CHAPTER 5

Extended Ethernet Segments over the WAN/MAN using EoMPLS

This chapter assists system engineers understand the various options available to extend an Ethernet segment using Ethernet over Multiprotocol Label Switching (EoMPLS) on the Cisco Sup720-3B. Special focus is placed on designs for geographically dispersed clusters.

Introduction

Several technologies can be used to extend Ethernet segments among multiple sites for different profiles of enterprises and networks.

This guide specifically focuses on enterprises that possess an MPLS metropolitan area network (MAN).

For disaster recovery purposes, data centers are hosted in multiple sites that are geographically distant and interconnected using a WAN or a MAN network, which in turn relies on dark fiber, dense wavelength division multiplexing (DWDM), and so on. The entire data center can be deployed using active/backup or active/active load distribution, ensuring that the critical applications such as CRM, ERP, and e-commerce are always available.

High availability (HA) clusters can often be geographically dispersed between various data centers. There are often two data centers; one active, and the other serving as a backup, with at least one member of the extended cluster in each data center. Most HA clusters require one or multiple VLANs to interconnect the nodes hosted in the various data centers. Generally, one VLAN is required for the heartbeat, also known as the private network, and another VLAN is used to carry the virtual IP address (VIP) managed by the application cluster itself, also known as the public network. One VLAN can provide both functions.

For more information, see Chapter 1, "Data Center High Availability Clusters," and Chapter 4, "FCIP over IP/MPLS Core."

EoMPLS can be used to carry Ethernet frames (native or dot1Q) across long distances. This chapter compares the available design options with EoMPLS that allow extending VLANs on top of an existing routed network.

Hardware Requirements

The following hardware is required:

- Aggregation layer—Cisco Catalyst 6000 Series switch or 7600 router with supervisor sup720-3B and Cisco Native IOS (tested with version 12.2(18) SX2)
- Line cards that support configurable MTU at aggregation or in the core
- Access layer—Any Layer 2 switch
- WAN-MAN edge (PE)—EoMPLS-aware devices (7x00, Cat65xx, 7600, 12000)

Note ISR routers do not yet support EoMPLS, but this feature will be available in the future. Layer 2 TPv3 can be an alternative to EoMPLS if ISR is used. Any type of links, such as Gigabit Ethernet, Fast Ethernet, POS, ATM, leased line, Frame Relay, and more can be used.

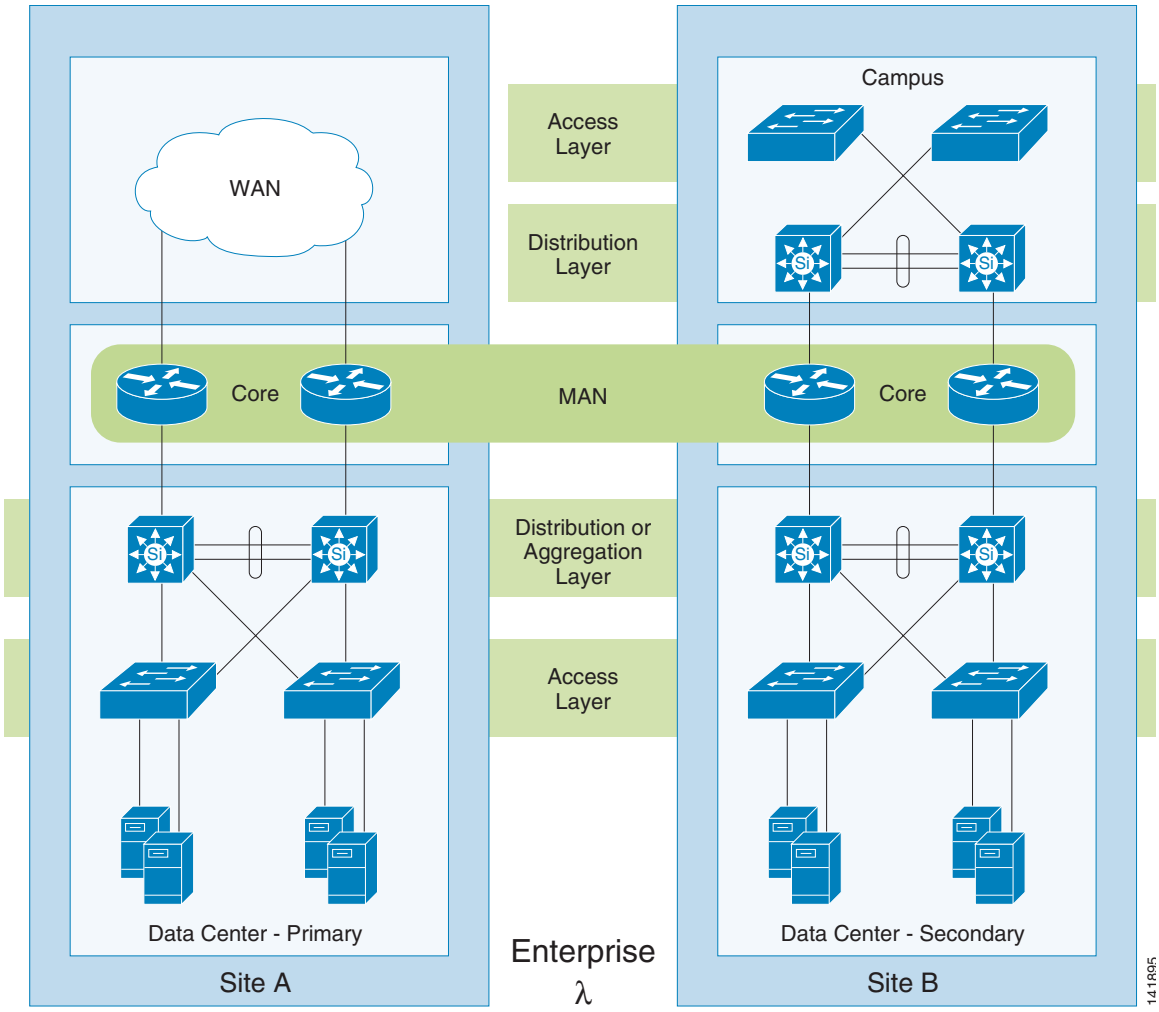
- WAN-MAN core (P)—MPLS router (7x00, 7600, 12000)

Note Sup720 with OSM or SIP cards is not within the scope of testing for this chapter.

Enterprise Infrastructure

Figure 5-1 shows a large enterprise network with two sites.

Figure 5-1 WAN MAN for Enterprise Core



The enterprise network is built around a hierarchical architecture of building blocks. Each building block supports specific functions, such as WAN access, campus, core, and data centers. The Intranet of large enterprises is often extended between multiple buildings or main offices using a MAN. The MAN is the extension of the core that exists at each site. For disaster recovery purposes, cluster members can be hosted by two or more data centers, which often requires that a Layer 2 network be deployed on top of a MAN.

EoMPLS Designs for Data Center Interconnectivity

During the last five years, MPLS has increasingly been the main WAN technology in the service provider arena. Multiple drivers have been key for this success. The two main ones are as follows:

- Switched IP-VPN—MPLS has given service providers the capability to create virtual IP networks over their physical high-speed backbone without the need to use encryption. Currently, approximately one million customer sites worldwide are connected through an MPLS-VPN network.

- Fast Reroute (FRR)—As a switching technology, MPLS offers an option to either logically replace optical fast rerouting techniques using pre-computed protected logical paths, or to replace the optical physical alternate path, ensuring a 50 ms backup at a low cost.

Large enterprises have recently started to adopt MPLS IP-VPN for their own needs. More than one hundred enterprises worldwide are currently deploying MPLS. The main interest for these enterprises is the virtualization capability of MPLS that facilitates the creation of private VPNs in the enterprise.

VLAN bridging can be easily done over an MPLS MAN. MPLS allows a core label to go from an ingress node to an egress node, which can then be used by any edge services.

The EoMPLS ingress node (PE) adds a label into the MPLS stack for any packet coming from a port or VLAN. This label has been previously negotiated with the egress node (PE) to point toward the egress port or VLAN. Thus, bridged packets are transported transparently over a Layer 3 core. This is a key benefit of EoMPLS because no Layer 2 technology, such as MAC address switching or spanning tree, has to be implemented in the MAN. In addition, core links and nodes are Layer 3 protected with a higher stability and faster convergence than any Layer 2 solution. However, packets are not routed, but instead are bridged between sites.

There are three types of EoMPLS. In two of them, attached-circuits are physical interfaces or sub-interfaces and are supported natively in the Sup720-3B. The third one, where the attached circuit is an internal VLAN, requires the core-facing card to be either an OSM or a SIP card.

The three types of EoMPLS are the following:

- Interface Xconnect, also called port-mode Xconnect (cross-connect)

  Interface Xconnect transports any packet getting into the physical port as is, transparently toward the egress associated port. This simulates a cross-connect cable (with an infinite length, as it is transported through the MPLS network). This approach allows flexibility, smooth integration, and full function transparency.

- Sub-interface Xconnect, also called VLAN-edge Xconnect

  Sub-interface Xconnect differs from Interface Xconnect because the ingress port is a trunk (such as dot1Q). The switch removes the 1Q header, extracts the VLAN number, determines the associated sub-interface, and performs the cross-connect with the associated VLAN at the other side of the network. This simulates VLAN-to-VLAN switching (at long distance through the MPLS network). VLAN renumbering is possible, but it adds complexity to the design.

- Internal VLAN Xconnect—This option is not covered in this document. Internal VLAN Xconnect requires OSM or SIP cards.

EoMPLS Termination Options

EoMPLS appears as a tunnel technology that connects ports, or sub-interfaces, or even VLANs located on both sides of a network. In the following diagrams, the EoMPLS tunnels are represented by a virtual link, shown in red, which is transported transparently by an MPLS core.

Note The red dots in the following figures represent the attachment point of the EoMPLS pseudowire.

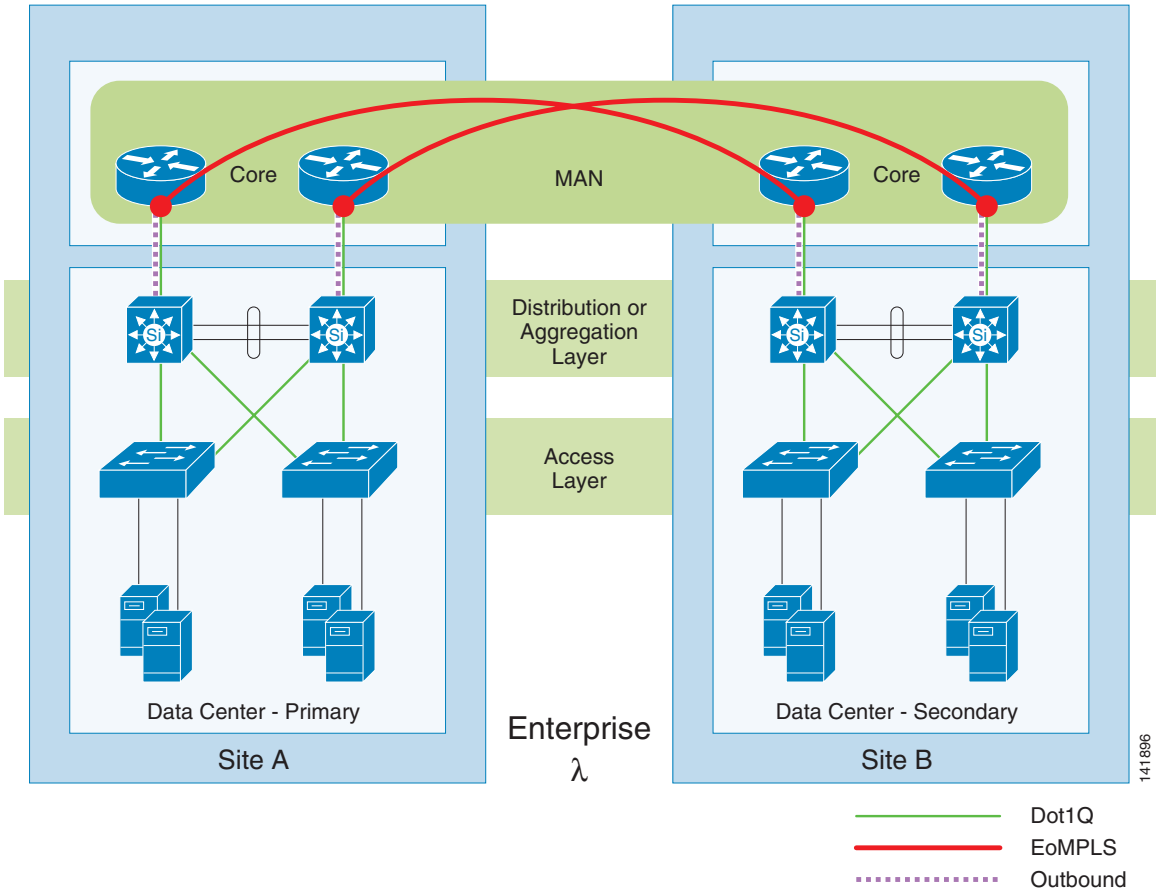
This MPLS core is composed of multiple sites, with a ring or a partially meshed topology. This design guide categorizes the designs based on where the MPLS MAN capability is located: in the MAN, at the data center premise, or within the aggregation layer.

This design guide describes the following four termination options, all based on a port-based Xconnect:

- EoMPLS termination on the MAN routers (see Figure 5-2)

This design relies on an MPLS-enabled core and does not require the data center routers to be configured for MPLS. This design typically relies on a managed Layer 2 service from a service provider. It is quite rare and difficult to use the same fiber from the data center to the POP to provide both Layer 3 VPN and Layer 2 VPN services. This would require deploying another technology, such as VLAN-based Xconnect or VPLS, which are not within the scope of this guide.

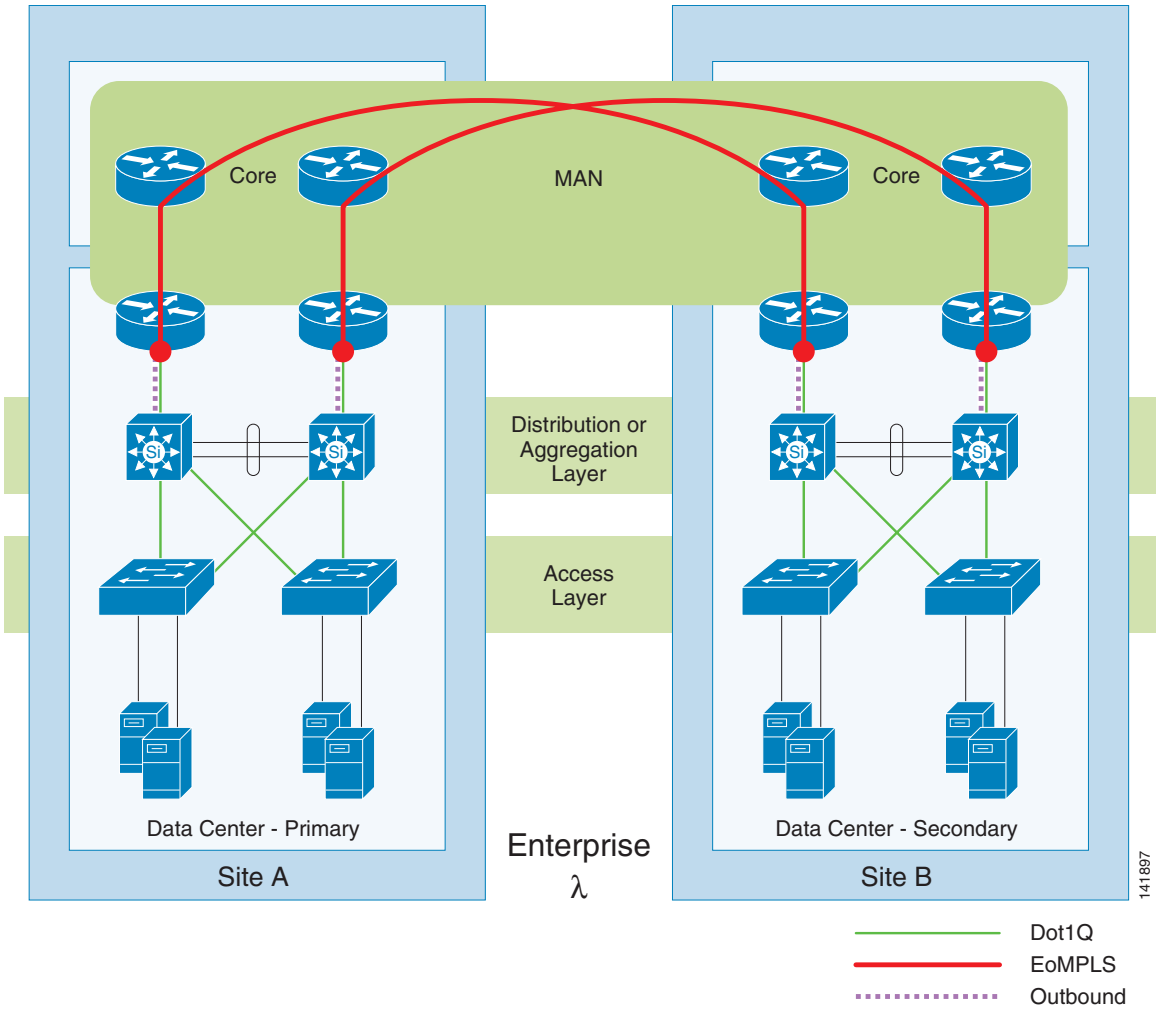
Figure 5-2 EoMPLS Termination on the MAN Routers



- EoMPLS termination on the WAN edge routers (see Figure 5-3)

  This design applies mainly to enterprise self-deployed MANs. With this design, a MAN PE is physically located at the data center boundary and terminates the EoMPLS pseudowire (indicated by the red dot).Two links exist between the aggregation and the PE; one link is used for the Layer 2 traffic between data centers, and one link is for Layer 3 traffic.

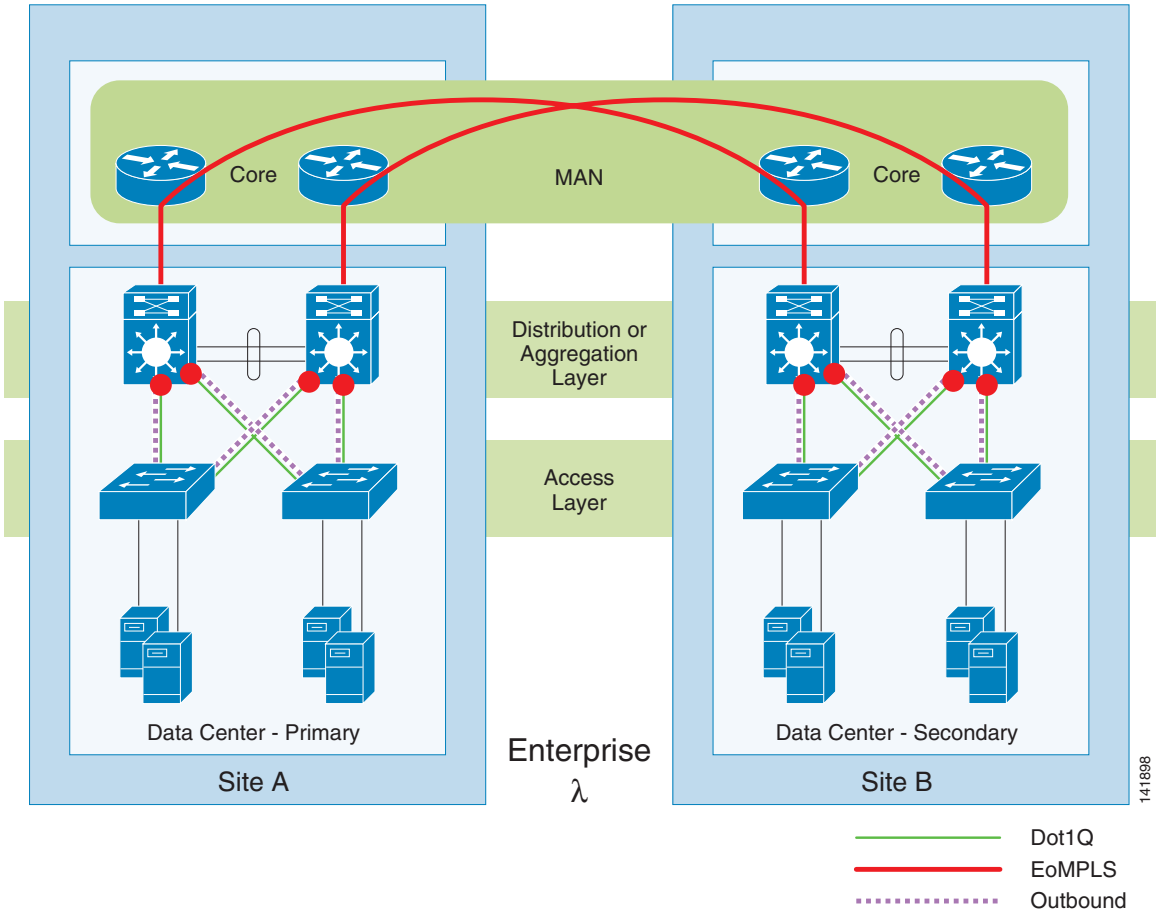
Figure 5-3 EoMPLS Termination on the WAN Edge Routers



- EoMPLS termination in the data center aggregation switches using a dedicated link to the access layer (see Figure 5-4)

  This design leverages the MAN capability of the aggregation layer devices. The aggregation switches in this design provide port-based Xconnect. The access switches need to connect to each aggregation switch with two cables: one cable going to the port cross-connect (which tunnels the VLANs between the two sites), and one cable providing regular routing and switching connectivity to the rest of the network.

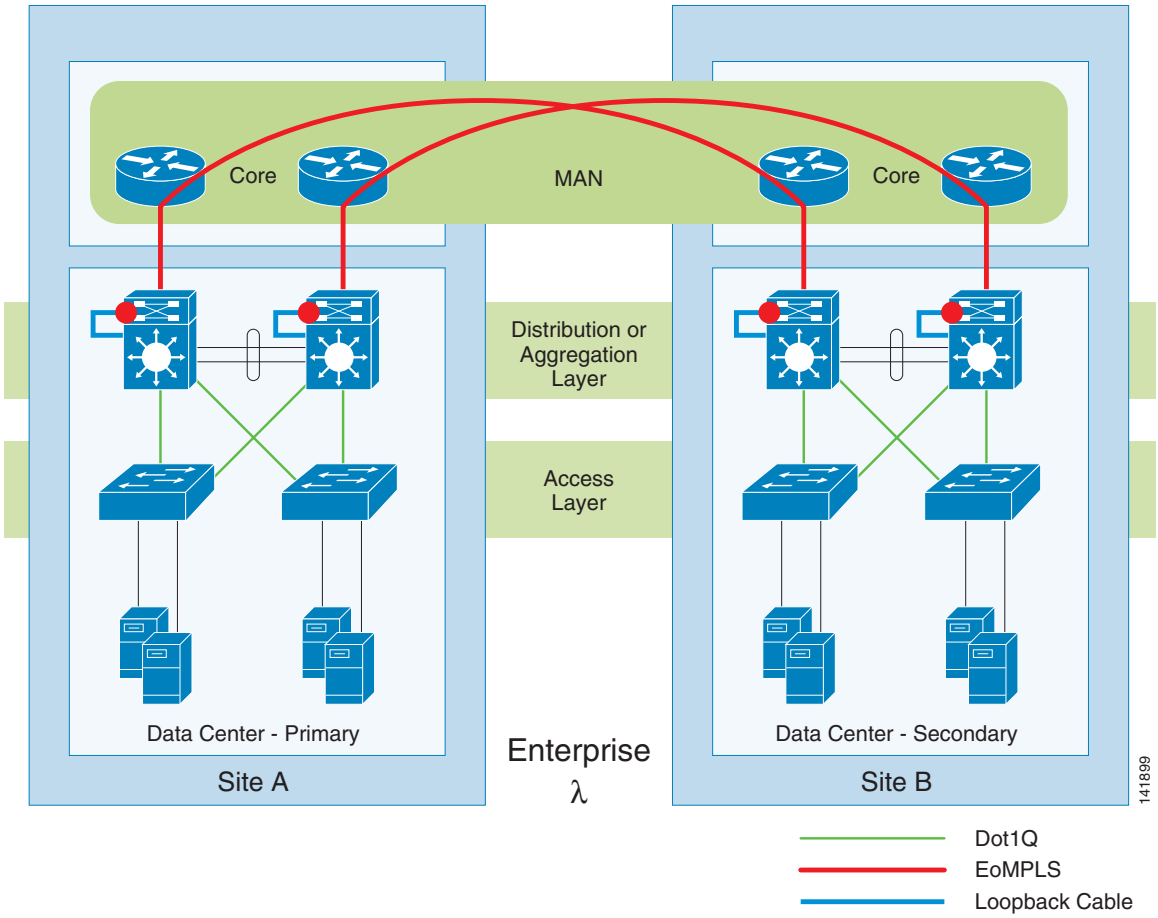
Figure 5-4 EoMPLS Termination at the Data Center Aggregation Using Dedicated Link to the Access Layer



- EoMPLS termination in the data center aggregation switches using a loopback cable (see Figure 5-5)

  To allow transport of the server farm VLAN of the aggregation switches through EoMPLS, a loopback cable is used to re-inject the internal VLAN on a physical port. From an EoMPLS point of view, this approach is very similar to the third option. The main difference is that it does not require multiple physical links from the access switch to the aggregation switch. Figure 5-5 presents a port-based Xconnect and requires a loopback cable at the aggregation layer to carry the VLAN independently.

Figure 5-5 EoMPLS Termination at the Data Center Aggregation using Loopback Cable



Several VLANs from the access layer can be carried through the Layer 2 VPN. Some of these VLANs are limited to data center-to-data center communication, such as the VLAN used for the heartbeat of the cluster. Other VLANs are used for the access to the outside of the data center (outbound), such the VLAN for the virtual IP address.

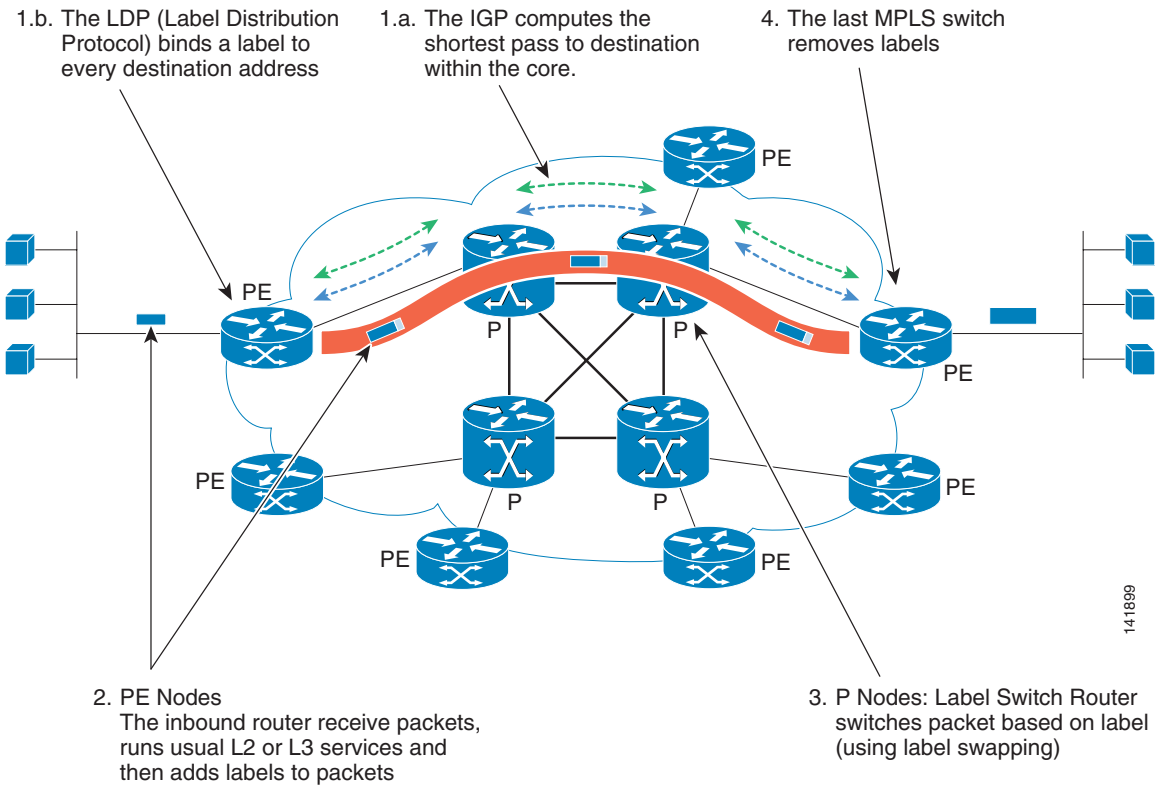
MPLS Technology Overview

The fundamental concepts in MPLS are the following:

- MPLS relies on IP connectivity.

  Figure 5-6 illustrates an MPLS-IP switching service.

Figure 5-6 MPLS-IP Switching Service



Before any labeling of transport and services, MPLS requires that the core be IP-enabled. Therefore, a core IGP must be selected first.

Any IGP is supported by MPLS, MPLS Layer 3VPN, MPLS Layer 2VPN (from static to ISIS, including RIPv2, IGRP, EIGRP, OSPF), but OSPF and ISIS are required to enable the most services, such as traffic-engineering, Fast Reroute, and fast convergence.

- MPLS is a label-based technology.

  At the first stage, labels are negotiated between peers, then subsequent packets are tagged by the ingress device and, at the egress layer, the labels are treated according with the negotiation.

  There are several protocols that are used to negotiate labels, depending on the architecture layer.

  Core label types are the following:

  - LDP (Label Distribution Protocol)
    Used to negotiate best path between to adjacent core nodes.
  - eBGP + labels
    Used to negotiate best path at an Autonomous System interconnect.
  - RSVP
    Used to negotiate deterministic path (with or without bandwidth reservation) along core nodes.

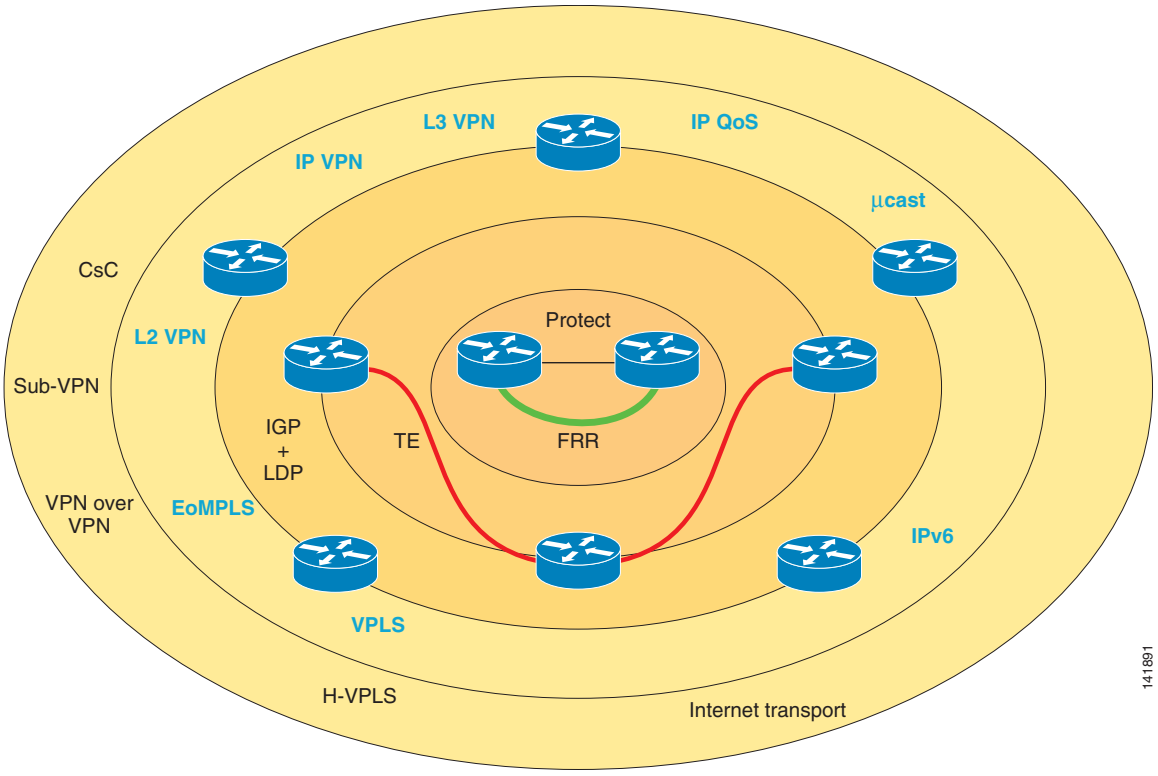
  Edge label types are the following:

  - Directed-LDP
    Used to negotiate Layer 2 virtual-circuit edge to edge.

- MP-BGP
  Used to negotiate Layer 3 multi-points connection between virtual routing instances between edge nodes.

- MPLS supports the stacking of labels (see Figure 5-7).

Figure 5-7 MPLS Recursivity



MPLS supports the overlay of architecture, each of them being independent of the others, and transported through a stack of labels.

Typical stacking includes the following:

- A Fast Reroute label
- A Traffic-Engineering label
- A Core IGP label
- A VPN label
- A Sub-VPN label

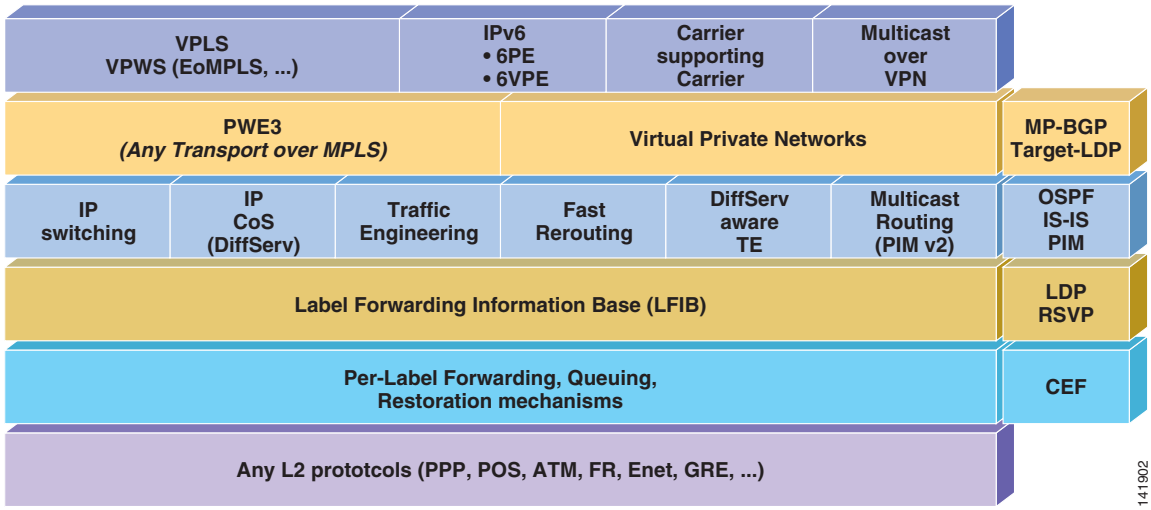
Within an enterprise, such a depth of stack is quite rare, and roughly a Core-IGP label and a VPN label is the most common, while a Fast Reroute label can also be useful.

- MPLS is constructed using three types of nodes:
  - A P node is a Layer 3 node that performs swapping of labels in-between interfaces.
  - A PE node is an edge node (Layer 3 or Layer 2) that imposes labels to plain ingress packet (and removes them at egress).
  - A CE node is any Layer 2 or Layer 3 node attached to a PE, performing IP routing, VLAN bridging, or any other switching technology.

The same device can be a pure P, a pure PE, a pure CE, or it can be some mixture of the three.

- MPLS is a layered technology (see Figure 5-8) consisting of the following:
  - A Data Link layer (Layer 2)—Can be any type of link and, in a MAN, very often an Ethernet transport
  - A core label-swapping layer—IGP or RSVP generated labels
  - Switching services—Core layer induced services
  - Routing virtualization—Layer 3 VPN or Layer 2 VPN virtualization over Layer 3 transport
  - Edge virtualization—Layer 2 or Layer 3 point-to-point/multipoint, IP multicast, IPv6 tunneling, sub-VPN, and so on

Figure 5-8 MPLS Layered Architecture

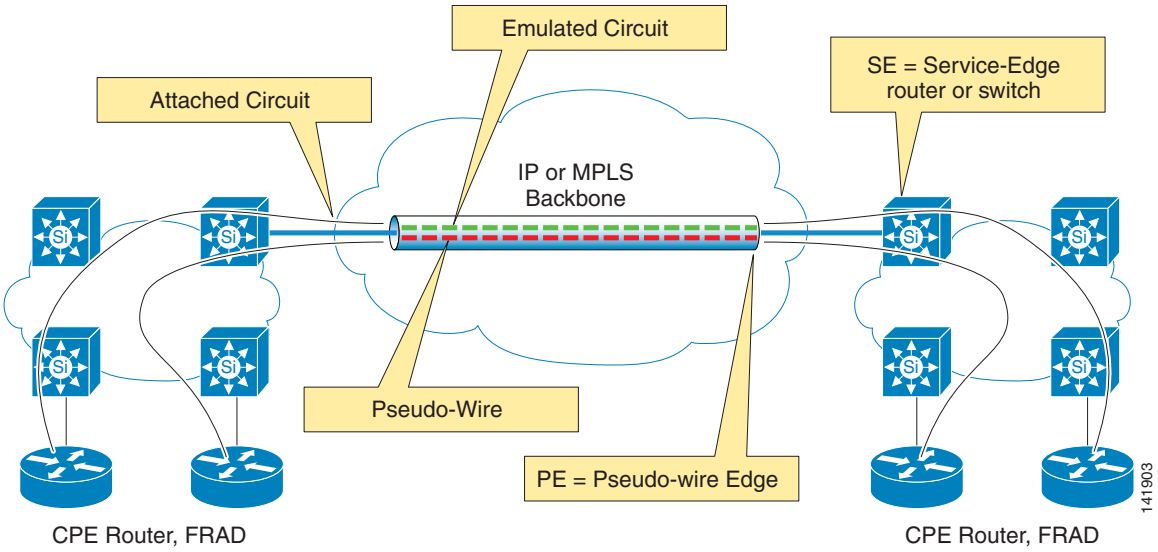


EoMPLS Design and Configuration

EoMPLS Overview

EoMPLS is a virtual circuit technology over an MPLS core (see Figure 5-9).

Figure 5-9 Pseudowire Emulation Edge-to-Edge—PWE3



The elements shown in Figure 5-9 are described as follows:

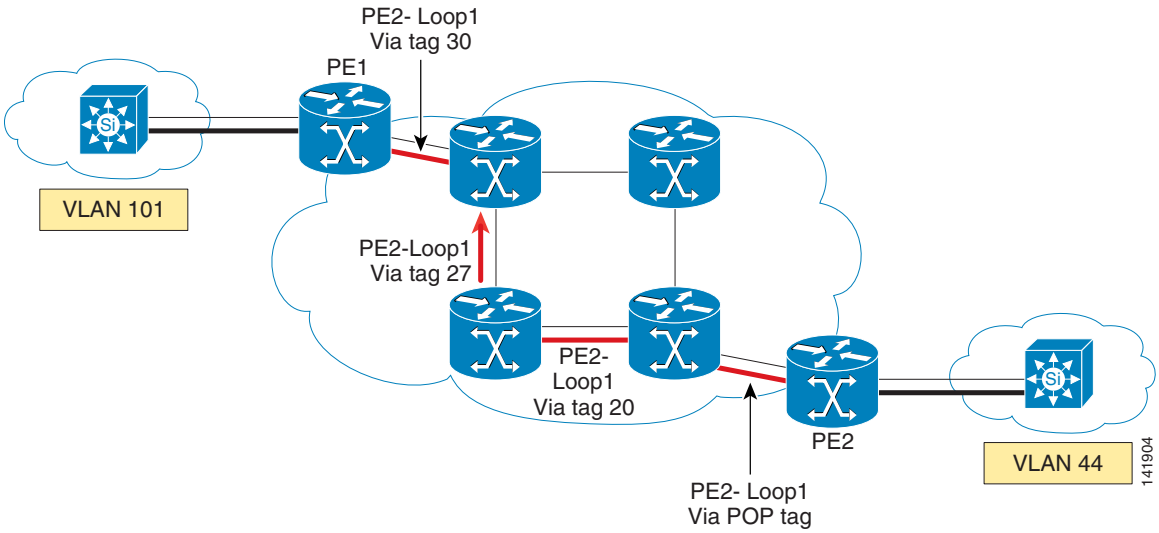
- Attached Circuit

  The purpose of EoMPLS is to transparently remotely interconnect attached circuits at both sides of the network.

- Type of Attachment Circuit with EoMPLS
  - Edge VLAN.
  - Edge Port.
- Emulated Circuit
  - The edge-to-edge transport virtual-circuit; associated with an edge Layer 2 label in the MPLS stack
- Pseudowire
  - The core PE-to-PE transport path (label switched path)
  - In general, established by LDP (sometimes RSVP) all along the core.
- Service edge
  - The Layer 2 device directly attached to the PE.

Figure 5-10 shows the core transport labels.

Figure 5-10 PWE3—Stage 1

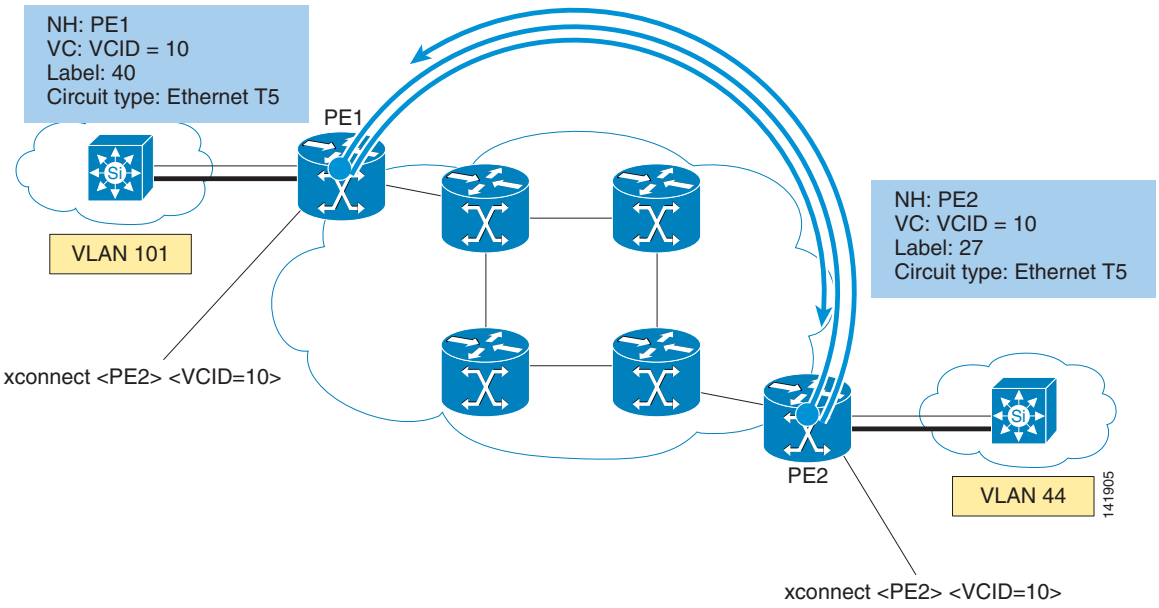


As with any MPLS service, the key element is the MPLS stack. The transport stack layers are used to interconnect PEs (with edge devices providing the upper layer services) and whatever the service is. The service stack layers then provide the labeling of the services themselves (Layer 2VPN, Layer 3VPN, and so on).

Stage 1 is common to any MPLS service, Layer 3 VPN, or Layer 2 VPN. LDP is the common way to distribute these core labels; RSVP can be another one. At this point, edge PEs are able to reach each other through a label switched path. This path can be traced and monitored using MPLS OAM services.

Figure 5-11 shows the labels being distributed through a directed LDP session.

Figure 5-11 PWE3—Stage 2



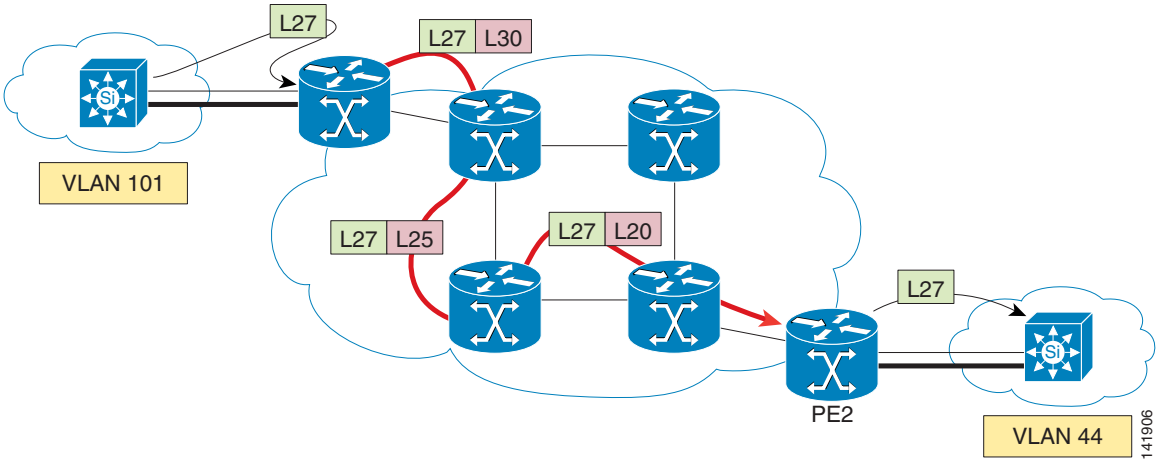
On overlay of the core, PEs are establishing a direct LDP session, completely independent from the core LDP. Each PE is advertising asynchronously to all of the VCs that have been configured.

Towards the specified PE:

- VC-ID number
  - Identifies the Emulated-circuit.
  - The key parameter that allows the other side PE to tie-up the emulated circuit with its local attachment circuit.
  - The VC-ID of each side must be the same.
- Label
  - Every packet at ingress on the attachment circuit of the other side is encapsulated with this label.
- Next-hop
  - Every packet at ingress on the attachment circuit of the other side receives a second label that is the core label leading to the next-hop.

Figure 5-12 shows label forwarding.

Figure 5-12 PWE3—Stage 3



The pseudowire (virtual path) from PE-to-PE is established. Any packet at ingress of the attachment circuit is encapsulated with two labels (the upper one in the stack is the Core label, the second one is the Emulated Circuit label, as shown in Figure 5-13.

The packet is then switched into the core using the top-most label, until it reaches the penultimate core device, (the last P). This one removes the top-most label, which has no further purpose, and the packet, along with only the Emulated Circuit label, is passed to the egress PE. The packet is then pushed toward the attachment circuit using the Emulated Circuit label, which is eventually removed. The removal of the Core label by the last P is an option that is called Penultimate Hop Popping. It is enabled by default.

When two PEs are directly connected, because of the Penultimate Hop Popping, the packet exchange on the direct link is always encapsulated with only the Emulated Circuit label, because the Core label is always empty.

EoMPLS—MTU Computation

This section describes how MTU is configured.

Core MTU

In EoMPLS, the full Ethernet packet is transported, except for the FCS (and the preamble and SFD, which would have no purpose). The maximum PDU size is then 1514.

As it is plain switching, the source and destination MAC address are key to be transported as is, but it is important to note that in Port-mode Xconnect, no bridging of the Ethernet frame is performed by any PE or P. No bridging function at all is performed on the PE ingress or egress ports, and therefore not in the core. That means that none of the MAC addresses of the customers, or the spanning tree, or other bridging features are handled by the PE, which is an important aspect for the stability of the MAN.

Figure 5-13 Ethernet v2—802.3 Encapsulation

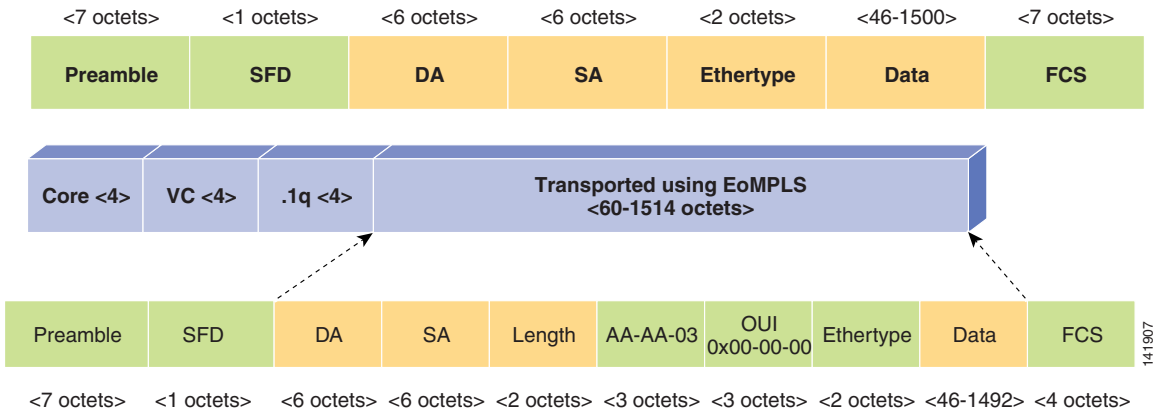
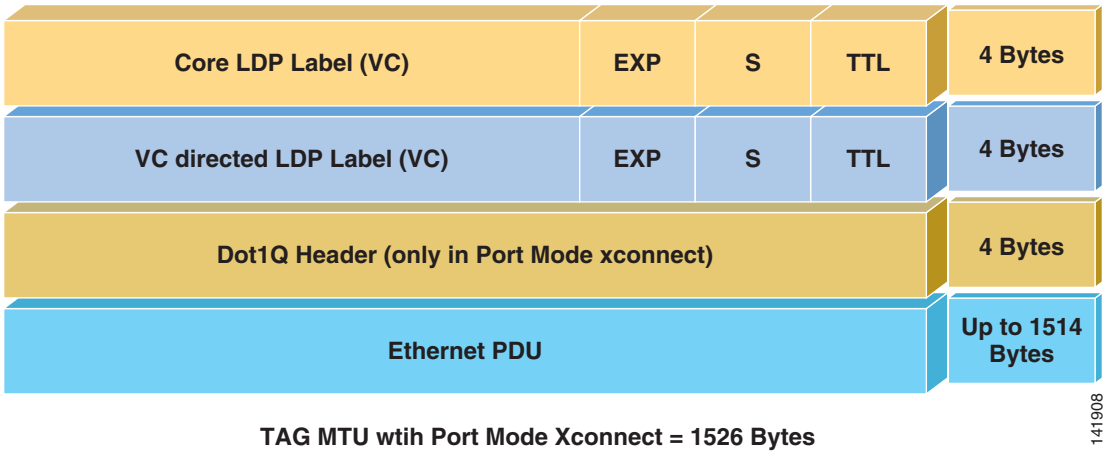


Figure 5-14 shows the MTU settings.

Figure 5-14 TAG MTU with Port-Based Xconnect



TAG MTU wtih Port Mode Xconnect = 1526 Bytes

As in any encapsulation technology in the Ethernet world, MTU is an important aspect to consider. The following is often the most common configuration failure:

Ethernet max PDU = 1514 (as the FCS is not transported).

In VLAN-edge Xconnect, there is no additional header; the Dot1Q header has been removed to determine the right subinterface.

In Port-based Xconnect, any ingress packet is encapsulated without any kind of processing:

- If the Service-edge device is sending plain packets, then the encapsulated PDU max size is 1514.
- If the Service-edge device has defined the link going to the PE as Trunk, then the PDU max size is 1518, as the Dot1Q header is transported as is.

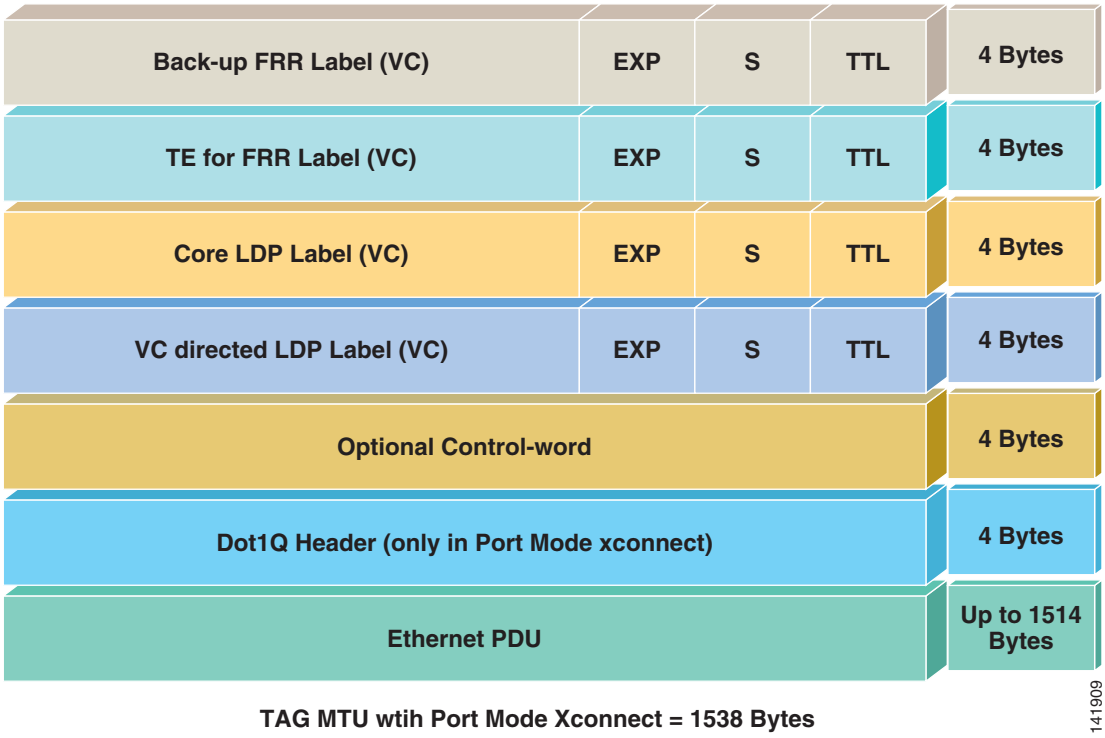
Then, four bytes are added for the Emulated Circuit label (VC-directed LDP label), and 4 more bytes for the core label (core LDP label), if any (remember that in back-to-back, the core label is empty).

The Tag MTU is the only active MTU setting for labeled packets. They are not checked against the Interface MTU. In addition, the Tag MTU includes the label stack.

In MPLS, you do not have to modify the interface MTU, the Emulated Circuit MTU is derived from the MPLS MTU minus the stack size (in this case, 1526 - 8 = 1518).

Figure 5-15 provides recommended MTU settings.

Figure 5-15 MPLS Links Recommended MTU Setting



TAG MTU wtih Port Mode Xconnect = 1538 Bytes

If the Fast Reroute option can be used in the MAN core to reach sub-50 ms backup time, two additional labels must be allowed.

Optionally, an additional control word can be added to the EoMPLS header to allow for features, such as mis-ordering detection. This would increase the size of the encapsulated PDU by 4, but in a common EoMPLS it is unused. A best practice would be to plan for these capabilities, even if they are not used today. The recommended core links MTU is then 1538.

Setting an MTU is always highly dependent on the physical card, or the physical connection used. In Gigabit Ethernet, this constraint is usually light, but in Fast-Ethernet, or with a service provider offering, physical limitation might be encountered.

If the core is transporting only MPLS, which means that no application traffic is being sent using the global routing table, then it is a good practice to increase the physical MTU to the tag value, to ensure transport.

Also, be careful when using a giant or jumbo frame, the previous recommendation is assuming that you are using only a plain Ethernet frame, or the Dot1Q frame has to be transported. If you transport larger Ethernet frames, the core link and tag MTUs must be increased as well (always with 24 additional bytes for EoMPLS).

Edge MTU

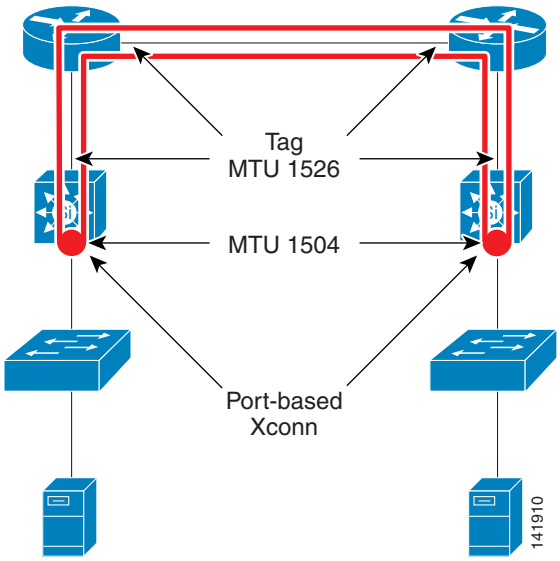
In a Port Cross-Connect configuration, the full traffic coming in is cross-connected toward the egress port and this port is declared as a routed port. If any 802.1Q frames have to be cross-connected, then this means that the port receives MTU frames at 1504 bytes. Therefore, the edge port MTU must be set to 1504. Keep in mind that some physical Fast-Ethernet or Ethernet cards cannot support configurable MTU.

Following are some examples of implementation MTU sizes:

- The physical Port-based Xconn Interface (Gig5/2) requires the 1500 + 4 bytes for tagged frames = 1504 bytes.
- The Uplink physical interface connected to the upstream router (MAN) requires the physical Ethernet max frame to be increased from 1518 by 4 bytes = 1522 bytes. This is required, as the PDU transported by EoMPLS is not 1500 bytes, but the full Ethernet frame (except for FCS), plus the 802.1Q header.
- On every core link, the TAG MTU should be set to 1514 + 4 bytes Dot1Q + 4 Bytes VCID + 4 bytes LDP = 1526 bytes. The reason for 1514 is that the encapsulated frame excludes the FCS (4 bytes).
- The Uplink physical interface of the upstream router (MAN) that is connected to the downstream aggregation switch requires that the TAG MTU be set to 1526 bytes and the physical interface connected to the MAN (remote sites) should have the TAG MTU set to 1526 bytes.

Figure 5-16 illustrates MTU size.

Figure 5-16 MTU Size



EoMPLS Configuration

This section provides some best practices to follow when setting up EoMPLS configurations.

Using Core IGP

To build a Label-Switched-Path, MPLS requires the IP connectivity into the MAN to be set. MPLS can use the current core IGP rather than a specific parallel IGP. If the choice is to use the current MAN WAN IGP to add the MPLS service, then the only real task is to enable MPLS on core links.

Note Ensure that the IGP path in-between edge nodes always cross links with MPLS enable.

Another choice that Cisco has made for the purpose of this best practice example is to use another instance of an IGP for MPLS. Therefore, on every core link that must support MPLS, ISIS has been enabled in addition to MPLS.

One loopback per PE per service is best:

```
interface Loopback99
 ip address 10.99.65.5 255.255.255.255
 ip router isis
 isis circuit-type level-1


router isis
 net 49.0001.0000.6500.5555.00
 is-type level-1
 metric-style wide
 passive-interface Loopback99
 advertise passive-only
```

Set MPLS Globally

```
MPLS label protocol ldp
MPLS ldp router-id Loopback99
```

Limit label advertisement to the only useful address, there is no need to transport other IP addresses in label mode.

For Layer 2VPN (as in Layer 3VPN), the only useful addresses are the PE loopback addresses, which are used for the targeted-LDP exchange.

Use a dedicated IP addressing class for all of the MPLS services loopbacks.

```
Best practice: Use different PE loopbacks for Layer 2 VPN service and Layer 3 VPN service
for a clear separation.

no MPLS ldp advertise-labels
MPLS advertise-tags for 1
access-list 1 permit 10.99.0.0 0.0.255.255
```

Enable MPLS on Core Links

On every link that comprises the MAN (or the WAN, if any), enable ISIS and MPLS transport.

```
interface FastEthernet1/1
 ip address …
 ip router isis
 tag-switching mtu 1526
 tag-switching ip
```

For the MTU setting, see Appendix A, “MTU Considerations.”.

Verify MPLS Connectivity

Without the use of the Traffic-Engineering capability, the MPLS path is the same as the IP path. Therefore, a plain IP-ping or IP-traceroute would still be useful for checking the data path. MPLS-ping and MPLS-traceroute will precisely check the label switched path for LDP.

In addition, MPLS-pseudowire-ping will allow packets to generate directly into the pseudowire to verify connectivity. These new MPLS OAMs are also able to generate packet in sub-second fashion, with a tunable tempo.

```
#ping mpls ipv4 10.99.65.5 255.255.255.255
Sending 5, 100-byte MPLS Echos to 10.99.65.5/32,
     timeout is 2 seconds, send interval is 0 msec:

Codes: '!' - success, 'Q' - request not transmitted,
       '.' - timeout, 'U' - unreachable,
       'R' - downstream router but not target,
       'M' - malformed request

Type escape sequence to abort.
!!!!!

#traceroute mpls ipv4 10.99.65.5 255.255.255.255 ttl 7
Tracing MPLS Label Switched Path to 10.99.65.5/32, timeout is 2 seconds
  0 10.0.0.6 MRU 1526 [Labels: 26 Exp: 0]
R 1 10.0.0.5 MRU 1526 [Labels: 16 Exp: 0] 4 ms
```

```
R 2 192.168.25.2 MRU 1530 [Labels: implicit-null Exp: 0] 1 ms
! 3 10.10.0.6 2 ms
```

Create EoMPLS Pseudowires

As previously stated, there are many ways to create pseudowires. In a data center environment, without any of the 6500/7600 OSM/SIP cards, the standard use is Port mode.

```
Cross-connect at port-level (physical interface level)
interface GigabitEthernet5/2
 mtu 1504
 no ip address
 Xconnect 10.99.65.2 100 encapsulation mpls
```

In Port mode, every packet ingress is transported through the pseudowire without any analysis. Therefore, if the interface might receive a DOT1Q packet, the PDU MTU must be sized accordingly.

Another option, that is not used in these designs, is to perform the cross-connect at the sub-interface level.

Verify EoMPLS Pseudowires

```
#show mpls l2 vc

Local intf     Local circuit        Dest address    VC ID      Status
-------------  -------------------- --------------- ---------- ------
Gi5/2          Ethernet             10.99.65.5      100        UP

#show mpls l2 vc detail
Local interface: Gi5/2 up, line protocol up, Ethernet up
  Destination address: 10.99.65.5, VC ID: 100, VC status: up
    Tunnel label: 26, next hop 10.0.0.5
    Output interface: Fa3/2, imposed label stack {26 16}
  Signaling protocol: LDP, peer 10.99.65.5:0 up
    MPLS VC labels: local 16, remote 16
    MTU: local 1504, remote 1504
  Sequencing: receive disabled, send disabled
  VC statistics:
    packet totals: receive 92471, send 349251
    byte totals:   receive 10962950, send 29963199
    packet drops:  receive 0, send 5

#ping mpls pseudowire 10.99.65.5 100
Sending 5, 100-byte MPLS Echos to 10.99.65.5,
     timeout is 2 seconds, send interval is 0 msec:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 2/3/5 ms
```

Optimize MPLS Convergence

Optimize to a sub-second or less. In a data center interconnection, the convergence time on a failure becomes very important.

In general, the weakest part of the MAN is the long distance link, but still it is a requirement that the data center be dual-connected to the MAN through two edge switches. One very good approach to increase the high-availability is to rely on the IGP capabilities in terms of convergence.

These days, in a network of a reasonable size a very few hundred of milliseconds is perfectly reachable in term of convergence after a link or node failure has been detected. If the network is more complex, then less than a second is a perfectly reasonable expectation.

```
router isis
 net 49.0001.0000.6500.5555.00
 is-type level-1
 metric-style wide! Allows Traffic-Engineering attribute propagation
 spf-interval 20 100 20! ISIS SPF fast reaction, but backoff protected, see below
 prc-interval 20 100 20! same for IP addresses changes
 lsg-gen-interval 1 1 20! Same for LSP advertisement
 fast-flood 15! Fast flooding for first LSP
```

Backoff Algorithm

The fast reaction of the IGP in terms of path re-computation is controlled by a backoff algorithm that prevents any instability.

In the previous setting, the first SPF is run right after the failure detection (20 ms), but a subsequent computation occurs only after a pace of 100 ms, a third one occurs after an interval of 200 ms, then 400 ms, then 800 ms, up to a max of 20 s.

Return to stability is considered okay after 20 s of perfect stability, and all timers are reset.

Carrier Delay

Fast computation of a new path can occur only after connection failure detection, the next step is to optimize link or node detection failure.

If the failure leads to a physical link down, which is clearly the case of fiber interconnection or even often in an Ethernet over SONET service, then the reaction can be almost instantaneous.

Carrier-delay must be set to 0. Carrier-delay is useful for links that are already protected with another Layer 1 technology, such as SONET or WDM protection.

When Ethernet is used as a point-to-point link, there is no need to waste time in a multi-points negotiation. To protect against any instability versus a fast convergence, dampening will control flapping of the link.

```
interface FastEthernet1/1
 ip address …
  carrier-delay msec 0
 ip router isis

 isis network point-to-point
 dampening
```

When the failure does not lead to a physical detection, the IGP timers will detect the loss of neighbor-ship in one second:

```
interface FastEthernet1/1
 ip address …
 ip router isis
 …
 isis circuit-type level-1
 isis hello-multiplier 10 level-1
 isis hello-interval minimal level-1
```

Following, is an example of convergence testing on link failure. To measure the convergence time, use a **ping mpls pseudowire** command with a frequency of 100 ms:

```
#ping mpls pseudowire 10.99.65.5 100 interval 100 repeat 600 size 1500 timeout 1
Sending 600, 1500-byte MPLS Echos to 10.99.65.5,
     timeout is 1 seconds, send interval is 100 msec:

Codes: '!' - success, '.' - timeout, 'U' - unreachable,

!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!.!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!UUUUUUU...!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
Success rate is 98 percent (589/600), round-trip min/avg/max = 1/1/20ms
```

When the link goes down physically, the IGP convergence time is less than a few 200 ms (one pseudowire ping is missing), but the link down detection can be longer, depending on the failure. The worst case is IGP hello detection + path re-computation, which should be below 1,5s. if no BFD is used.

It is important to notice that when the link came back up, the connectivity went down for roughly one second. This strange behavior is caused by a feature that is not yet supported on the Catalyst 6500 or Cisco 7600: MPLS LDP-IGP Synchronization.

The IGP converged well without any loss, but it took one second for LDP to recalculate the new labels and set them up into the line cards. Future enhancements will improve this recovery behavior.

A good way to decrease this back to normal synchronization problem is to use targeted-LDP in-between direct neighbors, instead of using automatic discovery. This allows the LDP session to not be dropped on neighboring failure, and to not wait for an LDP label advertisement when the neighbor comes back to normal.

So, at the global configuration level, you can optimize performance by entering the list of direct neighbors:

```
mpls ldp neighbor 10.99.65.2 targeted ldp
..
```

Consider that the MAN always recover s in less than 2s in an upper-layer convergence design.

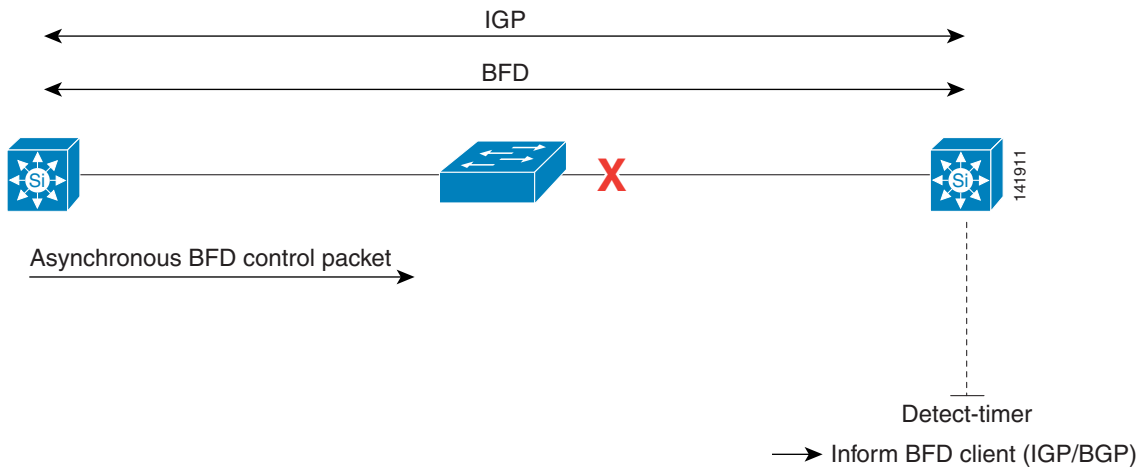
BFD (Bi-Directional Failure Detection)

In some specific environments, the connectivity failure might be difficult to detect in less than one second, and this is typically the case if the core links are provided using a plain Ethernet switching technology, such as QinQ, or even sometimes with some Ethernet over SONET devices.

As previously described, the IGP will not detect this kind of failure in less than one second. If sub-second detection is mandatory, then a new Layer 3 neighboring failure detection mechanism, BFD, can be used. BFD is a scalable asynchronous hello mechanism, which substitutes itself to the IGP hellos. As with any failure detection mechanism, timers must be tuned to avoid false detection. A few 100 ms would be, in many cases, adapted to enterprise requirements, while sub-100 ms detection is also feasible, it requires care.

As BFD is associated to IGP, flapping is managed by the back-off capabilities of the IGP, with BFD being just a link/Layer 2-path detection failure mechanism. Figure 5-17 illustrates the bi-directional failure detection process.

Figure 5-17 Bi-Directional Failure Detection




```
   interface Vlan600
 ip address …
 ip router isis
 bfd interval 10 min_rx 10 multiplier 3
 bfd neighbor 10.10.0.18
 dampening                ! protection against link flapping

router isis
 bfd all-interfaces

ping mpls pseudowire ipv4 10.99.65.1 255.255.255.255 time 1 int 100 repeat 1000
Sending 1000, 100-byte MPLS Echos to 10.99.65.1/32,
     timeout is 1 seconds, send interval is 100 msec:

!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
3w0d: %BFD_TEST-5-TRANSITION: Neighbor 10.10.0.17 DOWN - admin is FALSE
3w0d: %CLNS-5-ADJCHANGE: ISIS: Adjacency to Cat6k-DC2-left (vlan600)
Down, BFD !
3w0d: %LDP-5-NBRCHG: LDP Neighbor 10.99.65.4:0 is DOWN!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
    Codes: '!' - success, 'Q' - request not transmitted,
           '.' - timeout, 'U' - unreachable,
           'R' - downstream router but not target,
           'M' - malformed request

Type escape sequence to abort.
```

Note As you can see, no ping was dropped, meaning a sub-100 ms convergence time.

It is considered best practice to protect against any instability by setting BFD to 100 ms at the beginning:

```
bfd interval 100 min_rx 100 multiplier 3
```

Improving Convergence Using Fast Reroute

Another approach to fast convergence would be to use the capacities of MPLS in terms of Traffic-engineering. This approach is more complex than a plain IGP tuning, and, in general, it requires more design work.

Note IGP fast-convergence is often adequate to meet requirements, so it is best to start simple, before implementing Traffic-engineering Fast Reroute (FRR).

MPLS Traffic-Engineering FRR is based on a similar concept to Optical backup. It supposes that a pre-set alternate path as been prepared to diverge local traffic as soon as a failed link or node as been detected. This minimizes the propagation and new path recalculation time.

In a complex network, this time can be quite large, specifically in a MAN where the common topology is a dual ring including all sites. FRR reduces the convergence time down to a few tenths of milliseconds after failure detection. The failure detection might be a physical link down or RSVP fast-hello detection.

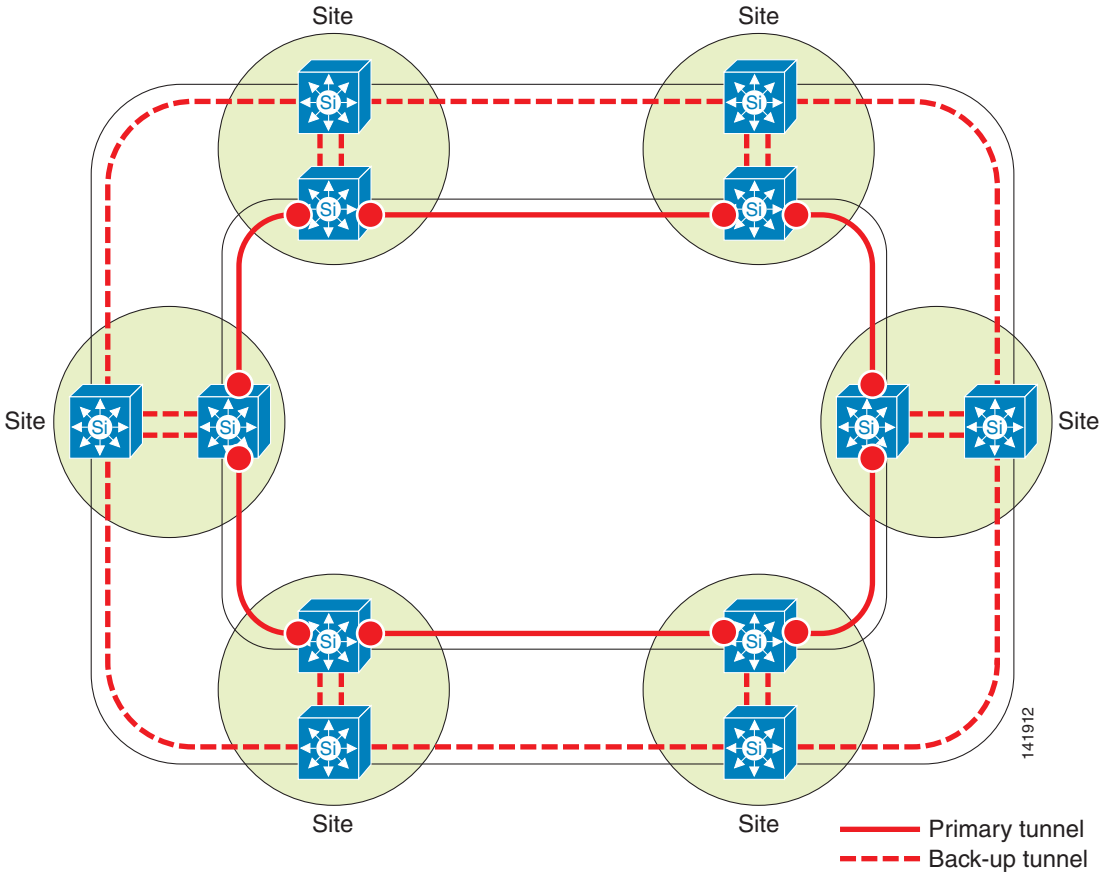
This guide does not focus on the usage of FRR that would require a separated analysis. FRR can protect link failures or even P-nodes failures. In a MAN design, where all nodes are PE, the FRR node protection can be useless. However, the weak part of a MAN is very often the link connection between sites.

The following three possible FRR configurations are possible:

- Manual settings:
  - Manual setting of primary tunnel.
  - Manual setting of the alternate tunnel.
- Automatic protection of each link:
  - Auto-setting of one-hop primary tunnels on all links.
  - Auto-building of back-up paths.
  - Not yet available on the Catalyst 6500 or the 7600 router.
- Automatic PE-PE full-meshing:
  - All PE are tunneled end-to-end to every other PE (automatic setting).
  - Auto-building of backup paths.
  - Not yet available on the Catalyst 6500 or the 7600 router.

Only one approach, the manual approach, is applicable to a network with only PE, such as in a MAN. Figure 5-18 illustrates the FRR design.

Figure 5-18 FRR Link-Protection Design - Dual Ring



Note that FRR protects TE tunnels, but not the plain IP or LDP packets. That means that traffic is protected only when pushed into Traffic-Engineering tunnels first, only then will these tunnels be FRR-protected.

Example of FRR setting:

Globally enable MPLS Traffic-Engineering:

```
mpls traffic-eng tunnels
```

Set traffic-engineering attributes on every core link:

```
Int …
 mpls traffic-eng tunnels
 ip rsvp bandwidth
```

Enable routing protocol transport of Traffic-engineering attributes:

```
router isis
     mpls traffic-eng router-id Loopback99
     mpls traffic-eng level-1
```

Manually create a Primary tunnel on the main path:

```
interface tunnel 25
     ip unnumbered Loopback99
     mpls ip                  ! Allows LDP over TE
     tunnel destination 10.99.72.5
     tunnel mode mpls traffic-eng
```

```
    tunnel mpls traffic-eng bandwidth 0
    tunnel mpls traffic-eng path-Option 1 dynamic
    tunnel mpls traffic-eng autoroute announce
    tunnel mpls traffic-eng fast-reroute
```

Create the backup path:

(Here, on the shortest path that does not use a primary link)

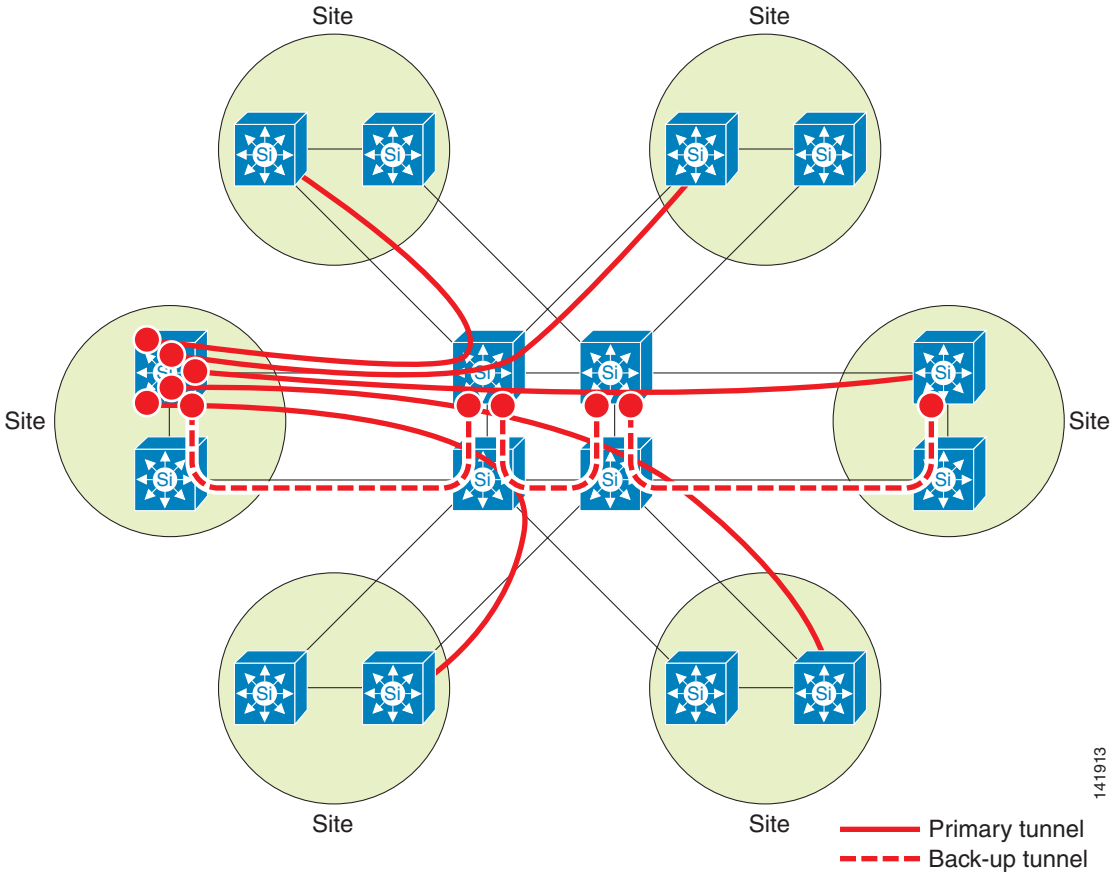
```
interface Tunnel 251
    ip unnumbered Loopback99
 tunnel destination 10.99.72.5
    tunnel mode mpls traffic-eng
 tunnel mpls traffic-eng path-Option 1 explicit name POS2

ip explicit-path name POS2 enable
    exclude-address 192.168.25.2
 exit
```

Associate a backup path to the Primary link:

```
interface pos 2/0
    mpls traffic-eng backup-path tunnel 251
```

An alternative design, especially applicable to a partially-meshed network, consists in the setting of a full-mesh of TE tunnels between PEs that must benefit from FRR, and then protect links or even core nodes along the path. Figure 5-19 illustrates an FRR link-protection design with a partially-meshed MAN.

Figure 5-19 FRR Link-Protection Design—Partially Meshed MAN



High Availability for Extended Layer 2 Networks

It is common practice to add redundancy to the interconnect between two data centers to avoid split-subnet scenarios and interruption of the communication between servers.

Note Proper design can address both problems. For example, the split-subnet is not necessarily a problem if the routing metric makes one site preferred over the other. Also, if the servers at each site are part of a cluster and the communication is lost, other mechanisms (such as the quorum disk) avoid a split-brain scenario.

Adding redundancy to an extended Ethernet network typically means relying on spanning tree to keep the topology free from loops. STP domains should be reduced as much as possible and limited inside the data center.

Cisco does not recommend that you deploy the legacy 802.1d because of its old timer-based mechanisms that make the recovery time too slow for most applications including typical clustering software. Consider using Rapid PVST+ or MST instead.

Etherchannel (with or without 802.3ad, Link Aggregation Control Protocol) provides an alternative to STP when using multiple links between sites for redundancy. With an EtherChannel, you can aggregate multiple physical links to shape a logical link while the traffic is load distributed over all available physical links. If one physical link fails, the switch redistributes all the traffic through the remaining active links.

EoMPLS Port-based Xconnect Redundancy with Multiple Spanning Tree Domains

This design uses the Multiple Spanning Tree (MST) protocol. The main enhancement introduced with MST is to allow several VLANs to be mapped into a single spanning tree instance. It is also easier to control a geographic region of a spanning tree domain containing multiple spanning tree instances (similar to an Autonomous System with a Layer 3 protocol such as Border Gateway Protocol (BGP).

With 802.1s, the bridges exchange a table containing the information that built the MST region:

- MST name
- Revision number
- VLAN mapping to instances

When a bridge receives this information, it compares it to the parameters of its MST region. If only one parameter differs from its database, the port receiving this digest is at the boundary of its MST region, which means that the remote spanning tree domain does not belong to the same MST region. Any communication to and from this interface outside of the MST region uses the IST-0 (Instance 0).

When connecting two MST regions together, Rapid Spanning Tree Protocol (RSTP) is used to control the spanning tree between two boundary ports of each MST region. IST (Instance 0) maps all VLANs to communicate outside of the MST region. Implicitly, the IST root is located inside the MST region. When two MST regions communicate with each other, they use the Instance 0; therefore, they use 802.1w BPDU (RSTP) to control any Layer 2 loop topology. All VLANs are mapped to the RSTP instance (IST).

The MST region, at its boundary interfaces, should be able to interoperate with any STP or RSTP devices as follows:

- RSTP (802.1w) to communicate with another MST region
- RSTP (802.1w) to communicate with a RSTP device connected to a boundary interface
- STP (802.1d) to communicate with an STP device connected to a boundary interface

Note The topology changes should not be affected by any WAN /MAN failure because they are masked by the MPLS convergence. Therefore, Instance 0 should only converge on aggregation switch failure.

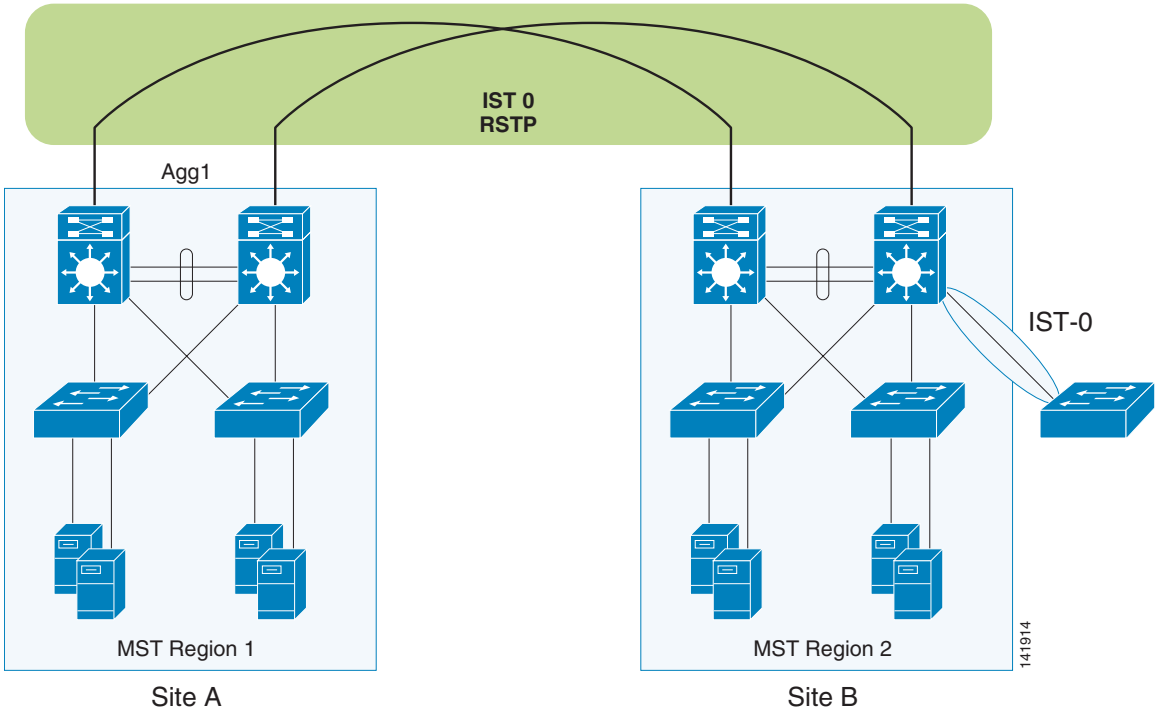
IST Everywhere

Only one single root bridge is elected for the IST - Instance 0 - for the whole Layer 2 network. In addition, any Layer 2 switch without MST enabled, or with MST configured to belong to different MST regions, will use the same Instance 0 to communicate with the original MST region, regardless of to which switch in the MST region it is connected.

This means that two data centers that belong to different MST regions are interconnected using the same IST. One of the data centers will support the primary root bridge for the whole extended Instance 0.

As shown in Figure 5-20, an additional switch, not configured to be in the same MST region (DC 2), is connected to one of the aggregation switches of an MST region. As the Instance 0 shares the same IST root bridge, the root bridge for that Instance 0 is the aggregation switch of DC 1.

Figure 5-20 MST and Instance 0



Interaction between IST and MST Regions

Root Bridges are defined per instance of spanning tree within a well delimited MST region. Any change in topology within the MST region is not propagated outside of its spanning tree instance, except for the Instance 0.

With IST (Instance 0) existing on all ports inside and outside of the MST region, there are some situations where a topology change can block the ports of the switches in a different MST region. For example, if the secondary root bridge is located in a different MST region and an alternate port in the same region begins forwarding traffic because of the topology change, then all downstream ports will go through the blocking stage (~1sec). To prevent this behavior, Cisco recommends enabling the secondary root switch inside the same MST region where the primary root bridge seats. In case of the IST root bridge failing, it will perform a flush on the remote CAM tables. However, the traffic is not disrupted.

First, a specific MST region is created on each data center: MST DC1 and MST DC2. Only the Instance 0 is used to communicate between two different MST regions using RSTP.

Both data centers are interconnected using two pseudowires (Layer 2 VPN) as shown in Figure 5-5.

Between two MST regions, the IST (Instance 0) cannot use PVST+ or Rapid PVST+, but only RSTP and eventually 802.1d in some rare cases. Therefore, it is not possible to load-share the traffic per instance of STP on different physical links, with the IST-0 being present on all interfaces. Figure 5-21 and Figure 5-22 illustrate MST that is dedicated to an external cluster.

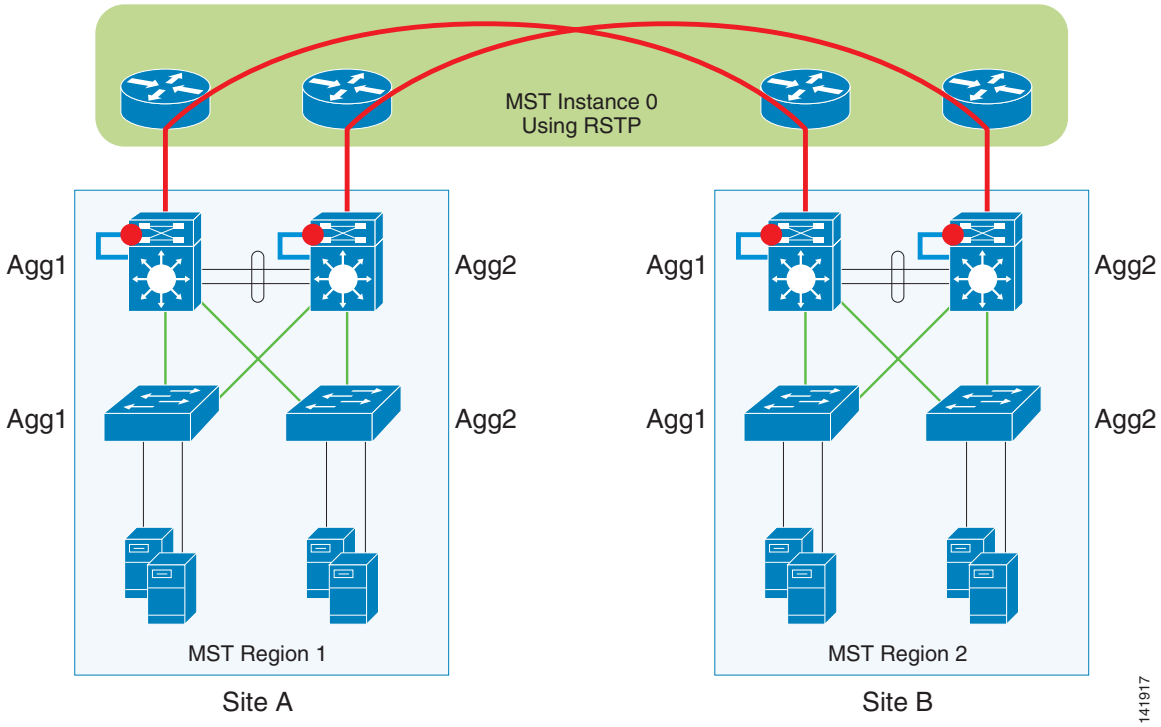
Figure 5-21 MST Dedicated for An Extended Cluster

From outside, the MST region is seen as a single logical switch, regardless the number of switches on each MST region, or boundary ports that belong to different devices.

Because of RSTP being used on the Instance 0, if the forwarding logical Layer 2 connection from site A to site B fails, the backup Layer 2 link will take a sub-second or so to failover - a sub-second failover in case of a direct physical link failure, otherwise it takes three times the BPDU hello timers before a timeout occurs (by default, 6 seconds).

Note The term "logical Layer 2 connection" is used in this guide because the extended Layer 2 is built on top of MPLS (EoMPLS), so any physical failure is fully controlled by the Layer 3 fast convergence protocol (IGP and BFD), as explained in MPLS Technology Overview, page 5-8.

Figure 5-22 EoMPLS at the DC Aggregation layer



This design assumes the aggregation switch 1 (Agg1) on DC1 is the root bridge for the MST region DC1 Instance 1 and Instance 2 and aggregation switch 1 (Agg1) on DC2 forms the root bridge for the MST region DC2 Instance 1 and Instance 2.

Therefore, you have a root bridge on each site for the Instance 1 and Instance 2; note that this is for the Instances limited inside each MST region.

The Instance 0 is used to communicate to outside the MST region and being present on each port of each bridge, only 1 root bridge for the whole IST 0 will exist inside the whole Layer 2 network. Aggr1 on DC1 have been chosen as the root bridge for the Instance 0 (IST Root) and Aggr2 on DC1 to be the secondary root bridge for the Instance 0.

Consequently, you have the following STP state on each switch:

- MST Interface State for Data Center 1 (MST Region 1—Instance 0):

```
Aggregation 1
Aggr1 - Interface 3/46 connected to Aggr2 is Designated Forwarding
Aggr1 - Interface 3/47 connected to the core (MAN) is Designated Forwarding and
boundary using RSTP
Aggr1 - Interface 3/48 connected to Access switch is Designated Forwarding

Interface        Role Sts Cost      Prio.Nbr Type
Fa3/46           Desg FWD 200000    128.302  P2p
Fa3/47           Boun FWD 200000    128.303  P2p Bound(RSTP)
Fa3/48           Desg FWD 200000    128.304  P2p

Aggregation 2
Aggr2 - Interface 3/46 connected to Aggr1 (root) is Root Forwarding
Aggr2 - Interface 3/47 connected to the core (MAN) is Designated Forwarding and
boundary using RSTP
Aggr2 - Interface 3/48 connected to Access switch is Designated Forwarding

Interface        Role Sts Cost      Prio.Nbr Type
Fa3/46           Root FWD 200000    128.302  P2p
Fa3/47           Boun FWD 200000    128.303  P2p Bound(RSTP)
Fa3/48           Desg FWD 200000    128.304  P2p

Access 1
Acc1 - Interface 2 connected to Aggr1 is Root Forwarding
Acc1 - Interface 4 connected to Aggr2 is Alternate
There is no Boundary interface on this switch.

Interface        Role Sts Cost      Prio.Nbr Type
Gi1/0/2          Root FWD 200000    128.2    P2p
Gi1/0/4          Altn BLK 200000    128.4    P2p
```

- MST Interface State for Data Center 2 (MST Region 2—Instance 0):

```
Aggregation 1
Aggr1 - Interface 1/46 connected to Aggr1 is Designated Forwarding
Aggr1 - Interface 1/47 connected to the core (MAN) is Root Forwarding and boundary
using RSTP
Aggr1 - Interface 1/48 connected to Access switch is Designated Forwarding

Interface        Role Sts Cost      Prio.Nbr Type
Fa1/46           Desg FWD 200000    128.46   P2p
Fa1/47           Root FWD 100000    128.47   P2p Bound(RSTP)
Fa1/48           Desg FWD 200000    128.48   P2p


Aggregation 2
Aggr2 - Interface 3/46 connected to Aggr2 is Root Forwarding (active path to the root)
```

```
Aggr2 - Interface 3/47 connected to the core (MAN) is Alternate blocking and boundary
using RSTP
Aggr2 - Interface 3/48 connected to Access switch is Designated Forwarding

Interface        Role Sts Cost      Prio.Nbr Type
Fa1/46           Root FWD 200000    128.46   P2p
Fa1/47           Altn BLK 200000    128.47   P2p Bound(RSTP)
Fa1/48           Desg FWD 200000    128.48   P2p

Access 1
Acc1 - Interface 2 connected to Aggr1 is Alternate
Acc1 - Interface 4 connected to Aggr2 is Root Forwarding
There is no Boundary interface on this switch.

Interface        Role Sts Cost      Prio.Nbr Type
Gi1/0/2          Altn BLK 200000    128.2    P2p
Gi1/0/4          Root FWD 200000    128.4    P2p
```

For VLAN mapping, VLAN 601 and 602 are mapped to Instance 1, all other VLANs are mapped to Instance 2.

In these tests, VLAN 601 is used for the cluster VIP and VLAN 602 is used for the health check of the cluster's members and these are extended up to the remote data center.

In this configuration, the boundaries of the MST regions are located at the inbound interface used for the loopback cable (n/47). On those boundary ports, IST0 is used to communicate between the two MST regions.

Note The loopback cables shown in Figure 5-5 are used to allow the VLAN dedicated for the pseudowire to be switched to outside of the data center (for example, the VLAN 601 used for VIP).

Configuration

This section provides examples for the MST configuration.

Aggregation Switch Left (Primary Root Bridge for MST Region DC1)

```
spanning-tree mode mst
no spanning-tree optimize bpdu transmission
spanning-tree extend system-id
spanning-tree mst configuration
 name DC1
 revision 10
 instance 1 vlan 601-602
 instance 2 vlan 1-600, 603-4094
!
spanning-tree mst 1-2 priority 24576
```

Aggregation Switch Right (Secondary Root Bridge for MST Region DC1)

```
spanning-tree mode mst
no spanning-tree optimize bpdu transmission
spanning-tree extend system-id
spanning-tree mst configuration
 name DC1
 revision 10
 instance 1 vlan 601-602
 instance 2 vlan 1-600, 603-4094
!
```

```
spanning-tree mst 1-2 priority 28672
```

Access Switch (MST Region DC1)

```
spanning-tree mode mst
no spanning-tree optimize bpdu transmission
spanning-tree extend system-id
spanning-tree mst configuration
 name DC1
 revision 10
 instance 1 vlan 601-602
 instance 2 vlan 1-600, 603-4094
```

Aggregation Switch Left (Primary Root Bridge for MST Region DC2)

```
spanning-tree mode mst
no spanning-tree optimize bpdu transmission
spanning-tree extend system-id
spanning-tree mst configuration
 name DC2
 revision 20
 instance 1 vlan 601-602
 instance 2 vlan 1-600, 603-4094
!
spanning-tree mst 1-2 priority 24576
```

Aggregation Switch Right (Secondary Root Bridge for MST Region DC2)

```
spanning-tree mode mst
no spanning-tree optimize bpdu transmission
spanning-tree extend system-id
spanning-tree mst configuration
 name DC2
 revision 20
 instance 1 vlan 601-602
 instance 2 vlan 1-600, 603-4094
!
spanning-tree mst 1-2 priority 28672
```

Access Switch (MST Region DC2)

```
spanning-tree mode mst
no spanning-tree optimize bpdu transmission
spanning-tree extend system-id
!
spanning-tree mst configuration
 name DC2
 revision 20
 instance 1 vlan 601-602
 instance 2 vlan 1-600, 603-4094
```

EoMPLS Port-based Xconnect Redundancy with EtherChannels

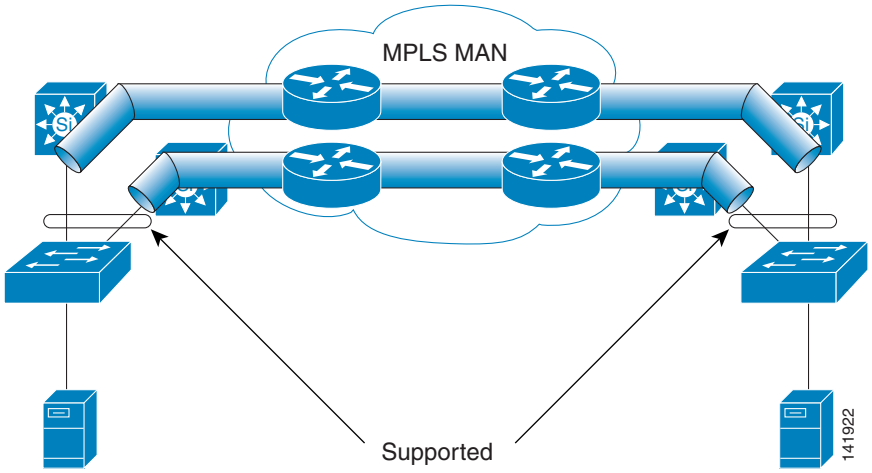
EtherChannel provides incremental trunk speeds from one to up to eight physical links, between Fast Ethernet, Gigabit Ethernet, and 10 Gigabit Ethernet. EtherChannel combines multiple Fast Ethernet up to 800 Mbps, Gigabit Ethernet up to 8 Gbps, and 10 Gigabit Ethernet up to 80 Gbps.

EtherChanneling provides an alternative to MST to keep redundant Layer 2 paths free from loops.

One option you can use to deploy an EtherChannel design while preventing a single point of failure is to encapsulate each physical link into a tunnel.

Figure 5-23 shows EtherChanneling between ports of remote switches. The switch at each data center is connected to each local aggregation switch. EtherChanneling end-to-end is possible because of the EoMPLS pseudowire that provides Layer 2 connectivity between the remote switches.

Figure 5-23 Split EtherChannel Over EoMPLS



As long as the Layer 2 VPN tunnel is maintained from end to end, the flow path can take multiple routes inside the WAN MAN, using Layer 3 or MPLS fast convergence algorithms. This keeps any convergences on the WAN MAN fully transparent for the edge devices (or access switches).

Figure 5-23 shows the logical equivalent of two access switches connected together through an EtherChannel built with two direct physical links.

Figure 5-24 Logical EtherChannel Over EoMPLS

Remote Failure Detection

In a native Layer 2 EtherChannel, if a segment within the channel fails, the traffic previously carried over the failed link switches to the remaining links within the EtherChannel.

However, this requires that the affected interface of the EtherChannel detects a physical link down. When using a Layer 2 VPN, the Logical Layer 2 link is built on top of existing transport layer (a pseudowire). This means that if one of the remote Layer 2 links (remote data center) fails for any reason, the local EtherChannel is not able to detect the remote failure as a direct physical link failure, therefore, the

EtherChannel protocol used to communicate (LACP or PAgP) times out after the certain time. With LACP (802.1ad), the timeout happens after 60 seconds, which means during 1 minute or so, the local site continues to send the traffic over this link, which is no longer terminated on the remote site.

You can use UDLD to make the detection faster.

Unidirectional Link Detection (UDLD)

UDLD has been designed and implemented by Cisco to detect unidirectional links and improve a Layer 1-Layer 2 Loop detection, usually controlled by a Layer 2 algorithm, such STP or RSTP.

UDLD is a Layer 2 protocol that works in conjunction with Layer 1 mechanisms to determine the physical status of a link. At Layer 1, auto-negotiation takes care of physical signaling and fault detection. UDLD detects the identities of neighbors and shuts down misconnected ports. When enabling both auto-negotiation and UDLD, Layer 1 and Layer 2 detections work together to prevent physical and logical unidirectional connections and the malfunctioning of other protocols.

UDLD works by exchanging protocol packets between the neighboring devices. In order for UDLD to work, both devices on the link must support UDLD and have it enabled on their respective ports.

Each switch port configured for UDLD sends UDLD protocol packets containing the port's own device or port ID, and the neighbor's device or port IDs, as seen by UDLD on that port. Neighboring ports should see their own device or port ID (echo) in the packets received from the other side. If the port does not see its own device or port ID in the incoming UDLD packets for a specific duration of time, the link is considered unidirectional. This heartbeat, based on a echo-algorithm, allows detection of several issues, such as damaged wiring, fiber mistakes, or, in this design, a remote link failure after a specific timeout.

To take advantage of EtherChannel, it is important to enable UDLD to improve the detection of a remote failure before the timeout. Port-based Xconnect of the Layer 2 VPN tunnel must be created at the ingress port directly linked to the access layer, as shown in Figure 5-23. This architecture is described in Figure 5-4. Until recently, the default message interval value was 7 seconds; therefore, the timeout before detecting a remote link failure was 21 seconds. This value is not fast enough for a cluster, which requires a maximum of 10 seconds to detect a remote link failure and keep the cluster recovery algorithm working as expected.

The minimum message interval has recently been reduced to 1. Deploying UDLD with this low interval can cause problems. Some tests through long distances (>20kms) have been conducted with the message interval value set to 1 and these have shown a very unstable behavior of UDLD. The minimum message interval value setting for UDLD, to keep it stable over long distances, is >3-4 seconds. Therefore, the minimum time to detect a remote failure becomes between 12 and 15 seconds, a value still insufficient for HA clusters. UDLD requires more analysis to understand why it becomes unstable when using 1 sec for the message interval.

UDLD Modes

UDLD can operate in two modes:

- Normal mode
- Aggressive mode

In normal mode, if the link state of the port is determined to be bi-directional and the UDLD information times out, no action is taken by UDLD. The port state for UDLD is marked as undetermined. The port behaves according to its STP state.

In aggressive mode, if the link state of the port is determined to be bi-directional and the UDLD information times out while the link on the port is still up, UDLD tries to re-establish the state of the port. If unsuccessful, the port is put into an errdisable state. This requires a manual action to re-enable the port using the following command:

```
switch#udld reset
```

It is up to the network manager to decide if any disruption on the tunnel should be transparent and dynamically re-initiate, or if it should force a manual reset.

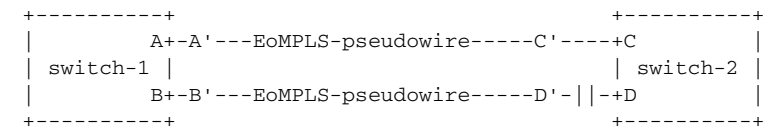
UDLD Configuration

From a global setting, define the same message interval on both access switches:

```
switch#udld message time 4

On each uplink interface enable UDLD

interface GigabitEthernet1/0/2
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 1,601,602
 switchport mode trunk
 load-interval 30
 udld port aggressive
 no mdix auto
```

EoMPLS Port-based Xconnect Redundancy with Spanning Tree

Spanning tree is an alternative to the use of EtherChannels, in that it can keep the topology free from loops. By using 802.1w, the convergence time for any failure is around ~4s, which is better than what you can achieve with EtherChannels combined with UDLD. The spanning tree BPDUs perform the role of UDLD frames by verifying the availability of the path between the sites (one BPDUs is sent every two seconds). The topology with spanning tree follows. The main drawback is that one link is blocking, while with EtherChannels, all links are used.



For example, consider two simple failures. When C' fails, there is almost no traffic loss. The reason is that switch-2 switches D into Forwarding immediately, while on switch-1 traffic gets flooded on both port A and port B.

On switch-1, before the link going down on the remote switch, you have the following entries in the Layer 2 forwarding table:

```
20    0000.0c07.ac01    DYNAMIC     Fa1/0/15
20    0005.5f0b.2800    DYNAMIC     Gi1/0/1 <<<<<<
20    0011.bb0f.b301    DYNAMIC     Gi1/0/1 <<<<<<
20    0030.4880.4d1f    DYNAMIC     Fa1/0/5
20    0030.4880.4d23    DYNAMIC     Gi1/0/1 <<<<<<
20    00d0.020e.7400    DYNAMIC     Fa1/0/15
```

After the failure of port C, you have the following entries:

```
20    0000.0c07.ac01    DYNAMIC     Fa1/0/15
20    0005.5f0b.2800    DYNAMIC     Gi2/0/2
20    0011.bb0f.b31b    DYNAMIC     Gi2/0/2
20    0030.4880.4d1f    DYNAMIC     Fa1/0/5
20    0030.4880.4d23    DYNAMIC     Gi2/0/2
20    00d0.020e.7400    DYNAMIC     Fa1/0/15
```

In the case of port A' failing, the downtime is higher; there is packet drop for ~4s. The reason is that it takes longer for switch-2 to put the alternate port into forwarding mode.

```
Interface        Role Sts Cost      Prio.Nbr Type
---------------- ---- --- --------- --------------
Gi1/0/1          Root FWD 20000     128.1    P2p
Fa1/0/3          Desg FWD 200000    128.5    Edge P2p
Fa1/0/15         Desg FWD 200000    128.17   P2p
Gi1/0/3          Altn BLK 20000     128.27   P2p

Switch-2#
Nov  1 16:17:02: %SPANTREE-5-TOPOTRAP: Topology Change Trap for vlan 21
Nov  1 16:17:03: %SPANTREE-2-LOOPGUARD_BLOCK: Loop guard blocking port
GigabitEthernet1/0/1 on VLAN0020.
Nov  1 16:17:03: %SPANTREE-5-TOPOTRAP: Topology Change Trap for vlan 20

Switch-2#show spanning-tree vlan 20

Interface        Role Sts Cost      Prio.Nbr Type
---------------- ---- --- --------- --------
Gi1/0/1          Root BKN*20000     128.1    P2p *LOOP_Inc
Fa1/0/3          Desg FWD 200000    128.5    Edge P2p
Fa1/0/15         Desg FWD 200000    128.17   P2p
Gi1/0/3          Root FWD 20000     128.27   P2p
```

Layer 2 loops with this design are a concern, but they are less disruptive than in a regular LAN design. The maximum bandwidth that they can use from the MPLS core is limited by the link bandwidth connecting the CE switch to the aggregation switch. As long as the EoMPLS link carries only the LAN extension traffic, and the FC traffic uses the MPLS network or another transport, a Layer 2 loop is going to cause a high bandwidth utilization on the local and remote LAN. However, it is not going to make the aggregation switches unmanageable, and it is not going to cause the storage arrays to be in a split state. Design the network to avoid loops (spanning tree is used for this purpose) because bugs or other mistakes (such as configuring the teamed NICs of a server for forwarding) can potentially introduce loops.