0 family bridge interface-mode trunk
user@host# set interfaces xe-10/2/0 unit 0 family bridge vlan-id-list 1-4094
```

3. Configure interface ae0.

NOTE: Interface ae0 is connected to Devices R-2 and R-3.

```
[edit]
user@host# set interfaces ae0 flexible-vlan-tagging
user@host# set interfaces ae0 unit 0 family bridge interface-mode trunk
user@host# set interfaces ae0 unit 0 family bridge vlan-id-list 1-4094
```

4. Configure ae0 member links.

```
[edit]
user@host# set interfaces xe-5/2/2 gigether-options 802.3ad ae0
user@host# set interfaces xe-7/0/0 gigether-options 802.3ad ae0
```

5. Configure interface ae10.

NOTE: Interface ae10 is connected to Device R-8.

```
[edit]
user@host# set interfaces ae10 flexible-vlan-tagging
user@host# set interfaces ae10 unit 0 family bridge interface-mode trunk
user@host# set interfaces ae10 unit 0 family bridge vlan-id-list 1-4094
```

6. Configure ae10 member links.

```
[edit]
user@host# set interfaces xe-9/0/1 gigether-options 802.3ad ae10
```

7. Configure interface ae11.

NOTE: Interface ae11 is connected to Device R-9.

```
[edit]
user@host# set interfaces ae11 flexible-vlan-tagging
user@host# set interfaces ae11 unit 0 family bridge interface-mode trunk
user@host# set interfaces ae11 unit 0 family bridge vlan-id-list 1-4094
```

8. Configure ae11 member links.

```
[edit]
user@host# set interfaces xe-2/1/0 gigether-options 802.3ad ae11
```

Configuring Device R-8 Interfaces

Step-by-Step Procedure

To configure interfaces for each device in the network:

1. Access the CLI for Device R-8.
2. Configure interface ge-0/0/25.

NOTE: Interface ge-0/0/25 is the interface configured on Device R-8 in POD 1. This interface provides access to server resources. This is configured in trunk mode and carries 4,094 VLANs

```
[edit]
user@host# set interfaces ge-0/0/25 description "R8 --> RT0"
user@host# set interfaces ge-0/0/25 unit 0 family ethernet-switching port-mode
  trunk
user@host# set interfaces ge-0/0/25 unit 0 family ethernet-switching vlan members
  all
```

3. Configure interface ae0.

```
[edit]
user@host# set interfaces ae0 aggregated-ether-options lacp active
user@host# set interfaces ae0 unit 0 family ethernet-switching port-mode trunk
user@host# set interfaces ae0 unit 0 family ethernet-switching vlan members all
```

4. Configure ae0 member links.

```
[edit]
user@host# set interfaces xe-0/0/6 ether-options 802.3ad ae0
user@host# set interfaces xe-0/0/17 ether-options 802.3ad ae0
```

5. Configure VLANs.

NOTE: Only two of the 4,094 configured VLANs are shown in detail here.

```
user@host# set vlans default vlan-id 4094
user@host# set vlans vlan-1 vlan-id 1
user@host# set vlans vlan-2 vlan-id 2
:
:
user@host# set vlans vlan-4093 vlan-id 4093
```

Configuring Device R-9 Interfaces

Step-by-Step Procedure

To configure interfaces for each device in the network:

1. Access the CLI for Device R-9.
2. Configure interface ge-0/0/5.

NOTE: Interfaces are configured on Device R-9 in POD 1 (top-of-rack).

```
[edit]
user@host# set interfaces ge-0/0/5 description "R8 --> RT0"
user@host# set interfaces ge-0/0/5 unit 0 family ethernet-switching port-mode
  trunk
user@host# set interfaces ge-0/0/5 unit 0 family ethernet-switching vlan members
  all
```

3. Configure interface ae0.

```
[edit]
user@host# set interfaces ae0 aggregated-ether-options lacp active
user@host# set interfaces ae0 unit 0 family ethernet-switching port-mode trunk
user@host# set interfaces ae0 unit 0 family ethernet-switching vlan members all
```

4. Configure ae0 member links.

```
[edit]
user@host# set interfaces xe-0/0/1 ether-options 802.3ad ae0
user@host# set interfaces xe-0/0/2 ether-options 802.3ad ae0
```

Configuring MC-LAG—Link and Node Redundancy

From the previous discussion if the link from Device R-8 to Device R-4 goes down, then the link from Device R-8 to Device R-5 is used for all forwarding. Similarly, if the link from Device R-8 to Device R-5 goes down, then the link from Device R-8 to Device R-4 is used for all forwarding. If Device R-4 goes down, then the traffic from Device R-8 is diverted to Device R-5. This way, both the link and node protection is achieved usingMC-LAG active/active.

The high availability between the data center core and edge networks is achieved by having MC-LAG active/active on the data center core routers for the aggregated Ethernet bundles that go to the edge and by having MC-LAG active/standby on the edge routers that go to the core side. On Device R-4 and Device R-5, interface ae0 is an MC-LAG active/active interface that is part of virtual switch vs1-1. The member links from interface ae0 on Device R-4 and Device R-5 are connected to Device R-2 and Device R-3.

Link and node redundancy is achieved by using multi-chassis link aggregation in data center core, data center edge, and WAN networks. As an example, coming from downstream to upstream:

- Interface ae0 on Device R-8 has two member links xe-0/0/6 and xe-0/0/17. Interface xe-0/0/6 goes to Device R-4 interface ae10, and xe-0/0/17 goes to Device R-5 interface ae10 in the data center core network.

  Similarly, interface ae0 on Device R-9 has two member links: xe-0/0/1 and xe-0/0/2. Interface xe-0/0/1 goes to Device R-4 interface ae11, and xe-0/0/2 goes to Device R-5 interface ae11 in the data center core network.

- Interface ae10 on Device R-4 and Device R-5 is configured with MC-LAG active/active. On each router, aggregated Ethernet bundles with two member links are configured and on interface ae10 on Device R-4 and Device R-5, there is a total of four member links.

- Device R-4 and Device R-5 act as active/active for MC-LAG and Layer 2 traffic is load-balanced across the aggregated Ethernet bundle member links that traverses the MC-LAG interfaces.

- Interface xe-9/2/0 on R-4 connects to interface xe-10/2/0 on Device R-5. Interface xe-10/2/0 acts as the inter-chassis link (ICL) between Device R-4 and Device R-5 , as well as part of the bridge domains configured in virtual switch vs1-1. Interface ae-9 is configured to run Inter-Chassis Control Protocol (ICCP) for MC-LAG.

The following sections describe how to configure MC-LAG on each device:

- Configuring MC-LAG on Device R-2 on page 31
- Configuring MC-LAG on Device R-3 on page 32
- Configuring MC-LAG on Device R-4 on page 33
- Configuring MC-LAG on Device R-5 on page 35

Configuring MC-LAG on Device R-2

Step-by-Step Procedure

To configure MC-LAG on Device R-2:

1. Configure an ICCP interface.

   ```
   [edit]
   user@host# set interfaces ae9 unit 0 family inet address 4.0.0.1/30
   user@host# set interfaces ae9 unit 0 family iso
   user@host# set interfaces ae9 unit 0 family inet6 address 2002::4.0.0.1/126
   user@host# set interfaces ae9 unit 0 family mpls
   ```

2. Configure the ICCP protocol.

```
[edit]
user@host# set protocols iccp local-ip-addr 4.0.0.1
user@host# set protocols iccp peer 4.0.0.2 redundancy-group-id-list 1
user@host# set protocols iccp peer 4.0.0.2 liveness-detection minimum-interval
  1000
user@host# set protocols iccp peer 4.0.0.2 liveness-detection detection-time
  threshold 2000000
user@host# set protocols iccp peer 4.0.0.2 liveness-detection single-hop
```

3. Configure the Layer 2 learning protocol.

```
[edit]
user@host# set protocols l2-learning global-mac-limit 1048575
```

4. Configure the Layer 2 control protocol.

```
[edit]
user@host# set protocols layer2-control nonstop-bridging
```

NOTE: During validation testing *PR 956847* was opened. Upon failover, device experienced high CPU utilization by PFE processes and packet loss, when nonstop-bridging and nonstop-routing were enabled.

5. Configure the MC-LAG aggregated Ethernet active/standby interface.

```
[edit]
user@host# set interfaces ae0 aggregated-ether-options lacp active
user@host# set interfaces ae0 aggregated-ether-options lacp periodic fast
user@host# set interfaces ae0 aggregated-ether-options lacp system-priority 100
user@host# set interfaces ae0 aggregated-ether-options lacp system-id
  00:00:00:00:00:20
user@host# set interfaces ae0 aggregated-ether-options lacp admin-key 1
user@host# set interfaces ae0 aggregated-ether-options mc-ae mc-ae-id 10
user@host# set interfaces ae0 aggregated-ether-options mc-ae redundancy-group
  1
user@host# set interfaces ae0 aggregated-ether-options mc-ae chassis-id 0
user@host# set interfaces ae0 aggregated-ether-options mc-ae mode
  active-standby
user@host# set interfaces ae0 aggregated-ether-options mc-ae status-control
  active
user@host# set interfaces ae0 aggregated-ether-options mc-ae events
  iccp-peer-down prefer-status-control-active
user@host# set interfaces ae0 unit 0 encapsulation vlan-bridge
user@host# set interfaces ae0 unit 0 vlan-id 1
user@host# set interfaces ae0 unit 1 encapsulation vlan-bridge
user@host# set interfaces ae0 unit 1 vlan-id 2
```

Configuring MC-LAG on Device R-3

Step-by-Step Procedure

To configure Device R-3:

1. Configure an ICCP interface.

```
[edit]
```

```
user@host# set interfaces ae9 unit 0 family inet address 4.0.0.2/30
user@host# set interfaces ae9 unit 0 family iso
user@host# set interfaces ae9 unit 0 family inet6 address 2002::4.0.0.2/126
user@host# set interfaces ae9 unit 0 family mpls
```

2. Configure the ICCP protocol attributes.

```
[edit]
user@host# set protocols iccp local-ip-addr 4.0.0.2
user@host# set protocols iccp peer 4.0.0.1 redundancy-group-id-list 1
user@host# set protocols iccp peer 4.0.0.1 liveness-detection minimum-interval
  1000
user@host# set protocols iccp peer 4.0.0.1 liveness-detection detection-time
  threshold 2000000
set protocols iccp peer 4.0.0.1 liveness-detection single-hop
```

3. Configure the Layer 2 learning protocol.

```
[edit]
user@host# set protocols l2-learning global-mac-limit 1048575
```

4. Configure the Layer 2 control protocol.

```
[edit]
user@host# set protocols layer2-control nonstop-bridging
```

5. Configure the MC-LAG aggregated Ethernet active/standby interface.

```
[edit]
user@host# set interfaces ae0 aggregated-ether-options lacp active
user@host# set interfaces ae0 aggregated-ether-options lacp periodic fast
user@host# set interfaces ae0 aggregated-ether-options lacp system-priority 100
user@host# set interfaces ae0 aggregated-ether-options lacp system-id
  00:00:00:00:00:20
user@host# set interfaces ae0 aggregated-ether-options lacp admin-key 1
user@host# set interfaces ae0 aggregated-ether-options mc-ae mc-ae-id 10
user@host# set interfaces ae0 aggregated-ether-options mc-ae redundancy-group
  1
user@host# set interfaces ae0 aggregated-ether-options mc-ae chassis-id 1
user@host# set interfaces ae0 aggregated-ether-options mc-ae mode
  active-standby
user@host# set interfaces ae0 aggregated-ether-options mc-ae status-control
  standby
user@host# set interfaces ae0 aggregated-ether-options mc-ae events
  iccp-peer-down prefer-status-control-active
user@host# set interfaces ae0 unit 0 encapsulation vlan-bridge
user@host# set interfaces ae0 unit 0 vlan-id 1
user@host# set interfaces ae0 unit 1 encapsulation vlan-bridge
user@host# set interfaces ae0 unit 1 vlan-id 2
```

Configuring MC-LAG on Device R-4

Step-by-Step Procedure

To configure Device R-4:

1. Configure an ICCP interface.

```
[edit]
user@host# set interfaces ae9 unit 0 family inet address 4.1.0.1/30
user@host# set interfaces ae9 unit 0 family iso
```

```
user@host# set interfaces ae9 unit 0 family inet6 address 2002::4.1.0.1/126
user@host# set interfaces ae9 unit 0 family mpls
```

2. Configure the ICCP protocol attributes.

```
[edit]
user@host# set protocols iccp local-ip-addr 4.1.0.1
user@host# set protocols iccp peer 4.1.0.2 redundancy-group-id-list 1
user@host# set protocols iccp peer 4.1.0.2 liveness-detection minimum-interval
  1000
user@host# set protocols iccp peer 4.1.0.2 liveness-detection detection-time
  threshold 2000000
user@host# set protocols iccp peer 4.1.0.2 liveness-detection single-hop
```

3. Configure interface ae0.

NOTE: This interface faces data center edge router Device R-2 and router Device R-3.

```
[edit]
user@host# set interfaces ae0 multi-chassis-protection 4.1.0.2 interface xe-9/2/0
user@host# set interfaces ae0 aggregated-ether-options lacp active
user@host# set interfaces ae0 aggregated-ether-options lacp periodic fast
user@host# set interfaces ae0 aggregated-ether-options lacp system-priority 100
user@host# set interfaces ae0 aggregated-ether-options lacp system-id
  00:00:00:00:00:60
user@host# set interfaces ae0 aggregated-ether-options lacp admin-key 1
user@host# set interfaces ae0 aggregated-ether-options mc-ae mc-ae-id 10
user@host# set interfaces ae0 aggregated-ether-options mc-ae redundancy-group
  1
user@host# set interfaces ae0 aggregated-ether-options mc-ae chassis-id 0
user@host# set interfaces ae0 aggregated-ether-options mc-ae mode active-active
user@host# set interfaces ae0 aggregated-ether-options mc-ae status-control
  active
user@host# set interfaces ae0 aggregated-ether-options mc-ae events
  iccp-peer-down prefer-status-control-active
```

4. Configure MC-LAG on interface ae10.

NOTE: This interface connects to Device R-8.

```
[edit]
user@host# set interfaces ae10 multi-chassis-protection 4.1.0.2 interface xe-9/2/0
user@host# set interfaces ae10 aggregated-ether-options lacp active
user@host# set interfaces ae10 aggregated-ether-options lacp periodic fast
user@host# set interfaces ae10 aggregated-ether-options lacp system-priority 100
user@host# set interfaces ae10 aggregated-ether-options lacp system-id
  00:00:00:00:00:40
user@host# set interfaces ae10 aggregated-ether-options lacp admin-key 1
user@host# set interfaces ae10 aggregated-ether-options mc-ae mc-ae-id 1
user@host# set interfaces ae10 aggregated-ether-options mc-ae redundancy-group
  1
```

```
user@host# set interfaces ae10 aggregated-ether-options mc-ae chassis-id 0
user@host# set interfaces ae10 aggregated-ether-options mc-ae mode active-active
user@host# set interfaces ae10 aggregated-ether-options mc-ae status-control
  active
user@host# set interfaces ae10 aggregated-ether-options mc-ae events
  iccp-peer-down prefer-status-control-active
```

5. Configure MC-LAG on interface ae11.

NOTE: This interface connects to Device R-9.

```
[edit]
user@host# set interfaces ae11 flexible-vlan-tagging
user@host# set interfaces ae11 multi-chassis-protection 4.1.0.2 interface xe-9/2/0
user@host# set interfaces ae11 encapsulation flexible-ethernet-services
user@host# set interfaces ae11 aggregated-ether-options lacp active
user@host# set interfaces ae11 aggregated-ether-options lacp periodic fast
user@host# set interfaces ae11 aggregated-ether-options lacp system-priority 100
user@host# set interfaces ae11 aggregated-ether-options lacp system-id
  00:00:00:00:00:41
user@host# set interfaces ae11 aggregated-ether-options lacp admin-key 1
user@host# set interfaces ae11 aggregated-ether-options mc-ae mc-ae-id 2
user@host# set interfaces ae11 aggregated-ether-options mc-ae redundancy-group
  1
user@host# set interfaces ae11 aggregated-ether-options mc-ae chassis-id 0
user@host# set interfaces ae11 aggregated-ether-options mc-ae mode active-active
user@host# set interfaces ae11 aggregated-ether-options mc-ae status-control
  active
user@host# set interfaces ae11 aggregated-ether-options mc-ae events
  iccp-peer-down prefer-status-control-active
user@host# set interfaces ae11 unit 0 family bridge interface-mode trunk
user@host# set interfaces ae11 unit 0 family bridge vlan-id-list 1-4094
```

Configuring MC-LAG on Device R-5

Step-by-Step Procedure

To configure Device R-5:

1. Configure interface ae9 services, including ICCP.

```
[edit]
user@host# set interfaces ae9 unit 0 family inet address 4.1.0.2/30
user@host# set interfaces ae9 unit 0 family iso
user@host# set interfaces ae9 unit 0 family inet6 address 2002::4.1.0.2/126
user@host# set interfaces ae9 unit 0 family mpls
```

2. Configure the ICCP protocol attributes.

```
[edit]
user@host# set protocols iccp local-ip-addr 4.1.0.2
user@host# set protocols iccp peer 4.1.0.1 redundancy-group-id-list 1
user@host# set protocols iccp peer 4.1.0.1 liveness-detection minimum-interval
  1000
user@host# set protocols iccp peer 4.1.0.1 liveness-detection detection-time
  threshold 2000000
```

```
user@host# set protocols iccp peer 4.1.0.1 liveness-detection single-hop
```

NOTE: BFD session timer values are aimed to ensure that the ICCP adjacency between routers is not lost during RE switchover on one of the routers.

3. Configure interface ae0.

NOTE: This interface faces data center edge Device R-2 and Device R-3.

```
[edit]
user@host# set interfaces ae0 multi-chassis-protection 4.1.0.1 interface xe-10/2/0
user@host# set interfaces ae0 aggregated-ether-options lacp active
user@host# set interfaces ae0 aggregated-ether-options lacp periodic fast
user@host# set interfaces ae0 aggregated-ether-options lacp system-priority 100
user@host# set interfaces ae0 aggregated-ether-options lacp system-id
  00:00:00:00:00:60
user@host# set interfaces ae0 aggregated-ether-options lacp admin-key 1
user@host# set interfaces ae0 aggregated-ether-options mc-ae mc-ae-id 10
user@host# set interfaces ae0 aggregated-ether-options mc-ae redundancy-group
  1
user@host# set interfaces ae0 aggregated-ether-options mc-ae chassis-id 1
user@host# set interfaces ae0 aggregated-ether-options mc-ae mode active-active
user@host# set interfaces ae0 aggregated-ether-options mc-ae status-control
  standby
user@host# set interfaces ae0 aggregated-ether-options mc-ae events
  iccp-peer-down prefer-status-control-active
user@host# set interfaces ae0 aggregated-ether-options mc-ae events
```

4. Configure MC-LAG active/active on interface ae10.

NOTE: This interface connects to Device R-8.

```
[edit]
user@host# set interfaces ae10 multi-chassis-protection 4.1.0.1 interface xe-10/2/0
user@host# set interfaces ae10 aggregated-ether-options lacp active
user@host# set interfaces ae10 aggregated-ether-options lacp periodic fast
user@host# set interfaces ae10 aggregated-ether-options lacp system-priority 100
user@host# set interfaces ae10 aggregated-ether-options lacp system-id
  00:00:00:00:00:40
user@host# set interfaces ae10 aggregated-ether-options lacp admin-key 1
user@host# set interfaces ae10 aggregated-ether-options mc-ae mc-ae-id 1
user@host# set interfaces ae10 aggregated-ether-options mc-ae redundancy-group
  1
user@host# set interfaces ae10 aggregated-ether-options mc-ae chassis-id 1
user@host# set interfaces ae10 aggregated-ether-options mc-ae mode active-active
user@host# set interfaces ae10 aggregated-ether-options mc-ae status-control
  standby
```

```
user@host# set interfaces ae10 aggregated-ether-options mc-ae events
  iccp-peer-down prefer-status-control-active
user@host# set interfaces ae10 aggregated-ether-options mc-ae events
```

NOTE: The prefer-status-control-active statement can be configured on the PE device along with the status-control standby statement, for the purpose of avoiding the LACP mc-ae system-id reverting to the LACP default system ID in the event of ICCP failure. We recommend using this configuration only when you can ensure that ICCP never goes down unless the remote PE device is going down.

5. Configure the interface hold time.

```
[edit]
user@host# set interfaces xe-10/2/0 flexible-vlan-tagging
user@host# set interfaces xe-10/2/0 hold-time up 100
user@host# set interfaces xe-10/2/0 hold-time down 10000
```

NOTE: Check the PE device configured using the status-control standby statement, should the PE device configured using the status-control active statement go down abruptly (for example, if it is powered off), we recommend that you configure the hold-time down statement interval for the interface configured as the inter-chassis control link (ICL) that is configured for "status-control standby" with a value greater than the ICCP BFD timeout value. Without this hold-time interval configuration on the ICL, the MC-LAG aggregated Ethernet Interface on the PE device configured with thestatus-control standby statement momentarily moves to standby when the PE device is powered off.

6. Configure MC-LAG for high availability on interface ae11.

NOTE: This interface connects to Device R-9.

```
[edit]
user@host# set interfaces ae11 flexible-vlan-tagging
user@host# set interfaces ae11 multi-chassis-protection 4.1.0.1 interface xe-10/2/0
user@host# set interfaces ae11 encapsulation flexible-ethernet-services
user@host# set interfaces ae11 aggregated-ether-options lacp active
user@host# set interfaces ae11 aggregated-ether-options lacp periodic fast
user@host# set interfaces ae11 aggregated-ether-options lacp system-priority 100
user@host# set interfaces ae11 aggregated-ether-options lacp system-id
  00:00:00:00:00:41
user@host# set interfaces ae11 aggregated-ether-options lacp admin-key 1
user@host# set interfaces ae11 aggregated-ether-options mc-ae mc-ae-id 2
user@host# set interfaces ae11 aggregated-ether-options mc-ae redundancy-group
  1
```

```
user@host# set interfaces ae11 aggregated-ether-options mc-ae chassis-id 1
user@host# set interfaces ae11 aggregated-ether-options mc-ae mode active-active
user@host# set interfaces ae11 aggregated-ether-options mc-ae status-control
  standby
user@host# set interfaces ae11 aggregated-ether-options mc-ae events
  iccp-peer-down prefer-status-control-active deactivate interfaces ae11
  aggregated-ether-options mc-ae events
user@host# set interfaces ae11 unit 0 family bridge interface-mode trunk
user@host# set interfaces ae11 unit 0 family bridge vlan-id-list 1-4094
```

Configuring IGP and BGP Protocols

Step-by-Step Procedure

In all Layer 2 data center deployment scenarios:

- OSPF is configured as the IGP on all core interfaces.
- BFD is configured to optimize convergence times during a core failure.
- MPLS and RSVP are enabled on each core interface.
- BGP is configured for L2VPN signaling with BFD detection.

NOTE: The following is a generic configuration. Interface values are represented by an asterisk (*). Description and IP address octet variables are represented by *x* values.

To configure IGP and BGP protocols:

1. Configure the router interfaces on all devices.

```
[edit]
user@host# set interfaces xe-*/*/* description "xxxx"
user@host# set interfaces xe-*/*/* unit 0 family inet address x.x.x.x/30
user@host# set interfaces xe-*/*/* unit 0 family mpls
```

2. Configure the OSPF protocol.

```
[edit]
user@host# set protocols ospf area 0.0.0.0 interface xe-*/*/*.0
  bfd-liveness-detection minimum-interval 500
user@host# set protocols ospf area 0.0.0.0 interface xe-*/*/*.0
  bfd-liveness-detection multiplier 3
```

3. Configure the RSVP and MPLS protocols.

```
[edit]
user@host# set protocols rsvp interface xe-*/*/*.0
user@host# set protocols mpls interface xe-*/*/*.0
user@host# set protocols mpls interface lo0.0
```

4. Configure BGP.

```
[edit]
user@host# set protocols bgp group vpls-bgp type internal
user@host# set protocols bgp group vpls-bgp local-address x.x.x.x
```

```
user@host# set protocols bgp group vpls-bgp bfd-liveness-detection
  minimum-interval 1000
user@host# set protocols bgp group vpls-bgp bfd-liveness-detection multiplier 3
user@host# set protocols bgp group vpls-bgp neighbor x.x.x.x family l2vpn signaling
```

Configuring VPLS

Step-by-Step Procedure

The virtual switch and VPLS instance contain only two bridge domains.

To configure VPLS:

1. Configure VPLS on Device R-0.

NOTE: The following example shows only customer vs1-1.

```
[edit]
user@host# set routing-instances vs1-1 instance-type virtual-switch
user@host# set routing-instances vs1-1 route-distinguisher 1000:1
user@host# set routing-instances vs1-1 vrf-target target:1000:10001
user@host# set routing-instances vs1-1 protocols vpls no-tunnel-services
user@host# set routing-instances vs1-1 protocols vpls site 1 site-identifier 1
user@host# set routing-instances vs1-1 bridge-domains bd-1 domain-type bridge
user@host# set routing-instances vs1-1 bridge-domains bd-1 vlan-id 1
user@host# set routing-instances vs1-1 bridge-domains bd-1 interface xe-0/3/1.0
user@host# set routing-instances vs1-1 bridge-domains bd-2 domain-type bridge
user@host# set routing-instances vs1-1 bridge-domains bd-2 vlan-id 2
user@host# set routing-instances vs1-1 bridge-domains bd-2 interface xe-0/3/1.1
user@host# set routing-instances vs1-1 switch-options service-id 1
```

2. Configure VPLS on Device R-2.

NOTE: The following example shows only customer vs1-1.

```
[edit]
user@host# set routing-instances vs1-1 instance-type virtual-switch
user@host# set routing-instances vs1-1 route-distinguisher 1000:8001
user@host# set routing-instances vs1-1 vrf-target target:1000:10001
user@host# set routing-instances vs1-1 protocols vpls no-tunnel-services
user@host# set routing-instances vs1-1 protocols vpls site 2 site-identifier 2
user@host# set routing-instances vs1-1 protocols vpls connectivity-type permanent
user@host# set routing-instances vs1-1 bridge-domains bd-1 domain-type bridge
user@host# set routing-instances vs1-1 bridge-domains bd-1 vlan-id 1
user@host# set routing-instances vs1-1 bridge-domains bd-1 interface ae0.0
user@host# set routing-instances vs1-1 bridge-domains bd-2 domain-type bridge
user@host# set routing-instances vs1-1 bridge-domains bd-2 vlan-id 2
user@host# set routing-instances vs1-1 bridge-domains bd-2 interface ae0.1
user@host# set routing-instances vs1-1 switch-options service-id 1
```

3. Configure VPLS on Device R-3.

NOTE: The following example shows only customer vs1-1.

```
[edit]
user@host# set routing-instances vs1-1 instance-type virtual-switch
user@host# set routing-instances vs1-1 interface ge-1/0/0.1
user@host# set routing-instances vs1-1 route-distinguisher 1000:16001
user@host# set routing-instances vs1-1 vrf-target target:1000:10001
user@host# set routing-instances vs1-1 protocols vpls no-tunnel-services
user@host# set routing-instances vs1-1 protocols vpls site 3 site-identifier 3
user@host# set routing-instances vs1-1 protocols vpls connectivity-type permanent
user@host# set routing-instances vs1-1 bridge-domains bd-1 domain-type bridge
user@host# set routing-instances vs1-1 bridge-domains bd-1 vlan-id 1
user@host# set routing-instances vs1-1 bridge-domains bd-1 interface ae0.0
user@host# set routing-instances vs1-1 bridge-domains bd-2 domain-type bridge
user@host# set routing-instances vs1-1 bridge-domains bd-2 vlan-id 2
user@host# set routing-instances vs1-1 bridge-domains bd-2 interface ae0.1
user@host# set routing-instances vs1-1 switch-options service-id 1
```

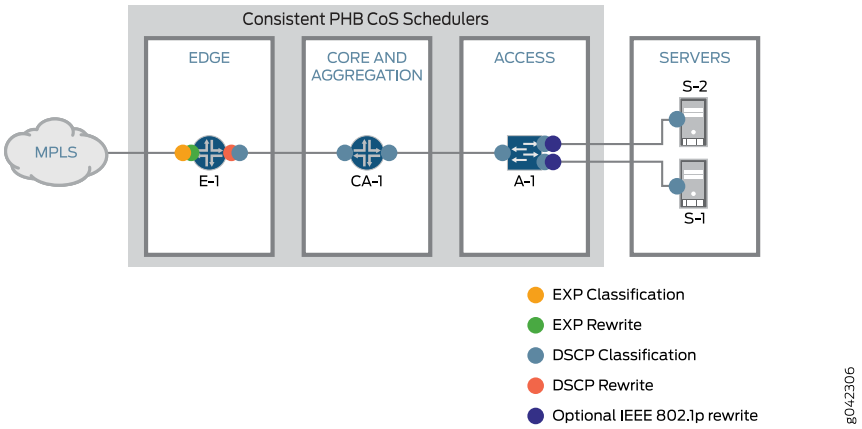
NOTE: The **connectivity-type permanent** statement is configured on PE devices that are configured for MC-LAG active/standby to ensure that VPLS is UP independently of the MC-LAG interface. This helps reduce the network convergence time.

Configuring Customer Class of Service Classifiers

To implement quality of service for data center edge and core routers, you must configure Class of Service (CoS) classifiers, re-write rules, schedulers, forwarding classes, and drop profiles for each device. All classifiers are configured on all devices where the specific classifier is used. However, the attachment of classifiers and re-write rules to interfaces is based on whether the traffic flowing through the circuit or interface are Layer 2 or Layer 3 data packets.

Figure 17 on page 41 shows the different customer Class of Service (CoS) points.

Figure 17: Layer 2 customer CoS points



The following sections detail how to configure customer class of service:

- Configure Class of Service Classifiers on page 41
- Configuring Drop Profiles on page 43
- Configuring Forwarding Classes on page 43
- Configuring Rewrite Rules on page 43
- Configuring Schedulers and Scheduler Maps on page 44

Configure Class of Service Classifiers

Step-by-Step Procedure

To configure class-of-service classifiers:

1. Configure EXP classifiers.

```
[edit]
user@host# set class-of-service classifiers exp DCN-exp-classifier forwarding-class
  voice loss-priority low code-points 101
user@host# set class-of-service classifiers exp DCN-exp-classifier forwarding-class
  interactive-video loss-priority low code-points 100
user@host# set class-of-service classifiers exp DCN-exp-classifier forwarding-class
  business-tier1 loss-priority low code-points 011
user@host# set class-of-service classifiers exp DCN-exp-classifier forwarding-class
  business-tier2 loss-priority low code-points 010
user@host# set class-of-service classifiers exp DCN-exp-classifier forwarding-class
  network-control loss-priority low code-points 110
user@host# set class-of-service classifiers exp DCN-exp-classifier forwarding-class
  network-control loss-priority high code-points 111
user@host# set class-of-service classifiers exp DCN-exp-classifier forwarding-class
  bulk loss-priority low code-points 000
user@host# set class-of-service classifiers exp DCN-exp-classifier forwarding-class
  bulk loss-priority high code-points 001
```

2. Configure dot1p classifiers.

```
[edit]
user@host# set class-of-service classifiers ieee-802.1 DCN-dot1p-classifier
  forwarding-class voice loss-priority low code-points 101
```

```
user@host# set class-of-service classifiers ieee-802.1 DCN-dot1p-classifier
  forwarding-class interactive-video loss-priority low code-points 100
user@host# set class-of-service classifiers ieee-802.1 DCN-dot1p-classifier
  forwarding-class business-tier1 loss-priority low code-points 011
user@host# set class-of-service classifiers ieee-802.1 DCN-dot1p-classifier
  forwarding-class business-tier2 loss-priority low code-points 010
user@host# set class-of-service classifiers ieee-802.1 DCN-dot1p-classifier
  forwarding-class network-control loss-priority low code-points 110
user@host# set class-of-service classifiers ieee-802.1 DCN-dot1p-classifier
  forwarding-class network-control loss-priority high code-points 111
user@host# set class-of-service classifiers ieee-802.1 DCN-dot1p-classifier
  forwarding-class bulk loss-priority low code-points 000
user@host# set class-of-service classifiers ieee-802.1 DCN-dot1p-classifier
  forwarding-class bulk loss-priority high code-points 001
```

3. Configure inet classifiers.

```
[edit]
user@host# set class-of-service classifiers inet-precedence DCN-inet
  forwarding-class voice loss-priority low code-points 101
user@host# set class-of-service classifiers inet-precedence DCN-inet
  forwarding-class interactive-video loss-priority low code-points 100
user@host# set class-of-service classifiers inet-precedence DCN-inet
  forwarding-class business-tier1 loss-priority low code-points 011
user@host# set class-of-service classifiers inet-precedence DCN-inet
  forwarding-class business-tier2 loss-priority low code-points 010
user@host# set class-of-service classifiers inet-precedence DCN-inet
  forwarding-class network-control loss-priority low code-points 110
user@host# set class-of-service classifiers inet-precedence DCN-inet
  forwarding-class network-control loss-priority high code-points 111
user@host# set class-of-service classifiers inet-precedence DCN-inet
  forwarding-class bulk loss-priority low code-points 000
user@host# set class-of-service classifiers inet-precedence DCN-inet
  forwarding-class bulk loss-priority high code-points 001
```

4. Configure TOS classifiers.

```
[edit]
user@host# set class-of-service classifiers inet-precedence DCN-TOS-classifier
  forwarding-class voice loss-priority low code-points 101
user@host# set class-of-service classifiers inet-precedence DCN-TOS-classifier
  forwarding-class interactive-video loss-priority low code-points 100
user@host# set class-of-service classifiers inet-precedence DCN-TOS-classifier
  forwarding-class business-tier1 loss-priority low code-points 011
user@host# set class-of-service classifiers inet-precedence DCN-TOS-classifier
  forwarding-class business-tier2 loss-priority low code-points 010
user@host# set class-of-service classifiers inet-precedence DCN-TOS-classifier
  forwarding-class network-control loss-priority low code-points 110
user@host# set class-of-service classifiers inet-precedence DCN-TOS-classifier
  forwarding-class network-control loss-priority high code-points 111
user@host# set class-of-service classifiers inet-precedence DCN-TOS-classifier
  forwarding-class bulk loss-priority low code-points 000
user@host# set class-of-service classifiers inet-precedence DCN-TOS-classifier
  forwarding-class bulk loss-priority high code-points 001
```

Configuring Drop Profiles

Step-by-Step Procedure

To configure drop profiles:

1. Access the CLI of each customer-facing device.
2. Configure the drop profiles.

```
[edit]
user@host# set class-of-service drop-profiles congest-drop-low fill-level 80
  drop-probability 100
user@host# set class-of-service drop-profiles congest-drop-high fill-level 60
  drop-probability 100
```

Configuring Forwarding Classes

Step-by-Step Procedure

To configure forwarding classes:

1. Access the CLI of each customer-facing device.
2. Configure forwarding classes.

```
[edit]
user@host# set class-of-service forwarding-classes class voice queue-num 2
user@host# set class-of-service forwarding-classes class interactive-video
  queue-num 1
user@host# set class-of-service forwarding-classes class business-tier1 queue-num
  4
user@host# set class-of-service forwarding-classes class business-tier2 queue-num
  5
user@host# set class-of-service forwarding-classes class network-control
  queue-num 3
user@host# set class-of-service forwarding-classes class bulk queue-num 0
```

Configuring Rewrite Rules

Step-by-Step Procedure

To configure rewrite rules:

1. Access the CLI of each customer-facing device.
2. Configure EXP rewrite rules.

```
[edit]
user@host# set class-of-service rewrite-rules exp DCN-exp-rewrite forwarding-class
  voice loss-priority low code-point 101
user@host# set class-of-service rewrite-rules exp DCN-exp-rewrite forwarding-class
  interactive-video loss-priority low code-point 100
user@host# set class-of-service rewrite-rules exp DCN-exp-rewrite forwarding-class
  business-tier1 loss-priority low code-point 011
user@host# set class-of-service rewrite-rules exp DCN-exp-rewrite forwarding-class
  business-tier2 loss-priority low code-point 010
user@host# set class-of-service rewrite-rules exp DCN-exp-rewrite forwarding-class
  network-control loss-priority low code-point 110
user@host# set class-of-service rewrite-rules exp DCN-exp-rewrite forwarding-class
  network-control loss-priority high code-point 111
```

```
user@host# set class-of-service rewrite-rules exp DCN-exp-rewrite forwarding-class
  bulk loss-priority low code-point 000
user@host# set class-of-service rewrite-rules exp DCN-exp-rewrite forwarding-class
  bulk loss-priority high code-point 001
```

3. Configure dot1p rewrite rules.

```
[edit]
user@host# set class-of-service rewrite-rules ieee-802.1 DCN-dot1p-rewrite
  forwarding-class voice loss-priority low code-point 101
user@host# set class-of-service rewrite-rules ieee-802.1 DCN-dot1p-rewrite
  forwarding-class interactive-video loss-priority low code-point 100
user@host# set class-of-service rewrite-rules ieee-802.1 DCN-dot1p-rewrite
  forwarding-class business-tier1 loss-priority low code-point 011
user@host# set class-of-service rewrite-rules ieee-802.1 DCN-dot1p-rewrite
  forwarding-class business-tier2 loss-priority low code-point 010
user@host# set class-of-service rewrite-rules ieee-802.1 DCN-dot1p-rewrite
  forwarding-class network-control loss-priority low code-point 110
user@host# set class-of-service rewrite-rules ieee-802.1 DCN-dot1p-rewrite
  forwarding-class network-control loss-priority high code-point 111
user@host# set class-of-service rewrite-rules ieee-802.1 DCN-dot1p-rewrite
  forwarding-class bulk loss-priority low code-point 000
user@host# set class-of-service rewrite-rules ieee-802.1 DCN-dot1p-rewrite
  forwarding-class bulk loss-priority high code-point 001
```

4. Configure TOS rewrite rules.

```
[edit]
user@host# set class-of-service rewrite-rules inet-precedence DCN-TOS-rewrite
  forwarding-class voice loss-priority low code-point 101
user@host# set class-of-service rewrite-rules inet-precedence DCN-TOS-rewrite
  forwarding-class interactive-video loss-priority low code-point 100
user@host# set class-of-service rewrite-rules inet-precedence DCN-TOS-rewrite
  forwarding-class business-tier1 loss-priority low code-point 011
user@host# set class-of-service rewrite-rules inet-precedence DCN-TOS-rewrite
  forwarding-class business-tier2 loss-priority low code-point 010
user@host# set class-of-service rewrite-rules inet-precedence DCN-TOS-rewrite
  forwarding-class network-control loss-priority low code-point 110
user@host# set class-of-service rewrite-rules inet-precedence DCN-TOS-rewrite
  forwarding-class network-control loss-priority high code-point 111
user@host# set class-of-service rewrite-rules inet-precedence DCN-TOS-rewrite
  forwarding-class bulk loss-priority low code-point 000
user@host# set class-of-service rewrite-rules inet-precedence DCN-TOS-rewrite
  forwarding-class bulk loss-priority high code-point 001
```

Configuring Schedulers and Scheduler Maps

Step-by-Step Procedure

To configure CoS schedulers and scheduler maps:

1. Access the CLI of each customer-facing device.

2. Configure schedulers.

```
[edit]
user@host# set class-of-service schedulers voice transmit-rate percent 25
user@host# set class-of-service schedulers voice transmit-rate exact
user@host# set class-of-service schedulers voice buffer-size temporal 25k
```

```
user@host# set class-of-service schedulers voice priority high
user@host# set class-of-service schedulers interactive-video transmit-rate percent
  25
user@host# set class-of-service schedulers interactive-video transmit-rate exact
user@host# set class-of-service schedulers interactive-video buffer-size temporal
  25k
user@host# set class-of-service schedulers interactive-video priority high
user@host# set class-of-service schedulers business-tier1 transmit-rate percent
  20
user@host# set class-of-service schedulers business-tier1 buffer-size percent 20
user@host# set class-of-service schedulers business-tier1 priority medium-high
user@host# set class-of-service schedulers network-control transmit-rate percent
  5
user@host# set class-of-service schedulers network-control buffer-size percent 5
user@host# set class-of-service schedulers network-control priority high
user@host# set class-of-service schedulers bulk transmit-rate remainder
user@host# set class-of-service schedulers bulk buffer-size remainder
user@host# set class-of-service schedulers bulk priority low
user@host# set class-of-service schedulers bulk drop-profile-map loss-priority low
  protocol any drop-profile congest-drop-low
user@host# set class-of-service schedulers bulk drop-profile-map loss-priority high
  protocol any drop-profile congest-drop-high
user@host# set class-of-service schedulers business-tier2 transmit-rate percent
  20
user@host# set class-of-service schedulers business-tier2 buffer-size percent 20
user@host# set class-of-service schedulers business-tier2 priority medium-high
```

3. Configure scheduler maps.

```
[edit]
user@host# set class-of-service scheduler-maps DCN-map forwarding-class voice
  scheduler voice
user@host# set class-of-service scheduler-maps DCN-map forwarding-class
  interactive-video scheduler interactive-video
user@host# set class-of-service scheduler-maps DCN-map forwarding-class
  business-tier1 scheduler business-tier1
user@host# set class-of-service scheduler-maps DCN-map forwarding-class
  business-tier2 scheduler business-tier2
user@host# set class-of-service scheduler-maps DCN-map forwarding-class bulk
  scheduler bulk
user@host# set class-of-service scheduler-maps DCN-map forwarding-class
  network-control scheduler network-control
```

Applying Class of Service Components

Step-by-Step Procedure

All classifiers are the same on all devices where a specific classifier is used. However, the attachment of classifiers and re-write rules to routing instances or interfaces is based on whether the traffic flowing through the circuit or interface are Layer 2 or Layer 3 data packets.

When applying class-of-service components, keep the following in mind:

- All XE interfaces access the MPLS network; that is, they are either connected to the P router or to the other PE router.
- All aggregated Ethernet interfaces are attached with the dot1p classifiers and re-write rules because the packets received by the AE interfaces are VLAN tagged packets. IRB interfaces are attached with the TOS classifier and TOS rewrite rules because the packets received by these interfaces contain IPv4 header packets.

To apply customer class of service to various interfaces on different devices sitting at the data center edge or core:

NOTE: Throughout this procedure, an asterisk (*) is used as a wildcard value to represent all static interface units.

1. Apply the EXP classifier to all customer routing instances.

```
[edit]
user@host# set class-of-service routing-instances vs1* classifiers exp
  DCN-exp-classifier
user@host# set class-of-service routing-instances vs2* classifiers exp
  DCN-exp-classifier
user@host# set class-of-service routing-instances vrf* classifiers exp
  DCN-exp-classifier
```

2. Apply the CoS configuration to each logical interface.

```
[edit]
user@host# set class-of-service interfaces ge-0/0/0 scheduler-map DCN-map
user@host# set class-of-service interfaces ge-0/0/0 unit * classifiers ieee-802.1
  DCN-dot1p-classifier
user@host# set class-of-service interfaces ge-0/0/0 unit * rewrite-rules ieee-802.1
  DCN-dot1p-rewrite
user@host# set class-of-service interfaces xe-0/2/0 scheduler-map DCN-map
user@host# set class-of-service interfaces xe-0/2/0 unit * classifiers ieee-802.1
  DCN-dot1p-classifier
user@host# set class-of-service interfaces xe-0/2/0 unit * rewrite-rules ieee-802.1
  DCN-dot1p-rewrite
user@host# set class-of-service interfaces xe-0/3/1 scheduler-map DCN-map
user@host# set class-of-service interfaces xe-0/3/1 unit * classifiers ieee-802.1
  DCN-dot1p-classifier
user@host# set class-of-service interfaces xe-0/3/1 unit * rewrite-rules ieee-802.1
  DCN-dot1p-rewrite
```

3. Apply the CoS configuration to each unit 0 logical static interface.

```
[edit]
user@host# set class-of-service interfaces xe-5/0/0 scheduler-map DCN-map
user@host# set class-of-service interfaces xe-5/0/0 unit 0 classifiers exp
  DCN-exp-classifier
user@host# set class-of-service interfaces xe-5/0/0 unit 0 rewrite-rules exp
  DCN-exp-rewrite
user@host# set class-of-service interfaces xe-5/0/1 scheduler-map DCN-map
user@host# set class-of-service interfaces xe-5/0/1 unit 0 classifiers exp
  DCN-exp-classifier
user@host# set class-of-service interfaces xe-5/0/1 unit 0 rewrite-rules exp
  DCN-exp-rewrite
user@host# set class-of-service interfaces xe-5/0/3 scheduler-map DCN-map
user@host# set class-of-service interfaces xe-5/0/3 unit 0 classifiers ieee-802.1
  DCN-dot1p-classifier
user@host# set class-of-service interfaces xe-5/0/3 unit 0 rewrite-rules ieee-802.1
  DCN-dot1p-rewrite
user@host# set class-of-service interfaces xe-5/1/0 scheduler-map DCN-map
user@host# set class-of-service interfaces xe-5/1/0 unit 0 classifiers exp
  DCN-exp-classifier
user@host# set class-of-service interfaces xe-5/1/0 unit 0 rewrite-rules exp
  DCN-exp-rewrite
user@host# set class-of-service interfaces xe-5/1/1 scheduler-map DCN-map
user@host# set class-of-service interfaces xe-5/1/1 unit 0 classifiers exp
  DCN-exp-classifier
user@host# set class-of-service interfaces xe-5/1/1 unit 0 rewrite-rules exp
  DCN-exp-rewrite
user@host# set class-of-service interfaces xe-5/1/2 scheduler-map DCN-map
user@host# set class-of-service interfaces xe-5/1/2 unit 0 classifiers exp
  DCN-exp-classifier
user@host# set class-of-service interfaces xe-5/1/2 unit 0 rewrite-rules exp
  DCN-exp-rewrite
user@host# set class-of-service interfaces xe-5/2/1 scheduler-map DCN-map
user@host# set class-of-service interfaces xe-5/2/1 unit 0 classifiers exp
  DCN-exp-classifier
user@host# set class-of-service interfaces xe-5/2/1 unit 0 rewrite-rules exp
  DCN-exp-rewrite
user@host# set class-of-service interfaces xe-5/3/1 scheduler-map DCN-map
user@host# set class-of-service interfaces xe-5/3/1 unit 0 classifiers exp
  DCN-exp-classifier
user@host# set class-of-service interfaces xe-5/3/1 unit 0 rewrite-rules exp
  DCN-exp-rewrite
user@host# set class-of-service interfaces xe-5/3/2 scheduler-map DCN-map
user@host# set class-of-service interfaces xe-5/3/2 unit 0 classifiers exp
  DCN-exp-classifier
user@host# set class-of-service interfaces xe-5/3/2 unit 0 rewrite-rules exp
  DCN-exp-rewrite
```

4. Attach the dot1p classifier to all aggregated Ethernet logical interfaces.

```
[edit]
user@host# set class-of-service interfaces ae0 scheduler-map DCN-map
user@host# set class-of-service interfaces ae0 unit * classifiers ieee-802.1
  DCN-dot1p-classifier
user@host# set class-of-service interfaces ae0 unit * rewrite-rules ieee-802.1
  DCN-dot1p-rewrite
```

5. Apply the CoS configuration to each aggregated Ethernet logical interface unit.

```
[edit]
user@host# set class-of-service interfaces ae1 scheduler-map DCN-map
user@host# set class-of-service interfaces ae1 unit 0 classifiers ieee-802.1
  DCN-dot1p-classifier
user@host# set class-of-service interfaces ae1 unit 0 classifiers inet-precedence
  DCN-TOS-classifier deactivate class-of-service interfaces ae1 unit 0 classifiers
  inet-precedence
user@host# set class-of-service interfaces ae1 unit 0 rewrite-rules ieee-802.1
  DCN-dot1p-rewrite
user@host# set class-of-service interfaces ae1 unit 0 rewrite-rules inet-precedence
  DCN-TOS-rewrite deactivate class-of-service interfaces ae1 unit 0 rewrite-rules
  inet-precedence DCN-TOS-rewrite
user@host# set class-of-service interfaces ae2 scheduler-map DCN-map
user@host# set class-of-service interfaces ae2 unit 0 classifiers ieee-802.1
  DCN-dot1p-classifier
user@host# set class-of-service interfaces ae2 unit 0 rewrite-rules ieee-802.1
  DCN-dot1p-rewrite
user@host# set class-of-service interfaces ae3 scheduler-map DCN-map
user@host# set class-of-service interfaces ae3 unit 0 classifiers ieee-802.1
  DCN-dot1p-classifier
user@host# set class-of-service interfaces ae3 unit 0 rewrite-rules ieee-802.1
  DCN-dot1p-rewrite
user@host# set class-of-service interfaces ae10 scheduler-map DCN-map
user@host# set class-of-service interfaces ae10 unit * classifiers ieee-802.1
  DCN-dot1p-classifier
user@host# set class-of-service interfaces ae10 unit * rewrite-rules ieee-802.1
  DCN-dot1p-rewrite
user@host# set class-of-service interfaces ae11 scheduler-map DCN-map
user@host# set class-of-service interfaces ae11 unit 0 classifiers ieee-802.1
  DCN-dot1p-classifier
user@host# set class-of-service interfaces ae11 unit 0 rewrite-rules ieee-802.1
  DCN-dot1p-rewrite
```

6. Attach the TOS classifier and TOS rewrite rules to all IRB interfaces.

```
[edit]
user@host# set class-of-service interfaces irb unit * classifiers inet-precedence
  DCN-TOS-classifier
user@host# set class-of-service interfaces irb unit * rewrite-rules inet-precedence
  DCN-TOS-rewrite
```

Configuring Virtual Switches

Step-by-Step Procedure

The virtual switch routing instance contains only two bridge domains.

To configure the virtual switch:

1. Configure a virtual switch routing instance on Device R-5.

NOTE: The following example shows only customer vs1-1.

```
[edit]
user@host# set routing-instances vs1-1 instance-type virtual-switch
```

```
user@host# set routing-instances vs1-1 interface xe-10/2/0.0
user@host# set routing-instances vs1-1 interface ae0.0
user@host# set routing-instances vs1-1 interface ae10.0
user@host# set routing-instances vs1-1 interface ae11.0
user@host# set routing-instances vs1-1 bridge-domains bd-1 domain-type bridge
user@host# set routing-instances vs1-1 bridge-domains bd-1 vlan-id 1
user@host# set routing-instances vs1-1 bridge-domains bd-1 bridge-options
  mac-table-size 1048575
user@host# set routing-instances vs1-1 bridge-domains bd-1 bridge-options
  interface-mac-limit 131071
user@host# set routing-instances vs1-1 bridge-domains bd-1 bridge-options
  mac-statistics
user@host# set routing-instances vs1-1 bridge-domains bd-2 domain-type bridge
user@host# set routing-instances vs1-1 bridge-domains bd-2 vlan-id 2
user@host# set routing-instances vs1-1 bridge-domains bd-2 bridge-options
  mac-table-size 1048575
user@host# set routing-instances vs1-1 bridge-domains bd-2 bridge-options
  interface-mac-limit 131071
user@host# set routing-instances vs1-1 bridge-domains bd-2 bridge-options
  mac-statistics
```

2. Configure a virtual switch routing instance on Device R-4.

NOTE: The following example shows only customer vs1-1.

```
[edit]
user@host# set routing-instances vs1-1 instance-type virtual-switch
user@host# set routing-instances vs1-1 interface xe-9/2/0.0
user@host# set routing-instances vs1-1 interface ae0.0
user@host# set routing-instances vs1-1 interface ae10.0
user@host# set routing-instances vs1-1 interface ae11.0
user@host# set routing-instances vs1-1 bridge-domains bd-1 domain-type bridge
user@host# set routing-instances vs1-1 bridge-domains bd-1 vlan-id 1
user@host# set routing-instances vs1-1 bridge-domains bd-1 bridge-options
  mac-table-size 1048575
user@host# set routing-instances vs1-1 bridge-domains bd-1 bridge-options
  interface-mac-limit 131071
user@host# set routing-instances vs1-1 bridge-domains bd-1 bridge-options
  mac-statistics
user@host# set routing-instances vs1-1 bridge-domains bd-2 domain-type bridge
user@host# set routing-instances vs1-1 bridge-domains bd-2 vlan-id 2
user@host# set routing-instances vs1-1 bridge-domains bd-2 bridge-options
  mac-table-size 1048575
user@host# set routing-instances vs1-1 bridge-domains bd-2 bridge-options
  interface-mac-limit 131071
user@host# set routing-instances vs1-1 bridge-domains bd-2 bridge-options
  mac-statistics
```

Verification

The following sections show how to verify the configuration in this example:

Verifying Device R-0 Interface Configuration

Purpose Verify proper interface configuration on Device R-0.

Action Verify that interface xe-0/3/1 is UP.

```
user@R0# run show interfaces terse xe-0/3/1
Interface               Admin Link Proto    Local                 Remote
xe-0/3/1                up    up
xe-0/3/1.0              up    up   bridge
xe-0/3/1.1              up    up   bridge
```

Meaning Indicates that the customer facing interface is UP on edge Device R-0.

NOTE: Interface status verification is essential to ensure that the data plane is in the proper state to forward traffic.

Verifying Device R-2 Interface Configuration

Purpose Verify proper interface configuration on Device R-2.

Action Verify that interface ae0 is UP.

```
user@R2# run show interfaces terse ae0
Interface               Admin Link Proto    Local                 Remote
ae0                     up    up
ae0.0                   up    up   bridge
ae0.1                   up    up   bridge
```

Meaning Indicates that the MC-AE active/standby interface is UP on edge Device R-2.

NOTE: On an MC-AE bundle in active/standby mode, the LAG on the active router must always be UP to ensure that the data plane is in the proper state to forward traffic.

Verifying Device R-3 Interface Configuration

Purpose Verify proper interface configuration on Device R-3.

Action

- Verify that interface ae0 is DOWN.

NOTE: This interface is DOWN because of the MC-LAG configuration.

```
user@R3# run show interfaces terse ae0
Interface               Admin    Link Proto    Local                 Remote
ae0                       up    down
ae0.0                     up    down bridge
ae0.1                     up    down bridge
```

- Verify the loopback interface configuration.

```
user@R3# run show interfaces terse lo0.0
Interface               Admin Link Proto    Local                 Remote
lo0.0                     up    up   inet     10.255.36.216          127.0.0.1

                                    iso
47.0005.80ff.f800.0000.0108.0001.0102.5503.6216
                                    inet6    abcd::10:255:36:216
                                             fe80::2a0:a50f:fc78:f69f
```

Meaning Indicates that the MCAE active-standby interface is DOWN on edge Device R-3. On an MCAE bundle in active-standby mode, the LAG on the standby router must always be down.

Verifying Device R-4 Interface Configuration

Purpose Verify proper interface configuration on Device R-4.

Action

- Verify Interface xe-9/2/0 is UP.

```
user@R4# run show interfaces terse xe-9/2/0
Interface               Admin Link Proto    Local                 Remote
xe-9/2/0                   up    up
xe-9/2/0.0                 up    up   bridge
xe-9/2/0.32767             up    up   multiservice
```

- Verify that Interface ae0 is UP.

```
user@R4# run show interfaces terse ae0
Interface               Admin   Link Proto    Local                 Remote
ae0                        up    up
ae0.0                      up    up   bridge
ae0.32767                  up    up   multiservice
```

- Verify that Interface ae10 is UP.

```
user@R4# run show interfaces terse ae10
Interface               Admin   Link Proto    Local                 Remote
ae10                       up    up
ae10.0                     up    up   bridge
ae10.32767                 up    up   multiservice
```

- Verify that Interface ae11 is UP.

```
user@R4# run show interfaces terse ae11
Interface             Admin   Link Proto    Local                 Remote
ae11                     up    up
ae11.0                   up    up bridge
ae11.32767               up    up multiservice
```

Meaning Indicates that the MC-AE active/active interfaces ae0, ae10, and ae11 are UP, and ICP interface xe-9/2/0 is UP on core router Device R-4. As both LAG bundles are capable of forwarding traffic in an MC-AE active/active configuration, it is important to verify that the bundles are UP to ensure that no traffic drops occur.

Verifying Device R-5 Interface Configuration

Purpose Perform the following actions to verify proper interface configuration on Device R-5.

Action

- Verify Interface xe-10/2/0 is UP.

```
user@R5# run show interfaces terse xe-10/2/0
Interface               Admin   Link Proto    Local                 Remote
xe-10/2/0                  up    up
xe-10/2/0.0                up    up   bridge
xe-10/2/0.32767            up    up   multiservice
```

- Verify Interface ae0 is UP.

```
user@R5# run show interfaces terse ae0
Interface               Admin   Link Proto    Local                 Remote
ae0                        up    up
ae0.0                      up    up   bridge
ae0.32767                  up    up   multiservice
```

- Verify Interface ae10 is UP.

```
user@R5# run show interfaces terse ae10
Interface               Admin   Link Proto    Local                 Remote
ae10                       up    up
```

```
ae10.0                  up    up   bridge
ae10.32767              up    up   multiservice
```

- Verify Interface ae11 is UP.

```
user@R5# run show interfaces terse ae11
Interface               Admin   Link Proto    Local                 Remote
ae11                         up    up
ae11.0                       up    up bridge
ae11.32767                   up    up multiservice
```

Meaning Indicates that the MCAE active-active interfaces ae0, ae10 and ae11 are UP and ICP interface xe-10/2/0 is UP on the core routers R-5. As both LAG bundles are capable of forwarding traffic in and MCAE active-active configuration, it is important to verify that the bundles are UP to ensure no traffic drops occur.

Verifying Device R-8 Interface Configuration

Purpose Verify proper interface and VLAN configuration on Device R-8.

Action

- Verify that interface ae0 is UP.

NOTE: This ae0 interface connects to devices R-4 and R-5.

```
user@R8# run show interfaces terse | match ae0
xe-0/0/6.0              up    up   aenet
xe-0/0/17.0              up    up   aenet
ae0                      up    up
ae0.0                    up    up   eth-switch
```

- Verify that interface ge-0/0/25 is UP.

```
user@R8# run show interfaces terse | match ge-0/0/25
ge-0/0/25               up    up
ge-0/0/25.0             up    up   eth-switch
```

- Verify that the configured interfaces are installed on the VLANs.

NOTE: Only 2 of 4094 VLANs are shown.

```
user@R8# run show vlans
default         4094
                        ae0.0*, ge-0/0/25.0*
vlan-1          1
                        ae0.0*, ge-0/0/25.0*
vlan-2          2
                        ae0.0*, ge-0/0/25.0*
```

- Verify Interface ae11 is UP.

```
user@R8# run show interfaces terse ae11
Interface               Admin Link Proto    Local                 Remote
ae11                         up    up
```

```
ae11.0                          up    up bridge
ae11.32767                      up    up multiservice
```

Meaning Indicates that interfaces ae0 and ae1 on router devices R-4 and R-5 (directly connected to ToR Device R-8) are UP. As both LAG bundles are capable of forwarding traffic in an MCAE active-active configuration, it is important to verify that the bundles are UP to ensure no traffic drops occur. Also indicates that interface ge-0/0/25 on R-8 (connected to the server) is UP. Interface ae0 (connecting to the core routers) and interface ge-0/0/25 (connecting to the server), are part of VLANs configured on ToR R-8.

Verifying Device R-9 Interface Configuration

Purpose Perform the following actions to verify proper interface and VLAN configuration on Device R-9.

Action

- Verify that interface ae0 is UP.

NOTE: This ae0 interface connects to devices R-4 and R-5.

```
user@R9# run show interfaces terse | match ae0
xe-0/0/1.0                 up    up   aenet
xe-0/0/2.0                 up    up   aenet
ae0                        up    up
ae0.0                      up    up   eth-switch
```

- Verify that interface ge-0/0/25 is UP.

```
user@R9# run show interfaces terse | match ge-0/0/5
ge-0/0/5                 up    up
ge-0/0/5.0               up    up   eth-switch
```

- Verify that the configured interfaces are installed on the VLANs.

NOTE: Only 2 of 4094 VLANs are shown.

```
user@R9# run show vlans
default         4094
                        ae0.0*, ge-0/0/5.0*
vlan-1          1
                        ae0.0*, ge-0/0/5.0*
vlan-2          2
                        ae0.0*, ge-0/0/5.0*
```

Meaning Indicates that interface ae0 on ToR Device R-9 (connected directly to core devices R-4 and R-5) is UP. Also indicates that Interface ge-0/0/5 on R-9 (connected to server) is UP and that ae0 and ge-0/0/5 are installed in the VLANs on R-9.

Verifying Device R-2 MC-LAG Configuration

Purpose Verify proper MC-LAG configuration on Device R-2.

Action

- Verify LACP statistics on interface ae0:

```
user@R2# run show lacp statistics interfaces ae0
Aggregated interface: ae0
    LACP Statistics:       LACP Rx     LACP Tx   Unknown Rx   Illegal Rx
      xe-3/1/3              266014      265813            0            0
      xe-3/2/2              265991      265813            0            0
```

- Verify LACP status on interface ae0:

```
user@R2# run show lacp interfaces ae0
Aggregated interface: ae0
    LACP state:       Role   Exp   Def  Dist  Col  Syn  Aggr  Timeout  Activity

      xe-3/1/3       Actor    No    No   Yes  Yes  Yes   Yes     Fast    Active

      xe-3/1/3     Partner    No    No   Yes  Yes  Yes   Yes     Fast    Active

      xe-3/2/2       Actor    No    No   Yes  Yes  Yes   Yes     Fast    Active

      xe-3/2/2     Partner    No    No   Yes  Yes  Yes   Yes     Fast    Active

    LACP protocol:        Receive State  Transmit State          Mux State
      xe-3/1/3                 Current   Fast periodic Collecting distributing

      xe-3/2/2                 Current   Fast periodic Collecting distributing
```

- Verify MCAE active/standby interface status:

```
user@R2# run show interfaces mc-ae id 10
Member Link                  : ae0
 Current State Machine's State: mcae active state
 Local Status                   : active
 Local State                    : up
 Peer Status                    : standby
 Peer State                      : up
     Logical Interface          : ae0.0
     Topology Type             : bridge
     Local State                   : up
     Peer State                    : up
     Peer Ip/MCP/State        : N/A
     Logical Interface           : ae0.1
     Topology Type              : bridge
     Local State                   : up
     Peer State                    : up
     Peer Ip/MCP/State         : N/A
```

Meaning Indicates that the LACP is running on interface ae0 and its mux state is in the distributing state. Interface ae0 is active locally and in stand-by mode on the peer device (R-3) for the ae0 bundle provisioned in active-standby mode.

Verifying Device R-3 MC-LAG Configuration

Purpose Verify proper MC-LAG configuration on Device R-3.

Action

- Verify LACP statistics on interface ae0:

```
user@R3# run show lacp statistics interfaces ae0
Aggregated interface: ae0
    LACP Statistics:       LACP Rx     LACP Tx   Unknown Rx   Illegal Rx
      xe-4/0/1             265801      265566            0            0
      xe-0/2/2             265758      265451            0            0
```

- Verify LACP status on interface ae0:

```
user@R3# run show lacp interfaces ae0
Aggregated interface: ae0
    LACP state:       Role   Exp   Def  Dist  Col  Syn  Aggr  Timeout  Activity

      xe-4/0/1       Actor    No    No    No   No   No   Yes     Fast    Active

      xe-4/0/1     Partner    No    No    No   No  Yes   Yes     Fast    Active

      xe-0/2/2       Actor    No    No    No   No   No   Yes     Fast    Active

      xe-0/2/2     Partner    No    No    No   No  Yes   Yes     Fast    Active

    LACP protocol:        Receive State  Transmit State          Mux State
      xe-4/0/1                  Current   Fast periodic           Waiting
      xe-0/2/2                  Current   Fast periodic           Waiting
```

- Verify MCAE active/standby interface status:

```
user@R3# run show interfaces mc-ae id 10
 Member Link                  : ae0
 Current State Machine's State: mcae standby state
 Local Status                   : standby
 Local State                    : up
 Peer Status                    : active
 Peer State                     : up
     Logical Interface          : ae0.0
     Topology Type              : bridge
     Local State                    : up
     Peer State                     : up
     Peer Ip/MCP/State        : N/A
     Logical Interface            : ae0.1
     Topology Type              : bridge
     Local State                    : up
     Peer State                     : up
     Peer Ip/MCP/State        : N/A
```

Meaning If the link between R-4 and R-2 goes down, then the alternative link between R-4 and R-3 will be utilized for data forwarding and vice-versa.

If the router Device R-2 goes down, then the router Device R-3 will became active for the vpls and traffic forwarding will take via router Device R-3. Connection-type permanent knob is configured on all the vpls instanes on R-2 and R-3 to make sure vpls will be UP always and convergence will be faster.

Verifying Device R-4 MC-LAG Configuration

Purpose Verify proper MC-LAG configuration on Device R-4.

Action

- Verify LACP statistics on interface ae0:

```
user@R4# run show lacp statistics interfaces ae0
Aggregated interface: ae0
    LACP Statistics:       LACP Rx     LACP Tx   Unknown Rx   Illegal Rx
      xe-4/3/0              338991      339297            0            0
      xe-0/0/0              338808      339088            0            0
```

- Verify LACP status on interface ae0:

```
user@R4# run show lacp interfaces ae0
Aggregated interface: ae0
    LACP state:       Role   Exp   Def  Dist  Col  Syn  Aggr  Timeout  Activity

      xe-4/3/0       Actor    No    No   Yes  Yes  Yes   Yes     Fast    Active

      xe-4/3/0     Partner    No    No   Yes  Yes  Yes   Yes     Fast    Active

      xe-0/0/0       Actor    No    No    No   No  Yes   Yes     Fast    Active

      xe-0/0/0     Partner    No    No    No   No   No   Yes     Fast    Active

    LACP protocol:        Receive State  Transmit State          Mux State
      xe-4/3/0                 Current   Fast periodic Collecting distributing

      xe-0/0/0                  Current   Fast periodic           Attached
```

- Verify MCAE interface ae0 status:

```
user@R4# run show interfaces mc-ae id 10
Member Link                  : ae0
 Current State Machine's State: mcae active state
 Local Status                : active
 Local State                 : up
 Peer Status                 : active
 Peer State                  : up
     Logical Interface       : ae0.0
     Topology Type           : bridge
     Local State             : up
     Peer State              : up
     Peer Ip/MCP/State       : 4.1.0.2 xe-9/2/0.0 up
```

If the link between R-4 and R-2 goes down, then the alternative link between R-4 and R-3 will be utilized for data forwarding and vice-versa.

If the router Device R-2 goes down, then the router Device R-3 will became active for the vpls and traffic forwarding will take via router Device R-3. Connection-type permanent knob is configured on all the vpls instanes on R-2 and R-3 to make sure vpls will be UP always and convergence will be faster.

- Verify the LACP state on interface ae10:

```
user@R4# run show lacp interfaces ae10
Aggregated interface: ae10
    LACP state:       Role   Exp   Def  Dist  Col  Syn  Aggr  Timeout  Activity

      xe-8/0/2       Actor    No    No   Yes  Yes  Yes   Yes     Fast    Active

      xe-8/0/2     Partner    No    No   Yes  Yes  Yes   Yes     Fast    Active

    LACP protocol:        Receive State  Transmit State          Mux State
      xe-8/0/2                 Current   Fast periodic Collecting distributing
```

- Verify the LACP statistics on interface ae10:

```
user@R4# run show lacp statistics interfaces ae10
Aggregated interface: ae10
    LACP Statistics:       LACP Rx     LACP Tx   Unknown Rx   Illegal Rx
      xe-8/0/2              338154      338769            0            0
```

- Verify the MCAE status of interface ae10:

```
user@R4# run show interfaces mc-ae id 1
 Member Link                  : ae10
 Current State Machine's State: mcae active state
 Local Status                   : active
 Local State                    : up
 Peer Status                   : active
 Peer State                     : up
     Logical Interface          : ae10.0
     Topology Type             : bridge
     Local State                   : up
     Peer State                    : up
     Peer Ip/MCP/State       : 4.1.0.2 xe-9/2/0.0 up
```

- Verify MC-LAG status of interface ae11:

```
user@R4# run show interfaces mc-ae id 2
 Member Link                  : ae11
 Current State Machine's State: mcae active state
 Local Status                   : active
 Local State                   : up
 Peer Status                    : active
 Peer State                     : up
     Logical Interface        : ae11.0
     Topology Type            : bridge
     Local State                   : up
     Peer State                    : up
     Peer Ip/MCP/State        : 4.1.0.2 xe-9/2/0.0 up
```

- Verify status of LACP on interface ae11:

```
user@R4# run show lacp interfaces ae11
Aggregated interface: ae11
    LACP state:       Role   Exp   Def  Dist  Col  Syn  Aggr  Timeout  Activity

      xe-2/2/2       Actor    No   Yes    No   No   No   Yes     Fast    Active

      xe-2/2/2     Partner    No   Yes    No   No   No   Yes     Fast   Passive

    LACP protocol:        Receive State  Transmit State          Mux State
      xe-2/2/2               Current   Fast periodic Collecting distributing
```

- Verify LACP statistics on interface ae11:

```
user@R4# run show lacp statistics interfaces ae11
Aggregated interface: ae11
    LACP Statistics:       LACP Rx     LACP Tx   Unknown Rx   Illegal Rx
      xe-2/2/2                 348900      348911            0            0
```

Meaning Indicates that the LACP is running on ae0, ae10 and ae11 and its mux state is in the appropriate state. Interface ae0 is active locally and functioning as stand-by on the peer device (R-5). Interfaces ae0, ae10, and ae11 are MCLAG active locally and on the peer device.

Verifying Device R-5 MC-LAG Configuration

Purpose Perform the following actions to verify proper MC-LAG configuration on Device R-5.

Action

- Verify LACP statistics on interface ae0:

```
user@R5# run show lacp statistics interfaces ae0
Aggregated interface: ae0
    LACP Statistics:       LACP Rx     LACP Tx   Unknown Rx   Illegal Rx
      xe-7/0/0              338921      339187            0            0
      xe-5/2/2              338482      338925            0            0
```

- Verify LACP status on interface ae0:

```
user@R5# run show lacp interfaces ae0
Aggregated interface: ae0
    LACP state:       Role   Exp   Def  Dist  Col  Syn  Aggr  Timeout  Activity

      xe-7/0/0       Actor    No    No   Yes  Yes  Yes   Yes     Fast    Active

      xe-7/0/0     Partner    No    No   Yes  Yes  Yes   Yes     Fast    Active

      xe-5/2/2       Actor    No    No    No   No  Yes   Yes     Fast    Active

      xe-5/2/2     Partner    No    No    No   No   No   Yes     Fast    Active

    LACP protocol:        Receive State  Transmit State          Mux State
      xe-7/0/0                 Current   Fast periodic Collecting distributing

      xe-5/2/2                  Current   Fast periodic           Attached
```

- Verify MCAE interface ae0 status:

```
user@R5# run show interfaces mc-ae id 10
 Member Link                  : ae0
 Current State Machine's State: mcae active state
 Local Status                  : active
 Local State                   : up
 Peer Status                   : active
 Peer State                    : up
     Logical Interface        : ae0.0
     Topology Type            : bridge
     Local State                  : up
     Peer State                  : up
     Peer Ip/MCP/State        : 4.1.0.1 xe-10/2/0.0 up
```

- Verify the LACP state on interface ae10:

```
user@R5# run show lacp interfaces ae10
Aggregated interface: ae10
    LACP state:       Role   Exp   Def  Dist  Col  Syn  Aggr  Timeout  Activity

      xe-9/0/1       Actor    No    No   Yes  Yes  Yes   Yes     Fast    Active

      xe-9/0/1     Partner    No    No   Yes  Yes  Yes   Yes     Fast    Active

    LACP protocol:        Receive State  Transmit State          Mux State
      xe-9/0/1                 Current   Fast periodic Collecting distributing
```

- Verify the LACP statistics on interface ae10:

```
user@R5# run show lacp statistics interfaces ae10
```

```
Aggregated interface: ae10
    LACP Statistics:       LACP Rx     LACP Tx   Unknown Rx   Illegal Rx
      xe-9/0/1              338170      338783            0            0
```

- Verify the MCAE status of interface ae10:

```
user@R5# run show interfaces mc-ae id 1
 Member Link                  : ae10
 Current State Machine's State: mcae active state
 Local Status                  : active
 Local State                   : up
 Peer Status                   : active
 Peer State                    : up
     Logical Interface         : ae10.0
     Topology Type             : bridge
     Local State                  : up
     Peer State                   : up
     Peer Ip/MCP/State      : 4.1.0.1 xe-10/2/0.0 up
```

- Verify MC-LAG status of interface ae11:

```
user@R5# run show interfaces mc-ae id 2
 Member Link                  : ae11
 Current State Machine's State: mcae active state
 Local Status                  : active
 Local State                   : up
 Peer Status                   : active
 Peer State                    : up
     Logical Interface         : ae11.0
     Topology Type             : bridge
     Local State                  : up
     Peer State                   : up
     Peer Ip/MCP/State        : 4.1.0.1 xe-10/2/0.0 up
```

- Verify status of LACP on interface ae11:

```
user@R5# run show lacp interfaces ae11
Aggregated interface: ae11
    LACP state:       Role   Exp   Def  Dist  Col  Syn  Aggr  Timeout  Activity

      xe-2/1/0       Actor    No    No   Yes  Yes  Yes   Yes     Fast    Active

      xe-2/1/0     Partner    No    No   Yes  Yes  Yes   Yes     Fast    Active

    LACP protocol:        Receive State  Transmit State          Mux State
      xe-2/1/0                Current   Fast periodic Collecting distributing
```

- Verify LACP statistics on interface ae11:

```
user@R5# run show lacp statistics interfaces ae11
Aggregated interface: ae11
    LACP Statistics:       LACP Rx     LACP Tx   Unknown Rx   Illegal Rx
      xe-2/1/0              348089      348763            0            0
```

Meaning Indicates that the LACP is running on interfaces ae0, ae10, and ae11 and its mux state is appropriate. Interface ae0 is active locally and functioning as stand-by on the peer device. Interfaces ae0, ae10, and ae11 are MC-LAG active locally and on the peer device.

Verifying VPLS Instances are Up on Each Router

Purpose Verify VPLS instances are functioning properly on each router.

Action

- Verify the VPLS instance on Device R-0:

```
user@R0# run show vpls connections instance vs1-1
Layer-2 VPN connections:
- - - - - - -  TRUNCATED - - - - - - -
Instance: vs1-1
  Local site: 1 (1)
    connection-site           Type  St     Time last up          # Up trans
    2                         rmt   Up     Nov  1 12:56:56 2013           1
      Remote PE: 10.255.32.193, Negotiated control-word: No
      Incoming label: 262162, Outgoing label: 262161
      Local interface: lsi.1048577, Status: Up, Encapsulation: VPLS
        Description: Intf - vpls vs1-1 local site 1 remote site 2
    3                         rmt   Up     Nov  1 13:02:14 2013           1
      Remote PE: 10.255.36.216, Negotiated control-word: No
      Incoming label: 262163, Outgoing label: 262161
      Local interface: lsi.1052580, Status: Up, Encapsulation: VPLS
        Description: Intf - vpls vs1-1 local site 1 remote site 3
```

- Verify the VPLS instance on Device R-2:

```
user@R2# run show vpls connections instance vs1-1
Layer-2 VPN connections:
- - - - - - -  TRUNCATED - - - - - - -
Instance: vs1-1
  Local site: 2 (2)
    connection-site           Type  St     Time last up          # Up trans
    1                         rmt   Up     Nov  1 12:57:33 2013           1
      Remote PE: 10.255.35.128, Negotiated control-word: No
      Incoming label: 262161, Outgoing label: 262162
      Local interface: lsi.1048577, Status: Up, Encapsulation: VPLS
        Description: Intf - vpls vs1-1 local site 2 remote site 1
    3                         rmt   Up     Nov  1 13:03:06 2013           1
      Remote PE: 10.255.36.216, Negotiated control-word: No
      Incoming label: 262163, Outgoing label: 262162
      Local interface: lsi.1052580, Status: Up, Encapsulation: VPLS
        Description: Intf - vpls vs1-1 local site 2 remote site 3
```

- Verify the VPLS instance on Device R-3:

```
user@R3# run show vpls connections instance vs1-1
Layer-2 VPN connections:
- - - - - - -  TRUNCATED - - - - - - -
Instance: vs1-1
  Local site: 3 (3)
    connection-site           Type  St     Time last up          # Up trans
    1                         rmt   Up     Nov  1 13:02:07 2013           1
      Remote PE: 10.255.35.128, Negotiated control-word: No
      Incoming label: 262161, Outgoing label: 262163
      Local interface: lsi.1048577, Status: Up, Encapsulation: VPLS
        Description: Intf - vpls vs1-1 local site 3 remote site 1
    2                         rmt   Up     Nov  1 13:03:57 2013           1
      Remote PE: 10.255.32.193, Negotiated control-word: No
      Incoming label: 262162, Outgoing label: 262163
      Local interface: lsi.1052580, Status: Up, Encapsulation: VPLS
        Description: Intf - vpls vs1-1 local site 3 remote site 2
```

Meaning Applied on Device R-0, R-2, and R-3, these verifications indicate that the VPLS instance on R-0 has two sites connecting router Device R-2 and R-3. The VPLS pseudowires should be in the UP state on all routers (that is, R-0, R-2, and R-3).

Verifying Routing Instance and Bridge Domain Information on Each Device

Purpose Verify the routing instance and bridge domain configuration on each router.

Action

- Verify the routing instance and bridge domain information on Device R-5:

```
user@R5# run show l2-learning instance vs1-1
Information for routing instance and bridge domain:
Flags (DL -disable learning, SE -stats enabled,
       AD -packet action drop, LH -mac limit hit)
Inst  Logical    Routing             Bridging            Index  IRB     Flags
     BD
Type  System     Instance            Domain                     Index
     vlan
RTT   Default   vs1-1                                        6          SE
BD    Default    vs1-1               bd-1                  2          SE
                1
BD    Default    vs1-1               bd-2                  3          SE
                2
```

- Verify the routing instance and bridge domain information on Device R-4:

```
user@R4# run show vpls connections instance vs1-1
Information for routing instance and bridge domain:
Flags (DL -disable learning, SE -stats enabled,
       AD -packet action drop, LH -mac limit hit)
Inst  Logical    Routing             Bridging            Index  IRB     Flags
     BD
Type  System     Instance            Domain                     Index
     vlan
RTT   Default   vs1-1                                     7             SE
BD    Default    vs1-1               bd-1                12            SE
                1
BD    Default    vs1-1               bd-2                13            SE
                2
```

Meaning Indicates that each VPLS routing instance has bridge domains configured and that they are active with an index for each bridge domain.

Verifying Customer Isolation on Devices R-2 and R-3

Purpose Verify the isolation of customers on Device R-2 and Device R-3. Tenant isolation can be provided by configuring VPLS services by bunching N number of communicating elements (for example, VLANs) for each tenant. We have virtual switch isolation for each tenant and bridge domain isolation for VLANs.

The following commands show the MAC addresses learned in each bridge domain for different routing instances.

NOTE: Tenant isolation is a critical requirement to ensure tenants do not receive traffic not intended for them. This isolation not only ensures privacy and security for the traffic generated by tenants, but also allows for the reuse of resources such as MAC and IP addresses.

Action

- Verify tenant isolation on Device R-2:

```
user@R2# run show bridge mac-table instance vs1-1
Routing instance : vs1-1
 Bridging domain : bd-1, VLAN : 1
   MAC                 MAC      Logical          NH     RTR
   address             flags    interface        Index  ID
   10:00:01:00:00:01   D        ae0.0
   10:00:01:00:0f:a1   D        ae0.0
   10:00:02:00:00:01   D        lsi.1048577
   10:00:02:00:0f:a1   D        lsi.1048577

Routing instance : vs1-1
 Bridging domain : bd-2, VLAN : 2
   MAC                 MAC      Logical          NH     RTR
   address             flags    interface        Index  ID
   10:00:01:00:00:02   D        ae0.1
   10:00:01:00:0f:a2   D        ae0.1
   10:00:02:00:00:02   D        lsi.1048577
   10:00:02:00:0f:a2   D        lsi.1048577
```

- Verify tenant isolation on Device R-2:

```
user@R2# run show bridge mac-table instance vs1-2
Routing instance : vs1-2
 Bridging domain : bd-1, VLAN : 3
   MAC                 MAC      Logical          NH     RTR
   address             flags    interface        Index  ID
   10:00:01:00:00:03   D        ae0.2
   10:00:01:00:0f:a3   D        ae0.2
   10:00:02:00:00:03   D        lsi.1049411
   10:00:02:00:0f:a3   D        lsi.1049411

Routing instance : vs1-2
 Bridging domain : bd-2, VLAN : 4
   MAC                 MAC      Logical          NH     RTR
   address             flags    interface        Index  ID
   10:00:01:00:00:04   D        ae0.3
   10:00:01:00:0f:a4   D        ae0.3
   10:00:02:00:00:04   D        lsi.1049411
   10:00:02:00:0f:a4   D        lsi.1049411
```

- Verify tenant isolation on Device R-2:

```
user@R2# run show route table vs1-1.l2vpn.0

vs1-1.l2vpn.0: 3 destinations, 3 routes (3 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

1000:1:1:1/96
                   *[BGP/170] 3d 02:14:00, localpref 65535, from 10.255.35.128
                      AS path: I, validation-state: unverified
                    > to 3.1.0.5 via xe-2/2/1.0, label-switched-path to-r0
1000:8001:2:1/96
                   *[L2VPN/170/-65536] 3d 02:23:27, metric2 1
                      Indirect
1000:16001:3:1/96
                   *[BGP/170] 3d 02:07:52, localpref 1, from 10.255.36.216
                      AS path: I, validation-state: unverified
                    > to 3.4.0.5 via xe-1/3/1.0, label-switched-path to-r3
```

- Verify tenant isolation on Device R-2:

```
user@R2# run show route table vs1-2.l2vpn.0

vs1-2.l2vpn.0: 3 destinations, 3 routes (3 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

1000:2:2:1/96
                   *[BGP/170] 3d 02:14:16, localpref 65535, from 10.255.35.128

                      AS path: I, validation-state: unverified
                    > to 3.1.0.5 via xe-2/2/1.0, label-switched-path to-r0
1000:8002:3:1/96
                   *[L2VPN/170/-65536] 3d 02:23:40, metric2 1
                      Indirect
1000:16002:4:1/96
                   *[BGP/170] 3d 02:08:07, localpref 1, from 10.255.36.216
                      AS path: I, validation-state: unverified
                    > to 3.4.0.5 via xe-1/3/1.0, label-switched-path to-r3
```

- Verify tenant isolation on Device R-3:

```
user@R3# run show route table vs1-1.l2vpn.0

vs1-1.l2vpn.0: 3 destinations, 3 routes (3 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

1000:1:1:1/96
                   *[BGP/170] 3d 02:09:33, localpref 65535, from 10.255.35.128

                      AS path: I, validation-state: unverified
                    > to 3.3.0.9 via xe-3/1/2.0, label-switched-path to-r0
1000:8001:2:1/96
                   *[BGP/170] 3d 02:08:31, localpref 65535, from 10.255.32.193

                      AS path: I, validation-state: unverified
                    > to 3.7.0.9 via xe-3/1/3.0, label-switched-path to-r2
1000:16001:3:1/96
                   *[L2VPN/170/-2] 3d 02:19:23, metric2 1
                      Indirect
```

- Verify tenant isolation on Device R-3:

```
user@R3# run show route table vs1-2.l2vpn.0

vs1-2.l2vpn.0: 3 destinations, 3 routes (3 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

1000:2:2:1/96
                   *[BGP/170] 3d 02:09:47, localpref 65535, from 10.255.35.128

                      AS path: I, validation-state: unverified
                    > to 3.3.0.9 via xe-3/1/2.0, label-switched-path to-r0
1000:8002:3:1/96
                   *[BGP/170] 3d 02:08:45, localpref 65535, from 10.255.32.193

                      AS path: I, validation-state: unverified
                    > to 3.7.0.9 via xe-3/1/3.0, label-switched-path to-r2
1000:16002:4:1/96
                   *[L2VPN/170/-2] 3d 02:19:36, metric2 1
                      Indirect
```

Meaning Indicates that eparate MAC tables are maintained for routing instance vs1-1 and vs1-2 and also separate MAC tables are maintained based on bridge domains bd-1 and bd-2 in routing instance vs1-1 and bridge domains bd-1 and bd-2 in routing instance vs1-2.

Verifying CoS Application to Each Core Facing Interface

Purpose Verify the CoS configuration on each core facing interface for each router.

Action

- Verify the CoS classification configuration for all core facing devices:

NOTE: Only the configuration for Device R-2 is shown.

```
user@R2# run show class-of-service classifier
Classifier: DCN-exp-classifier, Code point type: exp, Index: 42002
  Code point         Forwarding class                  Loss priority
  000                bulk                              low
  001                bulk                              high
  010                business-tier2                    low
  011                business-tier1                    low
  100                interactive-video                 low
  101                voice                             low
  110                network-control                   low
  111                network-control                   high

Classifier: DCN-dot1p-classifier, Code point type: ieee-802.1, Index: 43281
  Code point         Forwarding class                  Loss priority
  000                bulk                              low
  001                bulk                              high
  010                business-tier2                    low
  011                business-tier1                    low
  100                interactive-video                 low
  101                voice                             low
  110                network-control                   low
  111                network-control                   high

Classifier: DCN-TOS-classifier, Code point type: inet-precedence, Index: 56393

  Code point         Forwarding class                  Loss priority
  000                bulk                              low
  001                bulk                              high
  010                business-tier2                    low
  011                business-tier1                    low
  100                interactive-video                 low
  101                voice                             low
  110                network-control                   low
  111                network-control                   high
```

- Verify the CoS rewrite rule configuration for all core facing devices:

NOTE: Only the configuration for Device R-2 is shown.

```
user@R2# run show class-of-service rewrite-rule
```

```
Rewrite rule: DCN-exp-rewrite, Code point type: exp, Index: 2063
  Forwarding class                Loss priority       Code point
  bulk                            low                 000
  bulk                            high                001
  interactive-video               low                 100
  voice                           low                 101
  network-control                 low                 110
  network-control                 high                111
  business-tier1                  low                 011
  business-tier2                  low                 010

Rewrite rule: DCN-dot1p-rewrite, Code point type: ieee-802.1, Index: 11865
  Forwarding class                Loss priority       Code point
  bulk                            low                 000
  bulk                            high                001
  interactive-video               low                 100
  voice                           low                 101
  network-control                 low                 110
  network-control                 high                111
  business-tier1                  low                 011
  business-tier2                  low                 010

Rewrite rule: DCN-TOS-rewrite, Code point type: inet-precedence, Index: 28756
  Forwarding class                Loss priority       Code point
  bulk                            low                 000
  bulk                            high                001
  interactive-video               low                 100
  voice                           low                 101
  network-control                 low                 110
  network-control                 high                111
  business-tier1                  low                 011
  business-tier2                  low                 010
```

Meaning Indicates how code points are configured for each forwarding class and how loss priority is given to each forwarding class. Similarly, the re-write rules indicate how each forwarding class is given a loss priority value and assigned code points.

Verifying CoS Configuration on Each Device

Purpose Verify the CoS configuration on each interface for each router.

Action

- Verify the CoS configuration on Device R-3:

```
user@R3# run show class-of-service classifier
Classifier: DCN-exp-classifier, Code point type: exp, Index: 42002
  Code point         Forwarding class                  Loss priority
  000                bulk                              low
  001                bulk                              high
  010                business-tier2                    low
  011                business-tier1                    low
  100                interactive-video                 low
  101                voice                             low
  110                network-control                   low
  111                network-control                   high

Classifier: DCN-dot1p-classifier, Code point type: ieee-802.1, Index: 43281
  Code point         Forwarding class                  Loss priority
  000                bulk                              low
```

```
  001               bulk                              high
  010               business-tier2                    low
  011               business-tier1                    low
  100               interactive-video                 low
  101               voice                             low
  110               network-control                   low
  111               network-control                   high

Classifier: DCN-TOS-classifier, Code point type: inet-precedence, Index: 56393

  Code point        Forwarding class                  Loss priority
  000               bulk                              low
  001               bulk                              high
  010               business-tier2                    low
  011               business-tier1                    low
  100               interactive-video                 low
  101               voice                             low
  110               network-control                   low
  111               network-control                   high
```

```
user@R2# run show class-of-service rewrite-rule
Rewrite rule: DCN-exp-rewrite, Code point type: exp, Index: 2063
  Forwarding class                  Loss priority       Code point
  bulk                              low                 000
  bulk                              high                001
  interactive-video                 low                 100
  voice                             low                 101
  network-control                   low                 110
  network-control                   high                111
  business-tier1                    low                 011
  business-tier2                    low                 010

Rewrite rule: DCN-dot1p-rewrite, Code point type: ieee-802.1, Index: 11865
  Forwarding class                  Loss priority       Code point
  bulk                              low                 000
  bulk                              high                001
  interactive-video                 low                 100
  voice                             low                 101
  network-control                   low                 110
  network-control                   high                111
  business-tier1                    low                 011
  business-tier2                    low                 010

Rewrite rule: DCN-TOS-rewrite, Code point type: inet-precedence, Index: 28756
  Forwarding class                  Loss priority       Code point
  bulk                              low                 000
  bulk                              high                001
  interactive-video                 low                 100
  voice                             low                 101
  network-control                   low                 110
  network-control                   high                111
  business-tier1                    low                 011
  business-tier2                    low                 010
```

- Verify the CoS configuration on Device R-4:

```
user@R4# run show class-of-service classifier
Classifier: DSCP-classifier, Code point type: dscp, Index: 53201
  Code point        Forwarding class                  Loss priority
  000000            be                                low
  000001            af                                low
  000010            ef                                low
```

```
  000011            nc                                low
  000100            be1                               low
  000101            af1                               high
  000110            ef1                               low
  000111            nc1                               high

Classifier: DCN-exp-classifier, Code point type: exp, Index: 52412
  Code point        Forwarding class                  Loss priority
  000               be                                low
  001               af                                low
  010               ef                                low
  011               nc                                low
  100               be1                               low
  101               af1                               high
  110               ef1                               low
  111               nc1                               high

Classifier: DCN-dot1p-classifier, Code point type: ieee-802.1, Index: 62046
  Code point        Forwarding class                  Loss priority
  000               be                                low
  001               af                                low
  010               ef                                low
  011               nc                                low
  100               be1                               low
  101               af1                               high
  110               ef1                               low
  111               nc1                               high

Classifier: DCN-TOS-classifier, Code point type: inet-precedence, Index: 24735

  Code point        Forwarding class                  Loss priority
  000               be                                low
  001               af                                low
  010               ef                                low
  011               nc                                low
  100               be1                               low
  101               af1                               high
  110               ef1                               low
  111               nc1                               high
user@R4# run show class-of-service rewrite-rule
Rewrite rule: DSCP-rewrite, Code point type: dscp, Index: 23007
  Forwarding class                  Loss priority       Code point
  be                                low                 000000
  be                                high                000000
  af                                low                 000001
  af                                high                000001
  ef                                low                 000010
  ef                                high                000010
  nc                                low                 000011
  nc                                high                000011
  be1                               low                 000100
  be1                               high                000100
  ef1                               low                 000110
  ef1                               high                000110
  nc1                               low                 000111
  nc1                               high                000111

Rewrite rule: DCN-exp-rewrite, Code point type: exp, Index: 36206
  Forwarding class                  Loss priority       Code point
  be                                low                 000
  be                                high                000
```

```
  af                                low                 001
  af                                high                001
  ef                                low                 010
  ef                                high                010
  nc                                low                 011
  nc                                high                011
  be1                               low                 100
  be1                               high                100
  af1                               low                 101
  af1                               high                101
  ef1                               low                 110
  ef1                               high                110
  nc1                               low                 111
  nc1                               high                111

Rewrite rule: DCN-dot1p-rewrite, Code point type: ieee-802.1, Index: 58451
  Forwarding class                  Loss priority       Code point
  be                                low                 000
  be                                high                000
  af                                low                 001
  af                                high                001
  ef                                low                 010
  ef                                high                010
  nc                                low                 011
  nc                                high                011
  be1                               low                 100
  be1                               high                100
  af1                               low                 101
  af1                               high                101
  ef1                               low                 110
  ef1                               high                110
  nc1                               low                 111
  nc1                               high                111

Rewrite rule: DCN-TOS-rewrite, Code point type: inet-precedence, Index: 30353
  Forwarding class                  Loss priority       Code point
  be                                low                 000
  be                                high                000
  af                                low                 001
  af                                high                001
  ef                                low                 010
  ef                                high                010
  nc                                low                 011
  nc                                high                011
  be1                               low                 100
  be1                               high                100
  af1                               low                 101
  af1                               high                101
  ef1                               low                 110
  ef1                               high                110
  nc1                               low                 111
  nc1                               high                111
```

- Verify the CoS configuration on Device R-5:

```
user@R5# run show class-of-service classifier
Classifier: DSCP-classifier, Code point type: dscp, Index: 53201
  Code point        Forwarding class                  Loss priority
  000000            be                                low
  000001            af                                low
  000010            ef                                low
  000011            nc                                low
```

```
  000100            be1                               low
  000101            af1                               high
  000110            ef1                               low
  000111            nc1                               high

Classifier: DCN-exp-classifier, Code point type: exp, Index: 52412
  Code point        Forwarding class                  Loss priority
  000               be                                low
  001               af                                low
  010               ef                                low
  011               nc                                low
  100               be1                               low
  101               af1                               high
  110               ef1                               low
  111               nc1                               high

Classifier: DCN-dot1p-classifier, Code point type: ieee-802.1, Index: 62046
  Code point        Forwarding class                  Loss priority
  000               be                                low
  001               af                                low
  010               ef                                low
  011               nc                                low
  100               be1                               low
  101               af1                               high
  110               ef1                               low
  111               nc1                               high

Classifier: DCN-TOS-classifier, Code point type: inet-precedence, Index: 24735

  Code point        Forwarding class                  Loss priority
  000               be                                low
  001               af                                low
  010               ef                                low
  011               nc                                low
  100               be1                               low
  101               af1                               high
  110               ef1                               low
  111               nc1                               high
```

```
user@R5# run show class-of-service rewrite-rule
Rewrite rule: DSCP-rewrite, Code point type: dscp, Index: 23007
  Forwarding class                  Loss priority       Code point
  be                                low                 000000
  be                                high                000000
  af                                low                 000001
  af                                high                000001
  ef                                low                 000010
  ef                                high                000010
  nc                                low                 000011
  nc                                high                000011
  be1                               low                 000100
  be1                               high                000100
  ef1                               low                 000110
  ef1                               high                000110
  nc1                               low                 000111
  nc1                               high                000111

Rewrite rule: DCN-exp-rewrite, Code point type: exp, Index: 36206
  Forwarding class                  Loss priority       Code point
  be                                low                 000
  be                                high                000
  af                                low                 001
```

```
  af                                  high                001
  ef                                  low                 010
  ef                                  high                010
  nc                                  low                 011
  nc                                  high                011
  be1                                 low                 100
  be1                                 high                100
  af1                                 low                 101
  af1                                 high                101
  ef1                                 low                 110
  ef1                                 high                110
  nc1                                 low                 111
  nc1                                 high                111

Rewrite rule: DCN-dot1p-rewrite, Code point type: ieee-802.1, Index: 58451
  Forwarding class                    Loss priority       Code point
  be                                  low                 000
  be                                  high                000
  af                                  low                 001
  af                                  high                001
  ef                                  low                 010
  ef                                  high                010
  nc                                  low                 011
  nc                                  high                011
  be1                                 low                 100
  be1                                 high                100
  af1                                 low                 101
  af1                                 high                101
  ef1                                 low                 110
  ef1                                 high                110
  nc1                                 low                 111
  nc1                                 high                111

Rewrite rule: DCN-TOS-rewrite, Code point type: inet-precedence, Index: 30353
  Forwarding class                    Loss priority       Code point
  be                                  low                 000
  be                                  high                000
  af                                  low                 001
  af                                  high                001
  ef                                  low                 010
  ef                                  high                010
  nc                                  low                 011
  nc                                  high                011
  be1                                 low                 100
  be1                                 high                100
  af1                                 low                 101
  af1                                 high                101
  ef1                                 low                 110
  ef1                                 high                110
  nc1                                 low                 111
  nc1                                 high                111
```

- Verify the CoS configuration on Device R-6:

```
user@R6# run show class-of-service classifier
Classifier: DSCP-classifier, Code point type: dscp, Index: 53201
  Code point        Forwarding class                  Loss priority
  000000            be                                low
  000001            af                                low
  000010            ef                                low
  000011            nc                                low
  000100            be1                               low
```

```
  000101            af1                               high
  000110            ef1                               low
  000111            nc1                               high

Classifier: DCN-exp-classifier, Code point type: exp, Index: 52412
  Code point        Forwarding class                  Loss priority
  000               be                                low
  001               af                                low
  010               ef                                low
  011               nc                                low
  100               be1                               low
  101               af1                               high
  110               ef1                               low
  111               nc1                               high

Classifier: DCN-dot1p-classifier, Code point type: ieee-802.1, Index: 62046
  Code point        Forwarding class                  Loss priority
  000               be                                low
  001               af                                low
  010               ef                                low
  011               nc                                low
  100               be1                               low
  101               af1                               high
  110               ef1                               low
  111               nc1                               high

Classifier: DCN-TOS-classifier, Code point type: inet-precedence, Index: 24735

  Code point        Forwarding class                  Loss priority
  000               be                                low
  001               af                                low
  010               ef                                low
  011               nc                                low
  100               be1                               low
  101               af1                               high
  110               ef1                               low
  111               nc1                               high
```

```
user@R6# run show class-of-service rewrite-rule
Rewrite rule: DSCP-rewrite, Code point type: dscp, Index: 23007
  Forwarding class                  Loss priority       Code point
  be                                low                 000000
  be                                high                000000
  af                                low                 000001
  af                                high                000001
  ef                                low                 000010
  ef                                high                000010
  nc                                low                 000011
  nc                                high                000011
  be1                               low                 000100
  be1                               high                000100
  ef1                               low                 000110
  ef1                               high                000110
  nc1                               low                 000111
  nc1                               high                000111

Rewrite rule: DCN-exp-rewrite, Code point type: exp, Index: 36206
  Forwarding class                  Loss priority       Code point
  be                                low                 000
  be                                high                000
  af                                low                 001
  af                                high                001
```

```
  ef                                low                 010
  ef                                high                010
  nc                                low                 011
  nc                                high                011
  be1                               low                 100
  be1                               high                100
  af1                               low                 101
  af1                               high                101
  ef1                               low                 110
  ef1                               high                110
  nc1                               low                 111
  nc1                               high                111

Rewrite rule: DCN-dot1p-rewrite, Code point type: ieee-802.1, Index: 58451
  Forwarding class                  Loss priority       Code point
  be                                low                 000
  be                                high                000
  af                                low                 001
  af                                high                001
  ef                                low                 010
  ef                                high                010
  nc                                low                 011
  nc                                high                011
  be1                               low                 100
  be1                               high                100
  af1                               low                 101
  af1                               high                101
  ef1                               low                 110
  ef1                               high                110
  nc1                               low                 111
  nc1                               high                111

Rewrite rule: DCN-TOS-rewrite, Code point type: inet-precedence, Index: 30353
  Forwarding class                  Loss priority       Code point
  be                                low                 000
  be                                high                000
  af                                low                 001
  af                                high                001
  ef                                low                 010
  ef                                high                010
  nc                                low                 011
  nc                                high                011
  be1                               low                 100
  be1                               high                100
  af1                               low                 101
  af1                               high                101
  ef1                               low                 110
  ef1                               high                110
  nc1                               low                 111
  nc1                               high                111
```

- Verify the CoS configuration on Device R-7:

```
user@R7# run show class-of-service classifier
Classifier: DSCP-classifier, Code point type: dscp, Index: 53201
  Code point        Forwarding class                  Loss priority
  000000            be                                low
  000001            af                                low
  000010            ef                                low
  000011            nc                                low
  000100            be1                               low
  000101            af1                               high
```

```
  000110            ef1                               low
  000111            nc1                               high

Classifier: DCN-exp-classifier, Code point type: exp, Index: 52412
  Code point        Forwarding class                  Loss priority
  000               be                                low
  001               af                                low
  010               ef                                low
  011               nc                                low
  100               be1                               low
  101               af1                               high
  110               ef1                               low
  111               nc1                               high

Classifier: DCN-dot1p-classifier, Code point type: ieee-802.1, Index: 62046
  Code point        Forwarding class                  Loss priority
  000               be                                low
  001               af                                low
  010               ef                                low
  011               nc                                low
  100               be1                               low
  101               af1                               high
  110               ef1                               low
  111               nc1                               high

Classifier:DCN-TOS-classifier, Code point type: inet-precedence, Index: 24735
  Code point        Forwarding class                  Loss priority
  000               be                                low
  001               af                                low
  010               ef                                low
  011               nc                                low
  100               be1                               low
  101               af1                               high
  110               ef1                               low
  111               nc1                               high
```

```
user@R7# run show class-of-service rewrite-rule
Rewrite rule: DSCP-rewrite, Code point type: dscp, Index: 23007
  Forwarding class                  Loss priority       Code point
  be                                low                 000000
  be                                high                000000
  af                                low                 000001
  af                                high                000001
  ef                                low                 000010
  ef                                high                000010
  nc                                low                 000011
  nc                                high                000011
  be1                               low                 000100
  be1                               high                000100
  ef1                               low                 000110
  ef1                               high                000110
  nc1                               low                 000111
  nc1                               high                000111

Rewrite rule: DCN-exp-rewrite, Code point type: exp, Index: 36206
  Forwarding class                  Loss priority       Code point
  be                                low                 000
  be                                high                000
  af                                low                 001
  af                                high                001
  ef                                low                 010
  ef                                high                010
```

```
  nc                                low             011
  nc                                high            011
  be1                               low             100
  be1                               high            100
  af1                               low             101
  af1                               high            101
  ef1                               low             110
  ef1                               high            110
  nc1                               low             111
  nc1                               high            111

Rewrite rule: DCN-dot1p-rewrite, Code point type: ieee-802.1, Index: 58451
  Forwarding class                  Loss priority   Code point
  be                                low             000
  be                                high            000
  af                                low             001
  af                                high            001
  ef                                low             010
  ef                                high            010
  nc                                low             011
  nc                                high            011
  be1                               low             100
  be1                               high            100
  af1                               low             101
  af1                               high            101
  ef1                               low             110
  ef1                               high            110
  nc1                               low             111
  nc1                               high            111

Rewrite rule: DCN-TOS-rewrite, Code point type: inet-precedence, Index: 30353
  Forwarding class                  Loss priority   Code point
  be                                low             000
  be                                high            000
  af                                low             001
  af                                high            001
  ef                                low             010
  ef                                high            010
  nc                                low             011
  nc                                high            011
  be1                               low             100
  be1                               high            100
  af1                               low             101
  af1                               high            101
  ef1                               low             110
  ef1                               high            110
  nc1                               low             111
  nc1                               high            111
```

Meaning Indicates how the different classifiers and rewrite rules are configured on each network node in the topology.

Example: Configuring an Advanced Layer 2 Cloud Data Center Customer Deployment

The deployment scenario in this example deals with the connectivity and data forwarding requirement from POD 1 to POD 2 and vice versa. This can be achieved by placing the Layer 2 interfaces coming from POD 1 and POD 2 to the edge to be in the same routing instance and bridge domains.

The following sections explain the advanced Layer 2 configuration in more detail:

Requirements

Table 4 on page 19 lists the hardware used on each node/device in this example.

Table 6: Node / Device Hardware

Node/Device	Hardware
Remote Provider Edge Router (R-0)	Chassis: MX480 RE0: RE-S-1800x4 RE1: RE-S-1800x4 FPC0: MPCE Type 2 3D EQ FPC5: MPC 3D 16x 10GE
MPLS Provider Router (R-1)	Chassis: MX480 RE0: RE-S-2000 RE1: NONE FPC0: MPC 3D 16x 10GE FPC4: MPC 3D 16x 10GE FPC5: MPC Type 2 3D EQ
Data Center Edge Router (R-2)	Chassis: MX480 RE0: RE-S-1800x4 RE1: RE-S-1800x4 FPC0: MPC Type 2 3D EQ FPC1: MPC 3D 16x 10GE FPC2: MPC Type 2 3D EQ FPC3: MPC 3D 16x 10GE

Table 6: Node / Device Hardware *(continued)*

Node/Device	Hardware
Data Center Edge Router (R-3)	Chassis: MX960 RE0: RE-S-1800x4 RE1: RE-S-1800x4 FPC0: MPC 3D 16x 10GE FPC1: MPC Type 2 3D EQ FPC2: MPC 3D 16x 10GE FPC3: MPC 3D 16x 10GE FPC4: MPC 3D 16x 10GE FPC5: MPC Type 2 3D EQ
Data Center Core Router (R-4)	Chassis: MX960 RE0: RE-S-1800x4 RE1: RE-S-1800x4 FPC0: MPC 3D 16x 10GE FPC2: MPC 3D 16x 10GE FPC3: MPC 3D 16x 10GE FPC4: MPC Type 2 3D EQ FPC5: MPC Type 2 3D EQ FPC7: MPC Type 2 3D EQ FPC8: MPC 3D 16x 10GE FPC9: MPC Type 2 3D EQ

Table 6: Node / Device Hardware *(continued)*

Node/Device	Hardware
Data Center Core Router (R-5)	Chassis: MX960 RE0: RE-S-1800x4 RE1: RE-S-1800x4 FPC1: MPC Type 2 3D EQ FPC2: MPC 3D 16x 10GE FPC4: MPC Type 2 3D EQ FPC5: MPC Type 2 3D EQ FPC7: MPC Type 2 3D EQ FPC9: MPC Type 2 3D EQ FPC10: MPC Type 2 3D EQ FPC11: MPC 3D 16x 10GE
Data Center Core Router (R-6)	Chassis: MX480 RE0: RE-S-2000 RE1: NONE FPC0: DPCE 20x 1GE R EQ FPC2: MS-DPC FPC3: MPC Type 2 3D EQ
Data Center Core Router (R-7)	Chassis: MX240 RE0: RE-S-2000 RE1: RE-S-2000 FPC1: DPCE 20x 1GE R EQ FPC2: MS-DPC
Top Of Racks (TORs) (R-8 through R-13)	Chassis: EX4500-40F RE0: EX4500-40F RE1: NONE FPC0: EX4500-40F

All MX Series and EX Series devices in this example use Juniper Networks Junos OS Release 12.3R4. Table 7 on page 79 lists the scaling values used in configuring each device.

Table 7: Node / Device Scaling Targets

Node/Device	Targeted Feature Scale Values
Remote Provider Edge Router	Interfaces: ~25K IFL Protocols: OSPF - 8, OSPF3 - 8, IS-IS -8, BGP - 2, RSVP -4 Sessions, MPLS LSP - 2 Ingress LSPs + 2 Egress LSPs, BFD -22, VLAN -(1-4094) X 8 Services: VPLS - 4002, VRF - 4K, BD - 8012
MPLS Provider Router	Interfaces: 42 IFL Protocols: OSPF - 24, OSPF3 - 24, IS-IS - 24 , BFD - 48, RSVP LSP -4 Transit LSP
Data Center Edge Router	Interfaces: ~48630 IFL (8K IRB), AE - 8, MC-AE - 8 Protocols: OSPF - 8, OSPF3 - 8, IS-IS - 8, BFD – 23 sessions, MPLS -3 Ingress LSPs + 3 Egress LSPs, RSVP -3 Sessions, BGP - 3, VLAN -(1-4094) X 8, ICCP -1 Session Services: VPLS - 4002, VRF -4K, BD - 20200, MC-LAG active/standby - 3
Data Center Core Router	Interfaces: ~75 , AE - 8, MC-AE active/active - 8 Protocols: VLAN - (1-4094) X 8, ICCP - 1 Services: VIRTUAL-SWITCH -4
Top of Racks (ToRs)	Interfaces: ~10 IFL Protocols: VLAN -(1-4094) Services: BD (VLANs in EX) - (1-4094)

Before you configure the Layer 2 cloud data center customer:

- Make sure to configure loopback interface (lo0) on each routing device.

Before you configure the advanced Layer 2 cloud data center customer deployment center, be sure to perform the tasks defined the following sections:

- Example: Configuring a Simple Layer 2 Cloud Data Center Customer Deployment on a Juniper Networks MX Series Device on page 17

The following configuration example follows the same topology contained in the *Example: Configuring a Simple Layer 2 Cloud Data Center Customer Deployment on a Juniper Networks MX Series Device*. However, this configuration example will go further to also include detail on how to configure L2 access across PODs to provide transparent VM mobility.

Overview

This deployment scenario illustrates more clearly the ability of the cloud provider to assign core/access VLANs across PODs (in this example, ae0.400x and ae10.400x). All VLANs are assigned to a single bridge domain; thus, for the customer, each bridge domain appears to be part of the same VLAN. This is accomplished through the configuration of the vm-mobility routing instance and through the use of few bridge domains containing

10 interfaces each. The data from POD 1 can travel to the data center edge router (Device R-2 or Device R-3) and then divert to POD 2.

Topology

Figure 18 on page 80 details a logical view of the advanced deployment scenario topology.

Figure 18: Advanced deployment scenario topology (logical view)

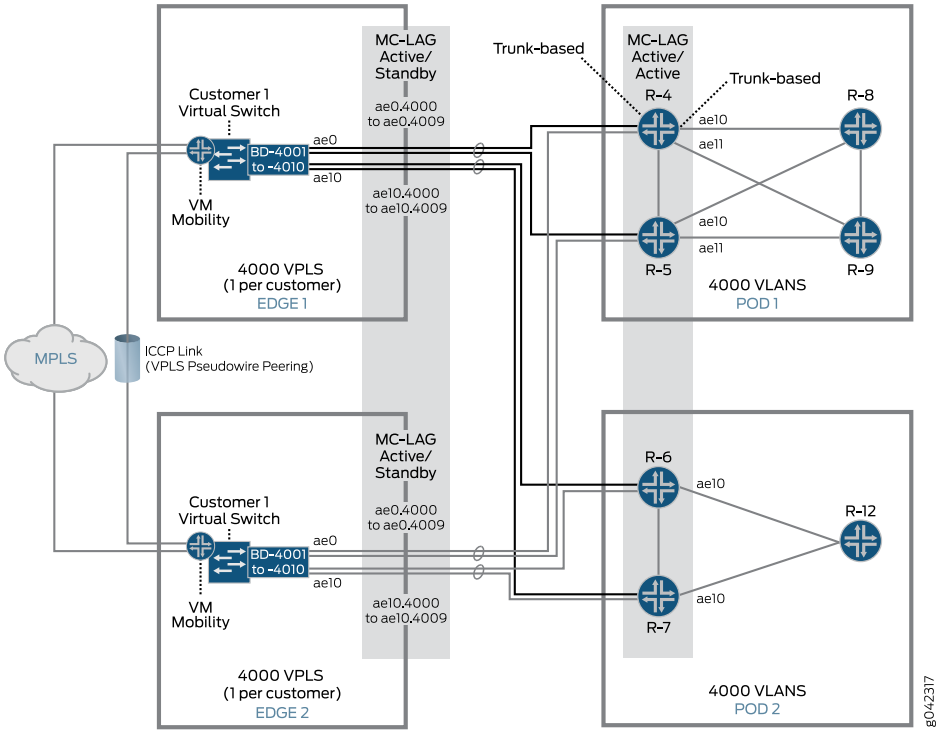


Figure 19 on page 81 shows how traffic flows between different PODs in the advanced deployment scenario topology.

Figure 19: Advanced deployment scenario topology (traffic flow between PODs)

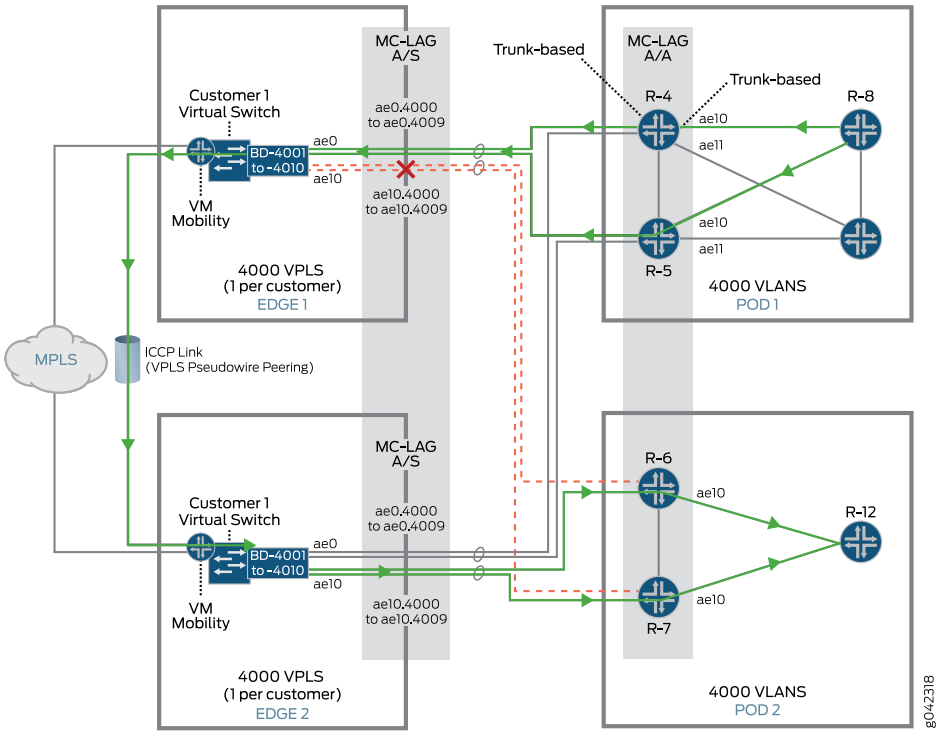
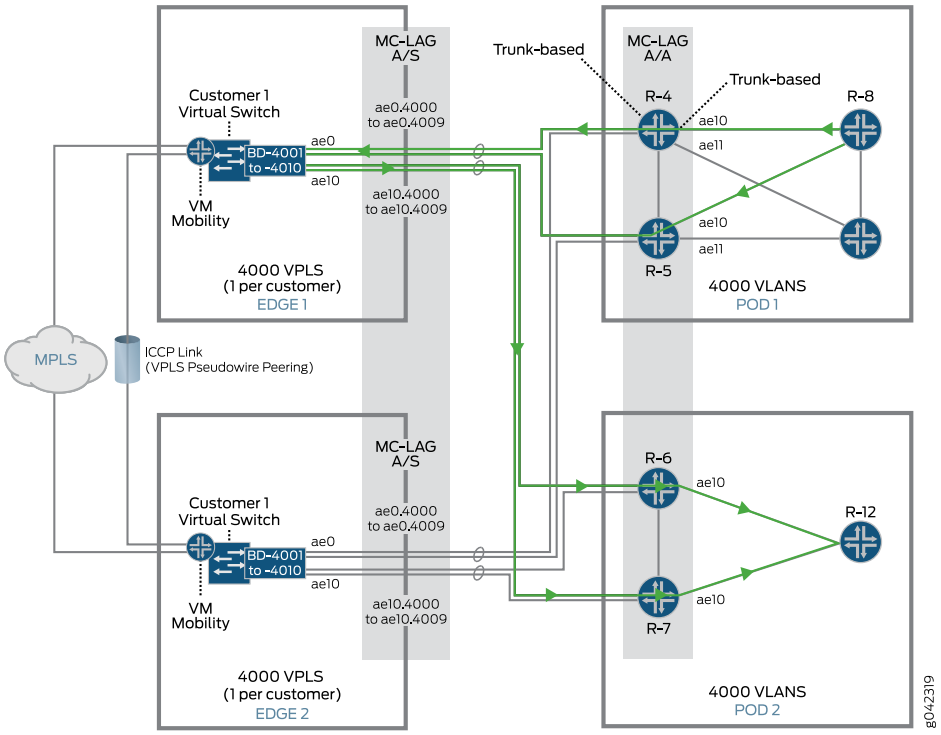


Figure 20 on page 82 shows the traffic flow in the advanced deployment scenario topology during interface ae10 failure on Device R-2.

Figure 20: Advanced deployment scenario topology (Device R-2 interface ae10 failure)



On Device R-0, interfaces xe-0/3/0.4000 to xe-0/3/0.4009 are placed in one bridge domain located in the vm-mobility routing instance. Interface xe-0/3/0 connects to router tester RT0. This interface sends traffic for POD 1 hosts. In addition, interfaces xe-0/2/0.4000 to xe-0/2/0.4009 are placed in another bridge domain in the same vm-mobility routing instance. This process is repeated, creating a total of 10 bridge domains, each with 10 interfaces, in the vm-mobility routing instance.

This configuration is mirrored on Device R-2 and Device R-3, except that the interfaces connect to the core and/or aggregation routers. In this example, the core Interface is ae0 for POD 1 and ae10 for POD 2. These interfaces are also contained in the vm-mobility routing instance as well as in the appropriate bridge domains.

Configuring Device Interfaces

The following sections show how to configure the interfaces on each device used in the advanced deployment scenario:

- Configuring Device R-2 Interfaces on page 83
- Configuring Device R-3 Interfaces on page 83

Configuring Device R-2 Interfaces

Step-by-Step Procedure

To configure interfaces for Device R-2:

1. Configure an interface on Device R-2 for the vm-mobility routing instance that faces POD 1.

NOTE: Only two VLANs are shown in this example. You must repeat this configuration for all logical interfaces in the bridge domain.

```
[edit]
user@host# set interfaces ae0 unit 3999 vlan-id 4000
user@host# set interfaces ae0 unit 4000 encapsulation vlan-bridge
user@host# set interfaces ae0 unit 4000 vlan-id 4001
```

2. Configure an interface on Device R-2 for the vm-mobility routing instance that faces POD 2.

NOTE: Only two VLANs of the ten that were provisioned are shown in this example. You must repeat this configuration for all logical interfaces in the bridge domain.

```
[edit]
user@host# set interfaces ae10 unit 3999 vlan-id 4000
user@host# set interfaces ae10 unit 4000 encapsulation vlan-bridge
user@host# set interfaces ae10 unit 4000 vlan-id 4001
:
user@host# set interfaces ae10 unit 4008 vlan-id 4009
```

Configuring Device R-3 Interfaces

Step-by-Step Procedure

To configure interfaces for Device R-3:

1. Configure an interface on Device R-3 for the vm-mobility routing instance that faces POD 1.

NOTE: Only two VLANs are shown in this example. You must repeat this configuration for all logical interfaces in the bridge domain.

```
[edit]
user@host# set interfaces ae0 unit 3999 vlan-id 4000
user@host# set interfaces ae0 unit 4000 encapsulation vlan-bridge
user@host# set interfaces ae0 unit 4000 vlan-id 4001
:
:
user@host# set interfaces ae0 unit 4008 vlan-id 4009
```

2. Configure an interface on Device R-3 for the vm-mobility routing instance that faces POD 2.

NOTE: Only two VLANs are shown in this example. You must repeat this configuration for all logical interfaces in the bridge domain.

```
[edit]
user@host# set interfaces ae10 unit 3999 encapsulation vlan-bridge
user@host# set interfaces ae10 unit 3999 vlan-id 4000
user@host# set interfaces ae10 unit 4000 encapsulation vlan-bridge
user@host# set interfaces ae10 unit 4000 vlan-id 4001
```

Configuring VPLS

Step-by-Step Procedure

This section describes how to configure the vm-mobility routing instance on Device R-2 for inter-POD connectivity. This configuration differs from the simple deployment scenario as follows:

- All VLANs are normalized to VLAN ID 4001.
- All interfaces configured for a specific POD VLAN are normalized to VLAN ID 4001 (the bridge domain VLAN ID).

This configuration enables VLANs in different PODs to communicate. Traffic within the same POD VLAN is still switched locally at the core and or aggregation layer and traffic between PODs or between access VLANs in a POD is switched at the data center edge within the bridge domain.

To configure VPLS:

1. Configure the vm-mobility routing instance on Device R-2 for inter-POD connectivity.

```
[edit]
user@host# set routing-instances vm-mobility instance-type virtual-switch
user@host# set routing-instances vm-mobility route-distinguisher 1000:64000
user@host# set routing-instances vm-mobility vrf-target target:1000:12000
user@host# set routing-instances vm-mobility protocols vpls no-tunnel-services
user@host# set routing-instances vm-mobility protocols vpls site 2 site-identifier
  2001
user@host# set routing-instances vm-mobility protocols vpls connectivity-type
  permanent
user@host# set routing-instances vm-mobility bridge-domains bd-4001-to-4010
  domain-type bridge
user@host# set routing-instances vm-mobility bridge-domains bd-4001-to-4010
  vlan-id 4001
user@host# set routing-instances vm-mobility bridge-domains bd-4001-to-4010
  interface ae0.4000
user@host# set routing-instances vm-mobility bridge-domains bd-4001-to-4010
  interface ae10.4000
user@host# set routing-instances vm-mobility bridge-domains bd-4001-to-4010
  interface ae0.4001
```

```
user@host# set routing-instances vm-mobility bridge-domains bd-4001-to-4010
  interface ae10.4001
user@host# set routing-instances vm-mobility bridge-domains bd-4001-to-4010
  interface ae0.4002
user@host# set routing-instances vm-mobility bridge-domains bd-4001-to-4010
  interface ae10.4002
user@host# set routing-instances vm-mobility bridge-domains bd-4001-to-4010
  interface ae0.4003
user@host# set routing-instances vm-mobility bridge-domains bd-4001-to-4010
  interface ae10.4003
user@host# set routing-instances vm-mobility bridge-domains bd-4001-to-4010
  interface ae0.4004
user@host# set routing-instances vm-mobility bridge-domains bd-4001-to-4010
  interface ae10.4004
user@host# set routing-instances vm-mobility bridge-domains bd-4001-to-4010
  interface ae0.4005
user@host# set routing-instances vm-mobility bridge-domains bd-4001-to-4010
  interface ae10.4005
user@host# set routing-instances vm-mobility bridge-domains bd-4001-to-4010
  interface ae0.4006
user@host# set routing-instances vm-mobility bridge-domains bd-4001-to-4010
  interface ae10.4006
user@host# set routing-instances vm-mobility bridge-domains bd-4001-to-4010
  interface ae0.4007
user@host# set routing-instances vm-mobility bridge-domains bd-4001-to-4010
  interface ae10.4007
user@host# set routing-instances vm-mobility bridge-domains bd-4001-to-4010
  interface ae0.4008
user@host# set routing-instances vm-mobility bridge-domains bd-4001-to-4010
  interface ae10.4008
user@host# set routing-instances vm-mobility bridge-domains bd-4001-to-4010
  interface ae0.4009
user@host# set routing-instances vm-mobility bridge-domains bd-4001-to-4010
  interface ae10.4009
```

2. Configure the vm-mobility routing instance on Device R-3 for inter-POD connectivity.

```
[edit]
user@host# set routing-instances vm-mobility instance-type virtual-switch
user@host# set routing-instances vm-mobility route-distinguisher 1000:64000
user@host# set routing-instances vm-mobility vrf-target target:1000:12000
user@host# set routing-instances vm-mobility protocols vpls no-tunnel-services
user@host# set routing-instances vm-mobility protocols vpls site 3 site-identifier
  2002
user@host# set routing-instances vm-mobility bridge-domains bd-4001-to-4010
  domain-type bridge
user@host# set routing-instances vm-mobility bridge-domains bd-4001-to-4010
  vlan-id 4001
user@host# set routing-instances vm-mobility bridge-domains bd-4001-to-4010
  interface ae0.4000
user@host# set routing-instances vm-mobility bridge-domains bd-4001-to-4010
  interface ae10.4000
user@host# set routing-instances vm-mobility bridge-domains bd-4001-to-4010
  interface ae0.4001user@host#
user@host# set routing-instances vm-mobility bridge-domains bd-4001-to-4010
  interface ae10.4001
```

```
user@host# set routing-instances vm-mobility bridge-domains bd-4001-to-4010
  interface ae0.4002
user@host# set routing-instances vm-mobility bridge-domains bd-4001-to-4010
  interface ae10.4002
user@host# set routing-instances vm-mobility bridge-domains bd-4001-to-4010
  interface ae0.4003
user@host# set routing-instances vm-mobility bridge-domains bd-4001-to-4010
  interface ae10.4003
user@host# set routing-instances vm-mobility bridge-domains bd-4001-to-4010
  interface ae0.4004
user@host# set routing-instances vm-mobility bridge-domains bd-4001-to-4010
  interface ae10.4004
user@host# set routing-instances vm-mobility bridge-domains bd-4001-to-4010
  interface ae0.4005
user@host# set routing-instances vm-mobility bridge-domains bd-4001-to-4010
  interface ae10.4005
user@host# set routing-instances vm-mobility bridge-domains bd-4001-to-4010
  interface ae0.4006
user@host# set routing-instances vm-mobility bridge-domains bd-4001-to-4010
  interface ae10.4006
user@host# set routing-instances vm-mobility bridge-domains bd-4001-to-4010
  interface ae0.4007
user@host# set routing-instances vm-mobility bridge-domains bd-4001-to-4010
  interface ae10.4007
user@host# set routing-instances vm-mobility bridge-domains bd-4001-to-4010
  interface ae0.4008
user@host# set routing-instances vm-mobility bridge-domains bd-4001-to-4010
  interface ae10.4008
user@host# set routing-instances vm-mobility bridge-domains bd-4001-to-4010
  interface ae0.4009
user@host# set routing-instances vm-mobility bridge-domains bd-4001-to-4010
  interface ae10.4009
```

Verification

The following sections provide examples used to verify the configuration in this example:

- Verifying VPLS Status on page 86
- Verifying Each Routing Instance has an Isolated Route Table on page 88
- Verifying Each Routing Instance has an Isolated MAC Table on page 90

Verifying VPLS Status

Purpose Verify the VPLS status on each router.

Action

- Verify the status of routing instance vm-mobility on router Device R-0.

```
user@R0# run show vpls connections instance vm-mobility
Instance: vm-mobility
  Local site: 1 (2000)
    connection-site           Type  St     Time last up          # Up trans
    2001                      rmt   Up     Nov  1 12:57:25 2013           1
      Remote PE: 10.255.32.193, Negotiated control-word: No
      Incoming label: 262153, Outgoing label: 8823
      Local interface: lsi.1051577, Status: Up, Encapsulation: VPLS
```

```
    Description: Intf - vpls vm-mobility local site 2000 remote site 2001
  2002                    rmt   Up     Nov  1 13:02:14 2013           1
    Remote PE: 10.255.36.216, Negotiated control-word: No
    Incoming label: 262154, Outgoing label: 8567
    Local interface: lsi.1052579, Status: Up, Encapsulation: VPLS
      Description: Intf - vpls vm-mobility local site 2000 remote site 2002
```

- Verify the status of routing instance vm-mobility on router Device R-2.

```
user@R2# run show vpls connections instance vm-mobility
Instance: vm-mobility
  Local site: 2 (2001)
    connection-site           Type  St     Time last up          # Up trans
    2000                      rmt   Up     Nov  1 12:58:01 2013           1
      Remote PE: 10.255.35.128, Negotiated control-word: No
      Incoming label: 8503, Outgoing label: 67328
      Local interface: lsi.1051577, Status: Up, Encapsulation: VPLS
        Description: Intf - vpls vm-mobility local site 2001 remote site 2000
    2002                      rmt   Up     Nov  1 13:03:06 2013           1
      Remote PE: 10.255.36.216, Negotiated control-word: No
      Incoming label: 262154, Outgoing label: 67328
      Local interface: lsi.1052579, Status: Up, Encapsulation: VPLS
        Description: Intf - vpls vm-mobility local site 2001 remote site 2002
```

- Verify the status of routing instance vm-mobility on router Device R-3.

```
user@R3# run show vpls connections instance vm-mobility
Instance: vm-mobility
  Local site: 3 (2002)
    connection-site           Type  St     Time last up          # Up trans
    2000                      rmt   Up     Nov  1 13:03:21 2013           1
      Remote PE: 10.255.35.128, Negotiated control-word: No
      Incoming label: 8503, Outgoing label: 67329
      Local interface: lsi.1048599, Status: Up, Encapsulation: VPLS
        Description: Intf - vpls vm-mobility local site 2002 remote site 2000
    2001                      rmt   Up     Nov  1 13:03:57 2013           1
      Remote PE: 10.255.32.193, Negotiated control-word: No
      Incoming label: 262153, Outgoing label: 67329
      Local interface: lsi.1052579, Status: Up, Encapsulation: VPLS
        Description: Intf - vpls vm-mobility local site 2002 remote site 2001
```

Meaning Indicates the following:

- Interfaces ae0 on routers R-2 and R-3 are connected to interfaces ae0 on routers R-4 and R-5 and part of routing instance vs1-1 in POD 1.
- Interfaces ae10 on routers R-2 and R-3 are connected to interfaces ae10 on routers R-6 and R-7 and part of instance vs1-1 in POD 2.

The end-to-end connectivity of from POD 1 and POD 2 to the WAN is as follows:

- WAN POD 1: R-0 to R-1 to R-2 *and* R-3 to R-4 *and* R-5 to R-8
- WAN POD 2: R-0 to R-1 to R-2 *and* R-3 to R-6 *and* R-7 to R-12

To send data traffic from POD 1 to POD 2 (and vice-versa), a traffic stream on Device R-0 is configured to exchange MAC addresses from the end points of devices R-8 and R-12.

Verifying Each Routing Instance has an Isolated Route Table

Purpose Verify the each router contains an isolated route table.

Action

- View the route table of VPLS routing instance vm-mobility on router Device R-0.

```
user@R0# run show route table vm-mobility.l2vpn.0
vm-mobility.l2vpn.0: 9 destinations, 9 routes (9 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both
1000:64000:2001:1993/96
                   *[BGP/170] 3d 03:30:36, localpref 65535, from 10.255.32.193

                      AS path: I, validation-state: unverified
                    > to 3.3.0.2 via xe-5/1/0.0, label-switched-path to-r2
1000:64000:2001:2001/96
                   *[BGP/170] 3d 03:30:44, localpref 65535, from 10.255.32.193

                      AS path: I, validation-state: unverified
                    > to 3.3.0.2 via xe-5/1/0.0, label-switched-path to-r2
1000:64000:2002:1993/96
                   *[BGP/170] 3d 03:25:29, localpref 1, from 10.255.36.216
                      AS path: I, validation-state: unverified
                    > to 3.8.0.2 via xe-5/0/1.0, label-switched-path to-r3
1000:64000:2002:2001/96
                   *[BGP/170] 3d 03:25:29, localpref 1, from 10.255.36.216
                      AS path: I, validation-state: unverified
                    > to 3.8.0.2 via xe-5/0/1.0, label-switched-path to-r3
1000:10000:2001:1993/96
                   *[BGP/170] 3d 03:30:36, localpref 65535, from 10.255.32.193

                      AS path: I, validation-state: unverified
                    > to 3.3.0.2 via xe-5/1/0.0, label-switched-path to-r2
1000:10000:2001:2001/96
                   *[BGP/170] 3d 03:30:44, localpref 65535, from 10.255.32.193

                      AS path: I, validation-state: unverified
                    > to 3.3.0.2 via xe-5/1/0.0, label-switched-path to-r2
1000:18000:2002:1993/96
                   *[BGP/170] 3d 03:25:27, localpref 1, from 10.255.36.216
                      AS path: I, validation-state: unverified
                    > to 3.8.0.2 via xe-5/0/1.0, label-switched-path to-r3
1000:18000:2002:2001/96
                   *[BGP/170] 3d 03:25:29, localpref 1, from 10.255.36.216
                      AS path: I, validation-state: unverified
                    > to 3.8.0.2 via xe-5/0/1.0, label-switched-path to-r3
1000:64998:2000:2001/96
                   *[L2VPN/170/-101] 3d 03:38:02, metric2 1
                      Indirect
```

- View the route table of VPLS routing instance vm-mobility on router Device R-2.

```
user@R2 run show route table vm-mobility.l2vpn.0
vm-mobility.l2vpn.0: 8 destinations, 8 routes (8 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both
1000:64000:2001:1993/96
                   *[L2VPN/170/-65536] 3d 03:30:57, metric2 1
                      Indirect
1000:64000:2001:2001/96
                   *[L2VPN/170/-65536] 3d 03:40:29, metric2 1
                      Indirect
```

```
1000:64000:2002:1993/96
                   *[BGP/170] 3d 03:24:54, localpref 1, from 10.255.36.216
                      AS path: I, validation-state: unverified
                    > to 3.4.0.5 via xe-1/3/1.0, label-switched-path to-r3
1000:64000:2002:2001/96
                   *[BGP/170] 3d 03:24:54, localpref 1, from 10.255.36.216
                      AS path: I, validation-state: unverified
                    > to 3.4.0.5 via xe-1/3/1.0, label-switched-path to-r3
1000:18000:2002:1993/96
                   *[BGP/170] 3d 03:24:53, localpref 1, from 10.255.36.216
                      AS path: I, validation-state: unverified
                    > to 3.4.0.5 via xe-1/3/1.0, label-switched-path to-r3
1000:18000:2002:2001/96
                   *[BGP/170] 3d 03:24:53, localpref 1, from 10.255.36.216
                      AS path: I, validation-state: unverified
                    > to 3.4.0.5 via xe-1/3/1.0, label-switched-path to-r3
1000:2000:2000:2001/96
                   *[BGP/170] 3d 03:31:02, localpref 65535, from 10.255.35.128

                      AS path: I, validation-state: unverified
                    > to 3.1.0.5 via xe-2/2/1.0, label-switched-path to-r0
1000:64998:2000:2001/96
                   *[BGP/170] 3d 03:31:02, localpref 100, from 10.255.35.128
                      AS path: I, validation-state: unverified
                    > to 3.1.0.5 via xe-2/2/1.0, label-switched-path to-r0
```

- View the route table of VPLS routing instance vm-mobility on router Device R-3.

```
user@R3 run show route table vm-mobility.l2vpn.0
vm-mobility.l2vpn.0: 8 destinations, 8 routes (8 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both
1000:64000:2001:1993/96
                   *[BGP/170] 3d 03:25:02, localpref 65535, from 10.255.32.193

                      AS path: I, validation-state: unverified
                    > to 3.7.0.9 via xe-3/1/3.0, label-switched-path to-r2
1000:64000:2001:2001/96
                   *[BGP/170] 3d 03:25:02, localpref 65535, from 10.255.32.193

                      AS path: I, validation-state: unverified
                    > to 3.7.0.9 via xe-3/1/3.0, label-switched-path to-r2
1000:64000:2002:1993/96
                   *[L2VPN/170/-2] 3d 03:25:59, metric2 1
                      Indirect
1000:64000:2002:2001/96
                   *[L2VPN/170/-2] 3d 03:35:54, metric2 1
                      Indirect
1000:10000:2001:1993/96
                   *[BGP/170] 3d 03:25:02, localpref 65535, from 10.255.32.193

                      AS path: I, validation-state: unverified
                    > to 3.7.0.9 via xe-3/1/3.0, label-switched-path to-r2
1000:10000:2001:2001/96
                   *[BGP/170] 3d 03:25:02, localpref 65535, from 10.255.32.193

                      AS path: I, validation-state: unverified
                    > to 3.7.0.9 via xe-3/1/3.0, label-switched-path to-r2
1000:2000:2000:2001/96
                   *[BGP/170] 3d 03:26:04, localpref 65535, from 10.255.35.128

                      AS path: I, validation-state: unverified
                    > to 3.3.0.9 via xe-3/1/2.0, label-switched-path to-r0
```

```
1000:64998:2000:2001/96
                    *[BGP/170] 3d 03:26:04, localpref 100, from 10.255.35.128
                       AS path: I, validation-state: unverified
                     > to 3.3.0.9 via xe-3/1/2.0, label-switched-path to-r0
```

Meaning Indicates the isolation of the route table for the vm-mobility routing instance on each router (that is, router Device R-0, R-2 and R-3).

Verifying Each Routing Instance has an Isolated MAC Table

Purpose Verify the MAC table on the router devices.

Action

- View the route table of VPLS routing instance vm-mobility on router Device R-2.

  ```
  user@R2 run show bridge mac-table instance vm-mobility

  MAC flags (S -static MAC, D -dynamic MAC, L -locally learned, C -Control MAC
             SE -Statistics enabled, NM -Non configured MAC, R -Remote PE MAC)

  Routing instance : vm-mobility
   Bridging domain : bd-4001-to-4010, VLAN : 4001
     MAC                 MAC      Logical          NH     RTR
     address             flags    interface        Index  ID
     30:00:01:00:00:01   D        ae10.4000
     30:00:01:00:00:02   D        ae10.4001
     30:00:01:00:00:03   D        ae10.4002
     30:00:01:00:00:04   D        ae10.4003
     30:00:01:00:00:05   D        ae10.4004
     30:00:01:00:00:06   D        ae10.4005
     30:00:01:00:00:07   D        ae10.4006
     30:00:01:00:00:08   D        ae10.4007
     30:00:01:00:00:09   D        ae10.4008
     30:00:01:00:00:0a   D        ae10.4009
     30:00:02:00:00:01   D        ae0.4000
     30:00:02:00:00:02   D        ae0.4001
     30:00:02:00:00:03   D        ae0.4002
     30:00:02:00:00:04   D        ae0.4003
     30:00:02:00:00:05   D        ae0.4004
     30:00:02:00:00:06   D        ae0.4005
     30:00:02:00:00:07   D        ae0.4006
     30:00:02:00:00:08   D        ae0.4007
     30:00:02:00:00:09   D        ae0.4008
     30:00:02:00:00:0a   D        ae0.4009
  ```

- View the route table of VPLS routing instance vm-mobility on router Device R-3.

  ```
  user@R3 run show bridge mac-table instance vm-mobility
  ```

  No MAC Table as VPLS in not learning addresses in router Device R-3

Meaning The MAC table from the vm-mobility routing instance with bridge domains bd-4001-to-4010 indicates that interfaces from ae0.4000 through ae0.4009 contain learned MAC addresses from POD 1 and interfaces from ae10.4000 through ae10.4009 contain learned MAC addresses from POD 2.

Similarly, the device can learn different POD 1 and POD 2 MAC addresses from other logical interfaces configured on interfaces ae0 and ae10.

Example: Configuring a Traditional Remote PE VPLS Deployment

The deployment scenario in this example is configured when a provider has existing VPLS customers, possibly at multiple remote sites that have used a traditional VPLS configuration, that wish to interoperate with the tenant design detailed throughout this document (using virtual-switches and bridge domains on the MX-series). .

A traditional VPLS configuration looks similar to the following:

```
interfaces ge-1/0/0 {
    encapsulation flexible-ethernet-services;
    flexible-vlan-tagging;
    unit 1 {
        encapsulation vlan-vpls;
        vlan-id-range 1-1000;
    }
}
routing-instances {
    green {
        instance-type vpls;
        vlan-id all;
        interface ge-1/0/0.1;
        route-distinguisher x.x.x.x;
        vrf-target target:11111:1;
        protocols {
            vpls {
                site-range 10;
                site greenPE1 {
                    site-identifier 1;
                }
            }
        }
    }
}
```

The following sections define how to configure a traditional remote PE VPLS deployment scenario:

Requirements

Before you configure customers and the data center in a network containing a traditional remote PE VPLS deployment, be sure to review the tasks to configure the core/aggreagation and top-of-rack access switches that are detailed in the following sections:

- Example: Configuring a Simple Layer 2 Cloud Data Center Customer Deployment on a Juniper Networks MX Series Device on page 17

NOTE: The deployment topology is the same in all configuration scenarios within this document. The main difference within this particular configuration scenario will be what will be configured on the tenant virtual switches and VPLS instances on the data center edge switches, Device E1 and Device E2.

Overview

When using the virtual switch style configuration with all VLANs (or bridge domains), the system dynamically assigns an ID. This raises the question of how to define the outer VLAN tag such that the end-to-end Layer 2 service becomes active.

To address this question, you must configure the remote PE as described in this section. When router Device R-0 is representing any edge router that has existing VPLS deployments, the dy1-1 configuration on router Device R-0 can be as shown below, that will allow access to the data center.

NOTE: No bridge domain configuration exists in routing instance dy1-1.

When using a dynamic profile on the data center edge device as described in this section, because outer tags must match, any remote site that is using the virtual switch bridge domain style VPLS configuration must have a dynamic profile associated with the VPLS instance. In addition, if multiple remote sites access the data center customer, you must configure a VLAN ID list or VLAN ID range as the outer tag to allow access from any remote sites where the outer VLAN ID is different. This configuration would appear similar to the following:

```
dynamic-profiles {
  dy1-1 {
    interfaces {
      "$junos-interface-ifd-name" {
        unit "$junos-underlying-interface-unit" {
          vlan-id-list [ 4091 4093 ];
          family bridge {
            interface-mode trunk;
            inner-vlan-id-list [ 1 10 100 ];
          }
        }
```

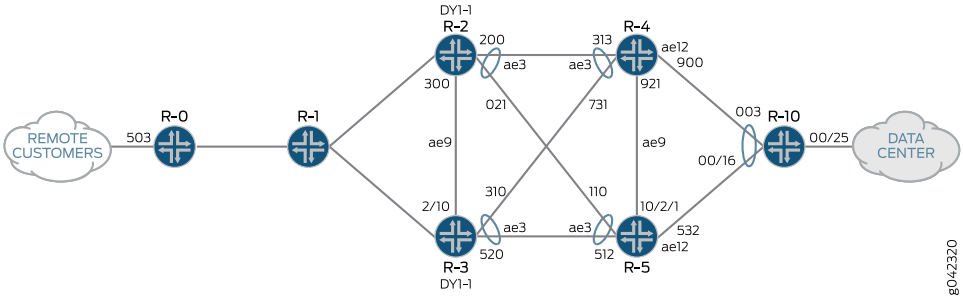
```
        }
      }
    }
  }
  dynamic-profiles {
    dy1-1 {
      interfaces {
        "$junos-interface-ifd-name" {
          unit "$junos-underlying-interface-unit" {
            vlan-id-range 4090-4094;
            family bridge {
              interface-mode trunk;
              inner-vlan-id-list [ 1 10 100 ];
            }
          }
        }
      }
    }
  }
```

NOTE: Notice the statement**vlan-id-list** in the first dynamic profile is the VLAN list. Then later in the next dynamic profile statement, the **vlan-id-range**command defines the Outer VLAN tags.

Topology

Figure 21 on page 93 shows the traditional remote PE VPLS deployment topology.

Figure 21: Traditional remote PE VPLS deployment topology



Configuring Remote PE Router Using Traditional Routing Instance and Interface Style VPLS

Step-by-Step Procedure

To configure customer CE-facing interfaces:

1. Configure the CE-facing interface.

```
[edit]
user@host# set interfaces xe-5/0/3 flexible-vlan-tagging
user@host# set interfaces xe-5/0/3 encapsulation flexible-ethernet-services
user@host# set interfaces xe-5/0/3 unit 0 encapsulation vlan-vpls <- Encapsulation
  of vlan-vpls on the CE interface
user@host# set interfaces xe-5/0/3 unit 0 vlan-id-list [110 100] <- Allowed customer
  VLANs – must match dynamic profile and remote
```

2. Configure the dynamic profile.

NOTE: The allowed inner VLANs must match the bridge domains remotely.

```
[edit]
user@host# set dynamic-profiles dy1-1 interfaces "$junos-interface-ifd-name" unit
  "$junos-u nderlying-interface-unit" vlan-id 4094
user@host# set dynamic-profiles dy1-1 interfaces "$junos-interface-ifd-name" unit
  "$junos-u nderlying-interface-unit" family bridge interface-mode trunk
user@host# set dynamic-profiles dy1-1 interfaces "$junos-interface-ifd-name" unit
  "$junos-u nderlying-interface-unit" family bridge inner-vlan-id-list 1
user@host# set dynamic-profiles dy1-1 interfaces "$junos-interface-ifd-name" unit
  "$junos-u nderlying-interface-unit" family bridge inner-vlan-id-list 10
user@host# set dynamic-profiles dy1-1 interfaces "$junos-interface-ifd-name" unit
  "$junos-u nderlying-interface-unit" family bridge inner-vlan-id-list 100
```

3. Configure the VPLS routing instance.

```
[edit]
user@host# set routing-instances dy1-1 instance-type vpls <- Routing instance type
  set to vpls
user@host# set routing-instances dy1-1 route-distinguisher 1000:6002
user@host# set routing-instances dy1-1 vrf-target target:1000:22001
user@host# set routing-instances dy1-1 interface xe-5/0/3.0
user@host# set routing-instances dy1-1 route-distinguisher 1000:64990
user@host# set routing-instances dy1-1 vrf-target target:1000:22001
user@host# set routing-instances dy1-1 protocols vpls no-tunnel-services
user@host# set routing-instances dy1-1 protocols vpls site 1 site-identifier 40010
user@host# set routing-instances dy1-1 protocols vpls associate-profile dy1-1
```

Configuring the Data Center Edge PE Router

Step-by-Step Procedure

This section describes how to configure the data center edge device, where bridge domains and virtual switches are used to enable VLAN normalization, inter-POD communication, MAC learning limits, and learning domain protections.

To configure the data center edge:

1. Configuring the CE interface for routing instance dy1-1 on router Device R-2.

```
[edit]
user@host# set interfaces ae3 flexible-vlan-tagging
user@host# set interfaces ae3 encapsulation flexible-ethernet-services
user@host# set interfaces ae3 unit 3999 encapsulation vlan-bridge
user@host# set interfaces ae3 unit 3999 vlan-id 1
user@host# set interfaces ae3 unit 4000 encapsulation vlan-bridge
user@host# set interfaces ae3 unit 4000 vlan-id 10
user@host# set interfaces ae3 unit 4001 encapsulation vlan-bridge
user@host# set interfaces ae3 unit 4001 vlan-id 100
```

2. Configure a dynamic profile to create dynamic interfaces with an outer tag of 4094 and an inner tag ranging from 1 through 4094 on router Device R-2.

NOTE: The outer VLAN tag set by the dynamic profile must match the remote side and the allowed inner VLAN tags must match bridge domains remotely.

```
[edit]
user@host# set dynamic-profiles dy1-1 interfaces "$junos-interface-ifd-name" unit
  "$junos-u nderlying-interface-unit" vlan-id 4094
user@host# set dynamic-profiles dy1-1 interfaces "$junos-interface-ifd-name" unit
  "$junos-u nderlying-interface-unit" family bridge interface-mode trunk
user@host# set dynamic-profiles dy1-1 interfaces "$junos-interface-ifd-name" unit
  "$junos-u nderlying-interface-unit" family bridge inner-vlan-id-list 1
user@host# set dynamic-profiles dy1-1 interfaces "$junos-interface-ifd-name" unit
  "$junos-u nderlying-interface-unit" family bridge inner-vlan-id-list 10
user@host# set dynamic-profiles dy1-1 interfaces "$junos-interface-ifd-name" unit
  "$junos-u nderlying-interface-unit" family bridge inner-vlan-id-list 100
```

3. Configure the VPLS routing instance (dy1-1) to deploy dynamic profiles for dual tag traffic on router Device R-2.

```
[edit]
user@host# set routing-instances dy1-1 instance-type virtual-switch
user@host# set routing-instances dy1-1 route-distinguisher 1000:6001
user@host# set routing-instances dy1-1 vrf-target target:1000:22001
user@host# set routing-instances dy1-1 protocols vpls no-tunnel-services
user@host# set routing-instances dy1-1 protocols vpls site 2 site-identifier 40011
user@host# set routing-instances dy1-1 protocols vpls associate-profile dy1-1
user@host# set routing-instances dy1-1 protocols vpls connectivity-type permanent
user@host# set routing-instances dy1-1 bridge-domains bd-1 domain-type bridge
user@host# set routing-instances dy1-1 bridge-domains bd-1 vlan-id 1
user@host# set routing-instances dy1-1 bridge-domains bd-1 interface ae0.3999
user@host# set routing-instances dy1-1 bridge-domains bd-1 bridge-options
  mac-table-size 1048575
user@host# set routing-instances dy1-1 bridge-domains bd-1 bridge-options
  interface-mac-limit 131071
user@host# set routing-instances dy1-1 bridge-domains bd-1 bridge-options
  mac-statistics
user@host# set routing-instances dy1-1 bridge-domains bd-10 domain-type bridge
user@host# set routing-instances dy1-1 bridge-domains bd-10 vlan-id 10
user@host# set routing-instances dy1-1 bridge-domains bd-1 interface ae0.4000
user@host# set routing-instances dy1-1 bridge-domains bd-10 bridge-options
  mac-table-size 1048575
user@host# set routing-instances dy1-1 bridge-domains bd-10 bridge-options
  interface-mac-limit 131071
user@host# set routing-instances dy1-1 bridge-domains bd-10 bridge-options
  mac-statistics
user@host# set routing-instances dy1-1 bridge-domains bd-100 domain-type bridge
user@host# set routing-instances dy1-1 bridge-domains bd-100 vlan-id 100
user@host# set routing-instances dy1-1 bridge-domains bd-1 interface ae0.4001
user@host# set routing-instances dy1-1 bridge-domains bd-100 bridge-options
  mac-table-size 1048575
user@host# set routing-instances dy1-1 bridge-domains bd-100 bridge-options
  interface-mac-limit 131071
```

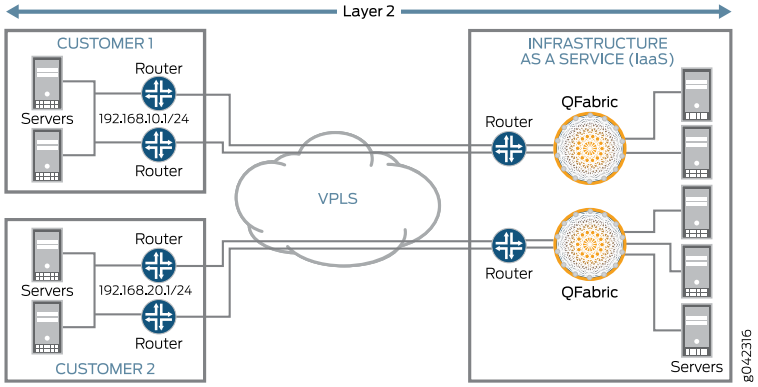
```
user@host# set routing-instances dy1-1 bridge-domains bd-100 bridge-options
    mac-statistics
```

Configuring Aggregated Ethernet Interfaces in Trunk Mode

The configuration in this document has focused on scaling the *tenant*, where the tenant is just a limited number of VLANs or bridge domains, each then assigned to a virtual switch instance and a corresponding VPLS instance. In this case, the aggregated Ethernet interfaces at the network edge must be “IFL-based” to provide the flexibility of assigning aggregated Ethernet units to different bridge domains.

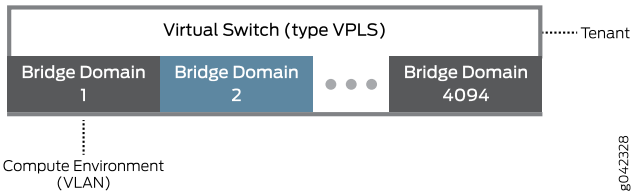
One negative impact of this configuration is that it consumes a large number of logical interfaces in the system. That is, each MC-LAG creates logical interfaces on each member link and also on the aggregated Ethernet logical interface. Due to this logical interface consumption, if a service provider wants to assign a large number of bridge domains to a single customer (that is, in a single virtual switch instance), or use this segmentation in a scaled data center interconnect (DCI) solution, he need not use logical interface-based trunks. Instead, the service provider can configure “trunk-mode” at the physical interface level and consume only a single logical interface in the system.

Figure 22: L2 Scaled Customer with Data Center Cloud Tenant Design



Each Customer (or remote data center VPLS instance) is represented as a virtual switch instance.

Figure 23: Customer VPLS Instance Presentation



This configuration method is already used on core and aggregation devices in this solution, but it is worth explaining how to configure it at the edge and map it to a virtual switch instance.

NOTE: The benefit of using this method of configuration when just transporting VLANs is that no bridge domains need be configured. Of course, though the configuration is minimized, using this configuration results in some benefits being unavailable (like the ease of normalization and the ease of use MAC limits).

- Configuring Trunk Mode Using Only Virtual Switch on page 97
- Configuring Trunk Mode Using a Dynamic Profile on page 98

Configuring Trunk Mode Using Only Virtual Switch

Step-by-Step Procedure

To configure aggregated Ethernet interfaces in trunk mode using only the virtual switch:

1. Configure the aggregated interface as a trunk.

```
[edit]
user@host# set interfaces ae3 flexible-vlan-tagging
user@host# set interfaces ae3 encapsulation flexible-ethernet-services
user@host# set interfaces ae3 unit 0 family bridge interface-mode trunk
user@host# set interfaces ae3 unit 0 family bridge vlan-id-list 1-4094
```

2. Configure routing instance vs1-1 by deploying dynamic profiles on Device R-2.

   Specific logical interface configurations are no longer required in each bridge domain.

   NOTE: Only 3 of 4094 bridge domains are shown.

```
[edit]
user@host# set routing-instances vs1-1 instance-type virtual-switch
user@host# set routing-instances vs1-1 interface ae3.0 Add this interface to the virtual
   switch. You can add more than one for interpod access.
user@host# set routing-instances vs1-1 route-distinguisher 1000:6001
user@host# set routing-instances vs1-1 vrf-target target:1000:22001
user@host# set routing-instances vs1-1 protocols vpls no-tunnel-services
user@host# set routing-instances vs1-1 protocols vpls site 2 site-identifier 40011
user@host# set routing-instances vs1-1 protocols vpls connectivity-type permanent
user@host# set routing-instances vs1-1 bridge-domains bd-1 domain-type bridge
user@host# set routing-instances vs1-1 bridge-domains bd-1 vlan-id 1
user@host# set routing-instances vs1-1 bridge-domains bd-1 bridge-options
   mac-table-size 1048575
user@host# set routing-instances vs1-1 bridge-domains bd-1 bridge-options
   interface-mac-limit 131071
user@host# set routing-instances vs1-1 bridge-domains bd-1 bridge-options
   mac-statistics
user@host# set routing-instances vs1-1 bridge-domains bd-10 domain-type bridge
user@host# set routing-instances vs1-1 bridge-domains bd-10 vlan-id 10
user@host# set routing-instances vs1-1 bridge-domains bd-10 bridge-options
   mac-table-size 1048575
user@host# set routing-instances vs1-1 bridge-domains bd-10 bridge-options
   interface-mac-limit 131071
```

```
user@host# set routing-instances vs1-1 bridge-domains bd-10 bridge-options
  mac-statistics
user@host# set routing-instances vs1-1 bridge-domains bd-100 domain-type bridge
user@host# set routing-instances vs1-1 bridge-domains bd-100 vlan-id 100
user@host# set routing-instances vs1-1 bridge-domains bd-100 bridge-options
  mac-table-size 1048575
user@host# set routing-instances vs1-1 bridge-domains bd-100 bridge-options
  interface-mac-limit 131071
user@host# set routing-instances vs1-1 bridge-domains bd-100 bridge-options
  mac-statistics
```

3. Configure a VPLS routing instance for the same customer on the remote PE, where the CE interface is now xe-5/0/3.

```
[edit]
user@host# set routing-instances dy1-1 instance-type vplsVPLS routing instance
  type
user@host# set routing-instances dy1-1 interface xe-5/0/3.0
user@host# set routing-instances dy1-1 route-distinguisher 1000:64990
user@host# set routing-instances dy1-1 vrf-target target:1000:22001
user@host# set routing-instances dy1-1 protocols vpls no-tunnel-services
user@host# set routing-instances dy1-1 protocols vpls site 1 site-identifier 40010
user@host# set routing-instances dy1-1 protocols vpls associate-profile dy1-1
```

4. Configure the remote PE interface to use VLAN-VPLS encapsulation.

```
[edit]
user@host# set interfaces xe-5/0/3 flexible-vlan-tagging
user@host# set interfaces xe-5/0/3 encapsulation flexible-ethernet-services
user@host# set interfaces xe-5/0/3 unit 0 interface-mode trunk
user@host# set interfaces xe-5/0/3 unit 0 encapsulation vlan-vpls VLAN-VPLS
  encapsulation on the CE interface
user@host# set interfaces xe-5/0/3 unit 0 vlan-id-range 1-4094
```

Configuring Trunk Mode Using a Dynamic Profile

Step-by-Step Procedure

To configure aggregated Ethernet interfaces in trunk mode using a dynamic profile:

1. Configure the aggregated interface as a trunk.

```
[edit]
user@host# set interfaces ae3 flexible-vlan-tagging
user@host# set interfaces ae3 encapsulation flexible-ethernet-services
user@host# set interfaces ae3 unit 0 family bridge interface-mode trunk
user@host# set interfaces ae3 unit 0 family bridge vlan-id-list 1-4094
```

2. Configure a dynamic profile with an outer VLAN tag of 100 and an inner VLAN tag range of 1 through 4094 on Device R-2.

NOTE: The dynamic profile is configured to transport all inner VLANs.

```
[edit]
user@host# set dynamic-profiles dy1-1 interfaces "$junos-interface-ifd-name" unit
  "$junos-underlying-interface-unit" vlan-id 100
```

```
user@host# set dynamic-profiles dy1-1 interfaces "$junos-interface-ifd-name" unit
  "$junos-underlying-interface-unit" family bridge interface-mode trunk
user@host# set dynamic-profiles dy1-1 interfaces "$junos-interface-ifd-name" unit
  "$junos-underlying-interface-unit" family bridge inner-vlan-id-list 1-4094
```

3. Configure routing instance vs1-1 by deploying dynamic profiles on Device R-2.

   Specific logical interface configurations are no longer required in each bridge domain.

   NOTE: Only 3 of 4094 bridge domains are shown.

```
[edit]
user@host# set routing-instances vs1-1 instance-type virtual-switch
user@host# set routing-instances vs1-1 interface ae3.0 Add this interface to the virtual
  switch. You can add more than one for interpod access.
user@host# set routing-instances vs1-1 route-distinguisher 1000:6001
user@host# set routing-instances vs1-1 vrf-target target:1000:22001
user@host# set routing-instances vs1-1 protocols vpls no-tunnel-services
user@host# set routing-instances vs1-1 protocols vpls site 2 site-identifier 40011
user@host# set routing-instances vs1-1 protocols vpls connectivity-type permanent
user@host# set routing-instances vs1-1 bridge-domains bd-1 domain-type bridge
user@host# set routing-instances vs1-1 bridge-domains bd-1 vlan-id 1
user@host# set routing-instances vs1-1 bridge-domains bd-1 bridge-options
  mac-table-size 1048575
user@host# set routing-instances vs1-1 bridge-domains bd-1 bridge-options
  interface-mac-limit 131071
user@host# set routing-instances vs1-1 bridge-domains bd-1 bridge-options
  mac-statistics
user@host# set routing-instances vs1-1 bridge-domains bd-10 domain-type bridge
user@host# set routing-instances vs1-1 bridge-domains bd-10 vlan-id 10
user@host# set routing-instances vs1-1 bridge-domains bd-10 bridge-options
  mac-table-size 1048575
user@host# set routing-instances vs1-1 bridge-domains bd-10 bridge-options
  interface-mac-limit 131071
user@host# set routing-instances vs1-1 bridge-domains bd-10 bridge-options
  mac-statistics
user@host# set routing-instances vs1-1 bridge-domains bd-100 domain-type bridge
user@host# set routing-instances vs1-1 bridge-domains bd-100 vlan-id 100
user@host# set routing-instances vs1-1 bridge-domains bd-100 bridge-options
  mac-table-size 1048575
user@host# set routing-instances vs1-1 bridge-domains bd-100 bridge-options
  interface-mac-limit 131071
user@host# set routing-instances vs1-1 bridge-domains bd-100 bridge-options
  mac-statistics
:
:
```

4. Configure a VPLS routing instance for the same customer on the remote PE, where the CE interface is now xe-5/0/3.

```
[edit]
set routing-instances dy1-1 instance-type vpls VPLS routing instance type
user@host# set routing-instances dy1-1 interface xe-5/0/3.0
user@host# set routing-instances dy1-1 route-distinguisher 1000:64990
```

```
user@host# set routing-instances dy1-1 vrf-target target:1000:22001
user@host# set routing-instances dy1-1 protocols vpls no-tunnel-services
user@host# set routing-instances dy1-1 protocols vpls site 1 site-identifier 40010
user@host# set routing-instances dy1-1 protocols vpls associate-profile dy1-1
```

5. Configure the remote PE interface to use VLAN-VPLS encapsulation.

```
[edit]
user@host# set interfaces xe-5/0/3 flexible-vlan-tagging
user@host# set interfaces xe-5/0/3 encapsulation flexible-ethernet-services
user@host# set interfaces xe-5/0/3 unit 0 interface-mode trunk
user@host# set interfaces xe-5/0/3 unit 0 encapsulation vlan-vpls VLAN-VPLS
  encapsulation on the CE interface
user@host# set interfaces xe-5/0/3 unit 0 vlan-id-range 1-4094
```

Verification

The following sections provide examples used to verify the configuration in this example:

Verify the Status of Each VPLS Routing Instance

Purpose Verify the status of VPLS routing instance dy1-1 on router Device R-2.

Action View VPLS routing instance dy1-1 status on router Device R-2.

```
user@R2 run show vpls connections instance dy1-1
Layer-2 VPN connections:
Instance: dy1-1
  Local site: 2 (40011)
    connection-site           Type  St     Time last up          # Up trans
    40010                     rmt   Up     Nov  1 12:57:33 2013           1
      Remote PE: 10.255.35.128, Negotiated control-word: No
      Incoming label: 262146, Outgoing label: 262147
      Local interface: lsi.1048576, Status: Up, Encapsulation: VPLS
      Dynamic profile: dy1-1   Dynamic profile deployed for this instance
        Description: Intf - vpls dy1-1 local site 40011 remote site 40010
    40012                     rmt   Up     Nov  1 13:03:06 2013           1
      Remote PE: 10.255.36.216, Negotiated control-word: No
      Incoming label: 262148, Outgoing label: 262147
      Local interface: lsi.1052578, Status: Up, Encapsulation: VPLS
      Dynamic profile: dy1-1   Dynamic profile deployed for this instance
        Description: Intf - vpls dy1-1 local site 40011 remote site 40012
```

Meaning Indicates that the VPLS routing instance is using dynamic profile dy1-1 for dual-tagging.

Verify the Installation of Bridge Domains

Purpose Verify bridge domains are installed in routing-instance dy1-1 on router Device R-2.

Action View each MAC table per bridge domain in routing instance dy1-1 on router Device R-2.

```
user@R2 run show bridge mac-table instance dy1-1
Routing instance : dy1-1
 Bridging domain : bd-1, VLAN : 1
   MAC                 MAC      Logical          NH     RTR
   address             flags    interface        Index  ID
   2c:6b:f5:47:0f:f0   D,SE     ae3.0
   50:20:10:00:00:01   D,SE     ae3.0
   50:20:10:00:0f:ff   D,SE     ae3.0

Routing instance : dy1-1
 Bridging domain : bd-10, VLAN : 10
   MAC                 MAC      Logical          NH     RTR
   address             flags    interface        Index  ID
   2c:6b:f5:47:0f:f0   D,SE     ae3.0
   50:20:10:00:00:0a   D,SE     ae3.0
   50:20:10:00:10:08   D,SE     ae3.0

Routing instance : dy1-1
 Bridging domain : bd-100, VLAN : 100
   MAC                 MAC      Logical          NH     RTR
   address             flags    interface        Index  ID
   2c:6b:f5:47:0f:f0   D,SE     ae3.0
   50:20:10:00:00:64   D,SE     ae3.0
   50:20:10:00:10:62   D,SE     ae3.0
```

Meaning Indicates that the appropriate MAC addresses are learned in each bridge domain configured under routing instance dy1-1.

Verify the Installation of Bridge Domains

Purpose Verify bridge domains are installed in routing-instance dy1-1 on router Device R-2. This verification shows all bridge domains configured in a VPLS instance alone with the index for each bridge domain in the VPLS routing instance.

Action View each bridge domain and routing instance association on router Device R-2.

```
user@R2 run show l2-learning instance dy1-1
Information for routing instance and bridge domain:
Inst  Logical    Routing              Bridging            Index  IRB     Flags
   BD
Type  System     Instance             Domain                     Index
   vlan
RTT   Default    dy1-1                                    4006
BD    Default    dy1-1                bd-1                2              SE
   1
BD    Default    dy1-1                bd-10               3              SE
   10
BD    Default    dy1-1                bd-100              4              SE
   100
```

Meaning Indicates that bridge domains bd-1, bd-10, bd-100, and so on are installed in routing instance dy1-1.

Verify MAC Tables

Purpose Verify each MAC table per bridge domain in routing-instance dy1-1 on router Device R-2.

Action View the MAC table on router Device R-2.

```
user@R2 run show bridge mac-table instance dy1-1
Routing instance : dy1-1
 Bridging domain : bd-4094, VLAN : 4094
   MAC                 MAC      Logical          NH     RTR
   address             flags    interface        Index  ID
   00:00:4f:c3:17:5b   D,SE     lsi.1052578
   00:00:4f:c3:17:5d   D,SE     lsi.1052578
   00:00:4f:c3:17:67   D,SE     lsi.1052578
   00:00:4f:c3:17:6b   D,SE     lsi.1052578
   00:00:4f:c3:33:f9   D,SE     lsi.1052578
   00:00:4f:c3:34:17   D,SE     lsi.1052578
   00:00:4f:c3:34:23   D,SE     lsi.1052578
   00:00:4f:c3:34:27   D,SE     lsi.1052578
    - - - - 1000's lines truncated - - -
   00:00:4f:c3:34:29   D,SE     lsi.1052578
   00:00:4f:c3:34:3b   D,SE     lsi.1052578
   00:00:4f:c3:34:3d   D,SE     lsi.1052578
   00:00:4f:c3:34:3f   D,SE     lsi.1052578
   00:00:4f:c3:34:51   D,SE     lsi.1052578
   00:00:4f:c3:34:79   D,SE     lsi.1052578
   00:00:4f:c3:34:7b   D,SE     lsi.1052578
   00:00:4f:c3:34:87   D,SE     lsi.1052578
   2c:6b:f5:47:0f:f0   D,SE     lsi.1052578
   50:10:10:00:00:01   D,SE     lsi.1048576
   50:10:10:00:00:02   D,SE     lsi.1048576
   50:10:10:00:00:03   D,SE     lsi.1048576
   50:10:10:00:00:04   D,SE     lsi.1048576
   50:10:10:00:00:05   D,SE     lsi.1048576
       - - - - 1000's lines truncated - - -
   50:10:10:00:00:57   D,SE     lsi.1048576
   50:10:10:00:00:58   D,SE     lsi.1048576
   50:10:10:00:00:59   D,SE     lsi.1048576
   50:10:10:00:00:5a   D,SE     lsi.1048576
   50:10:10:00:00:5b   D,SE     lsi.1048576
```

Meaning Indicates that VLAN ID 4094 is used as an outer tag for the dynamic profile configuration and that all MAC addresses are learned under bridge domain bd-4094.