Introduction to MPLS

APNIC

Issue Date: [201609]
Revision: [01]

APNIC

What is MPLS?

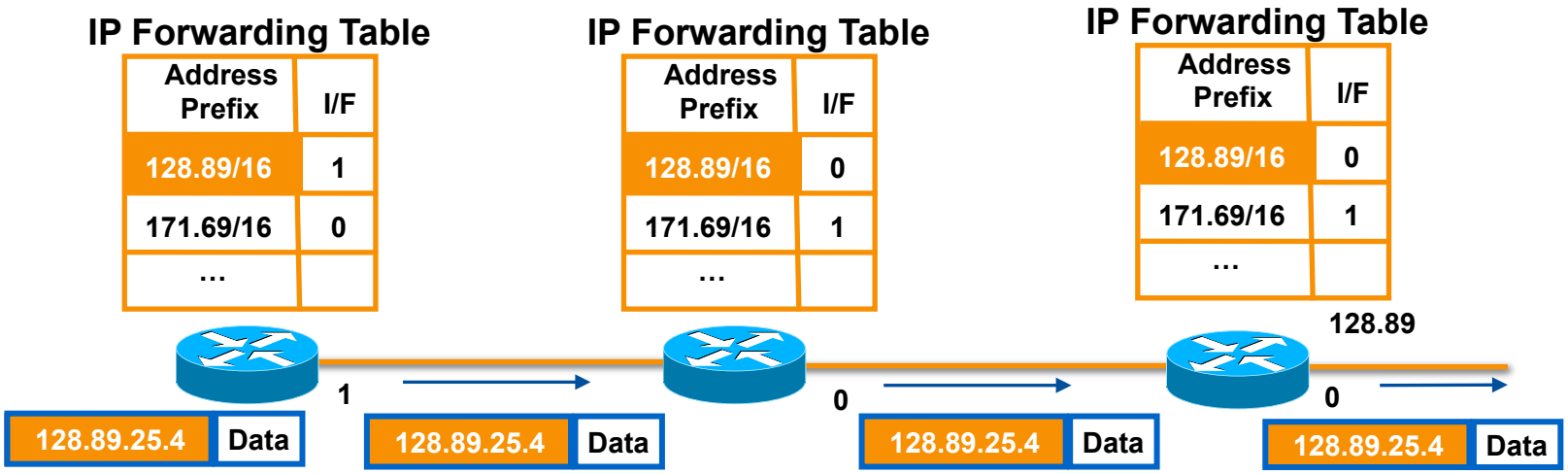
APNIC

Definition of MPLS

- **M**ulti **P**rotocol **L**abel **S**witching
  - Multiprotocol, it supports ANY network layer protocol, i.e. IPv4, IPv6, IPX, CLNP, etc.
  - A short label of fixed length is used to encapsulate packets
  - Packets are forwarded by label switching instead of by IP switching

Initial Motivation of MPLS

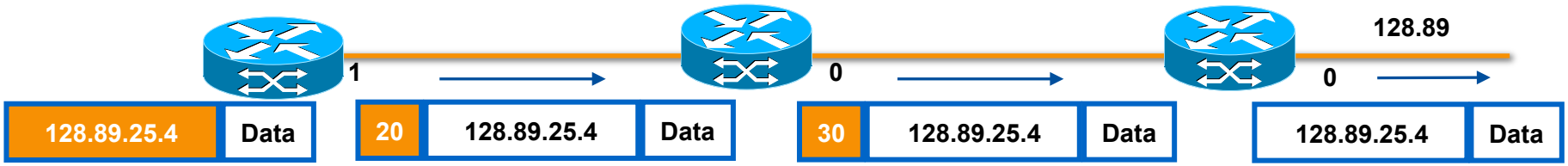
- In mid 1990s, IP address lookup was considered more complex and take longer time.
  - Longest matching



A label-swapping protocol was the need for speed.

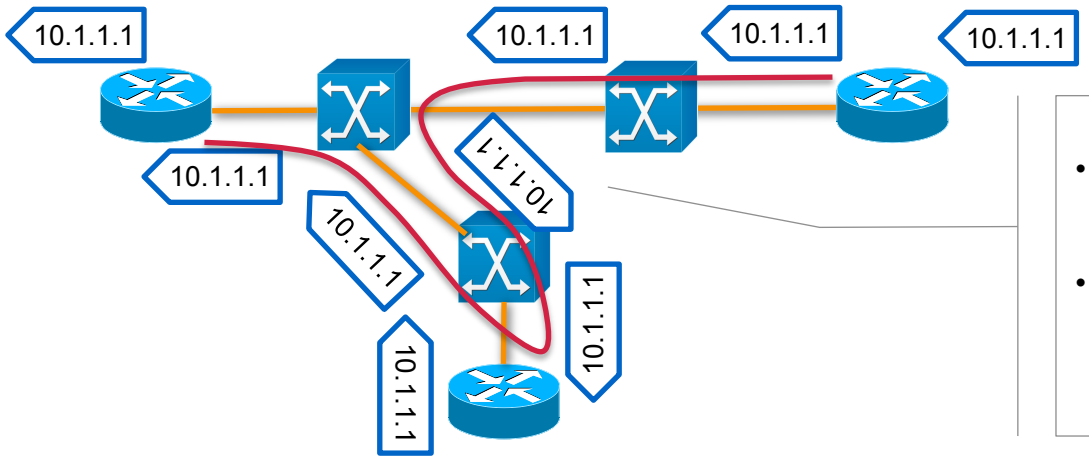
Decoupling Routing and Forwarding

- MPLS can allow core routers to switch packets based on some simplified header.



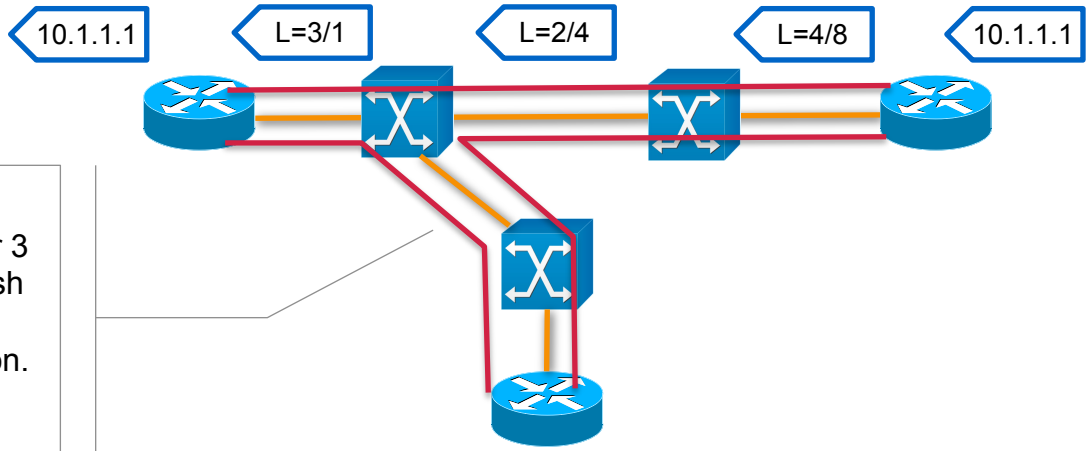
- But, hardware of routers became better and looking up longest best match was no longer an issue.
- More importantly, MPLS de-couples forwarding from routing, and support multiple service models.

MPLS vs IP over ATM



IP over ATM

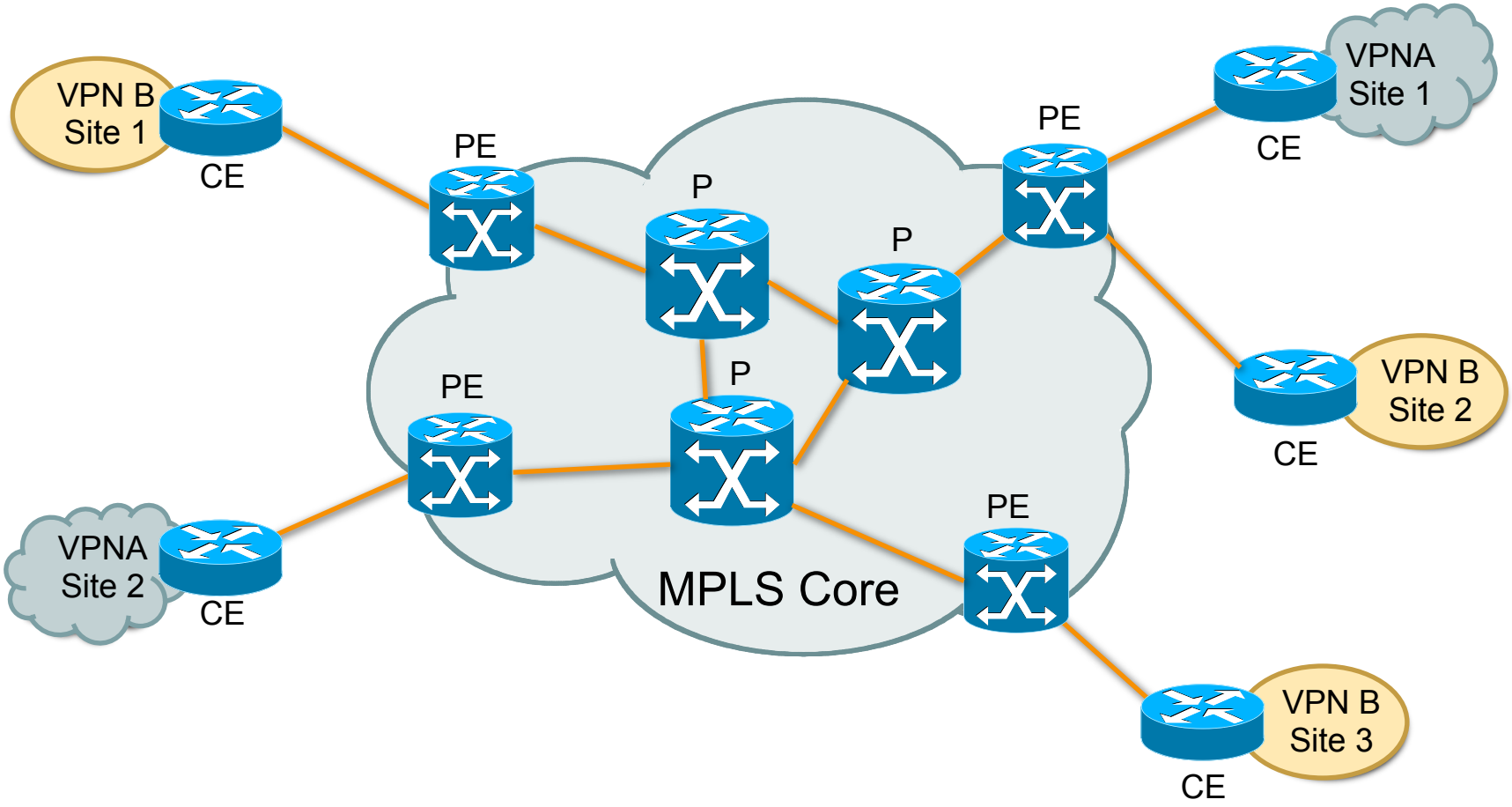
- Layer 2 topology may be different from Layer 3 topology, resulting in suboptimal paths.
- Layer 2 devices have no knowledge of Layer 3 routing – virtual circuits must be manually established.



MPLS

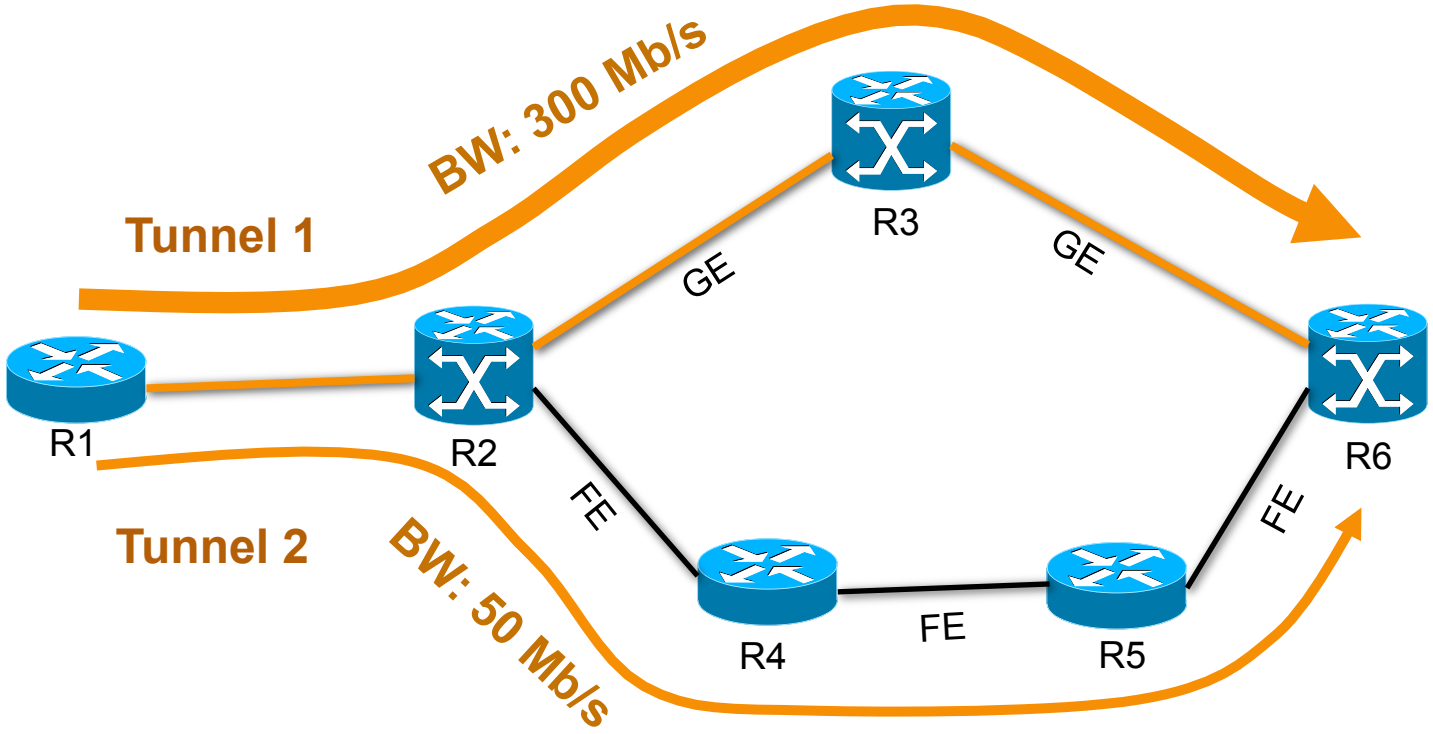
- Layer 2 devices run a Layer 3 routing protocol and establish virtual circuits dynamically based on Layer 3 information.
- MPLS provides a virtual full mesh topology.

MPLS VPN



- MPLS Layer 3/ Layer 2 VPN

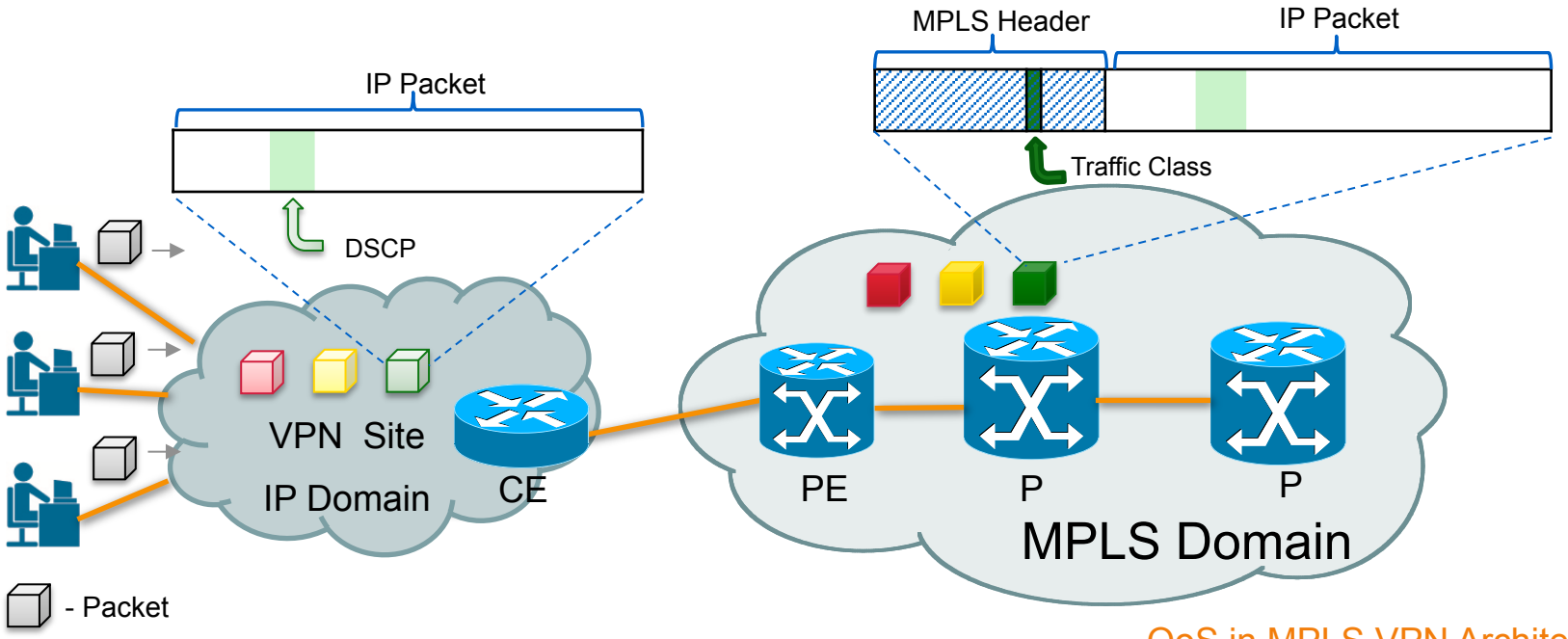
Optimal Traffic Engineering



IP TE	MPLS TE
Shortest path	Determines the path at the source based on additional parameters (available resources and constraints, etc.)
Equal cost load balancing	Load sharing across unequal paths can be achieved.

MPLS QoS

- MPLS does NOT define a new QoS architecture.
  - Similar parts with IP DiffServ: functional components and where they are used.(such as marking and traffic policing at network edge, etc)
  - Difference: packets are differentiated by MPLS Traffic Class bits



QoS in MPLS VPN Architecture

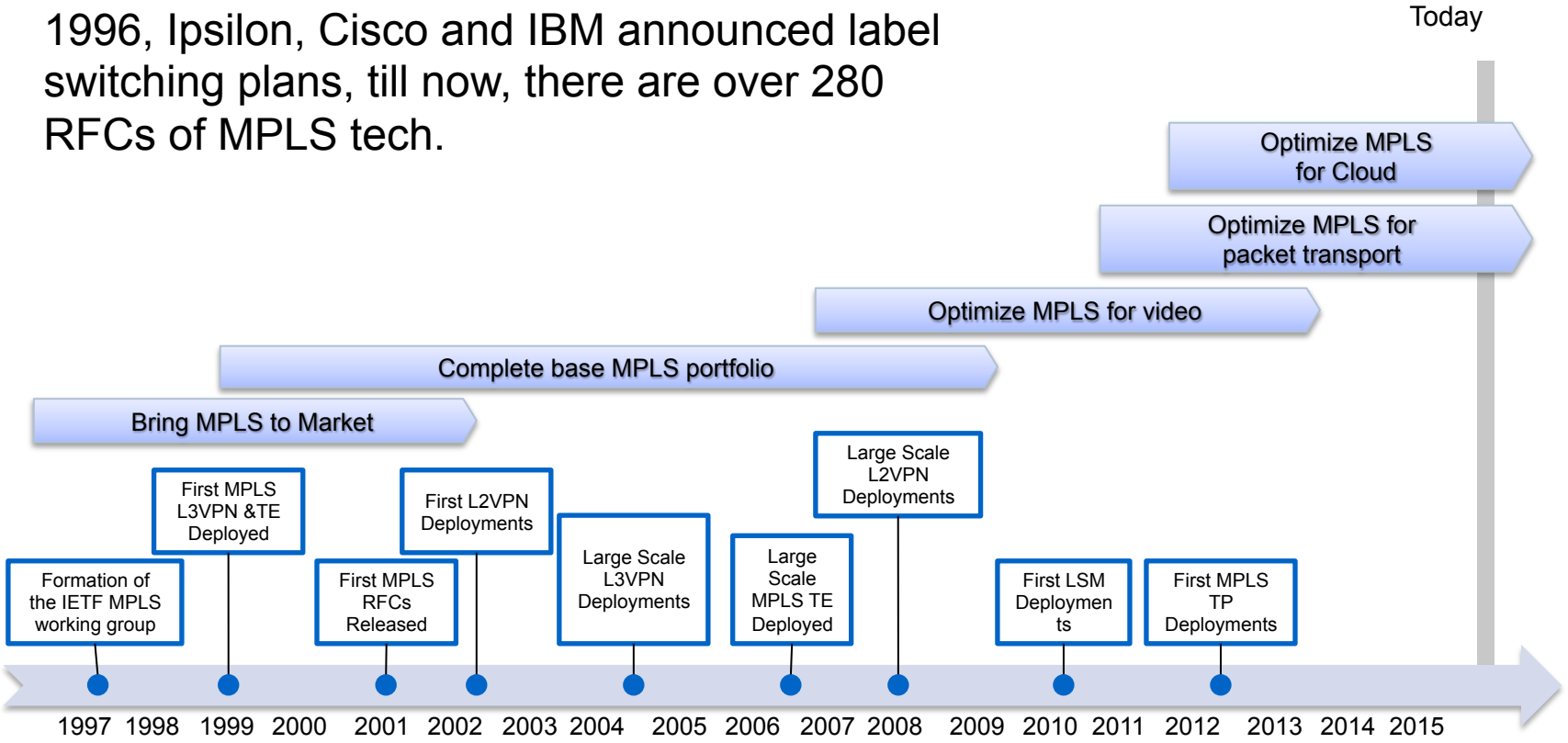
Technology Comparison

	IP	Native Ethernet	MPLS
Forwarding	• Destination address based • Forwarding table learned from control plane • TTL support	• Destination address based • Forwarding table learned from data plane • No TTL support	• Label based • Forwarding table learned from control plane • TTL support
Control Plane	Routing protocols	Ethernet loop avoidance and signaling protocols	Routing protocols Label distribution protocols
Packet Encapsulation	IP header	802.3 header	MPLS Header
QoS	8 bit TOS in IP header	3 bit 802.1p in VLAN tag	3 bit TC in label
OAM	IP Ping, traceroute	E-OAM	MPLS Ping, traceroute

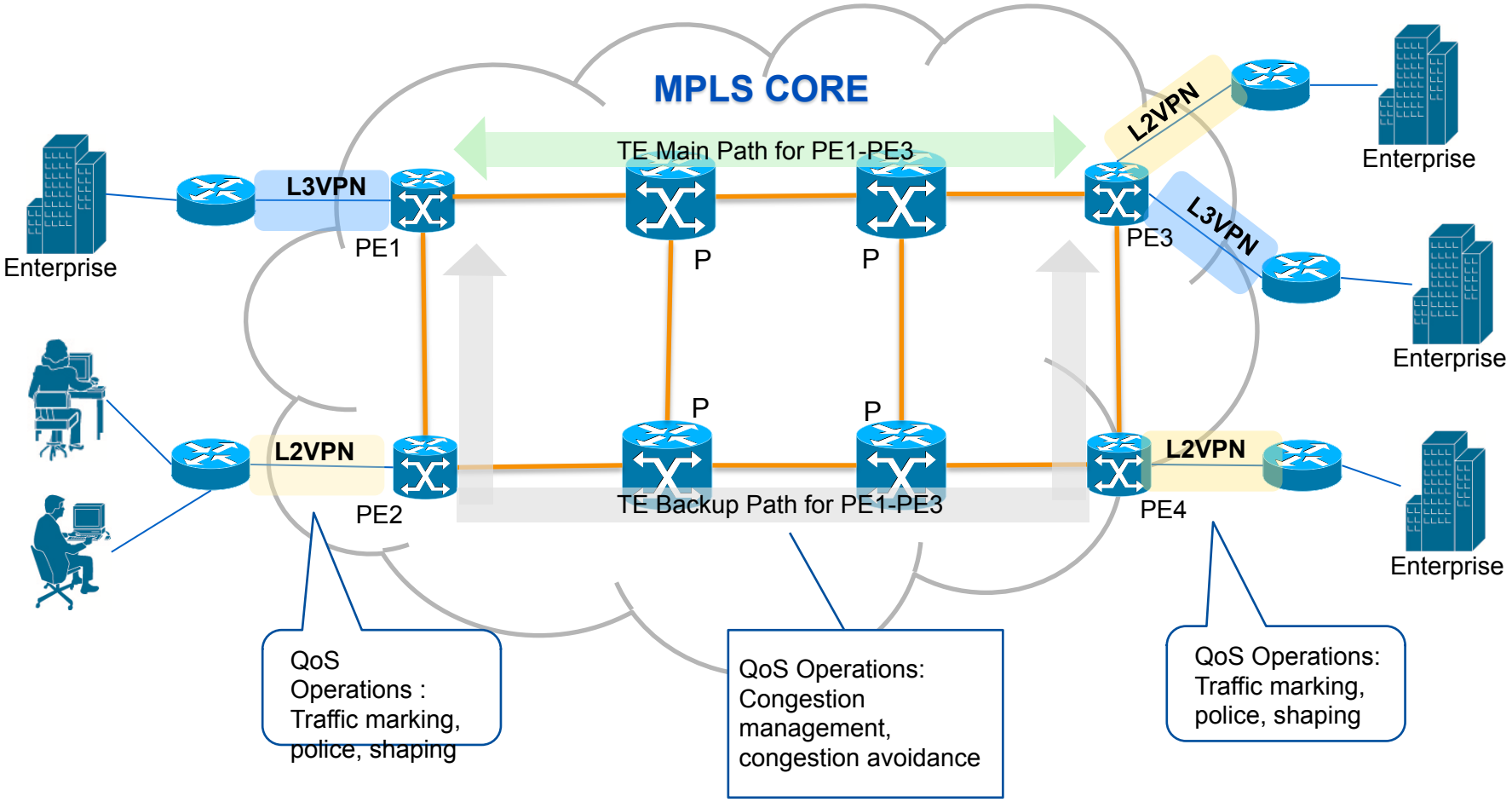
Evolution of MPLS

- Technology Evolution and Main Growth Areas

1996, Ipsilon, Cisco and IBM announced label switching plans, till now, there are over 280 RFCs of MPLS tech.



MPLS Application Scenario



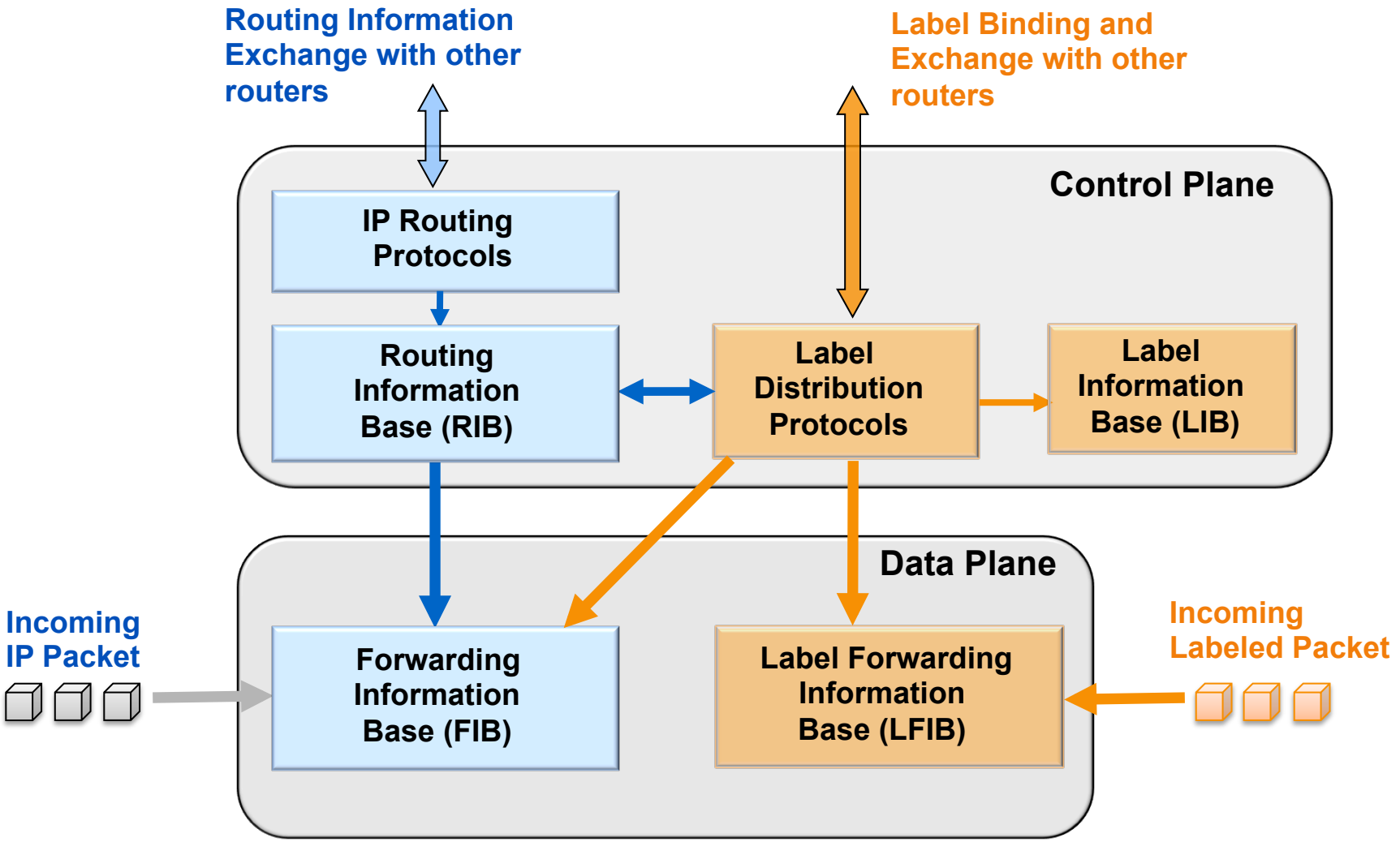
Questions?

APNIC

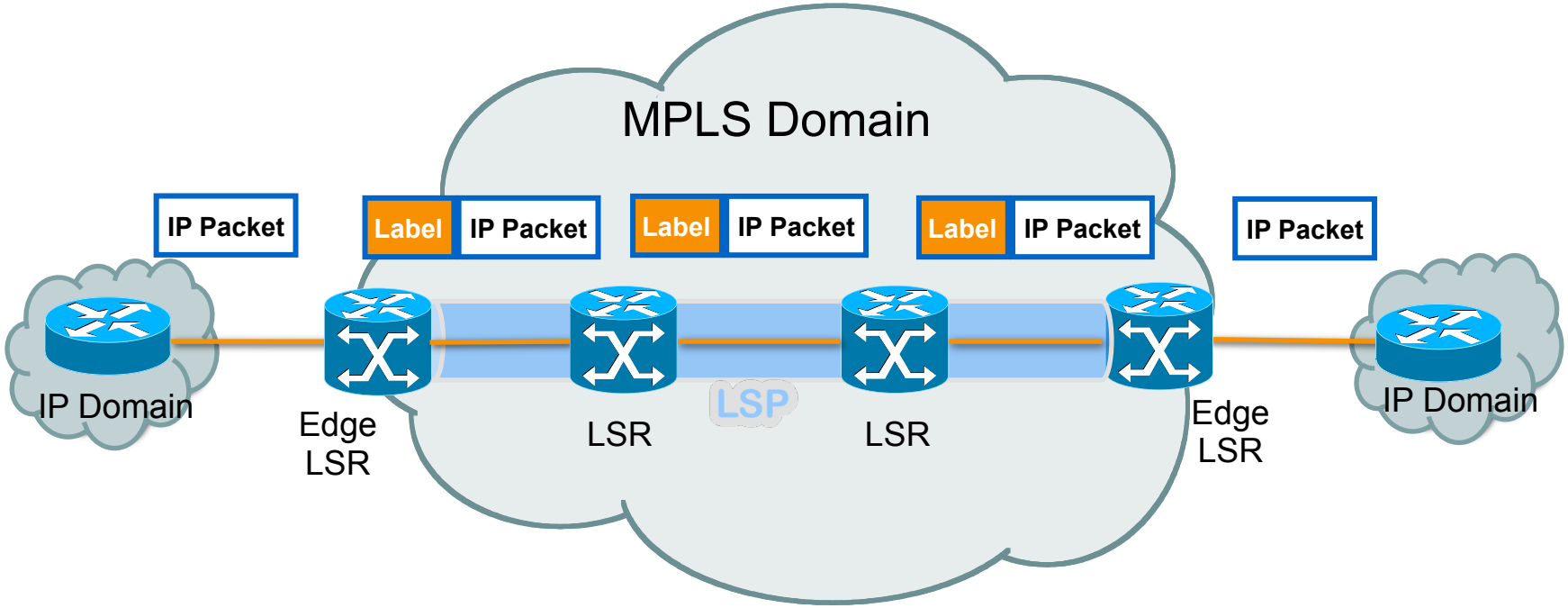
MPLS Technology Basics

APNIC

MPLS Architecture

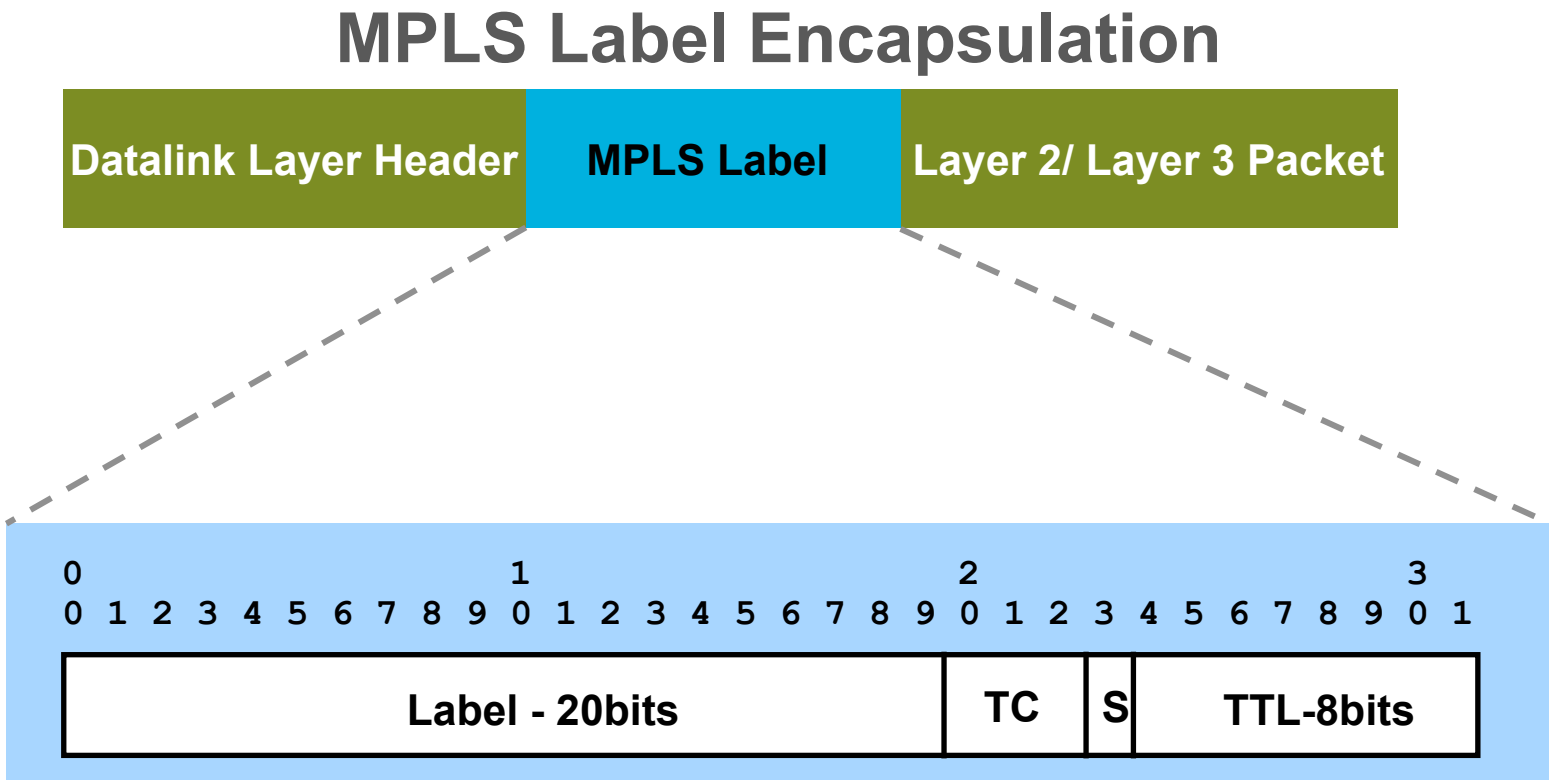


MPLS Topology



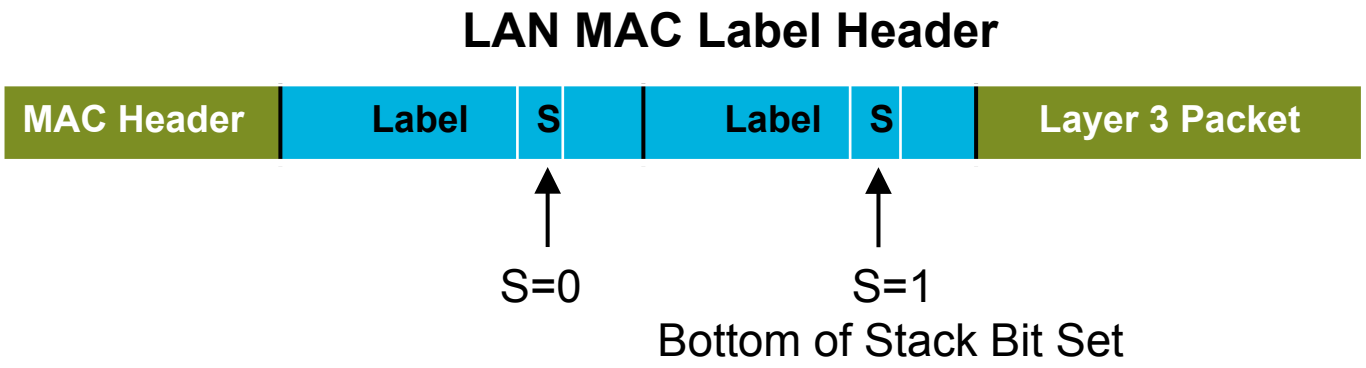
- LSR (Label Switch Router) is a router that supports MPLS.
- LER (Label Edge Router), also called edge LSR, is an LSR that operates at the edge of an MPLS network.
- LSP (Label Switched Path) is the path through the MPLS network or a part of it that packets take.

MPLS Label



TC = Traffic Class: 3 Bits; S = Bottom of Stack: 1 Bit; TTL = Time to Live

MPLS Label Stacking

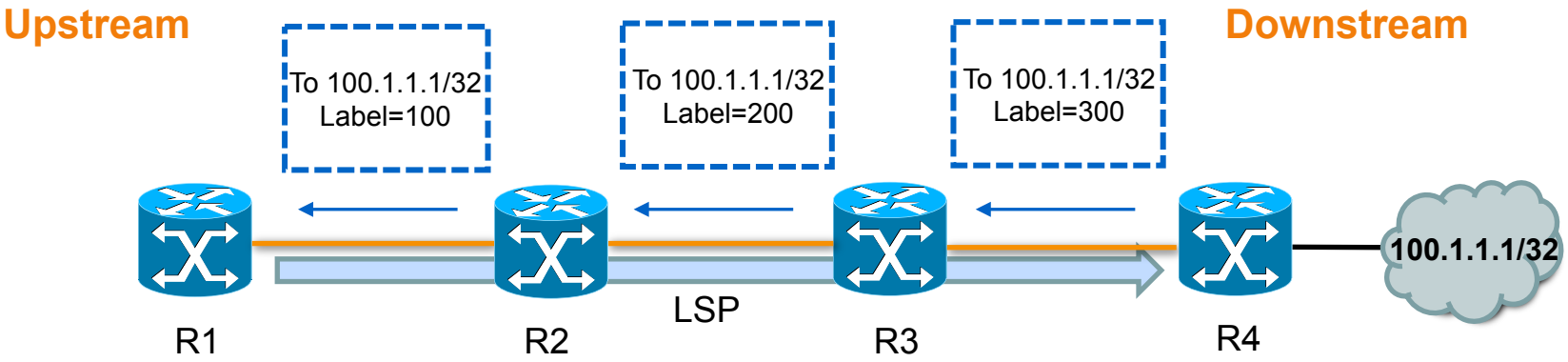


MPLS Label Stack

- Multiple labels can be used for MPLS packet encapsulation. network. This is done by packing the labels into a stack.
- Some MPLS applications (VPN, etc.) actually need more than one labels in the label stack to forward the labeled packets.

LSP Setup Overview

- Before forwarding packets, labels must be allocated to establish an LSP.
- Protocols for label distribution: LDP, RSVP-TE, MP-BGP.



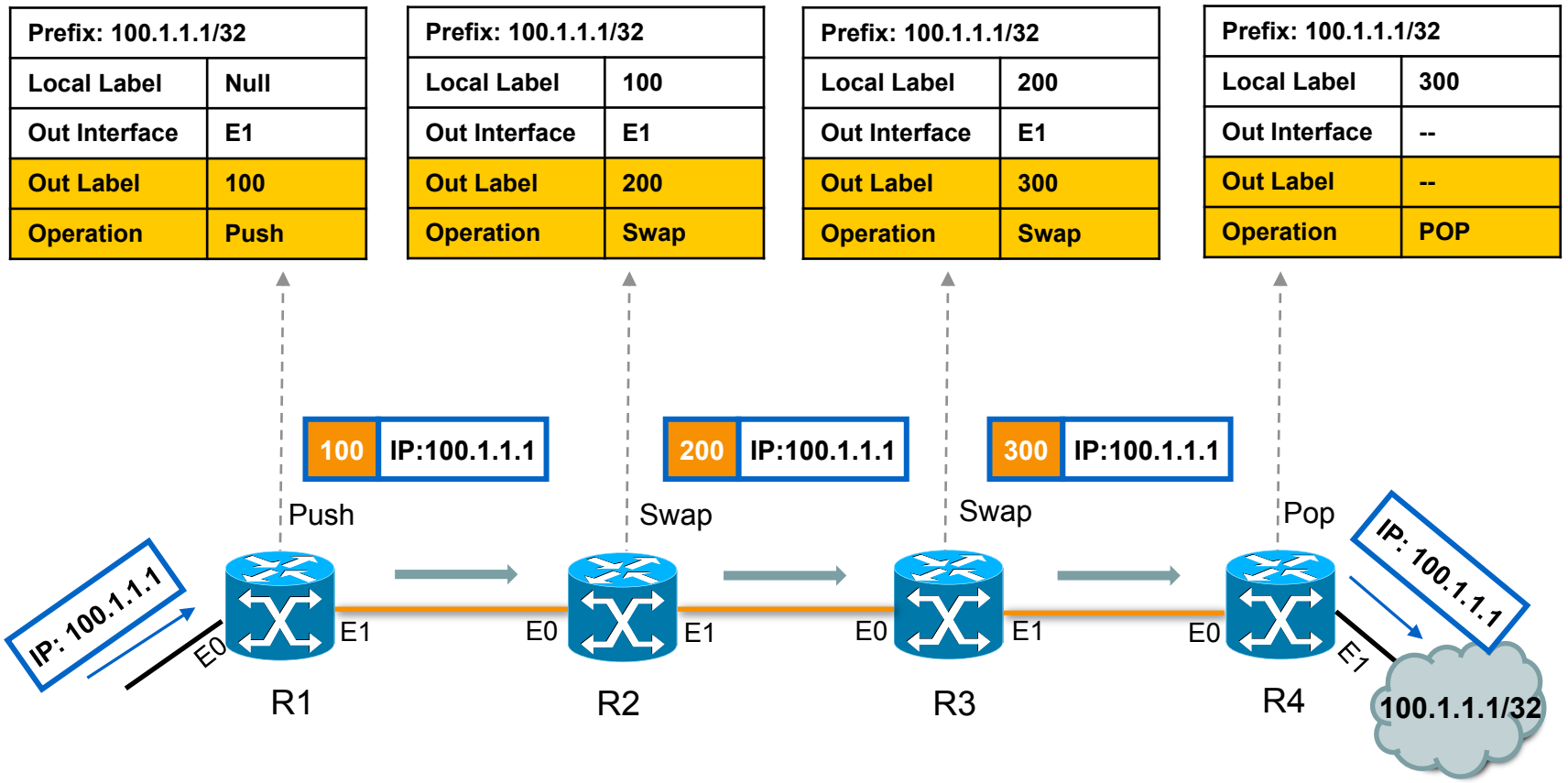
Establishing an LSP

Labels are allocated from downstream LSRs to upstream LSRs.

Basic Concepts of MPLS Forwarding

- **FEC**
  - Forwarding Equivalence Class, is a group or flow of packets that are forwarded along the same path and are treated the same with regard to the forwarding treatment.
  - For example, packets with Layer 3 destination IP address matching a certain prefix.
- **Push**
  - A new label is added to the packet between the Layer 2 header and the IP header or to the top of the label stack.
- **Swap**
  - The top label is removed and replaced with a new label.
- **Pop**
  - The top label is removed. The packet is forwarded with the remaining label stack or as an unlabeled packet.

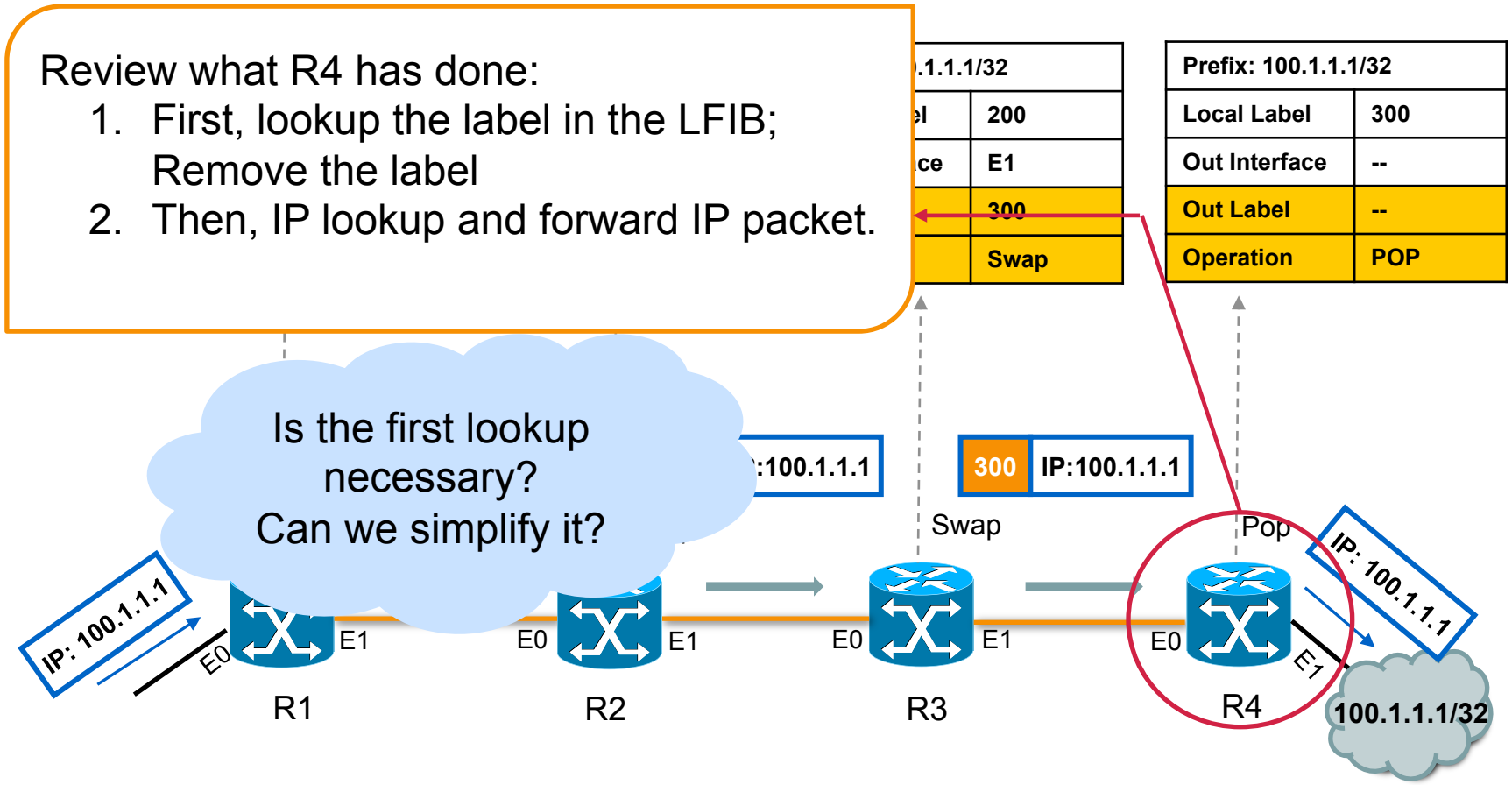
MPLS Forwarding Operations



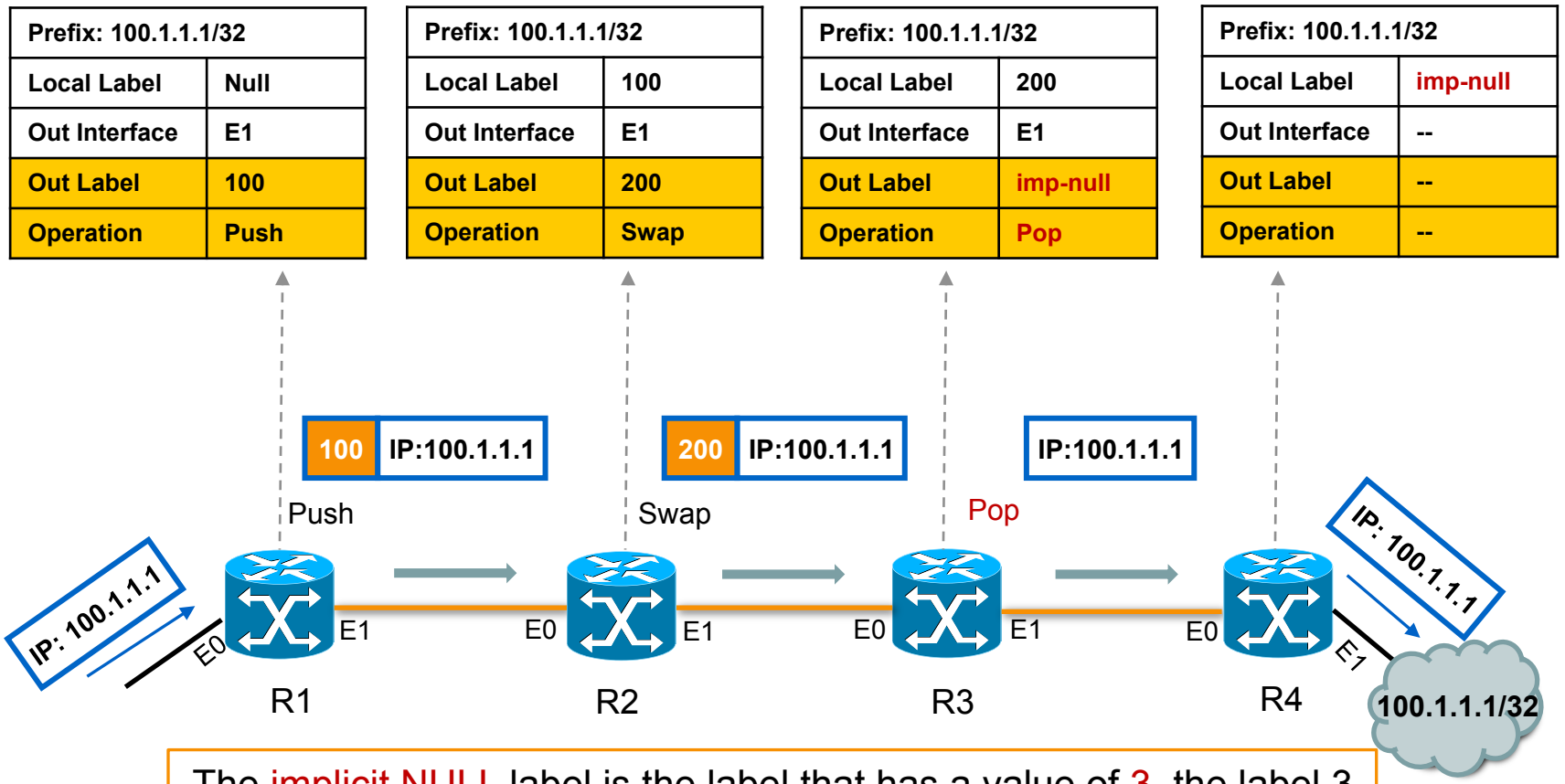
Why Penultimate Hop Popping?

Review what R4 has done:

1. First, lookup the label in the LFIB; Remove the label
2. Then, IP lookup and forward IP packet.

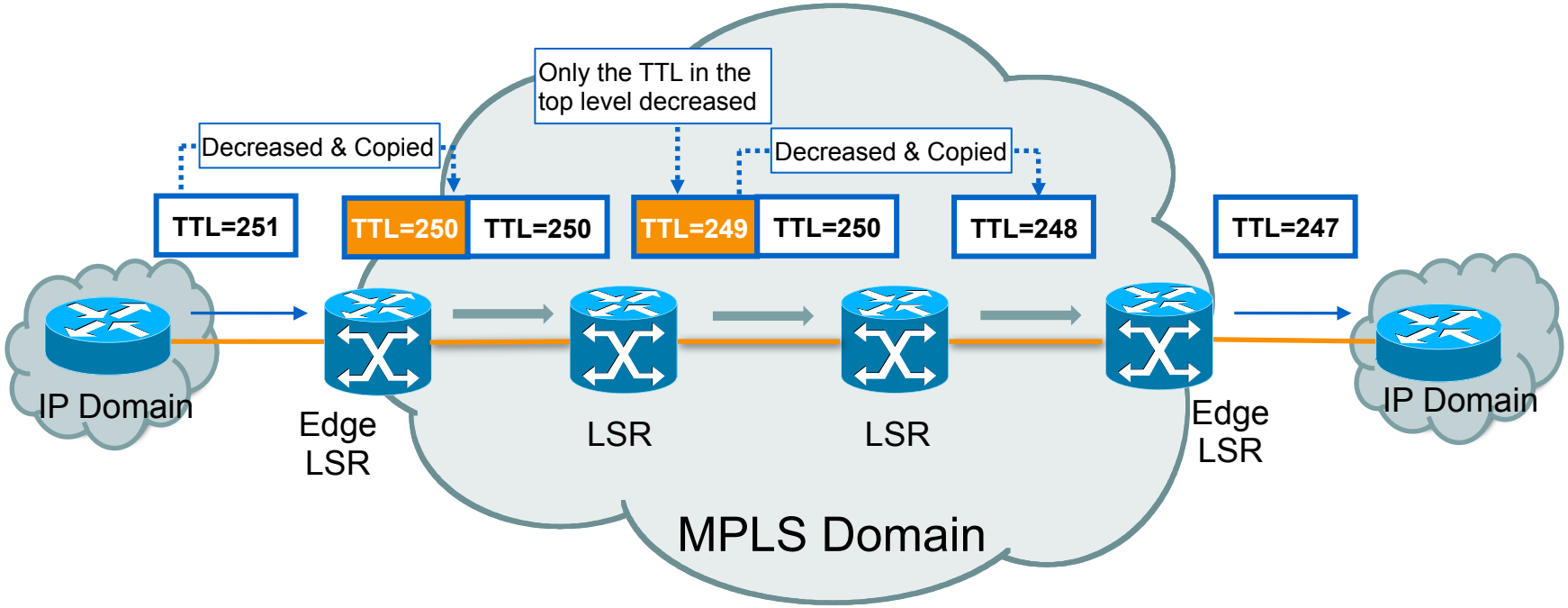


Penultimate Hop Popping



The implicit NULL label is the label that has a value of 3, the label 3 will never be seen as a label in the label stack of an MPLS packet.

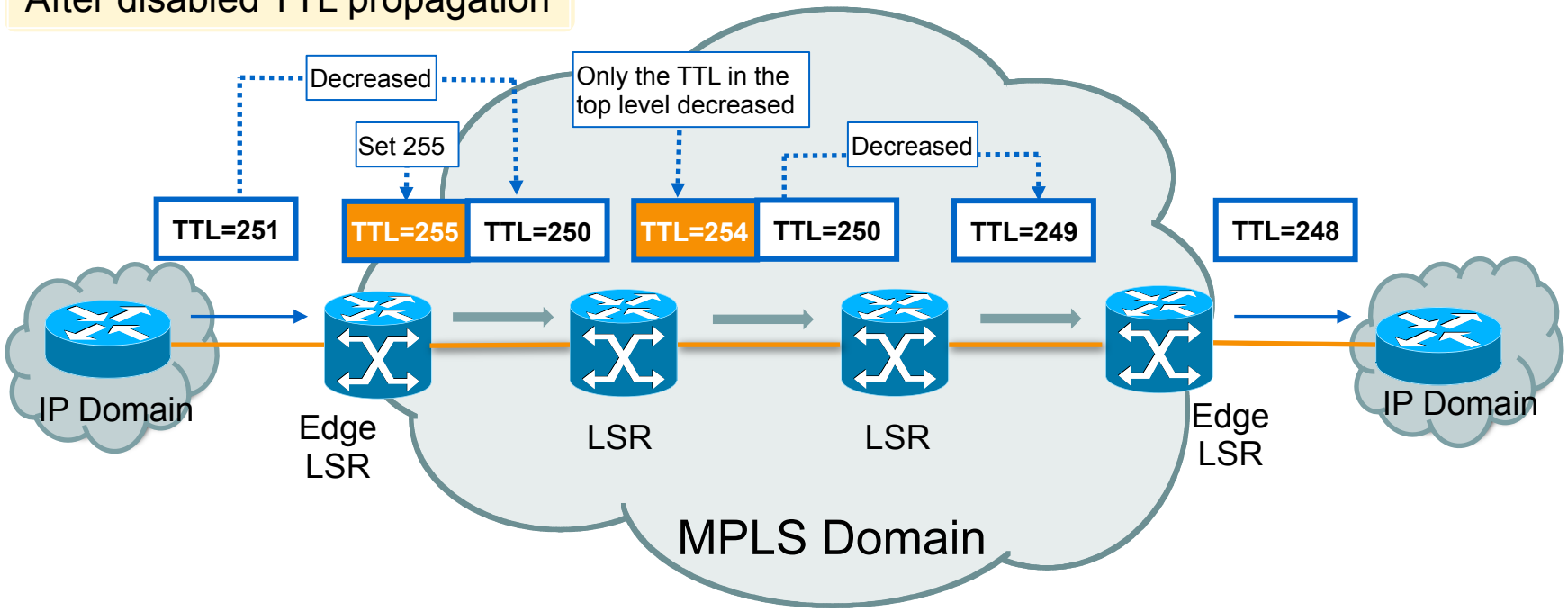
MPLS TTL Processing (1)



- MPLS processes the TTL to prevent loops and implement traceroute.
- By default, TTL propagation is enabled as above.

MPLS TTL Processing (2)

After disabled TTL propagation

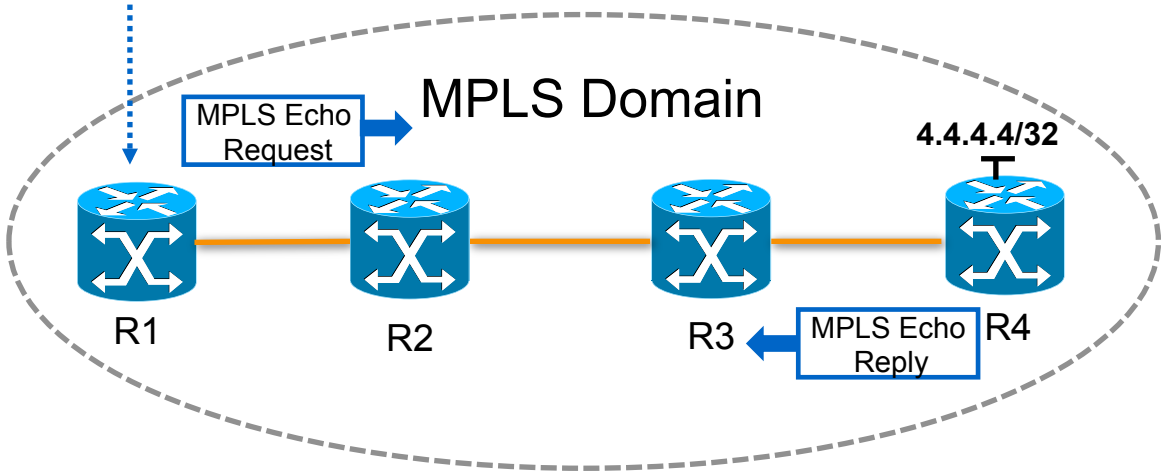


- TTL propagation can be disabled to hide the MPLS network topology.
- Disabling TTL propagation makes routers set the value 255 into the TTL field of the label when an IP packet is labeled.

MPLS LSP Ping

```
R1#ping mpls ipv4 4.4.4.4/32
Sending 5, 100-byte MPLS Echos to 4.4.4.4/32,
     timeout is 2 seconds, send interval is 0 msec:
Codes: '!' - success, 'Q' - request not sent, '.' - timeout,
  'L' - labeled output interface, 'B' - unlabeled output interface,
  'D' - DS Map mismatch, 'F' - no FEC mapping, 'f' - FEC mismatch,
  'M' - malformed request, 'm' - unsupported tlvs, 'N' - no label entry,
  'P' - no rx intf label prot, 'p' - premature termination of LSP,
  'R' - transit router, 'I' - unknown upstream index,
  'l' - Label switched with FEC change, 'd' - see DDMAP for return code,
  'X' - unknown return code, 'x' - return code 0

Type escape sequence to abort.
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 12/14/16 ms
 Total Time Elapsed 128 ms
```

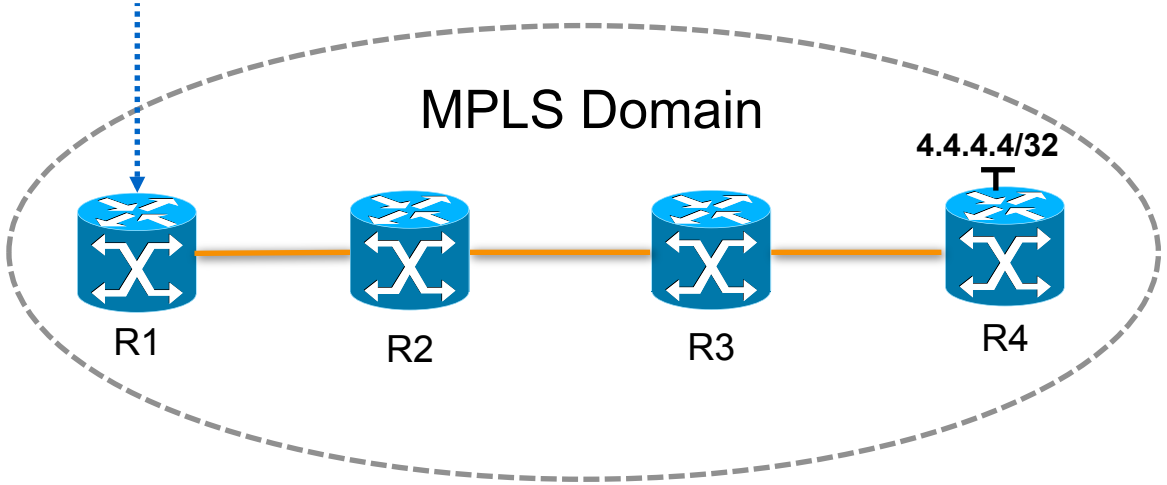
Cisco IOS



MPLS LSP Trace

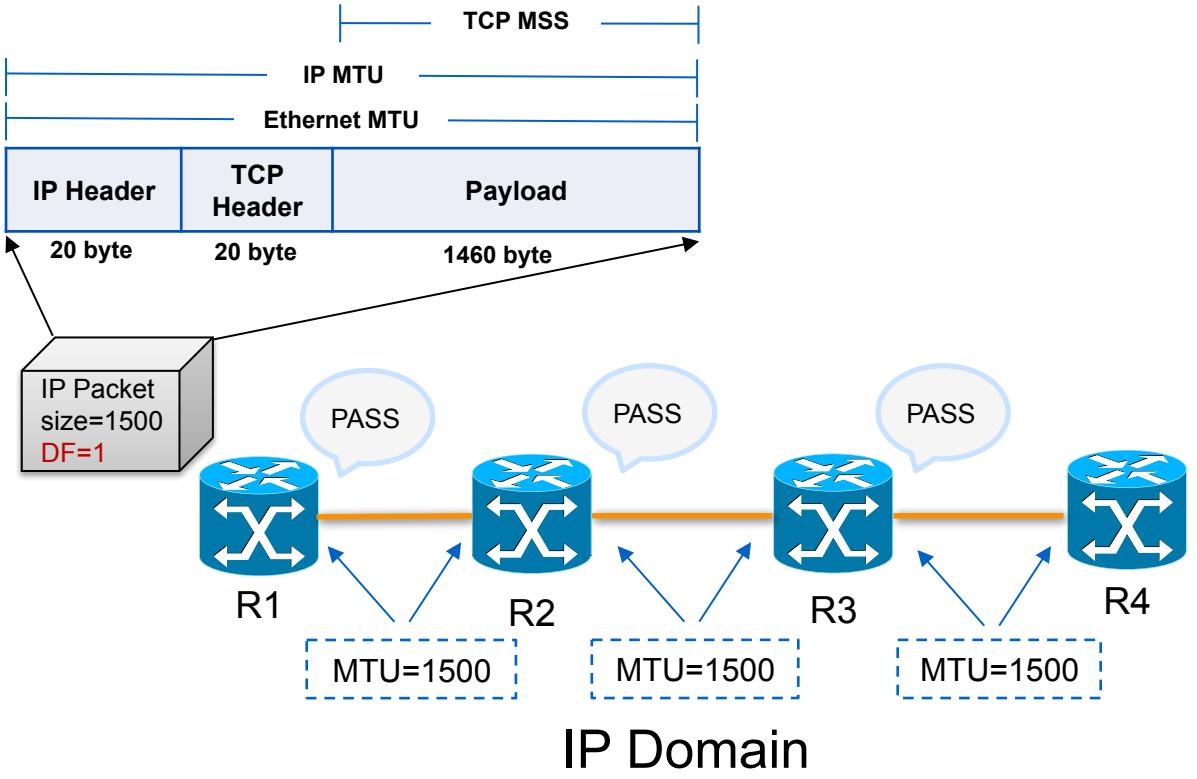
```
R1#traceroute mpls ipv4 4.4.4.4/32
Tracing MPLS Label Switched Path to 4.4.4.4/32, timeout is 2 seconds
Codes: '!' - success, 'Q' - request not sent, '.' - timeout,
  'L' - labeled output interface, 'B' - unlabeled output interface,
  'D' - DS Map mismatch, 'F' - no FEC mapping, 'f' - FEC mismatch,
  'M' - malformed request, 'm' - unsupported tlvs, 'N' - no label entry,
  'P' - no rx intf label prot, 'p' - premature termination of LSP,
  'R' - transit router, 'I' - unknown upstream index,
  'l' - Label switched with FEC change, 'd' - see DDMAP for return code,
  'X' - unknown return code, 'x' - return code 0

Type escape sequence to abort.
  0 12.1.1.1 MRU 1500 [Labels: 200 Exp: 0]
L 1 12.1.1.2 MRU 1500 [Labels: 19 Exp: 0] 16 ms
L 2 23.1.1.2 MRU 1504 [Labels: implicit-null Exp: 0] 12 ms
! 3 34.1.1.2 12 ms
```

Cisco IOS



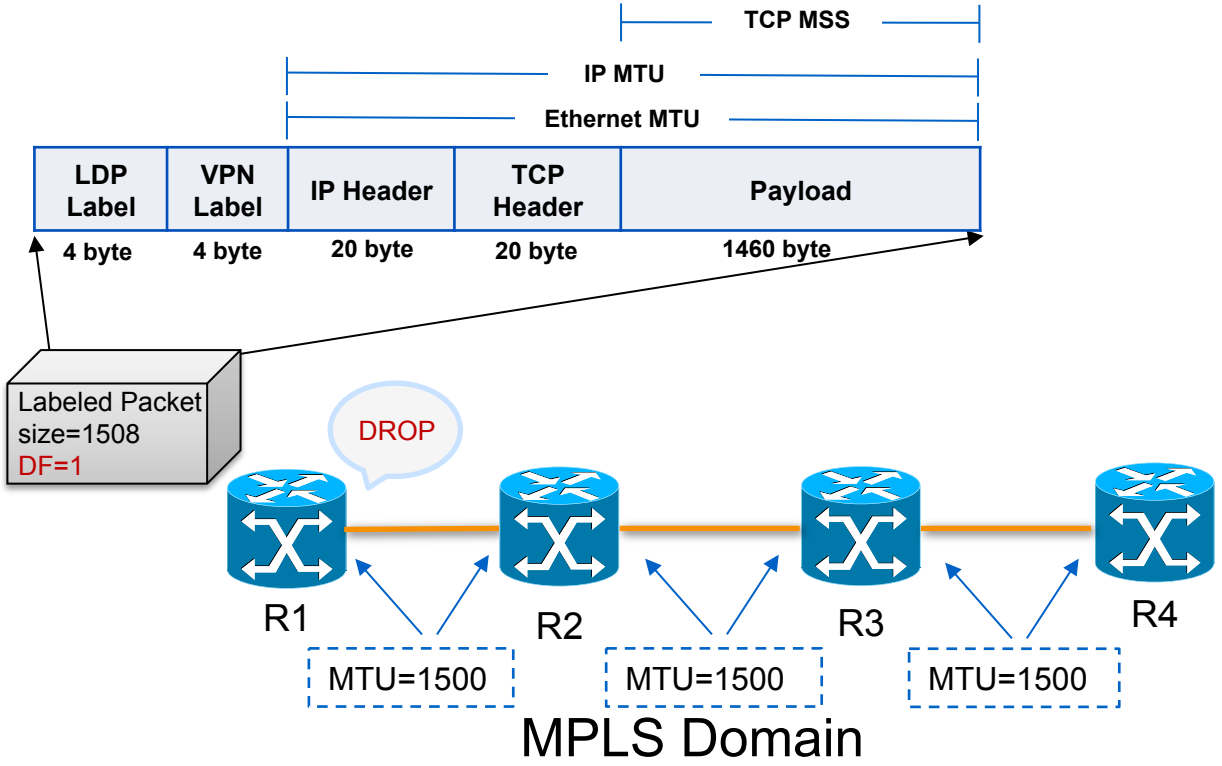
IP MTU

- MTU indicates the maximum size of the IP packet that can still be sent on a data link, without fragmenting the packet.



MPLS MTU Issue

- In MPLS L3VPN network, 2 labels are added into the packet, the labeled packets are slightly bigger than the IP packets. This would lead to the need to fragment the packet.



How to Optimize Fragmentation?

- Solution 1. Change MPLS MTU: Make sure that you configure this value on all the links in the path so that the packets are not dropped.

```
R1(config)#interface ethernet1/0
R1(config-if)#mpls mtu 1508
R1#show mpls interfaces Ethernet 1/0 detail
Interface Ethernet1/0:
 IP labeling enabled
 LSP Tunnel labeling not enabled
 BGP labeling not enabled
 MPLS not operational
 MTU = 1508
```

- Solution 2. Change the TCP MSS to be smaller:

```
R1(config)#interface ethernet 1/0
R1(config-if)#ip tcp adjust-mss 1452
```

Questions?

APNIC

Label Distribution Protocol and Basic MPLS Configuration

APNIC

Issue Date: [201609]
Revision: [01]

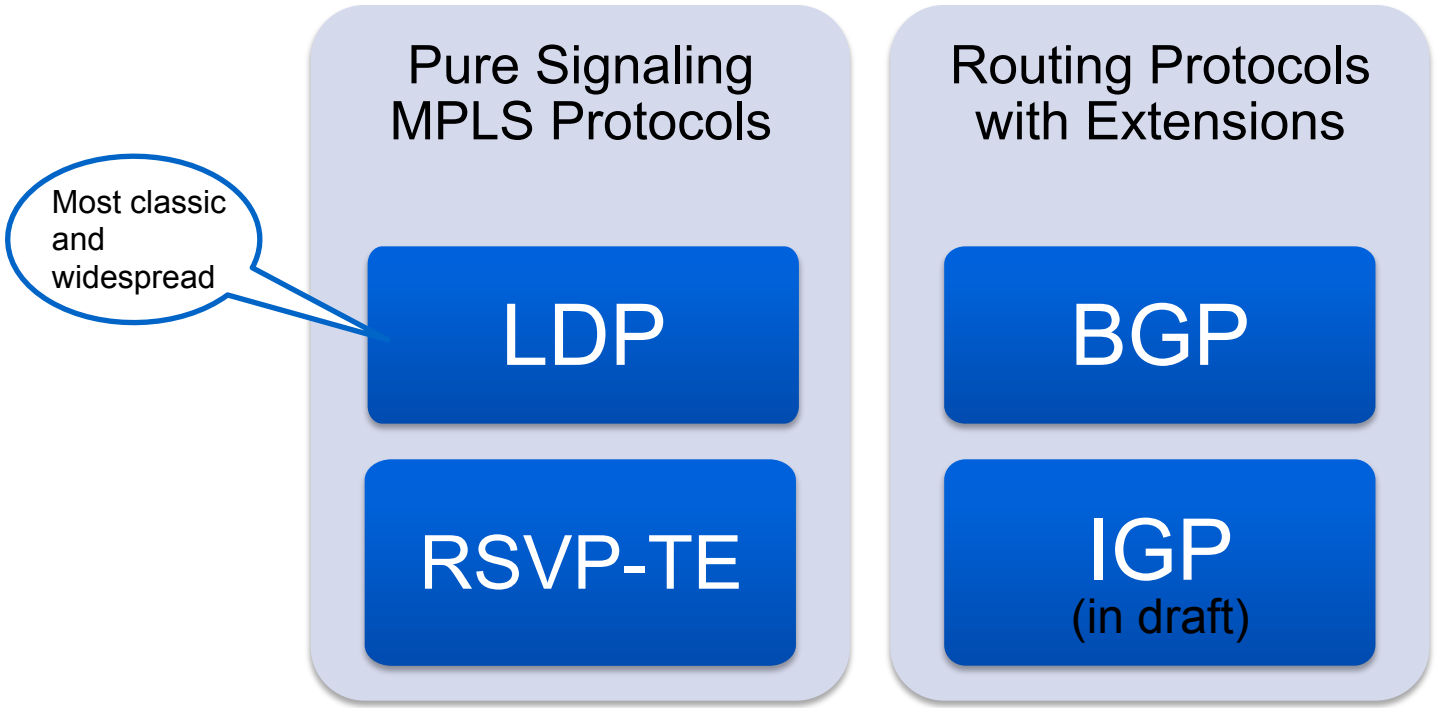
APNIC

Label Distribution Protocol

APNIC

MPLS Builders

Which protocols can set up Label Switched Path?



Advantages of LDP

- **Reliability**
  - LDP uses reliable TCP as the transport protocol for all but the discovery messages.
- **Auto provision**
  - Abilities to set up LSPs dynamically based on routing information
- **Plug-and-play**
  - Simple deployment and configuration
- **Support for a large number of LSPs**

LDP Identifier

- An LDP Identifier is a six octet quantity used to identify an LSR label space.

4 byte	2 byte
LSR ID	**Label Space ID**
10.10.1.1	0
20.20.20.2	6

Label Space ID = 0
Label space is per platform

Label Space ID ≠ 0
Label space is per interface

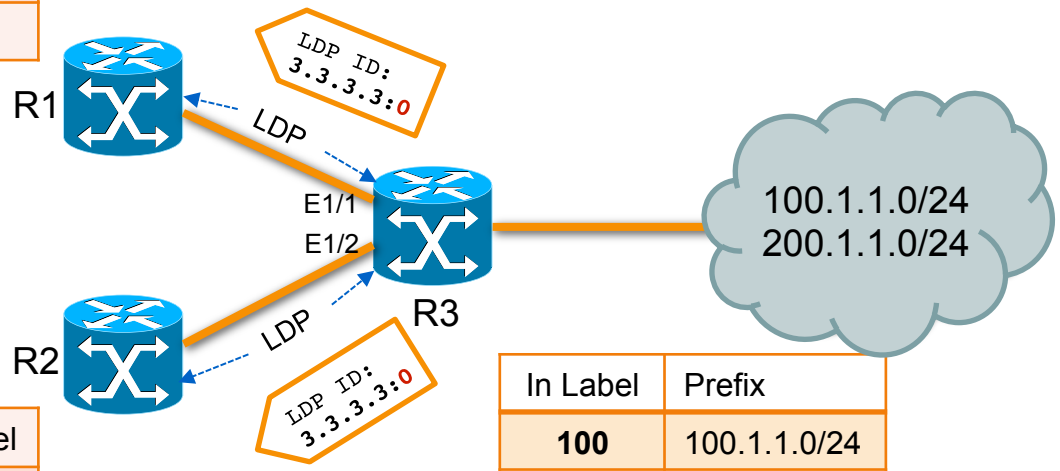
```
R2#show mpls ldp discovery
 Local LDP Identifier:
    2.2.2.2:0
    Discovery Sources:
    Interfaces:
          FastEthernet0/0 (ldp): xmit/recv
              LDP Id: 3.3.3.3:0
          Ethernet1/0 (ldp): xmit/recv
              LDP Id: 1.1.1.1:0
```

Cisco IOS

Label Space – Per Platform

- In per-platform label space, one single label is assigned to a destination network and announced to all neighbors. The label must be locally unique and valid on all incoming interfaces.

Prefix	Out Label
100.1.1.0/24	**100**
200.1.1.0/24	**200**



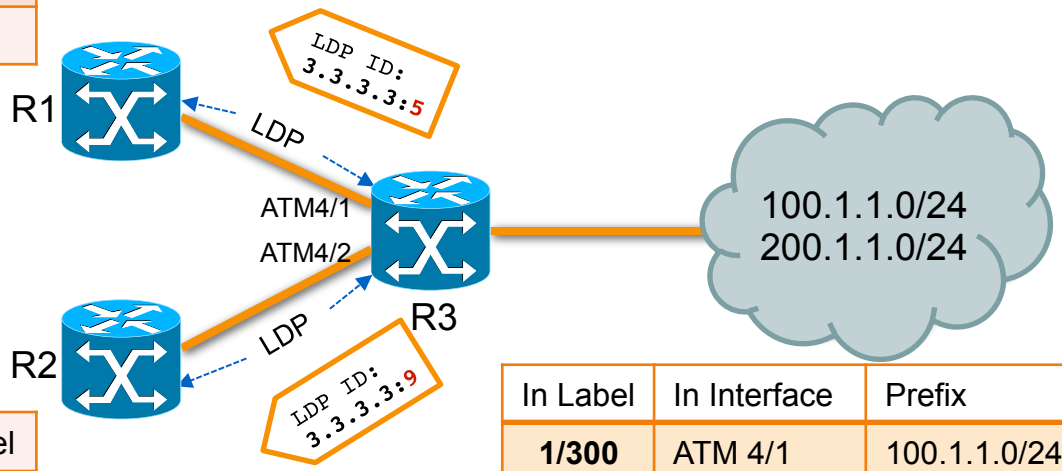
Prefix	Out Label
100.1.1.0/24	**100**
200.1.1.0/24	**200**

In Label	Prefix
100	100.1.1.0/24
200	100.1.1.0/24

Label Space – Per Interface

- In per-interface label space, local labels are assigned to IP destination prefixes on a per-interface basis. These labels must be unique on a per-interface basis.

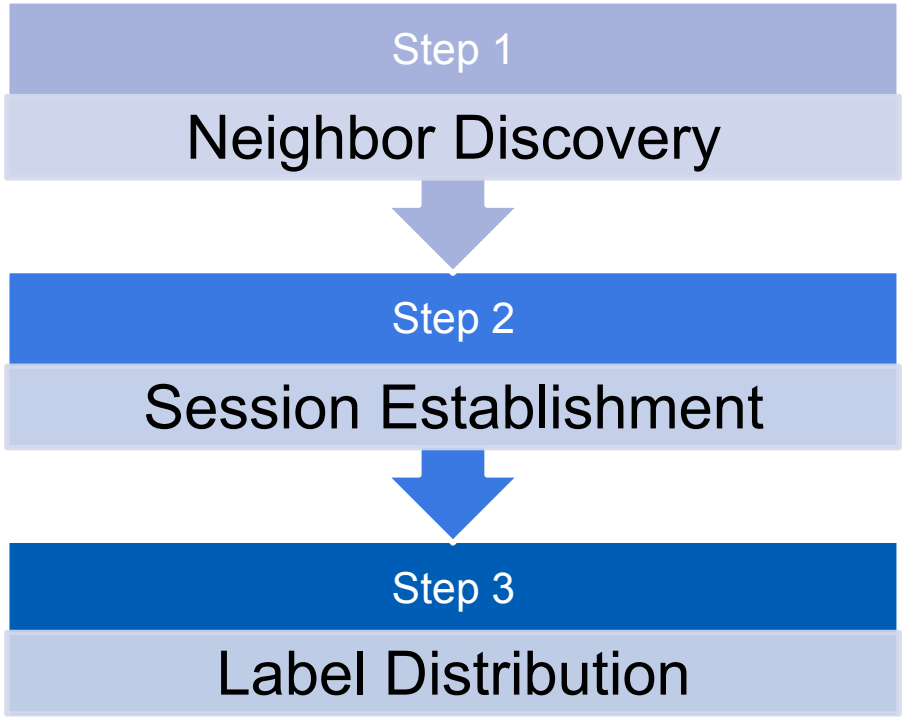
Prefix	Out Label
100.1.1.0/24	**1/300**
200.1.1.0/24	**1/200**



Prefix	Out Label
100.1.1.0/24	**1/400**
200.1.1.0/24	**1/500**

In Label	In Interface	Prefix
1/300	ATM 4/1	100.1.1.0/24
1/200	ATM 4/1	200.1.1.0/24
1/400	ATM 4/2	100.1.1.0/24
1/500	ATM 4/2	200.1.1.0/24

LDP Operations



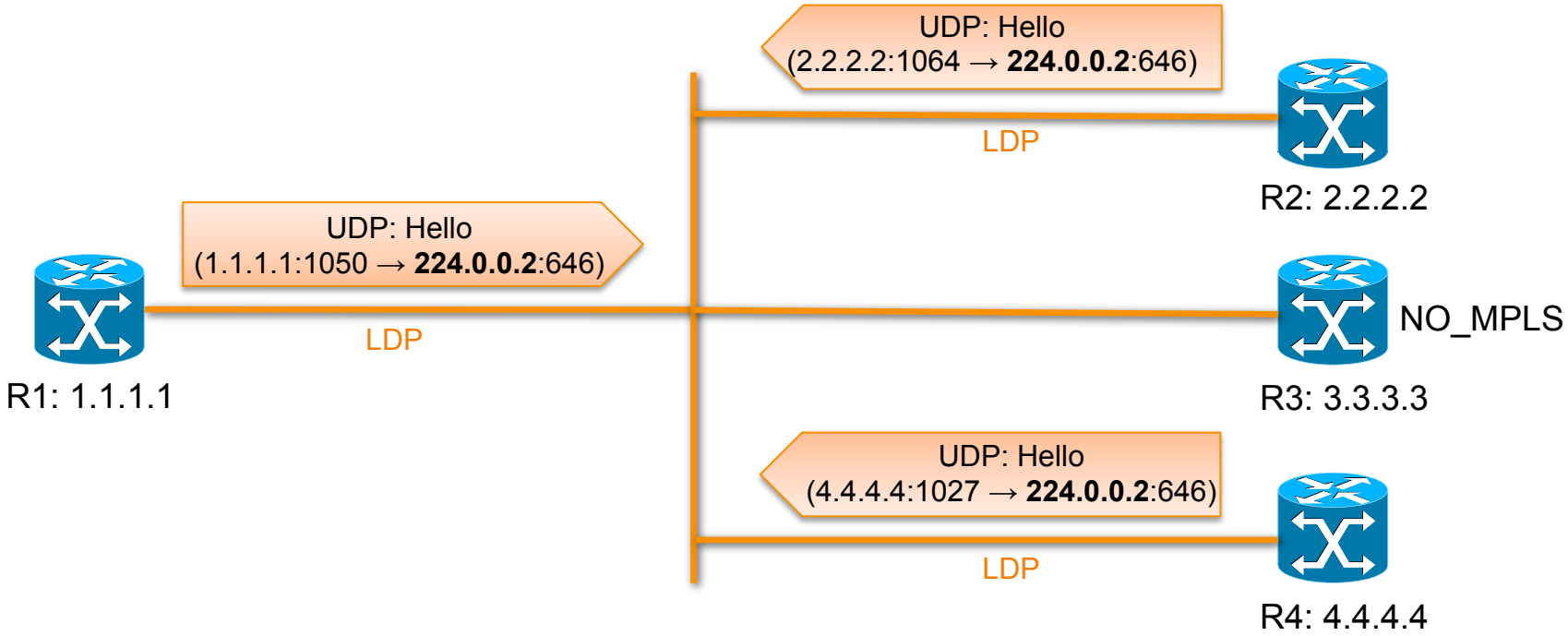
LDP Messages

Category	Function	Message Name
Discovery	Announce and maintain the presence of an LSR in a network	Hello
Session	Establish, maintain, and terminate sessions between LDP peers	Initialization
		Keepalive
Label Distribution	Create, change, and delete label mappings for FECs	Label Release
		Label Request
		Label Abort Request
		Label Mapping
		Label Withdrawal
Notification	Provide advisory information and to signal error information	Notification

(Not list all the messages)

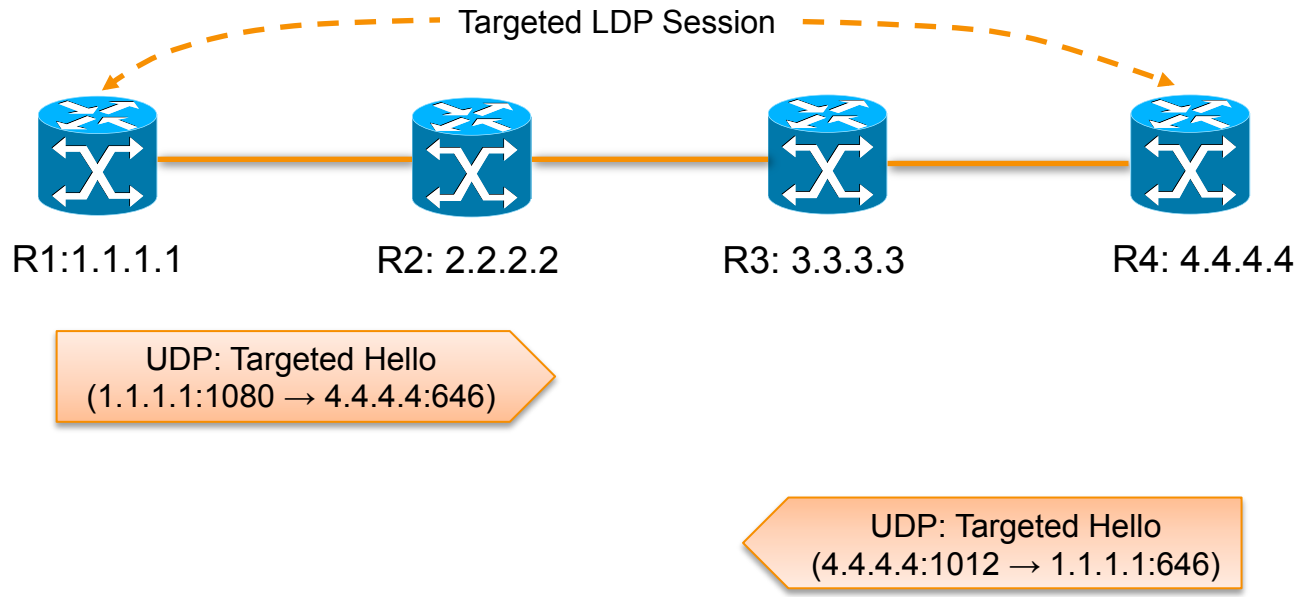
LDP Neighbor Discovery (1)

- Basic Discovery – Directly connected peer
  - LDP Hello messages are UDP messages that are sent on the links to the “all routers on this subnet” multicast IP address - 224.0.0.2. The UDP port used for LDP is 646.

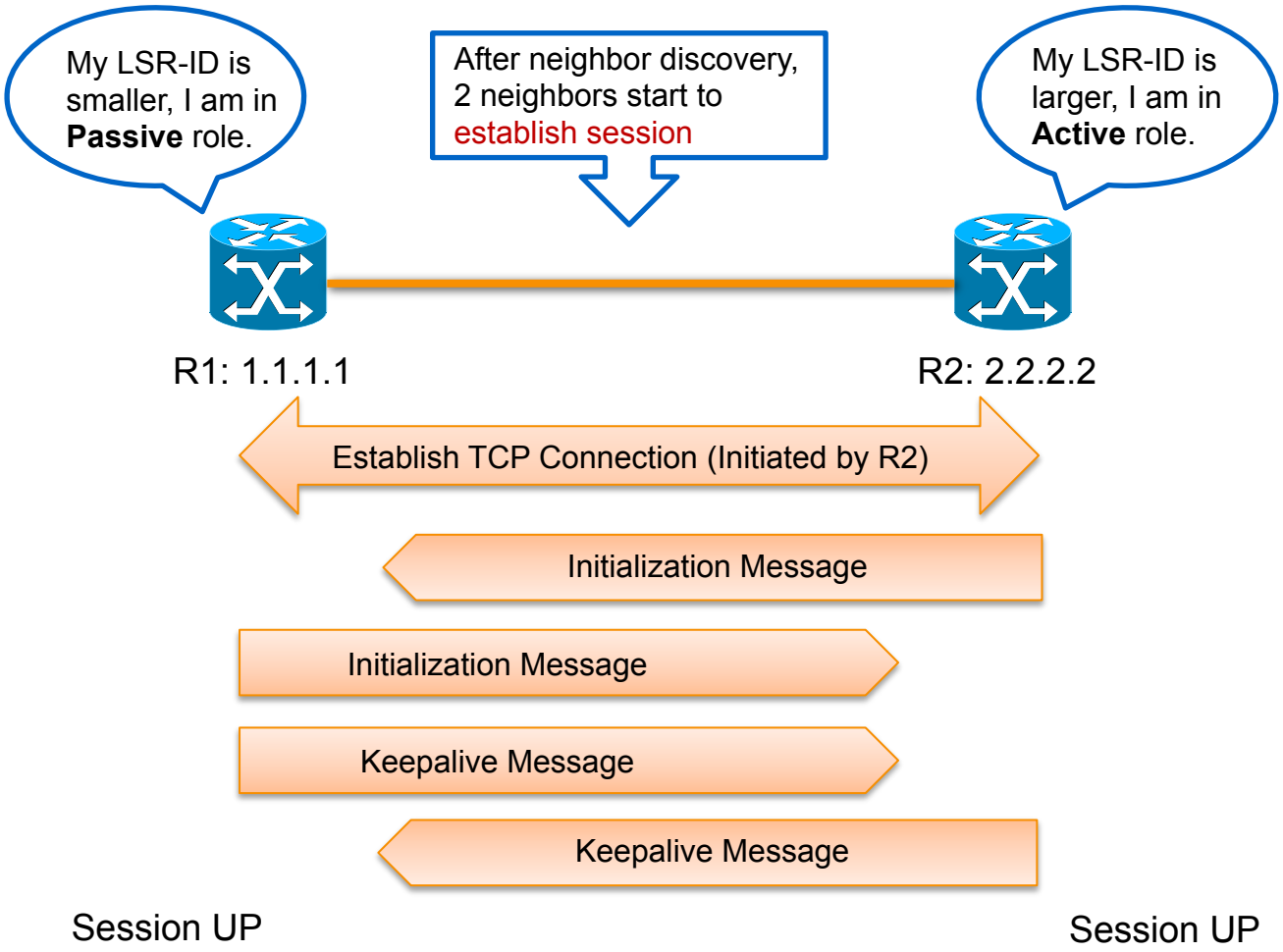


LDP Neighbor Discovery (2)

- Extended Discovery – Non-directly connected peer
  - LDP sessions between non-directly connected LSRs are supported by LDP Extended Discovery.



LDP Session Establishment and Maintenance



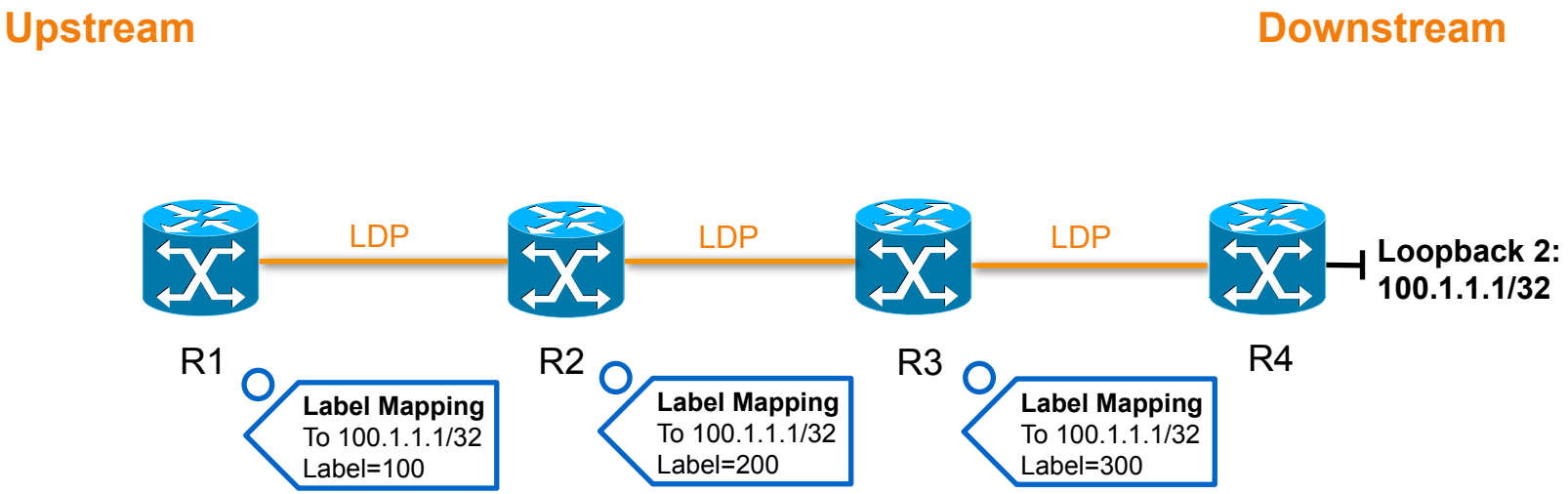
Label Distribution and Management

- After LDP sessions are established, labels will be distributed between LDP peers. The label distribution mode used depends on the interface and the implementation.

Label Distribution Control Mode	**Ordered**
	Independent
Label Advertisement Mode	**DoD (Downstream on Demand)**
	DU (Downstream Unsolicited)
Label Retention Mode	**Liberal**
	Conservative

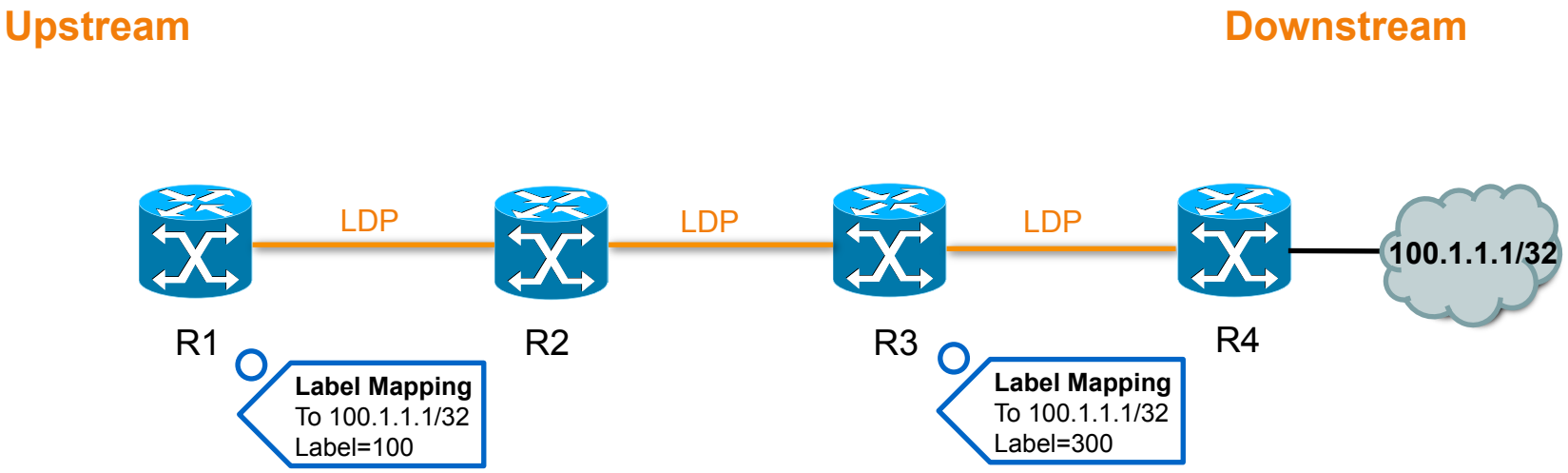
Label Distribution Control Mode - Ordered

- In Ordered control mode, an LSR would only assign a local label for the IGP prefixes that are marked as directly connected in its routing table or also for the IGP prefixes for which it has already received a label from the nexthop router.



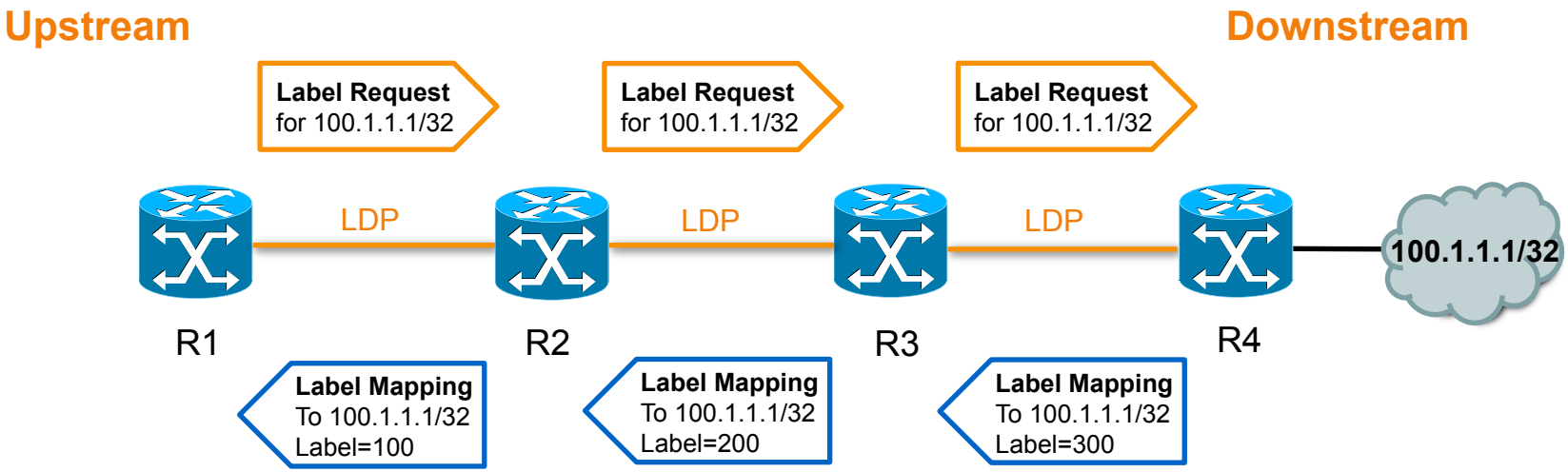
Label Distribution Control Mode - Independent

- In the independent mode, each LSR creates a local binding for a particular FEC as soon as it recognizes the FEC. Usually, this means that the prefix for the FEC is in its routing table.



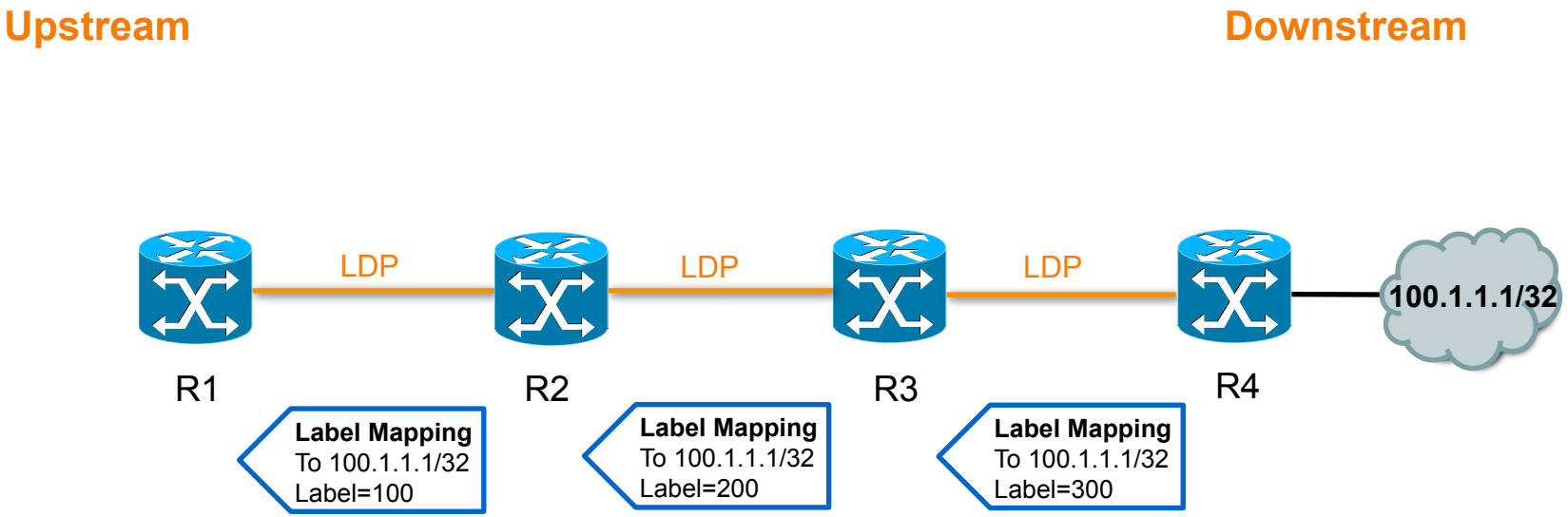
Label Advertisement Mode - Downstream on Demand

- In the DoD mode, an LSR distributes labels to a specified FEC only after receiving Label Request messages from its upstream LSR.



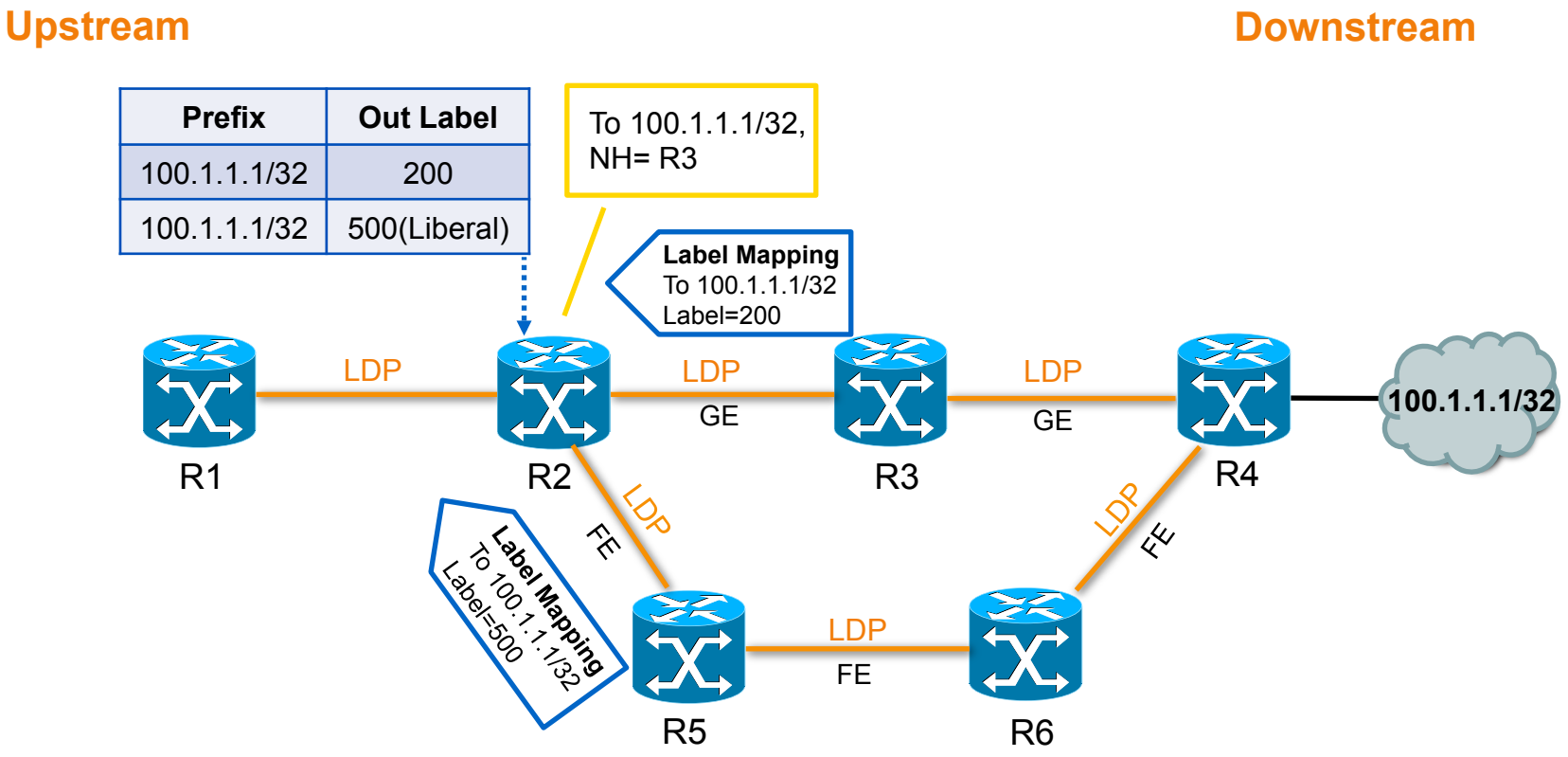
Label Advertisement Mode - Downstream Unsolicited

- In the DU mode, each LSR distributes a label to its upstream LSRs, without those LSRs requesting a label.



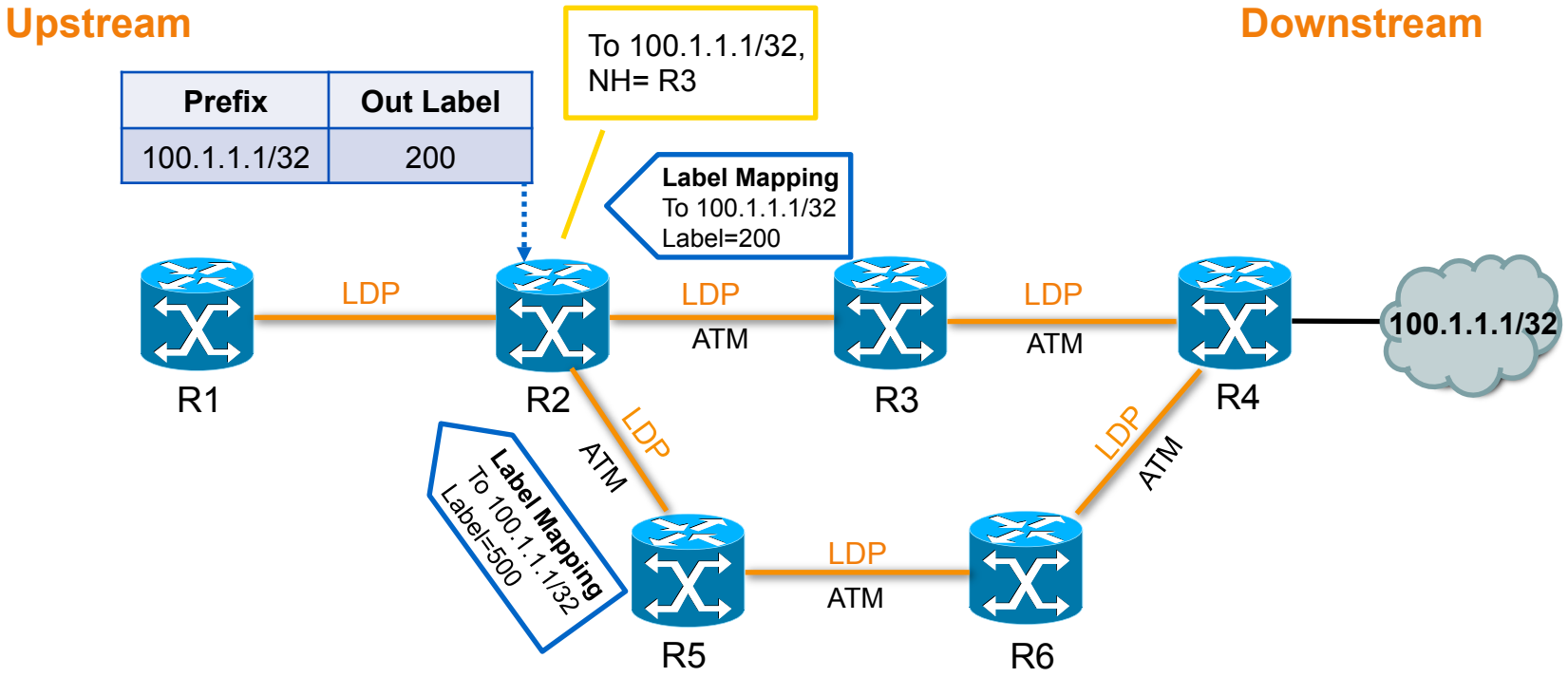
Label Retention Mode - Liberal

- In the liberal mode, an LSR keeps all received remote labels in the LIB, but not all are used to forward packets.



Label Retention Mode - Conservative

- An LSR that is running this mode does not store all remote labels in the LIB, but it stores only the remote label that is associated with the next-hop LSR for a particular FEC.



Label Distribution Scheme Summary

- Cisco IOS can support:

	Control	Distribution	Retention	Label Space
Frame Mode	Independent	DU	Liberal	Per Platform
Cell Mode (LC ATM)	Ordered	DoD	Conservation	Per Interface

- Junos can support:

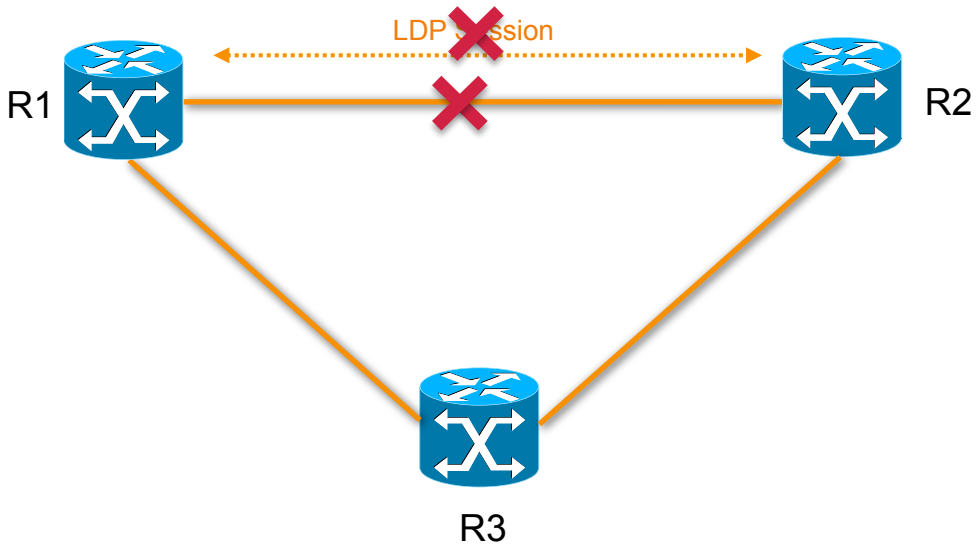
Control	Distribution	Retention
Ordered	DU	Liberal
	DoD	

- Huawei VRP can support:

Control	Distribution	Retention	
Ordered	DU	Liberal	By default
Ordered	DoD	Conservation	Also support

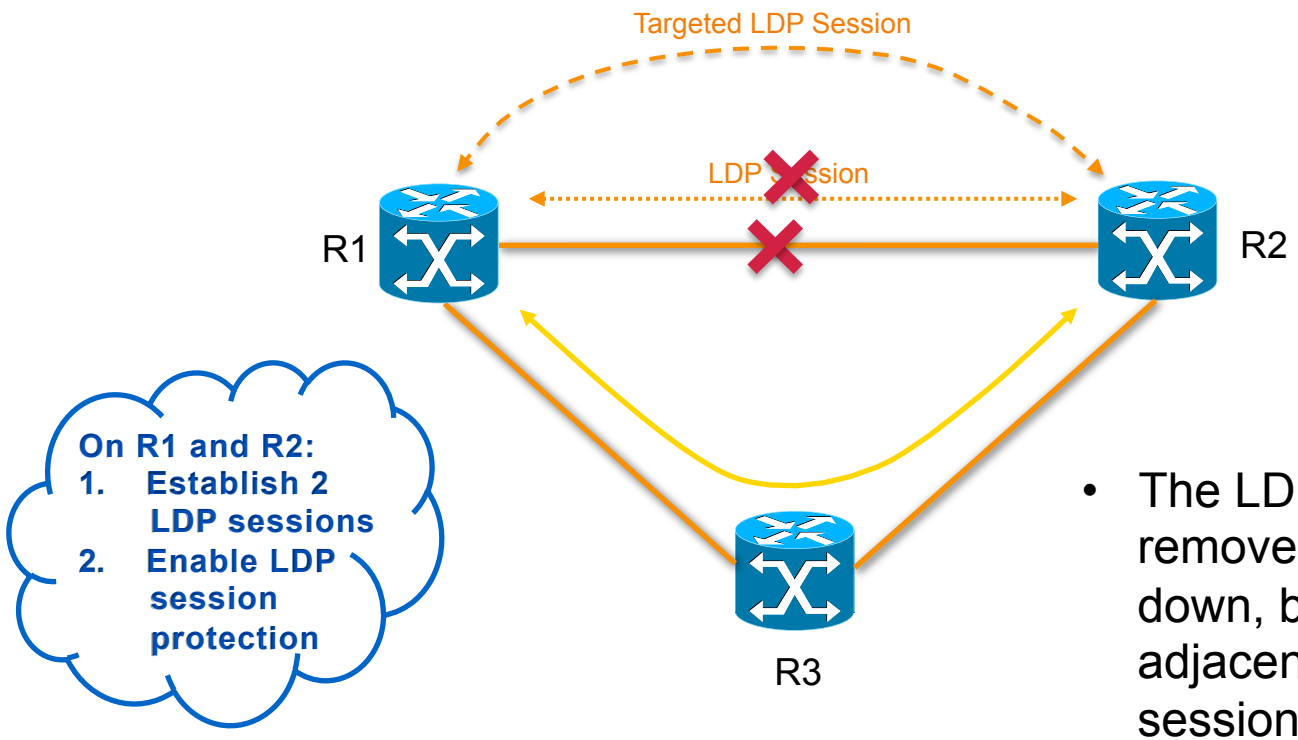
LDP Session Protection (1)

- Without LDP session protection, if the link between R1 and R2 fails, the LDP direct link adjacency fails.



LDP Session Protection (2)

- MPLS LDP Session Protection uses LDP Targeted Hellos to protect LDP sessions.



- The LDP link adjacency is removed when the link goes down, but the targeted adjacency keeps the LDP session up.

Questions?

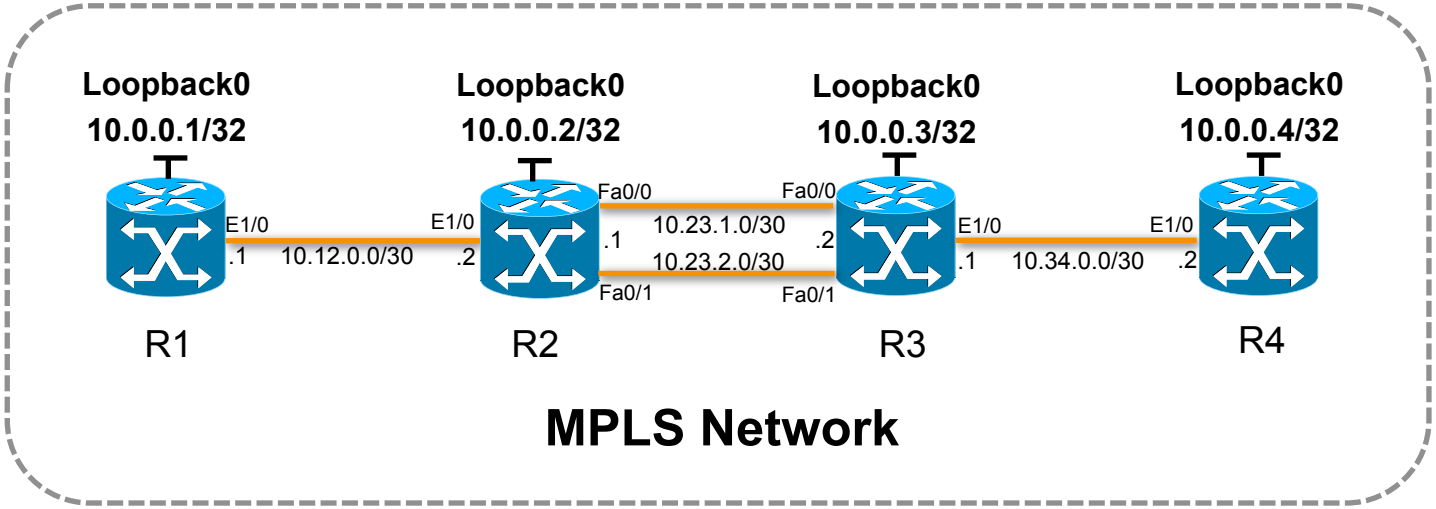
APNIC

Basic MPLS Configuration

APNIC

Configuration Example

- Task: Configure MPLS LDP on Cisco IOS (Version 15.2) to set up MPLS LSP only for loopback addresses.
- Prerequisite configuration:
  - 1. IP address configuration on all the routers
  - 2. IGP configuration on all the routers



Step 1: Enable MPLS & LDP

- Configuration steps:
  - 1. Configure basic MPLS and LDP on all the routers.

    R1 configuration:

```
R1(config)# ip cef
```
To make MPLS work, CEF switching is mandatory.
```
R1(config)# mpls label range 100 199
```
Specifying the label range for this router start from 100 to 199.
```
R1(config)# mpls ldp router-id loopback 0 force
```
Forcing LDP router ID to be loopback 0 address.
```
R1(config)# interface ethernet 1/0
R1(config-if)# mpls ip
```
IP MPLS is enabled on the interface.
```
R1(config-if)# mpls label protocol ldp
```
Label distribution protocol is LDP.

Check MPLS Interface

- Using **show mpls interfaces** command verifies that interfaces have been configured to use LDP:

```
R1#show mpls interfaces
Interface              IP             Tunnel   BGP Static Operational
Ethernet1/0            Yes (ldp)      No       No  No     Yes
```

```
R2#show mpls interfaces
Interface              IP             Tunnel   BGP Static Operational
FastEthernet0/0        Yes (ldp)      No       No  No     Yes
FastEthernet0/1        Yes (ldp)      No       No  No     Yes
Ethernet1/0            Yes (ldp)      No       No  No     Yes
```

Check LDP Discovery

- Check LDP discovery

```
R2#show mpls ldp discovery
 Local LDP Identifier:
    10.0.0.2:0
    Discovery Sources:
    Interfaces:
        FastEthernet0/0 (ldp): xmit/recv
            LDP Id: 10.0.0.3:0
        FastEthernet0/1 (ldp): xmit/recv
            LDP Id: 10.0.0.3:0
        Ethernet1/0 (ldp): xmit/recv
            LDP Id: 10.0.0.1:0
```

Local LDP ID

Neighbor's LDP ID

R2 has received Hello messages from routers whose ID are 10.0.0.3 and 10.0.0.1

Check LDP Neighbors

- Check LDP neighbors on R1

```
R1#show mpls ldp neighbor
    Peer LDP Ident: 10.0.0.2:0; Local LDP Ident 10.0.0.1:0
         TCP connection: 10.0.0.2.48548 - 10.0.0.1.646
         State: Oper; Msgs sent/rcvd: 34/34; Downstream
         Up time: 00:09:57
         LDP discovery sources:
           Ethernet1/0, Src IP addr: 10.12.0.2
        Addresses bound to peer LDP Ident:
          10.23.1.1       10.23.2.1       10.12.0.2       10.0.0.2
```

LDP session is a TCP session (port = 646)

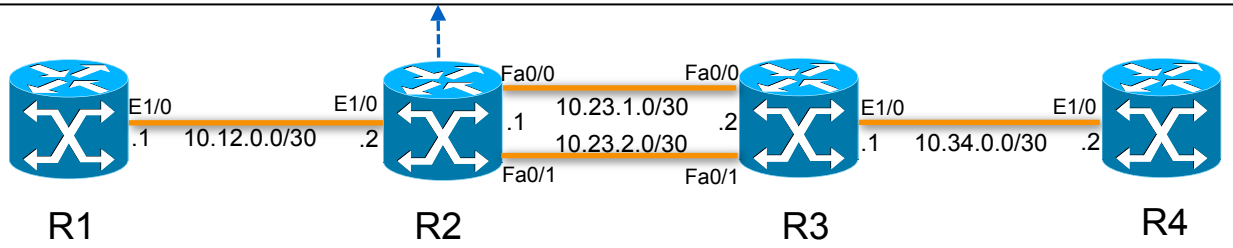
Operational is the stable state of LDP session.

Check LDP Neighbors

- Check LDP neighbors on R2

```
R2#show mpls ldp neighbor
    Peer LDP Ident: 10.0.0.3:0; Local LDP Ident 10.0.0.2:0
           TCP connection: 10.0.0.3.28664 - 10.0.0.2.646
           State: Oper; Msgs sent/rcvd: 36/36; Downstream
           Up time: 00:12:12
           LDP discovery sources:
             FastEthernet0/0, Src IP addr: 10.23.1.2
             FastEthernet0/1, Src IP addr: 10.23.2.2
         Addresses bound to peer LDP Ident:
           10.23.1.2       10.23.2.2       10.34.0.1       10.0.0.3
    Peer LDP Ident: 10.0.0.1:0; Local LDP Ident 10.0.0.2:0
           TCP connection: 10.0.0.1.646 - 10.0.0.2.48548
           State: Oper; Msgs sent/rcvd: 36/36; Downstream
           Up time: 00:11:40
           LDP discovery sources:
             Ethernet1/0, Src IP addr: 10.12.0.1
         Addresses bound to peer LDP Ident:
           10.12.0.1       10.0.0.1
```

Multiple links between two routers still mean single LDP session.



Check LDP Label Information Base

- Check LDP LIB

```
R2#show mpls ldp bindings
  lib entry: 10.0.0.1/32, rev 24
          local binding:  label: 206
          remote binding: lsr: 10.0.0.3:0, label: 302
          remote binding: lsr: 10.0.0.1:0, label: imp-null
  lib entry: 10.0.0.2/32, rev 18
          local binding:  label: imp-null
          remote binding: lsr: 10.0.0.3:0, label: 301
          remote binding: lsr: 10.0.0.1:0, label: 101
  lib entry: 10.0.0.3/32, rev 22
          local binding:  label: 205
          remote binding: lsr: 10.0.0.3:0, label: imp-null
          remote binding: lsr: 10.0.0.1:0, label: 100
          ..........(omitted)
```

```
R2#show mpls ldp bindings 10.0.0.4 32
 lib entry: 10.0.0.4/32, rev 20
          local binding:  label: 204
          remote binding: lsr: 10.0.0.3:0, label: 300
          remote binding: lsr: 10.0.0.1:0, label: 107
```

Check Label Forwarding Table

- Check label forwarding table

```
R2#show mpls forwarding-table
Local      Outgoing   Prefix           Bytes Label   Outgoing    Next Hop
Label      Label      or Tunnel Id     Switched      interface
203        Pop Label  10.34.0.0/30     0             Fa0/0       10.23.1.2
           Pop Label  10.34.0.0/30     0             Fa0/1       10.23.2.2
204        300        10.0.0.4/32      0             Fa0/0       10.23.1.2
           300        10.0.0.4/32      0             Fa0/1       10.23.2.2
205        Pop Label  10.0.0.3/32      0             Fa0/0       10.23.1.2
           Pop Label  10.0.0.3/32      0             Fa0/1       10.23.2.2
206        Pop Label  10.0.0.1/32      0             Et1/0       10.12.0.1
```

Step 2: Configure Conditional Label Distribution

- Only set up LSP for loopback addresses.
  - 2.1 Create prefix-list on each router, all the loopback addresses are in 10.0.0.0/24 block.

```
R1(config)#ip prefix-list ALL-LOOPBACK seq 5 permit
10.0.0.0/24 le 32
```

  - 2.2 Apply the prefix-list

```
R1(config)#mpls ldp label
R1(config-ldp-lbl)#allocate global prefix-list ALL-LOOPBACK
```

Allocate labels for the routes matching ALL-LOOPBACK prefix-list.

Verify the Results of Conditional Label Distribution

- **Before** the configuration.

```
R1#show mpls ldp bindings
  lib entry: 10.0.0.1/32, rev 51
          local binding:  label: imp-null
          remote binding: lsr: 10.0.0.2:0, label: 206
  lib entry: 10.0.0.2/32, rev 52
          local binding:  label: 101
          remote binding: lsr: 10.0.0.2:0, label: imp-null
  lib entry: 10.0.0.3/32, rev 53
          local binding:  label: 100
          remote binding: lsr: 10.0.0.2:0, label: 205
  lib entry: 10.0.0.4/32, rev 54
          local binding:  label: 107
          remote binding: lsr: 10.0.0.2:0, label: 204
  lib entry: 10.12.0.0/30, rev 71
          local binding:  label: imp-null
          remote binding: lsr: 10.0.0.2:0, label: imp-null
  lib entry: 10.23.1.0/30, rev 72
          local binding:  label: 102
          remote binding: lsr: 10.0.0.2:0, label: imp-null
  lib entry: 10.23.2.0/30, rev 73
          local binding:  label: 103
          remote binding: lsr: 10.0.0.2:0, label: imp-null
  lib entry: 10.34.0.0/30, rev 75
          local binding:  label: 104
        remote binding: lsr: 10.0.0.2:0, label: 200
```

Entries for all the prefixes in IP routing table.

Verify the Results of Conditional Label Distribution

- **After** configure on all the routers.

```
R1#show mpls ldp bindings
  lib entry: 10.0.0.1/32, rev 51
          local binding:  label: imp-null
          remote binding: lsr: 10.0.0.2:0, label: 206
  lib entry: 10.0.0.2/32, rev 52
          local binding:  label: 101
          remote binding: lsr: 10.0.0.2:0, label: imp-null
  lib entry: 10.0.0.3/32, rev 53
          local binding:  label: 100
          remote binding: lsr: 10.0.0.2:0, label: 205
  lib entry: 10.0.0.4/32, rev 54
          local binding:  label: 107
          remote binding: lsr: 10.0.0.2:0, label: 204
```

Only the entries for loopback addresses.

Check the LSP

- Check the LSP for 10.0.0.4 from R1 to R3

```
R1#show mpls forwarding-table 10.0.0.4 32
Local      Outgoing   Prefix           Bytes Label   Outgoing    Next Hop
Label      Label      or Tunnel Id     Switched      interface
107        204        10.0.0.4/32      0             Et1/0       10.12.0.2
```

```
R2#show mpls forwarding-table 10.0.0.4 32
Local      Outgoing   Prefix           Bytes Label   Outgoing    Next Hop
Label      Label      or Tunnel Id     Switched      interface
204        300        10.0.0.4/32      0             Fa0/0       10.23.1.2
           300        10.0.0.4/32      0             Fa0/1       10.23.2.2
```

```
R3# show mpls forwarding-table 10.0.0.4 32
Local      Outgoing   Prefix
Label      Label      or Tunnel
300        Pop Label  10.0.0.4/32                                .34.0.2
```

Implicit Null

If I want to use an outgoing label at the penultimate hop for keeping QoS info. What can I do?

Explicit-null Label

- Explicit Null label can be used to keep the QoS information.

Explicit Null (IPv4) = Label 0

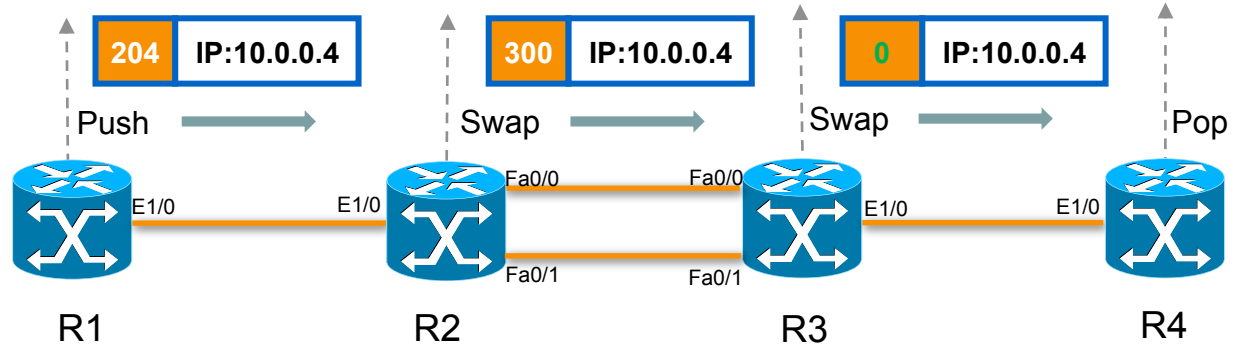
Explicit Null (IPv6) = Label 2

Prefix: 10.0.0.4/32	
Local Label	Null
Out Interface	E1/0
Out Label	204
Operation	Push

Prefix: 10.0.0.4/32	
Local Label	204
Out Interface	Fa0/0 Fa0/1
Out Label	300
Operation	Swap

Prefix: 10.0.0.4/32	
Local Label	300
Out Interface	E1/0
Out Label	Explicit-n
Operation	Swap

Prefix: 10.0.0.4/32	
Local Label	Explicit-n
Out Interface	--
Out Label	--
Operation	POP



Additional Task: Using Explicit-null Label

- Explicit-null configuration on R4:

```
R4(config)# mpls ldp explicit-null
```

- After configuring this command, check the label forwarding table on R3.

```
R3#show mpls forwarding-table 10.0.0.4 32
Local      Outgoing   Prefix           Bytes Label   Outgoing    Next Hop
Label      Label      or Tunnel Id     Switched      interface
300        explicit-n 10.0.0.4/32      0             Et1/0
10.34.0.2
```

Out label is explicit-null

Questions?

APNIC

Deploy MPLS L3 VPN

Issue Date: [201609]
Revision: [01]

APNIC

Acknowledgement

- Cisco Systems

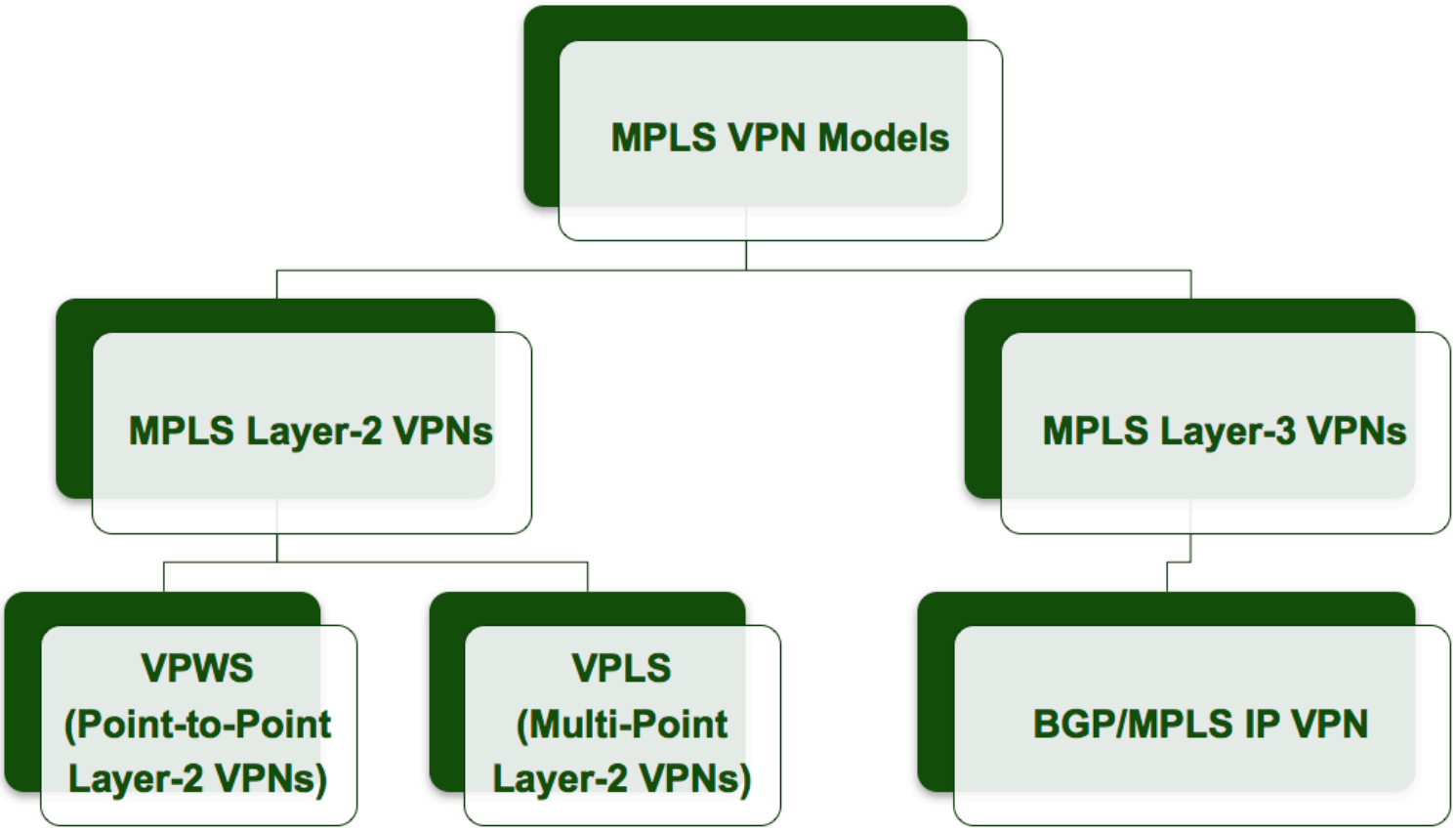
Course Outline

- MPLS L3 VPN Models
- L3 VPN Terminologies
- MPLS VPN Operation
  - Control Panel
  - Data Plane
  - Forwarding function
- Function of RD and RT
- Configuration Examples
- MPLS L3 VPN Service Deployment
  - Multi-homed VPN Sites
  - Hub and Spoke
  - Extranet VPN
  - Internet Access Services

MPLS L3VPN Principle

APNIC

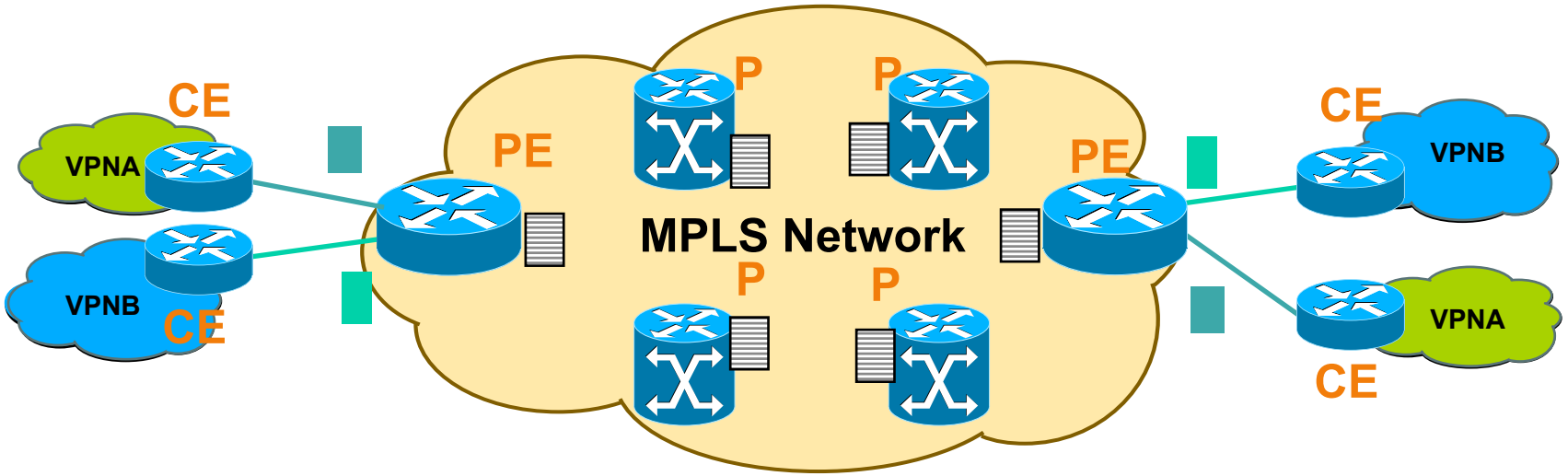
MPLS VPN Models



Advantages of MPLS Layer-3 VPN

- Scalability
- Security
- Easy to Create
- Flexible Addressing
- Integrated Quality of Service (QoS) Support
- Straightforward Migration

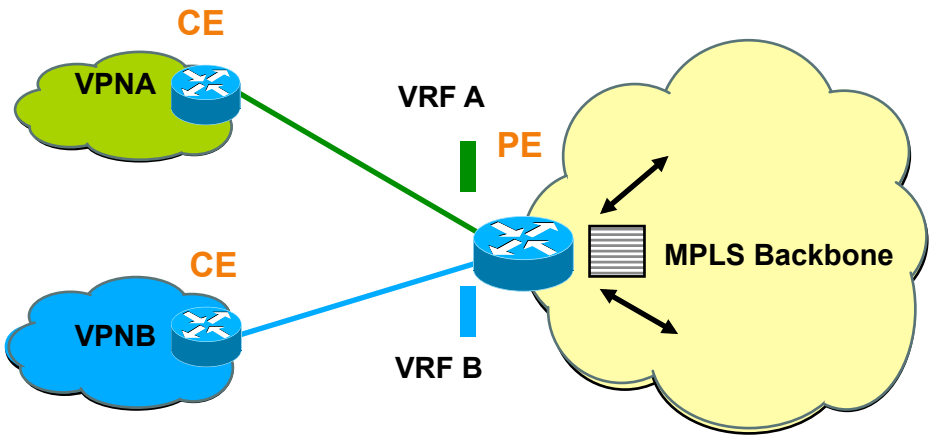
MPLS L3VPN Topology



- PE: Provider Edge Router
- P : Provider Router
- CE: Customer Edge Router

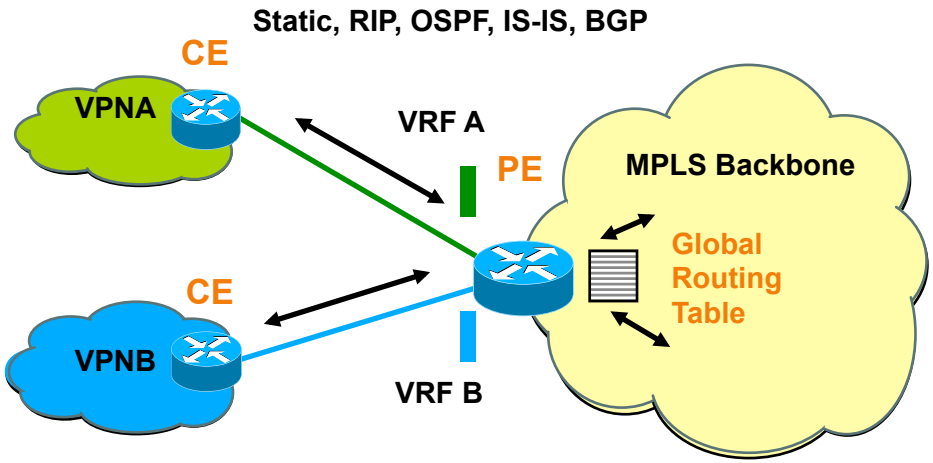
Virtual Routing and Forwarding Instance

- Virtual routing and forwarding table
  - On PE router
  - Separate instance of routing (RIB) and forwarding table
- A VRF defines the VPN membership of a customer site attached to a PE device.
- VRF associated with one or more customer interfaces



Routes Transfer between CE and PE

- PE installs the internal routes (IGP) in global routing table
- PE installs the VPN customer routes in VRF routing tables
  - VPN routes are learned from CE routers or remote PE routers
  - VRF-aware routing protocol (static, RIP, BGP, OSPF, IS-IS) on each PE



Control Plane: Multi-Protocol BGP

- PE routers distribute VPN routes to each other via MP-BGP.

- MP-BGP customizes the VPN Customer Routing Information as per the Locally Configured VRF Information at the PE using:
  - Route Distinguisher (RD)
  - Route Target (RT)
  - VPN Label

What is RD

- Route distinguisher is an 8-octet field prefixed to the customer's IPv4 address. RD makes the customer's IPv4 address unique inside the SP MPLS network.
- RD is configured in the VRF at PE

VPNv4 Address:

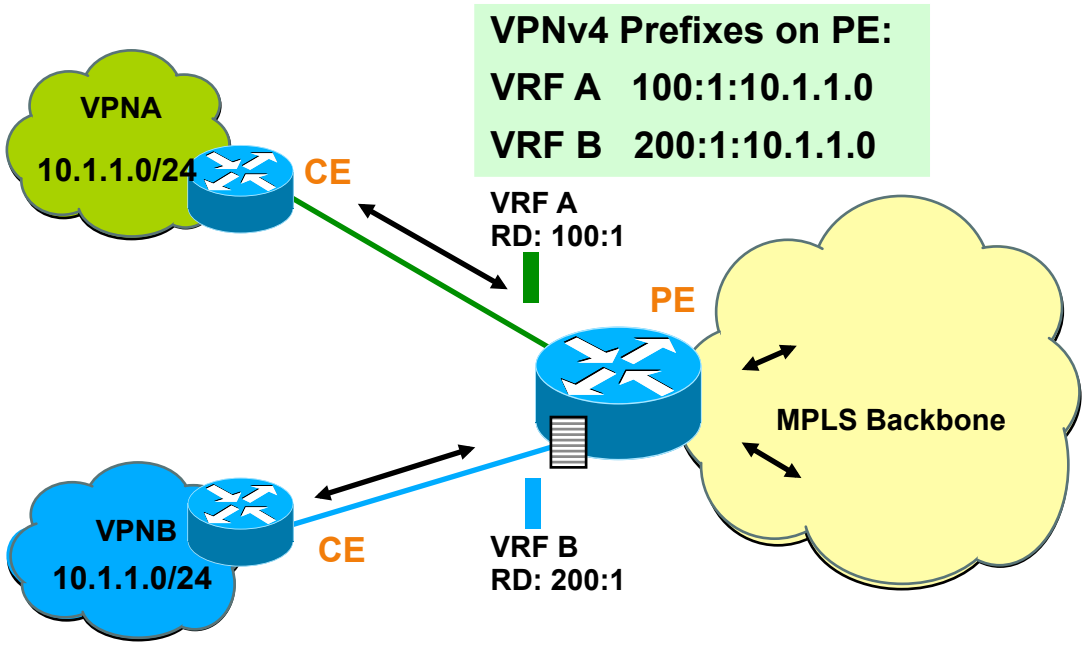
Route Distinguisher (8 bytes)		IPv4 Address (4 bytes)

Example:

Type 0	100:1	10.1.1.1
Type 1	192.168.19.1:1	10.1.1.1
Type 2	65538:10	10.1.1.1

Route Advertisement: RD

- VPN customer IPv4 prefix is converted into a VPNv4 prefix by appending the RD to the IPv4 address
- PE devices use MP-BGP to advertise the VPNv4 address



What is RT

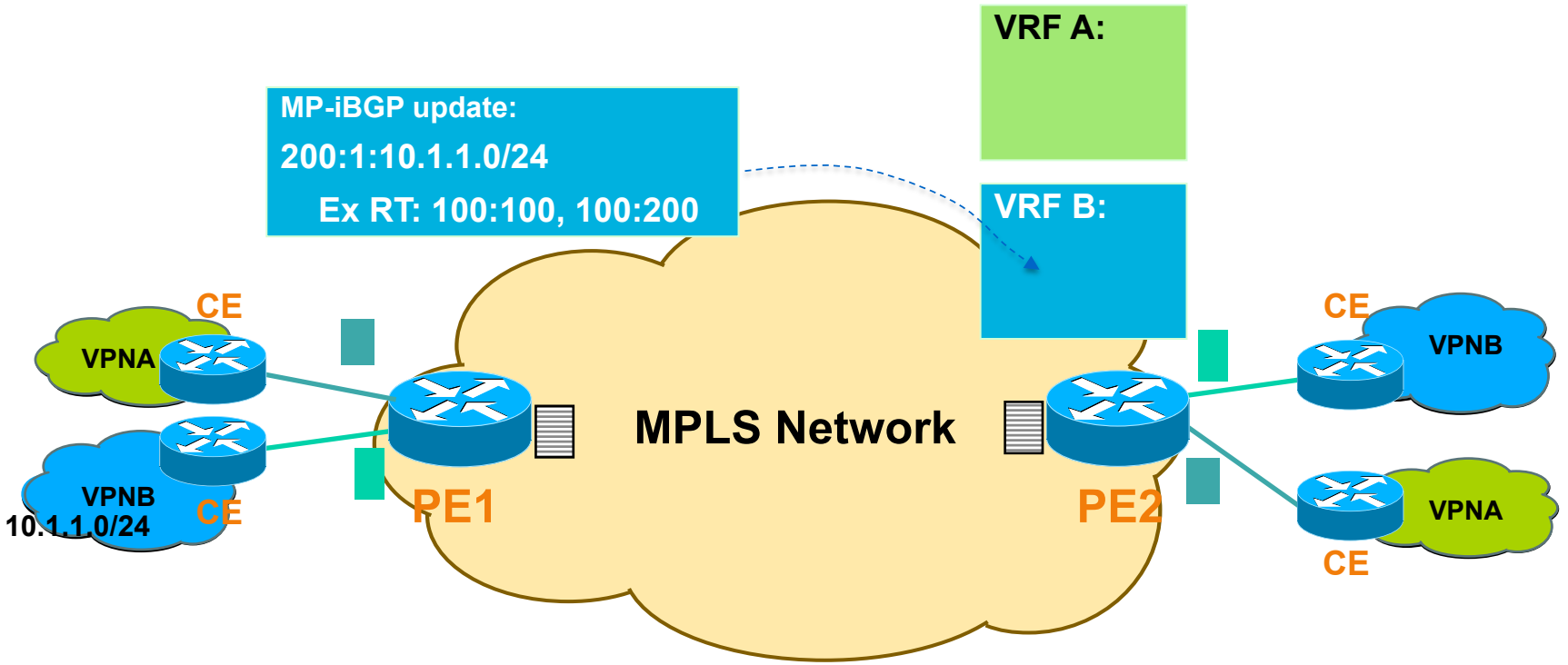
- Route Target is a BGP extended community attribute, is used to control VPN routes advertisement.

Example:

Route Target (8 bytes)	
Type 0	100:1
Type 1	192.168.1.1:1
Type 2	65538:10

- Two types of RT:
  - Export RT
  - Import RT

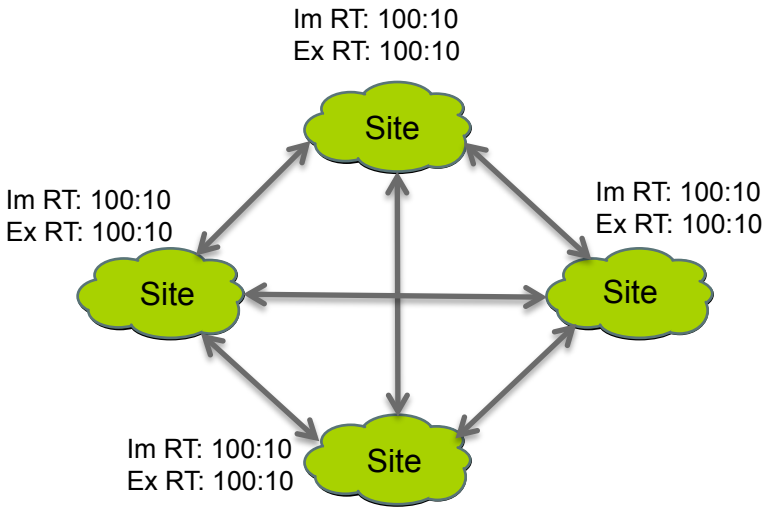
Route Advertisement: RT



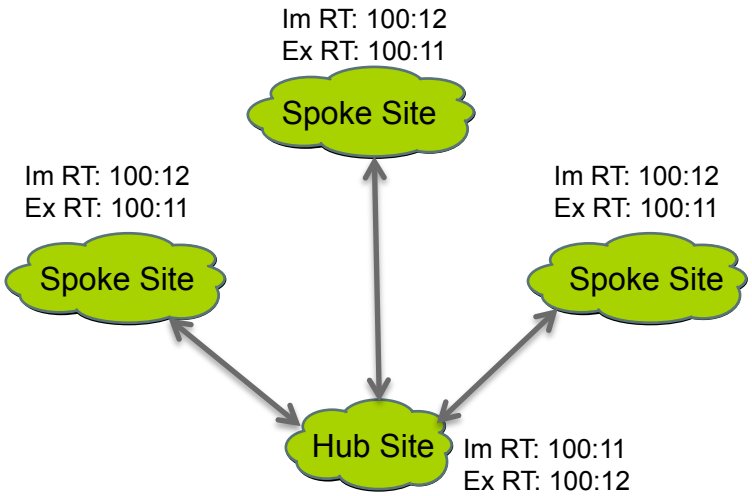
	Import RT	Export RT
VRF A	100:1	100:1
VRF B	100:100 100:200	**100:100** 100:200

	Import RT	Export RT
VRF A	100:1 100:2 100:3	100:1 100:2
VRF B	**100:100**	100:100

Using RT to Build VPN Topologies



Full Mesh

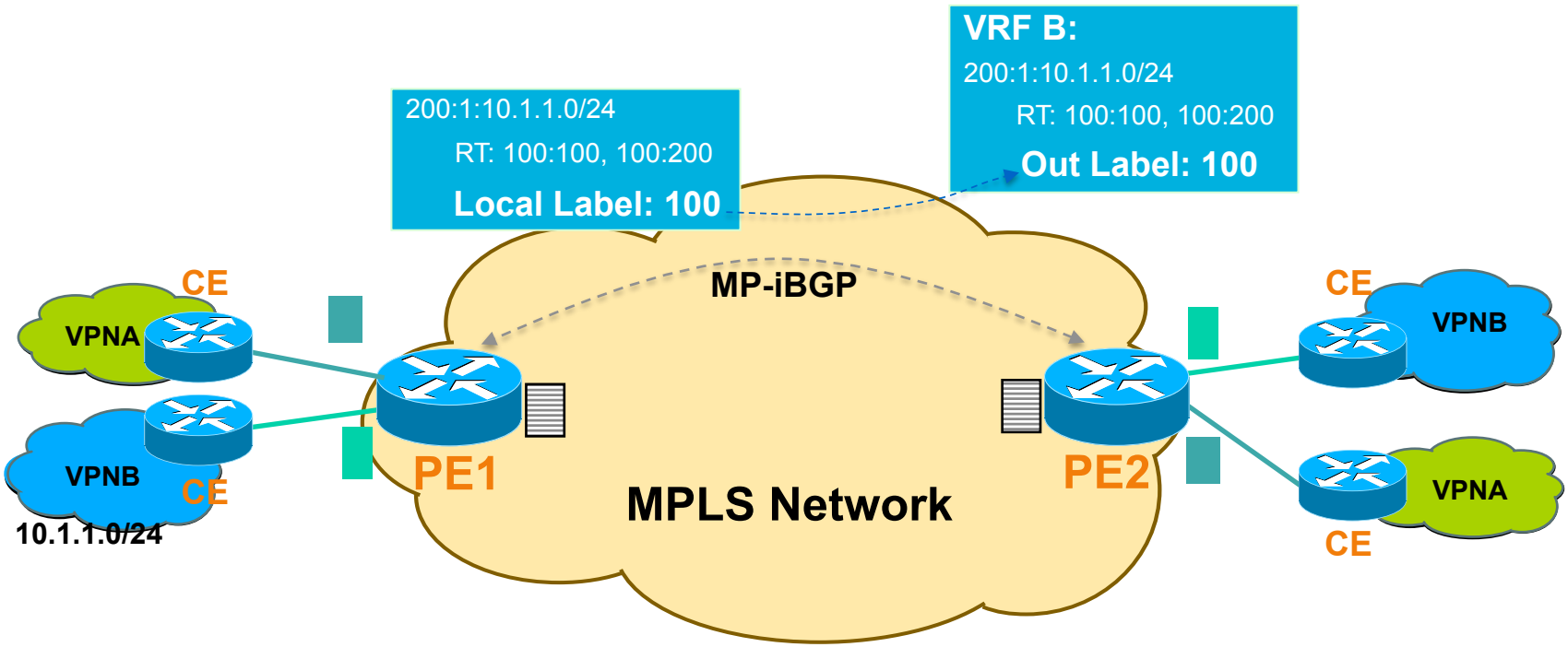


Hub Spoke

In a full-mesh VPN, each site in the VPN can communicate with every other site in that same VPN.

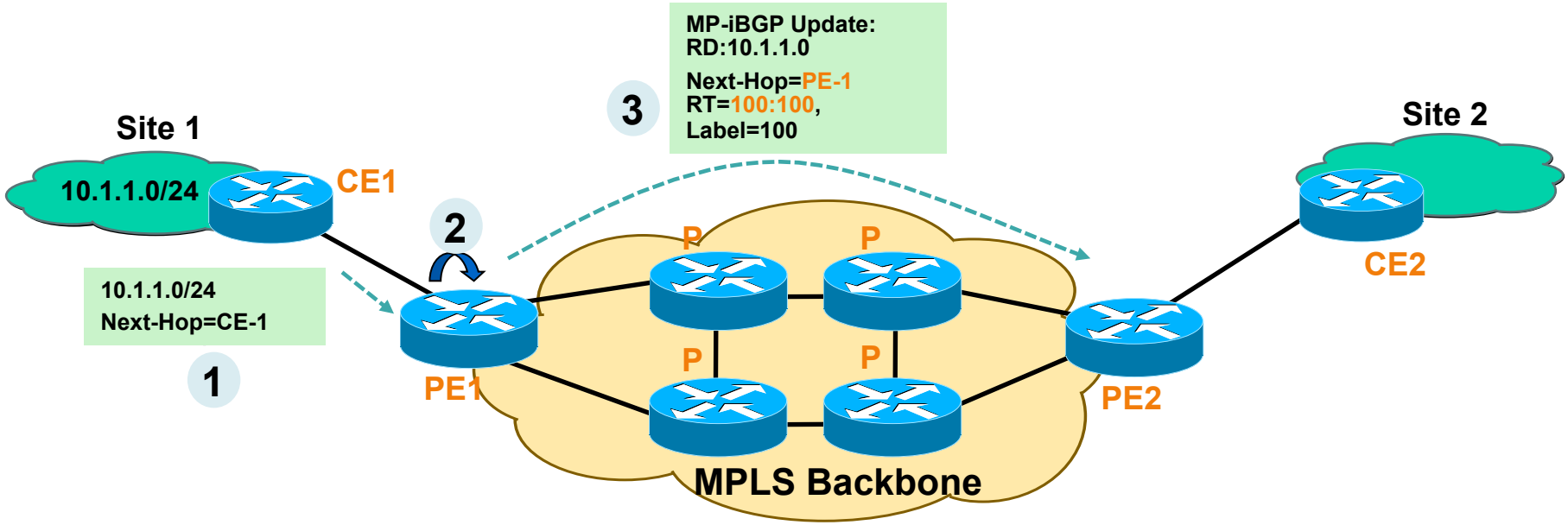
In a hub-and-spoke VPN, the spoke sites in the VPN can communicate only with the hub sites; they cannot communicate with other spoke sites.

VPN Label



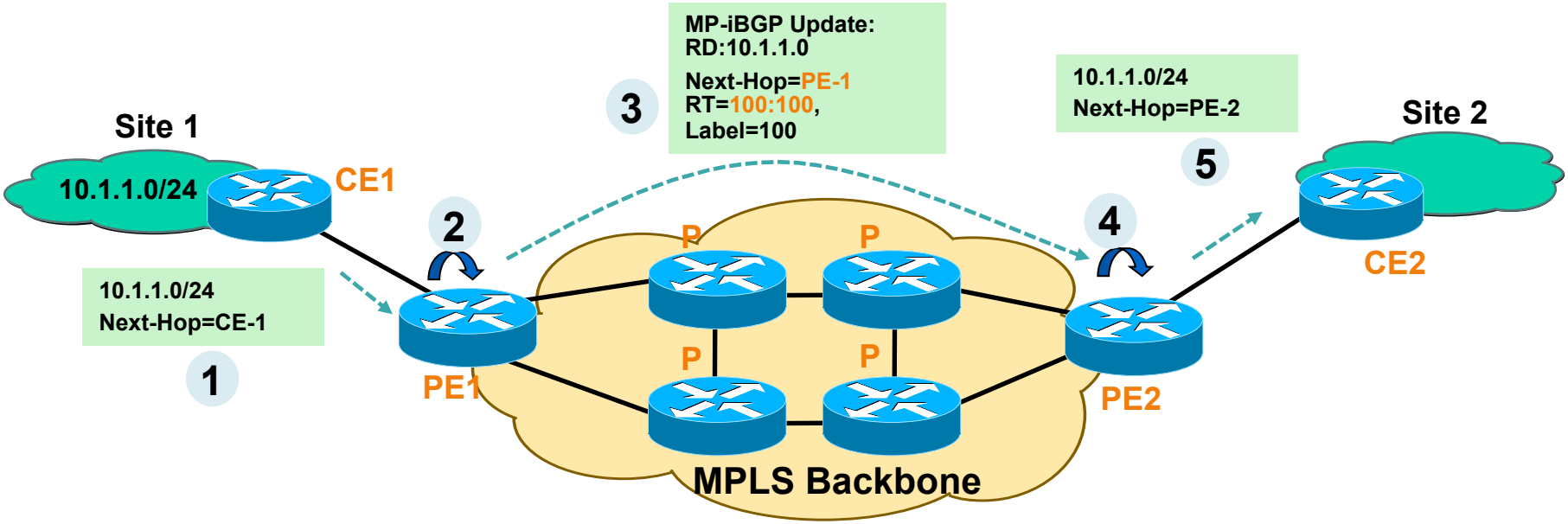
- PE adds the label to the NLRI field.

Control Plane Walkthrough(1/2)



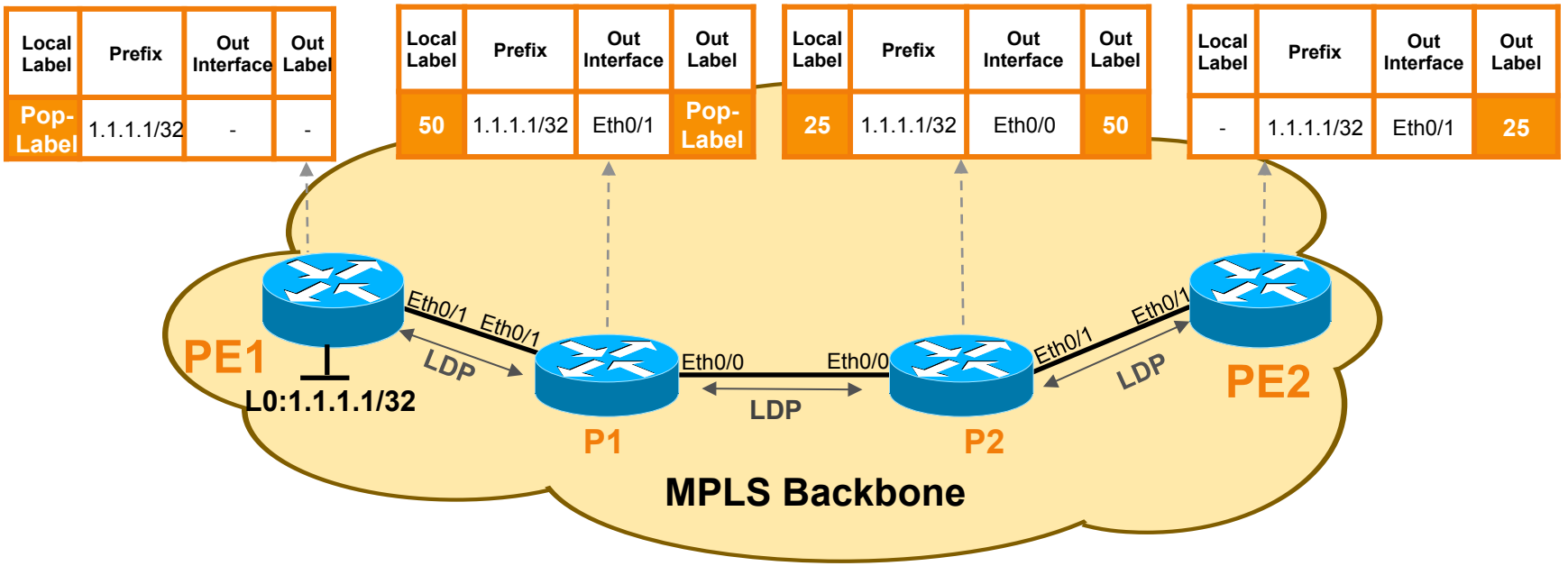
1. PE1 receives an IPv4 update (eBGP/OSPF/ISIS/RIP)
2. PE1 converts it into VPNv4 address and constructs the MP-iBGP UPDATE message
   - Associates the RT values (export RT =100:100) per VRF configuration
   - Rewrites next-hop attribute to itself
   - Assigns a label (100); Installs it in the MPLS forwarding table.
3. PE1 sends MP-iBGP update to other PE routers

Control Plane Walkthrough(2/2)



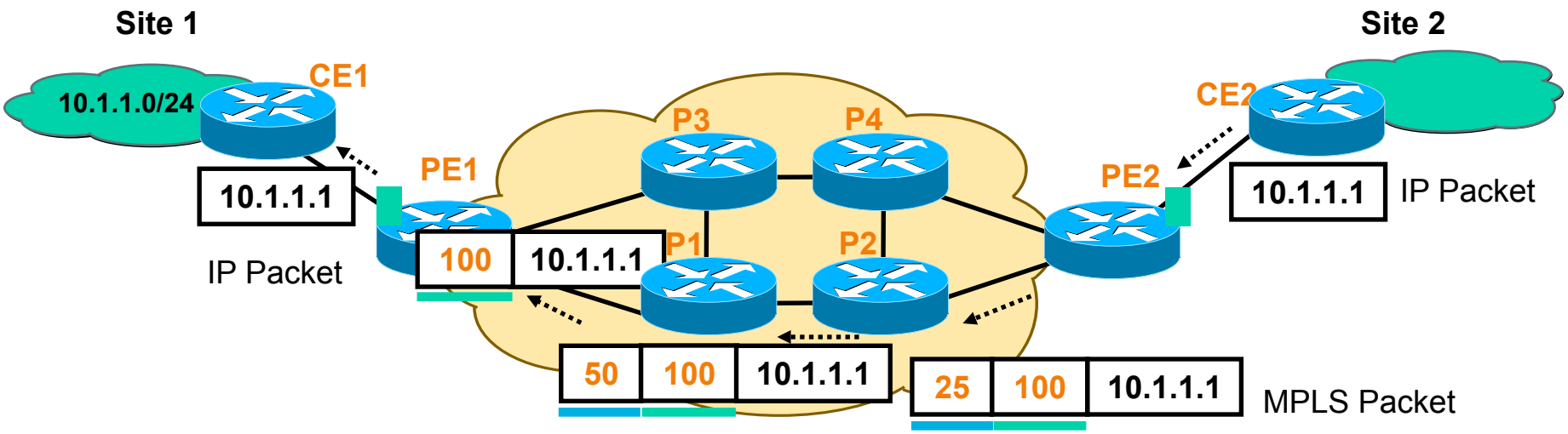
4. PE2 receives and checks whether the RT=200:1 is locally configured as import RT within any VRF, if yes, then
   - PE2 translates VPNv4 prefix back to IPv4 prefix
   - Updates the VRF CEF table for 10.1.1.0/24 with label=100
5. PE2 advertises this IPv4 prefix to CE2

Control Plane: Tunnel Label



- LDP runs on the MPLS backbone network to build the public LSP. The tunnel label is also called transport label or public label.
- Local label mapping are sent to connected nodes. Receiving nodes update forwarding table.

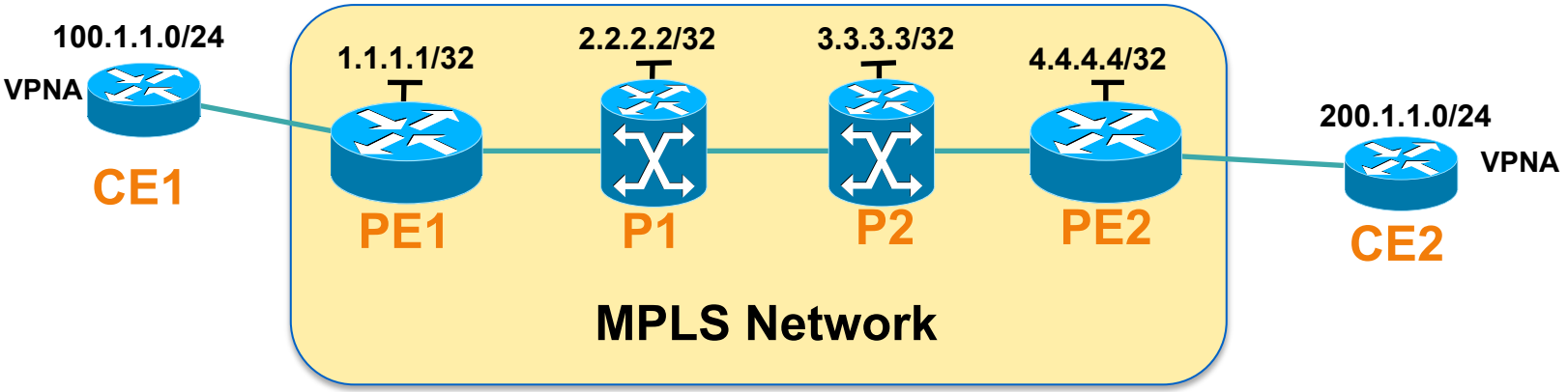
Data Plane



- PE2 imposes two labels for each IP packet going to site2
  - Tunnel label is learned via LDP; corresponds to PE1 address
  - VPN label is learned via BGP; corresponds to the VPN address
- P1 does the Penultimate Hop Popping (PHP)
- PE1 retrieves IP packet (from received MPLS packet) and forwards it to CE1.

Configuration Example

- Task: Configure MPLS L3VPN on Cisco **IOS** (Version 15.2) to make the following CEs communication with each other.
- Prerequisite configuration:
  - 1. IP address configuration on PE & P routers
  - 2. IGP configuration on PE & P routers
    - Make sure all the routers in public network can reach each other.



Configure MPLS & LDP

- Configuration steps:
  - 1. Configure MPLS and LDP on PE & P routers

```
ip cef
mpls ldp router-id loopback 0

interface ethernet1/0
mpls ip
mpls label protocol ldp

interface ethernet1/1
mpls ip
mpls label protocol ldp
```

Configure VRF

- Configuration steps:
  - 2. Configure VRF instance on PE routers

```
vrf definition VPNA
 rd 100:10
 route-target export 100:100
 route-target import 100:100
 !
 address-family ipv4
 exit-address-family
!
```

  - Bind PE-CE interface under VRF

```
interface FastEthernet0/0
 vrf forwarding VPNA
 ip address 10.1.1.1 255.255.255.252
```

Configure MP-iBGP

- Configuration steps:
  - 3. Enable MP-iBGP neighbors in vpnv4 address-family on PE routers

```
router bgp 100
 neighbor 4.4.4.4 remote-as 100
 neighbor 4.4.4.4 update-source loopback 0
 !
 address-family vpnv4
  neighbor 4.4.4.4 activate
  neighbor 4.4.4.4 send-community both
 exit-address-family
!
```

Configure PE-CE eBGP Neighbour

- Configuration steps:
  - 4. Adding PE-CE eBGP neighbour in VRF context of BGP on PE

```
router bgp 100
 address-family ipv4 vrf VPNA
  neighbor 10.1.1.2 remote-as 65001
  neighbor 10.1.1.2 activate
 exit-address-family
!
```

Adding PE-CE eBGP neighbour in BGP on CE

```
router bgp 65001
 neighbor 10.1.1.1 remote-as 100
 !
 address-family ipv4
  network 100.1.1.0 mask 255.255.255.0
  neighbor 10.1.1.1 activate
 exit-address-family
!
ip route 100.1.1.0 255.255.255.0 null 0
```

Verify Results – VRF Routing Table

- Check the routes of VRF VPNA on PE.

```
PE1#show bgp vpnv4 unicast vrf VPNA
BGP table version is 4, local router ID is 1.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
Route Distinguisher: 100:10 (default for vrf VPNA)
 *>  100.1.1.0/24    10.1.1.2                 0             0 65001 i
 *>i 200.1.1.0       4.4.4.4                  0    100      0 65002 i
```

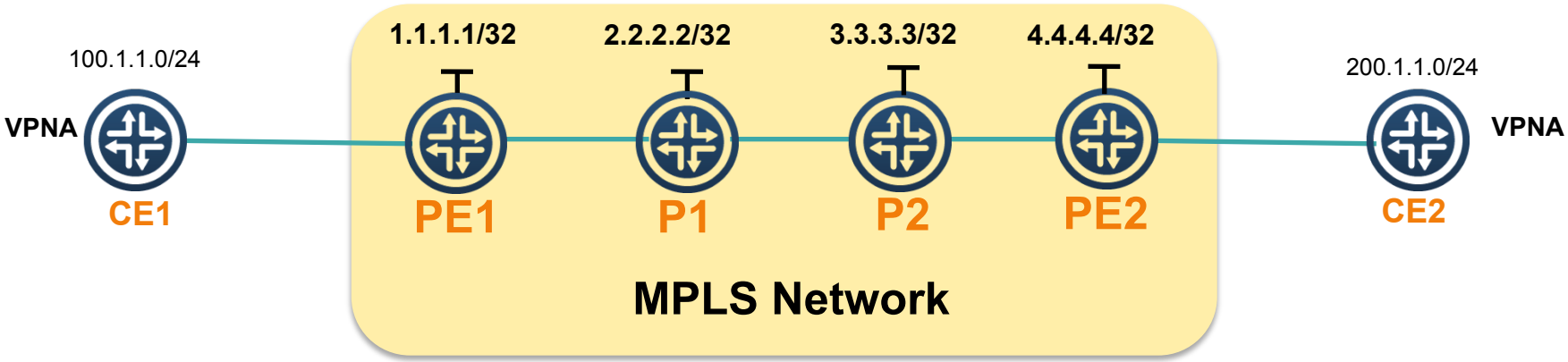
Verify Results – VPN Reachability

- CE can learn the routes from each other:

```
CE2#show ip route
....
      10.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C        10.1.2.0/30 is directly connected, FastEthernet0/1
L        10.1.2.2/32 is directly connected, FastEthernet0/1
      100.0.0.0/24 is subnetted, 1 subnets
B        100.1.1.0 [20/0] via 10.1.2.1, 00:38:26
      200.1.1.0/24 is variably subnetted, 2 subnets, 2 masks
S        200.1.1.0/24 is directly connected, Null0
C        200.1.1.1/32 is directly connected, Loopback1
```

Configuration Example

- Task: Configure MPLS L3VPN on **Juniper Junos** (Version 12.1) to make the following CEs communication with each other.
- Prerequisite configuration:
  - 1. IP address configuration on PE & P routers
  - 2. IGP configuration on PE & P routers
    - Make sure all the routers in public network can reach each other.



Configure MPLS & LDP

- Configuration steps:
  - 1. Configure MPLS and LDP on PE & P routers
  - This is the example on PE1.

```
interfaces {
    em0 {
        unit 0 {
            family inet {
                address 10.0.12.1/30;
            }
            family mpls;
        }
    }
}
```

```
protocols {
    mpls {
        interface em0.0;
    }
     ldp {
        interface em0.0;
}}
```

Configure VRF

- Configuration steps:
  - 2. Configure VRF instance on PE routers.

```
routing-instances {
    VPNA {
        instance-type vrf;
        interface em1.0;
        route-distinguisher 100:10;
        vrf-target target:100:100;
    }
}
```

VPN instance and parameters, Interface em1.0 has been added in the VPNA

```
em1 {
        unit 0 {
            family inet {
                address 10.0.1.2/30;
            }
        }
    }
```

This is the interface configuration from PE to CE, as a normal interface

Configure MP-iBGP

- Configuration steps:
  - 3. Enable MP-iBGP neighbors in vpnv4 address-family on PE routers

```
routing-options {
    router-id 1.1.1.1;
    autonomous-system 100;
}
```

```
protocols {
    bgp {
        local-address 1.1.1.1;
        family inet-vpn {
            unicast;
        }
        group PE1-PE2 {
            type internal;
            neighbor 4.4.4.4;
} }
```

Configure PE-CE eBGP Neighbour

- Configuration steps:
  - 4. Adding PE-CE eBGP neighbour in VPN on PE

```
routing-instances {
    VPNA {
        instance-type vrf;
        interface em1.0;
        route-distinguisher 100:10;
        vrf-target target:100:100;
        protocols {
            bgp {
                group PE1-CE1 {
                    type external;
                    peer-as 65001;
                    neighbor 10.0.1.1;
}}}}}
```

Configure PE-CE eBGP Neighbour

- Configuration steps:
  - 4. Adding CE-PE eBGP neighbour in BGP on CE

```
routing-options {
    autonomous-system 65001;
}
protocols {
    bgp {
        group CE1-PE1 {
            type external;
            peer-as 100;
            neighbor 10.0.1.2;
        }
    }
}
```

CE1 is in AS 65001, sets up the neighbor with AS100.

Advertise Static Route on CE

- Configuration steps:
  - 5. Advertise routes on CE routers, CE1 advertises 100.1.1.0/24, CE2 advertises 200.1.1.0/24

```
routing-options {
    generate {
        route 100.1.1.0/24 passive;
}}
```

Generate a static route.

```
policy-options {
    policy-statement ADVERTISE-PREFIX {
        from {
            route-filter 100.1.1.0/24 exact;
        }
        then accept;
}}
```

Define the route policy

```
protocols {
    bgp {
        group CE1-PE1 {
            export ADVERTISE-PREFIX;
}}}
```

Apply the policy in eBGP neighbor, only advertise 100.1.1.0/24

Verify Results – VRF Routing Table

- Check the routes of VRF VPNA on PE.

```
root@PE1> show route receive-protocol bgp 4.4.4.4

inet.0: 10 destinations, 10 routes (10 active, 0 holddown, 0 hidden)

inet.3: 3 destinations, 3 routes (3 active, 0 holddown, 0 hidden)

VPNA.inet.0: 5 destinations, 5 routes (5 active, 0 holddown, 0 hidden)
  Prefix   Nexthop       MED     Lclpref    AS path
* 10.0.2.0/30            4.4.4.4                      100        I
* 200.1.1.0/24           4.4.4.4                      100        65002 I


mpls.0: 9 destinations, 9 routes (9 active, 0 holddown, 0 hidden)


bgp.l3vpn.0: 2 destinations, 2 routes (2 active, 0 holddown, 0 hidden)
  Prefix   Nexthop       MED     Lclpref    AS path
  100:20:10.0.2.0/30
*                         4.4.4.4                      100        I
  100:20:200.1.1.0/24
*                         4.4.4.4                      100        65002 I
```

RD on PE2 is 100:20

Check VPN Routes in BGP

- Check the detailed route of VRF VPNA on PE received from remote PE.

```
root@PE1> show route receive-protocol bgp 4.4.4.4 detail
......(Omitted)
bgp.l3vpn.0: 2 destinations, 2 routes (2 active, 0 holddown, 0 hidden)

......(Omitted)


* 100:20:200.1.1.0/24 (1 entry, 0 announced)
     Import Accepted
     Route Distinguisher: 100:20
     VPN Label: 300016
     Nexthop: 4.4.4.4
     Localpref: 100
     AS path: 65002 I
     Communities: target:100:100
```

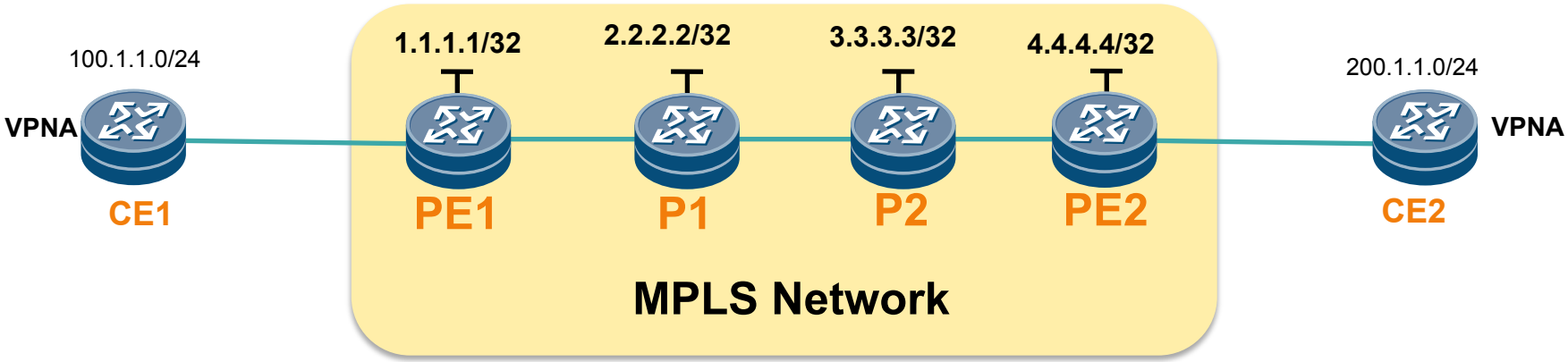
Verify Results – VPN Reachability

- CE can learn the routes from each other:

```
root@CE1> show route


inet.0: 5 destinations, 5 routes (5 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both


10.0.1.0/30        *[Direct/0] 00:44:34
                    > via em1.0
10.0.1.1/32        *[Local/0] 00:44:34
                      Local via em1.0
10.0.2.0/30        *[BGP/170] 00:04:23, localpref 100
                      AS path: 100 I
                    > to 10.0.1.2 via em1.0
100.1.1.0/24       *[Aggregate/130] 00:16:30
                      Reject
200.1.1.0/24       *[BGP/170] 00:04:24, localpref 100
                      AS path: 100 65002 I
                    > to 10.0.1.2 via em1.0
```

Configuration Example

- Task: Configure MPLS L3VPN on **Huawei VRP** (Version 5.1) to make the following CEs communication with each other.
- Prerequisite configuration:
  - 1. IP address configuration on PE & P routers
  - 2. IGP configuration on PE & P routers
    - Make sure all the routers in public network can reach each other.



Configure MPLS & LDP

- Configuration steps:
  - 1. Configure MPLS and LDP on PE & P routers

```
[PE1] mpls lsr-id 1.1.1.1
[PE1] mpls
Info: Mpls starting, please wait... OK!
[PE1-mpls] quit
[PE1] mpls ldp
[PE1-mpls-ldp] quit
[PE1] interface gigabitethernet 0/0/0
[PE1-GigabitEthernet0/0/0] mpls
[PE1-GigabitEthernet0/0/0] mpls ldp
[PE1-GigabitEthernet0/0/0] quit
```

Configure VRF

- Configuration steps:
  - 2. Configure VRF instance on PE routers

```
[PE1] ip vpn-instance VPNA
[PE1-vpn-instance-VPNA] ipv4-family
[PE1-vpn-instance-VPNA-af-ipv4] route-distinguisher 100:10
[PE1-vpn-instance-VPNA-af-ipv4] vpn-target 100:100 both
 IVT Assignment result:
Info: VPN-Target assignment is successful.
 EVT Assignment result:
Info: VPN-Target assignment is successful.
[PE1-vpn-instance-VPNA-af-ipv4] quit
```

  - Bind PE-CE interface under VRF

```
[PE1] interface gigabitethernet 0/0/1
[PE1-GigabitEthernet0/0/1] ip binding vpn-instance vpna
Info: All IPv4 related configurations on this interface are removed!
Info: All IPv6 related configurations on this interface are removed!
[PE1-GigabitEthernet0/0/1] ip address 10.1.1.1 30
[PE1-GigabitEthernet0/0/1] quit
```

Configure MP-iBGP

- Configuration steps:
  - 3. Enable MP-iBGP neighbors in vpnv4 address-family on PE routers

```
[PE1] bgp 100
[PE1-bgp] peer 4.4.4.4 as-number 100
[PE1-bgp] peer 4.4.4.4 connect-interface loopback 0
[PE1-bgp] ipv4-family vpnv4
[PE1-bgp-af-vpnv4] peer 4.4.4.4 enable
[PE1-bgp-af-vpnv4] quit
[PE1-bgp] quit
```

Configure PE-CE eBGP Neighbour

- Configuration steps:
  - 4. Adding PE-CE eBGP neighbour in VRF context of BGP on PE

```
[PE1] bgp 100
[PE1-bgp] ipv4-family vpn-instance VPNA
[PE1-bgp-vpna] peer 10.1.1.2 as-number 65001
[PE1-bgp-vpna] quit
```

Adding CE-PE eBGP neighbour in BGP on CE

```
[CE1] ip route-static 100.1.1.0 24 null 0
[CE1] bgp 65001
[CE1-bgp] peer 10.1.1.2 as-number 100
[CE1-bgp] network 100.1.1.0 24
[CE1-bgp] quit
```

Verify Results – VRF Routing Table

- Check the routes of VRF VPNA on PE.

```
<PE1> display bgp vpnv4 vpn-instance VPNA routing-table

 BGP Local router ID is 10.0.0.1
 Status codes: * - valid, > - best, d - damped,
               h - history,  i - internal, s - suppressed, S - Stale
               Origin : i - IGP, e - EGP, ? - incomplete


 VPN-Instance VPNA, Router ID 10.0.0.1:

 Total Number of Routes: 2
      Network            NextHop        MED        LocPrf    PrefVal Path/Ogn

 *>   100.1.1.0/24       10.1.1.2        0                      0      65001i
 *>i  200.1.1.0          4.4.4.4         0          100         0      65002i
```

Check VPN Routes in BGP

- Check the detailed route of VRF VPNA on PE.

```
<PE1> display bgp vpnv4 vpn-instance VPNA routing-table 200.1.1.0

 BGP local router ID : 1.1.1.1
 Local AS number : 100

 VPN-Instance VPNA, Router ID 1.1.1.1:
 Paths:   1 available, 1 best, 1 select
 BGP routing table entry information of 200.1.1.0/24:
 Label information (Received/Applied): 1028/NULL
 From: 4.4.4.4 (4.4.4.4)
 Route Duration: 00h00m04s
 Relay Tunnel Out-Interface: GigabitEthernet0/0/0
 Relay token: 0x18
 Original nexthop: 4.4.4.4
 Qos information : 0x0
 Ext-Community:RT <100 : 100>
 AS-path 65002, origin igp, MED 0, localpref 100, pref-val 0, valid, internal, b
est, select, active, pre 255, IGP cost 3
 Advertised to such 1 peers:
    10.1.1.2
```

Verify Results – VPN Reachability

- CE can learn the routes from each other:

```
[CE2]display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 7        Routes : 7

Destination/Mask    Proto   Pre  Cost      Flags NextHop         Interface

        10.1.2.0/30  Direct  0    0           D   10.1.2.2        GigabitEthernet
0/0/1
        10.1.2.2/32  Direct  0    0           D   127.0.0.1       GigabitEthernet
0/0/1
      100.1.1.0/24   EBGP    255  0           D   10.1.2.1        GigabitEthernet
0/0/1
      127.0.0.0/8    Direct  0    0           D   127.0.0.1       InLoopBack0
      127.0.0.1/32   Direct  0    0           D   127.0.0.1       InLoopBack0
      200.1.1.0/24   Static  60   0           D   0.0.0.0         NULL0
      200.1.1.1/32   Direct  0    0           D   127.0.0.1       LoopBack0
```

Questions?

Issue Date:

Revision:

APNIC

MPLS L3VPN Services

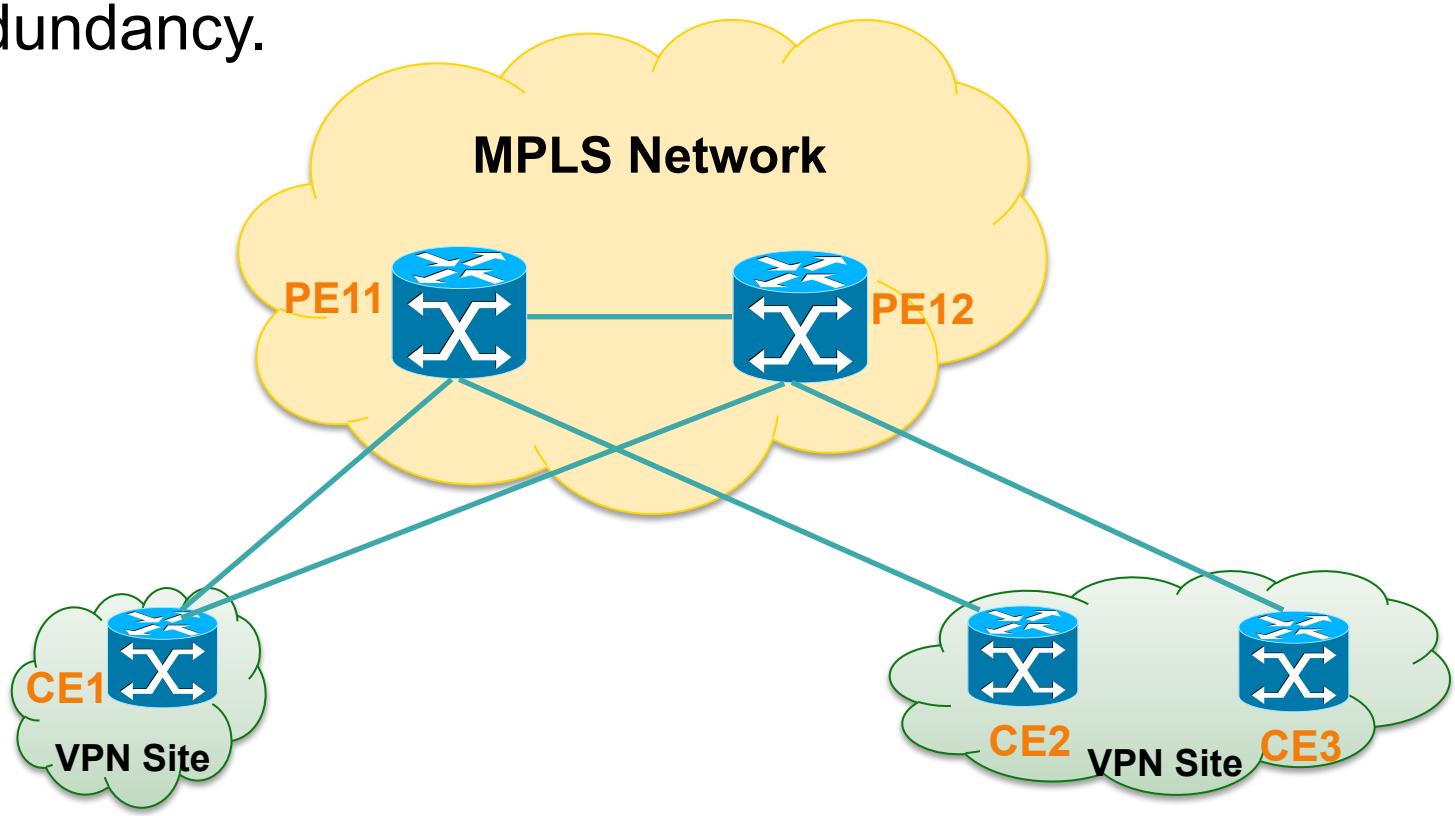
APNIC

MPLS L3VPN Services

- Multi-homed VPN Sites
- Hub and Spoke Service
- Extranet Service
- Internet Access Service

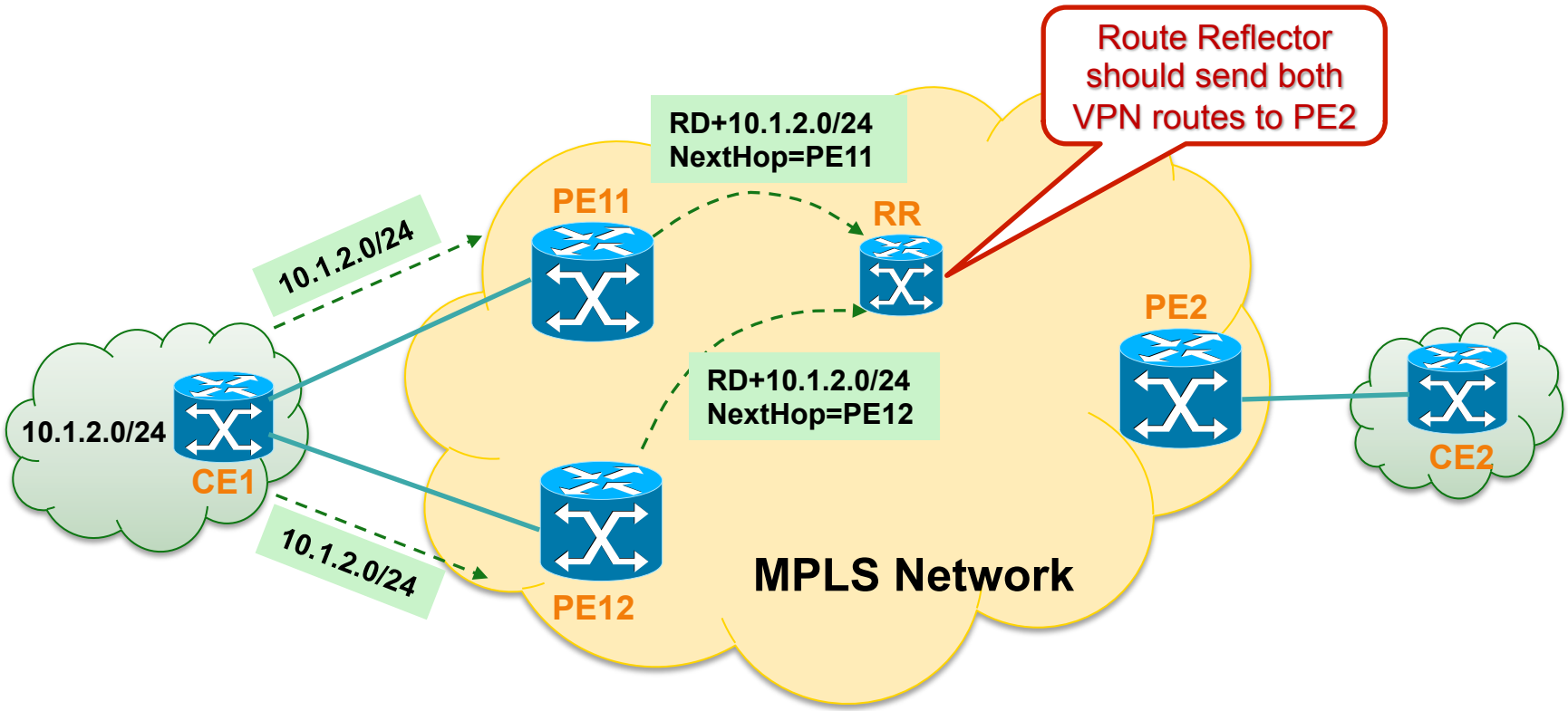
VPN Multihoming Scenarios

- In an MPLS VPN Layer 3 environment, it is common for customers to multihome their networks to provide link redundancy.



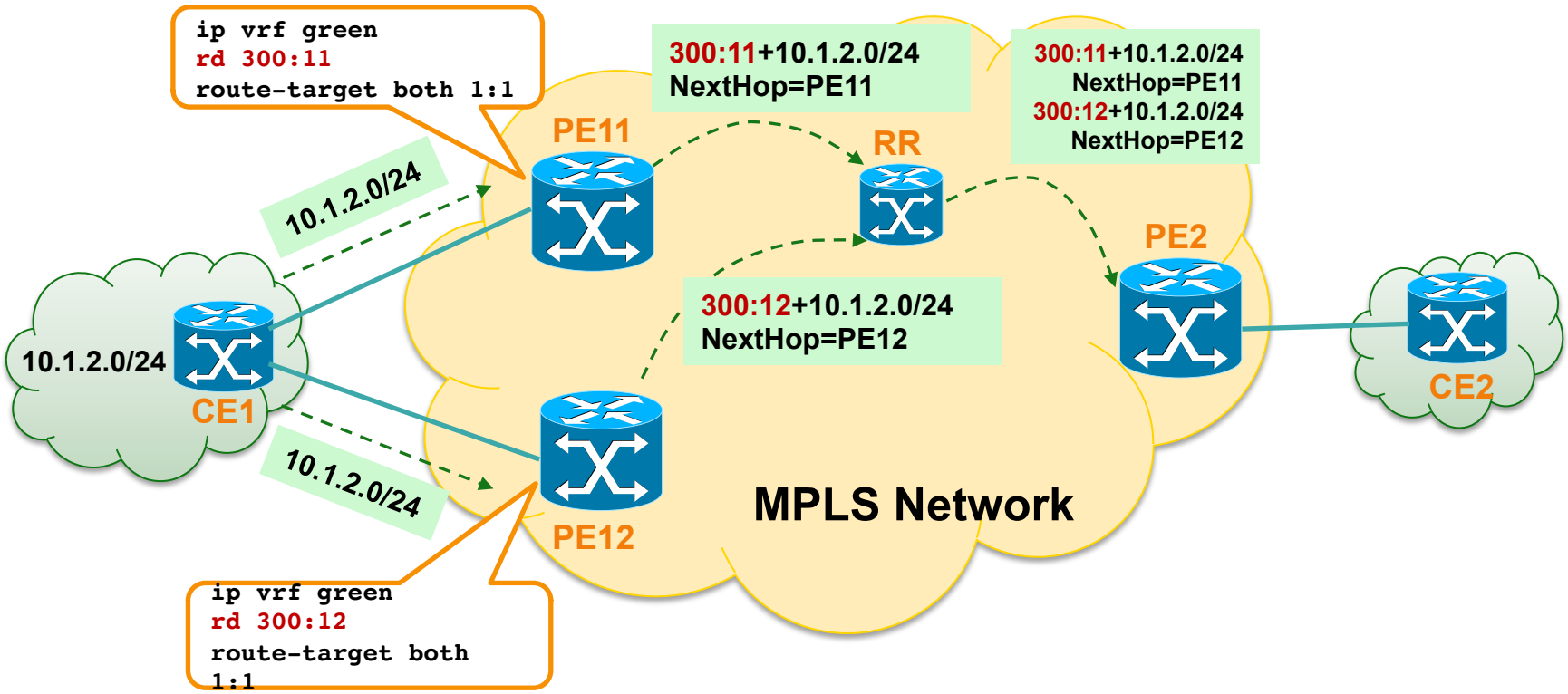
VPN Route Advertisement

- VPN route advertisement from multihomed VPN site.



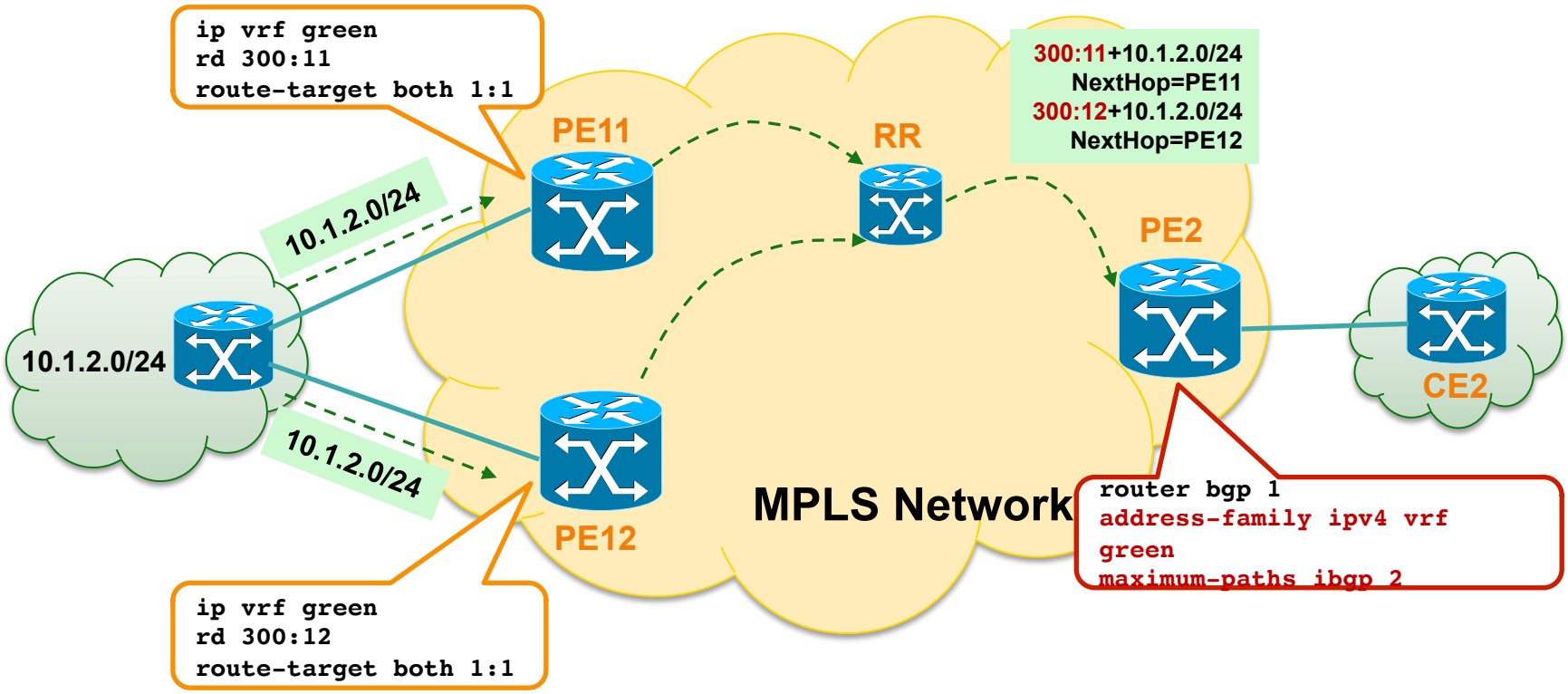
VPN Route Advertisement– Unique RD

- Configure unique RD per VRF per PE for multihomed site/ interfaces



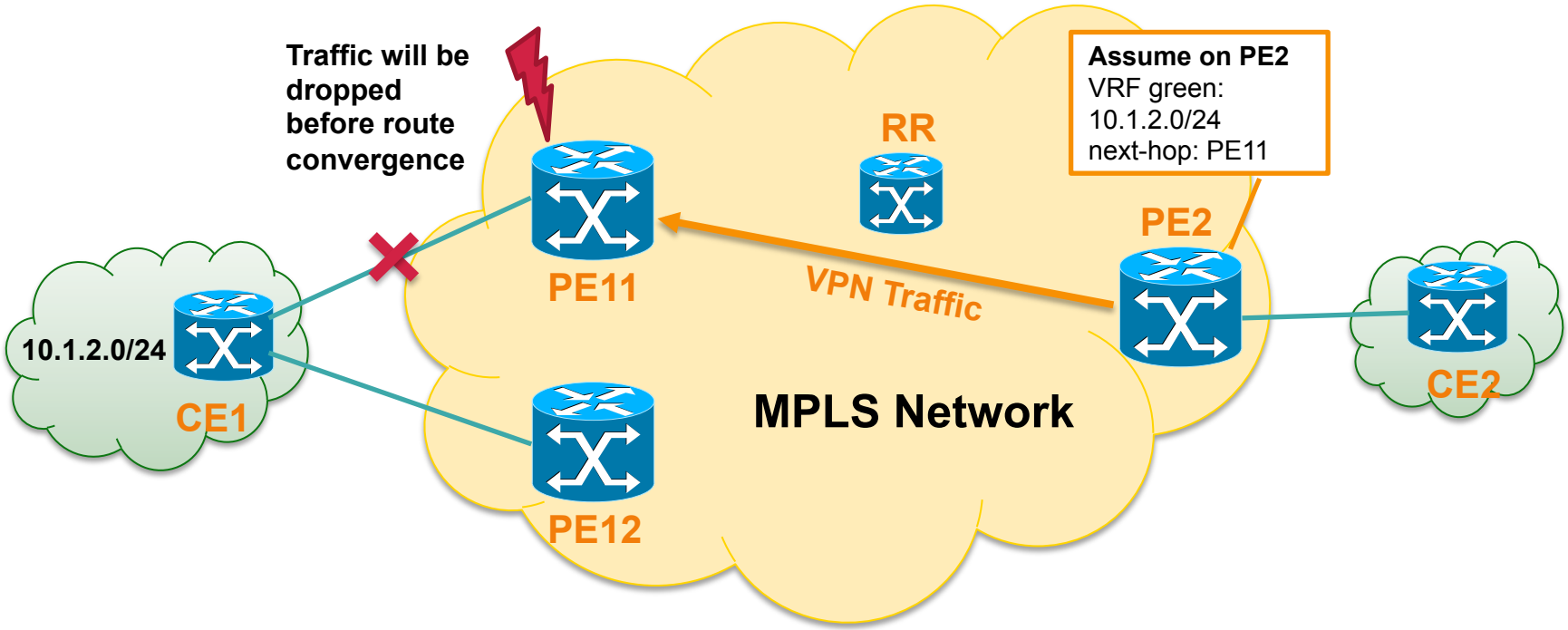
Load Sharing Configuration

- To implement load sharing between PE11 and PE12, enable BGP multipath at remote PE routers such as PE2.



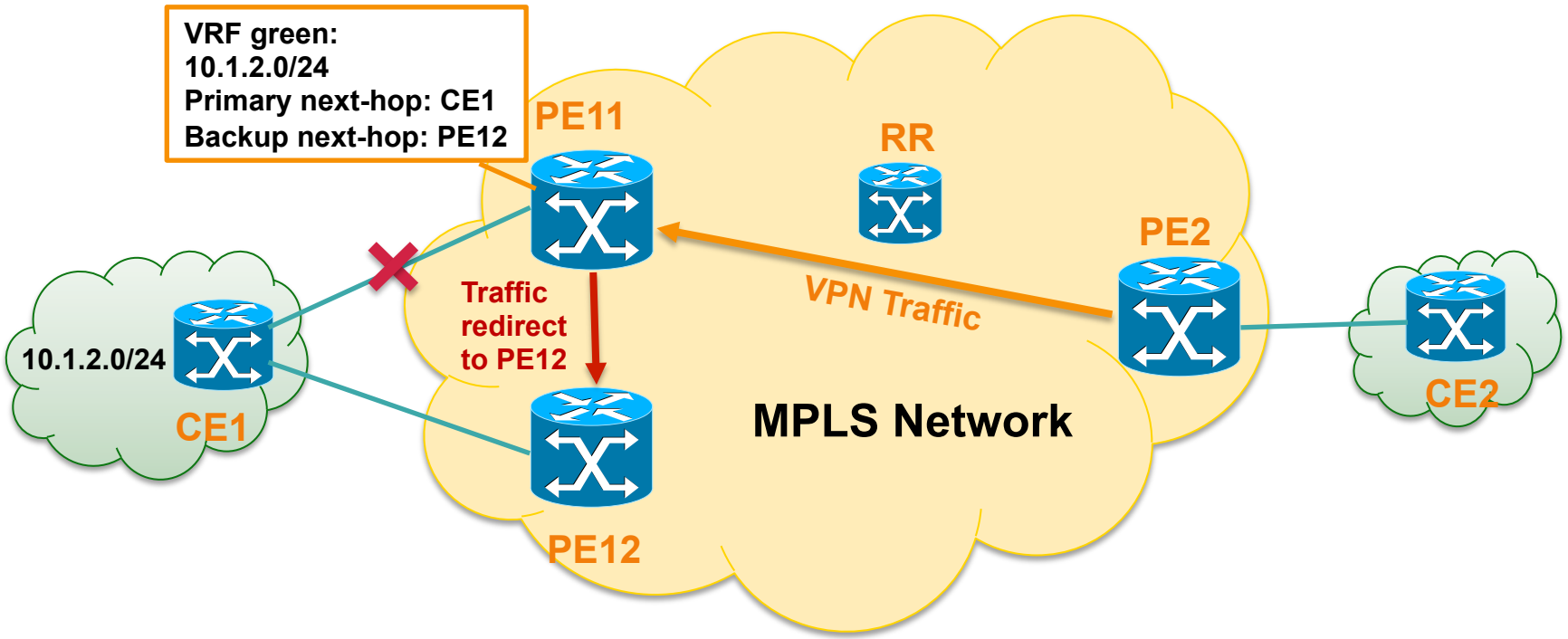
PE-CE Link Failure

- After detecting the PE-CE link failure, PE11 sends BGP message to withdraw the VPN routes, traffic will be dropped on PE11 before PE2 completes BGP route convergence.



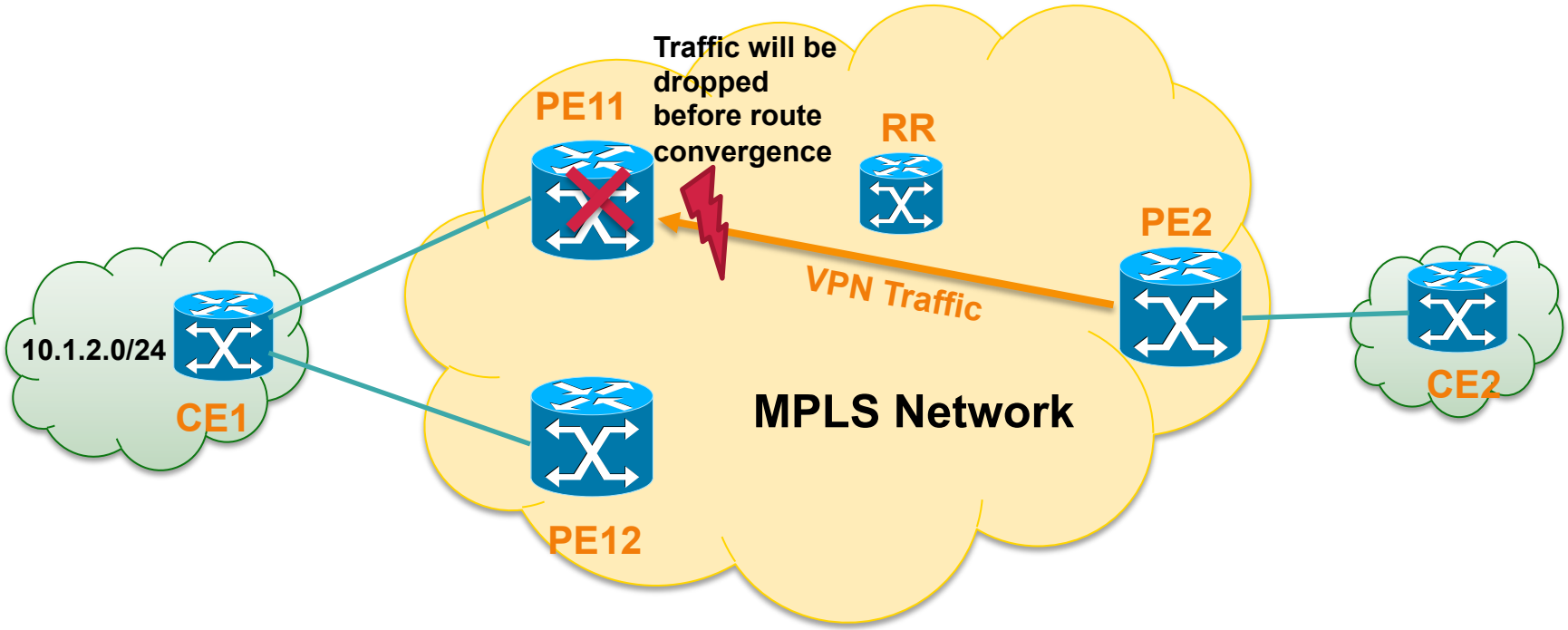
VPN Fast Convergence – PIC Edge

- Use **PIC Edge** feature to minimize the loss due to the PE-CE link failure from sec to msec.
- **P**refix **I**ndependent **C**onvergence is a method for speeding up convergence of the FIB under failover conditions.



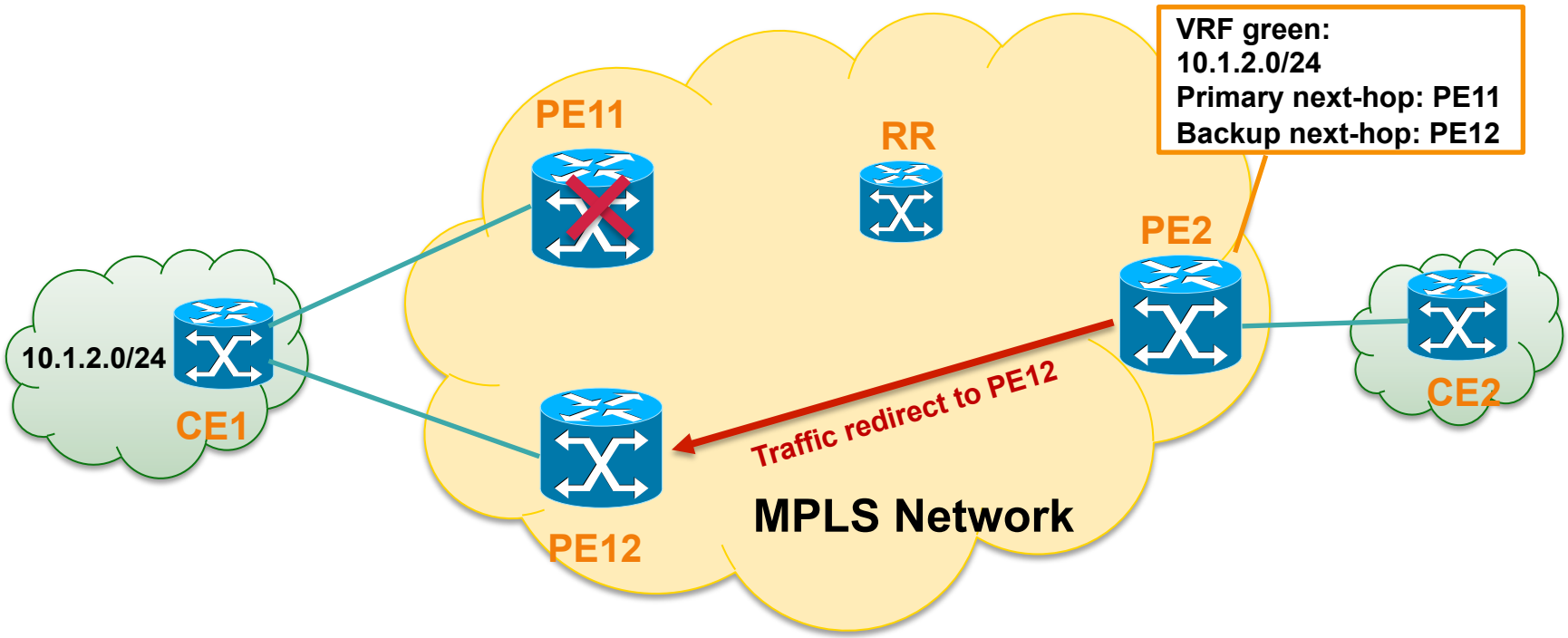
PE Node Failure

- When PE11 router fails, traffic will be lost before PE2 completes BGP route convergence.

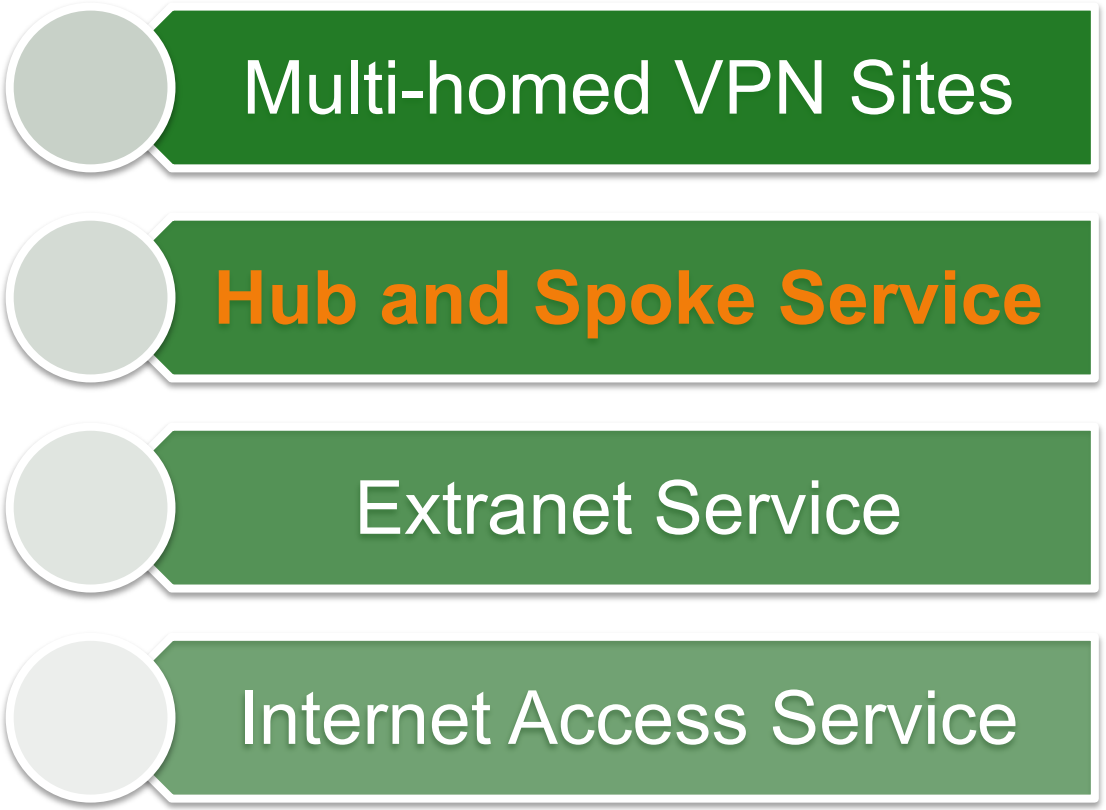


VPN Fast Convergence – PIC Edge

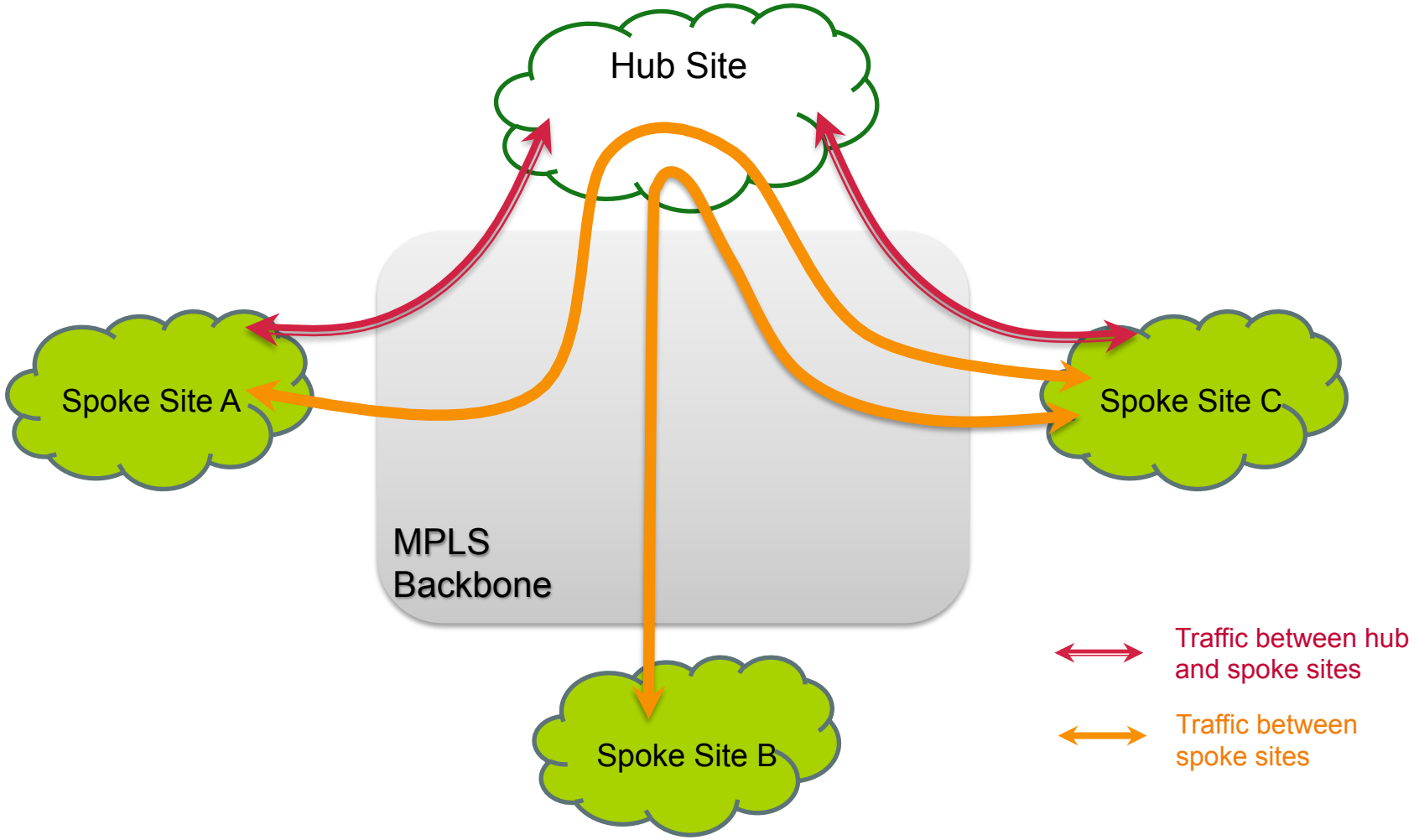
- PE2 uses the alternative VPN route for forwarding until global convergence is complete, this reduces traffic loss.



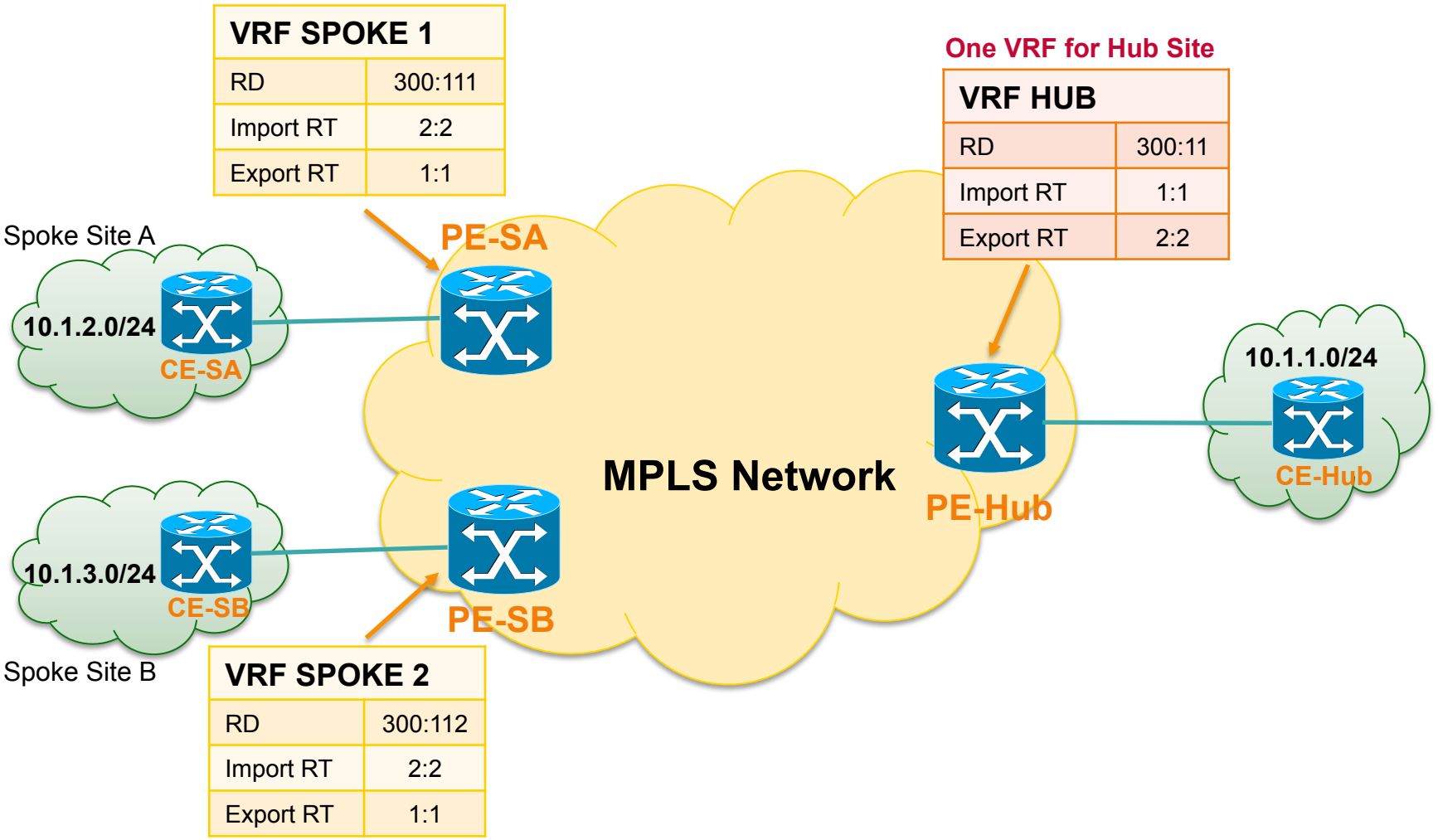
MPLS L3VPN Services



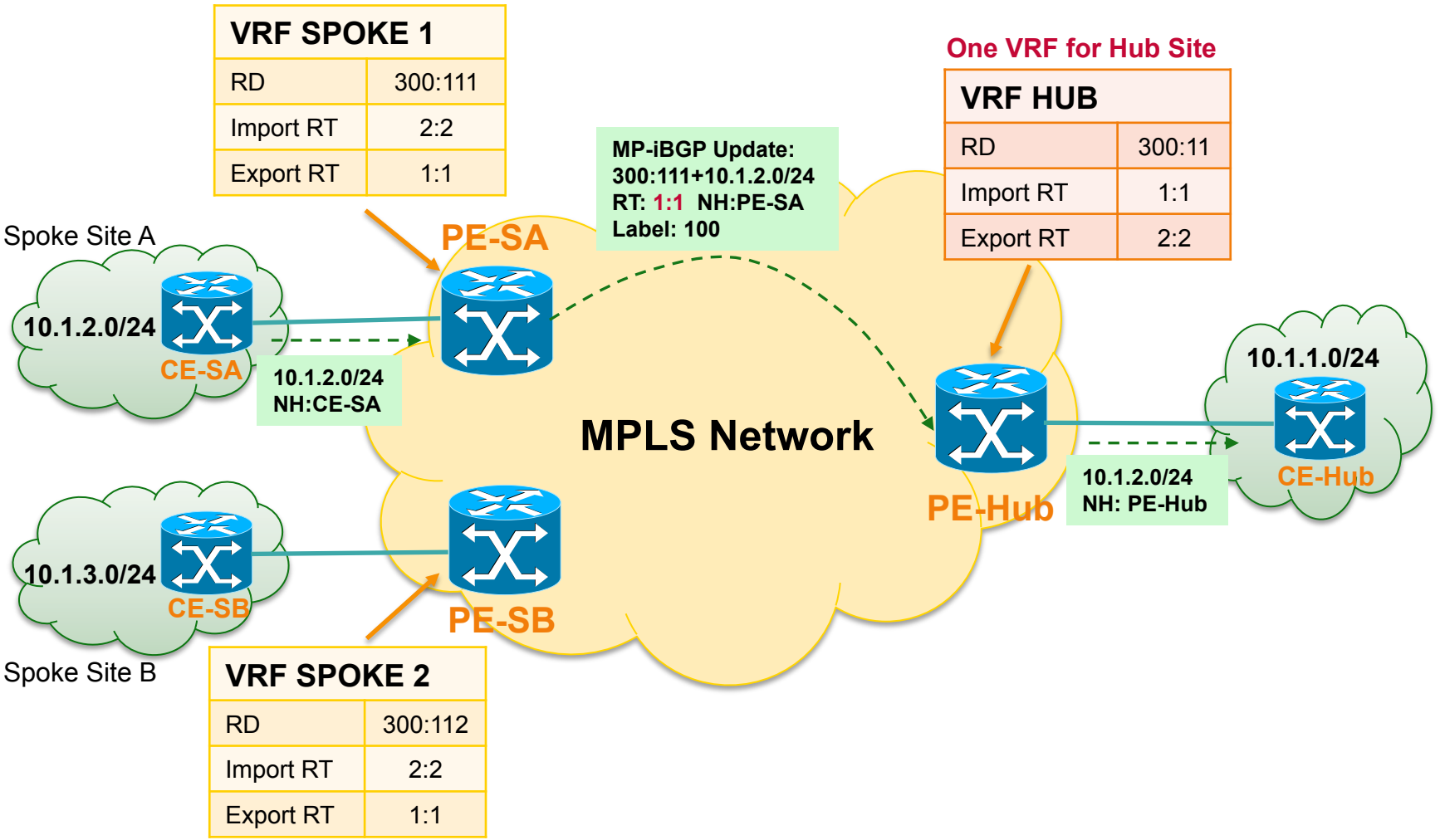
Hub and Spoke Service



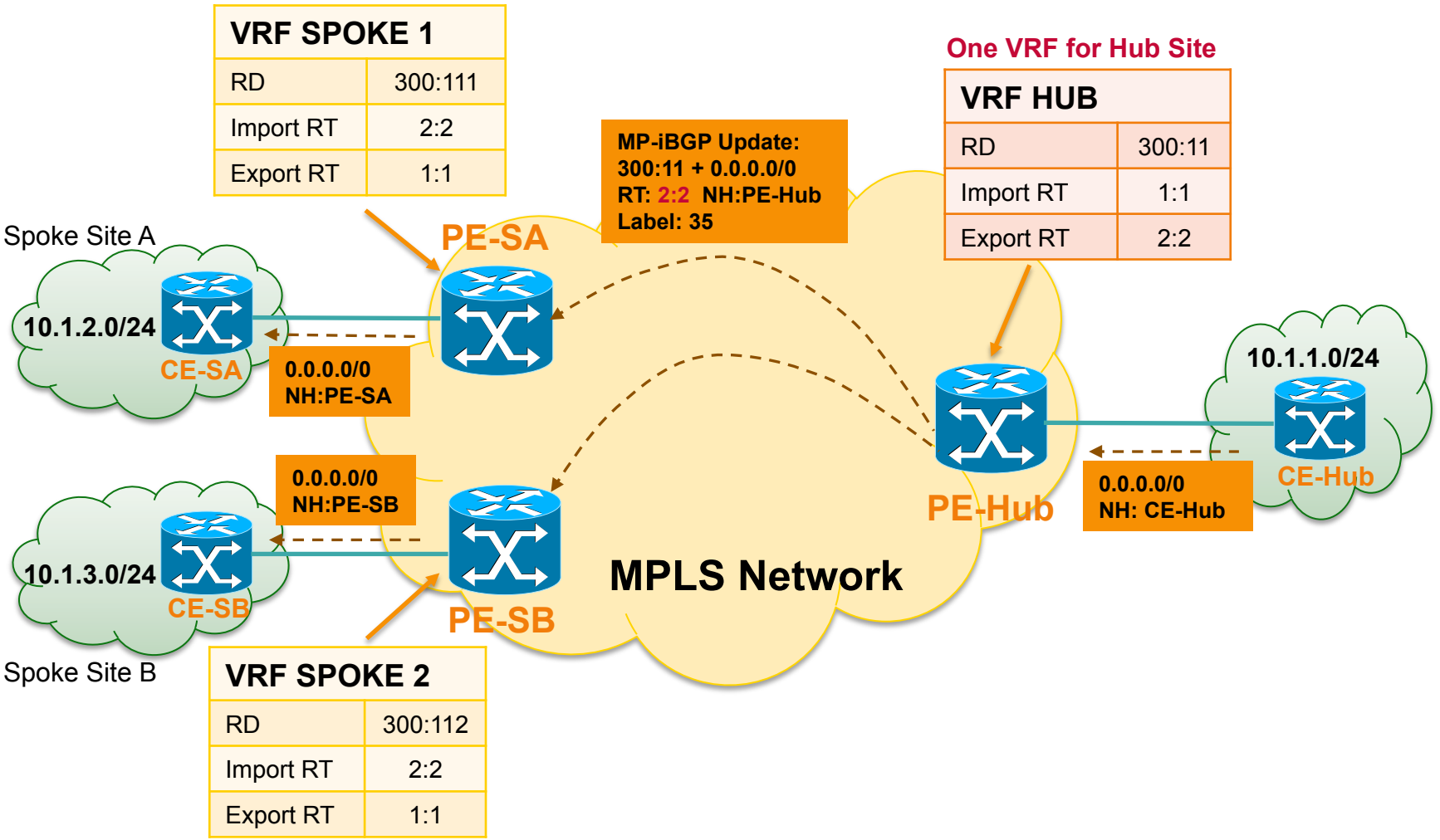
Option 1 - Single Interface



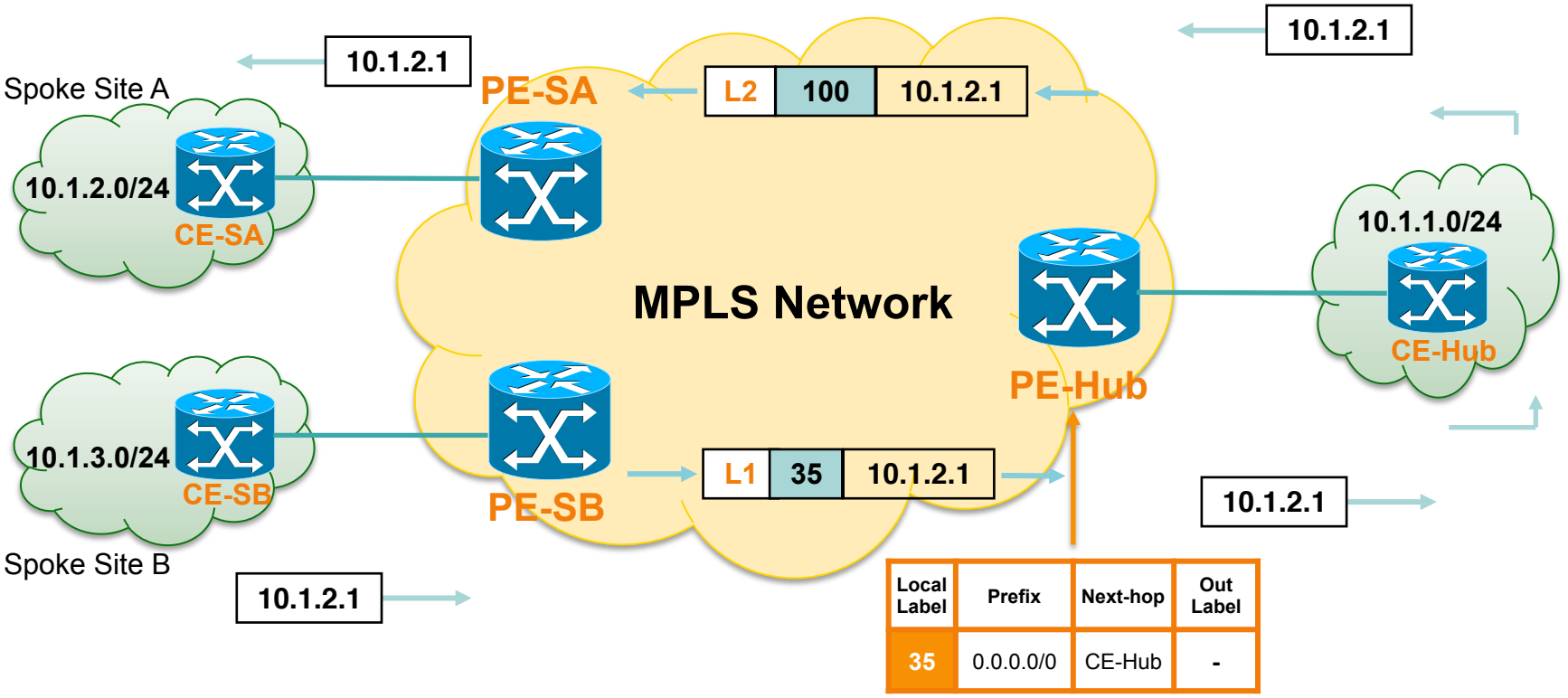
Control Plane – from Spoke to Hub



Control Plane – from Hub to Spoke

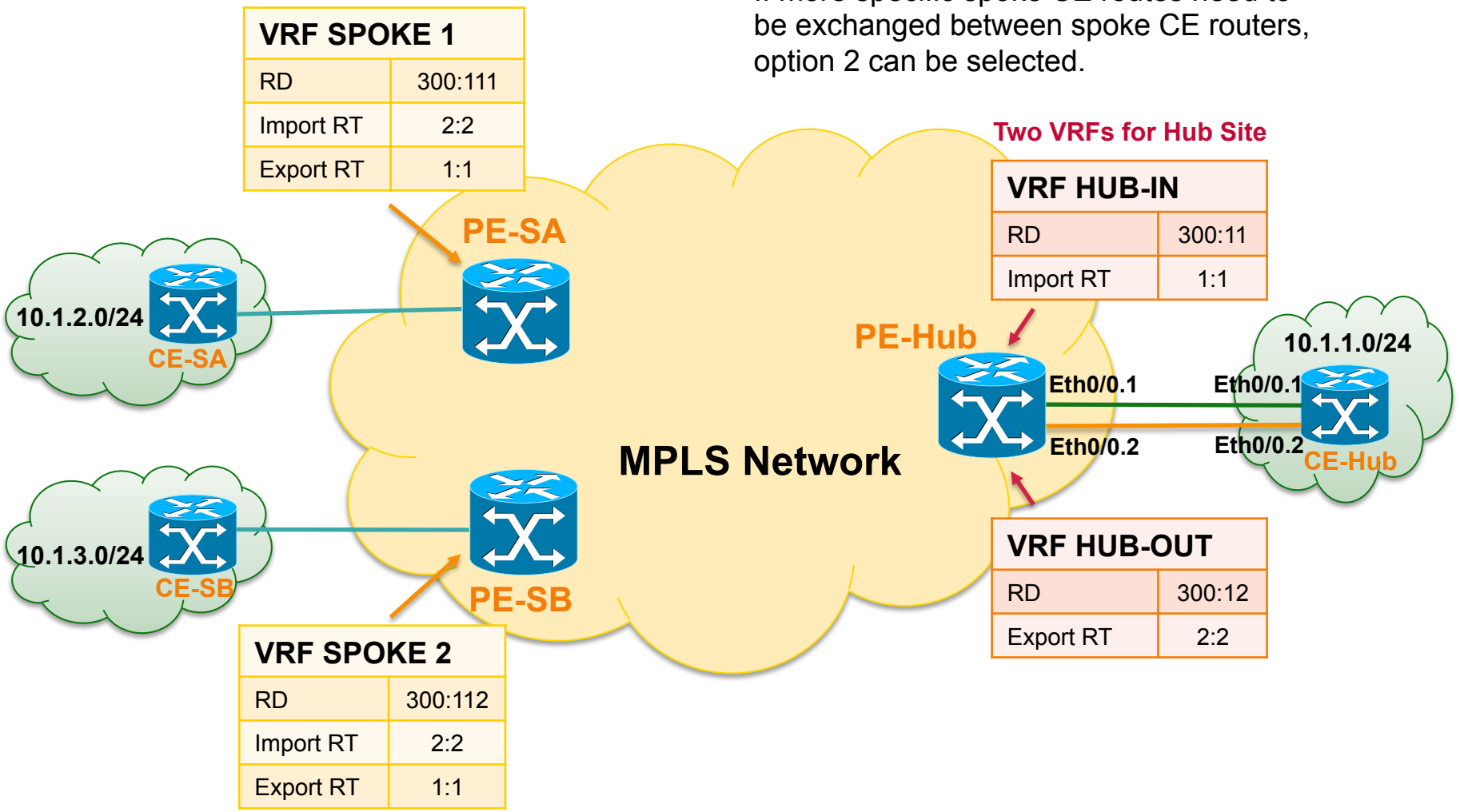


Data Plane – Traffic between Spoke Sites

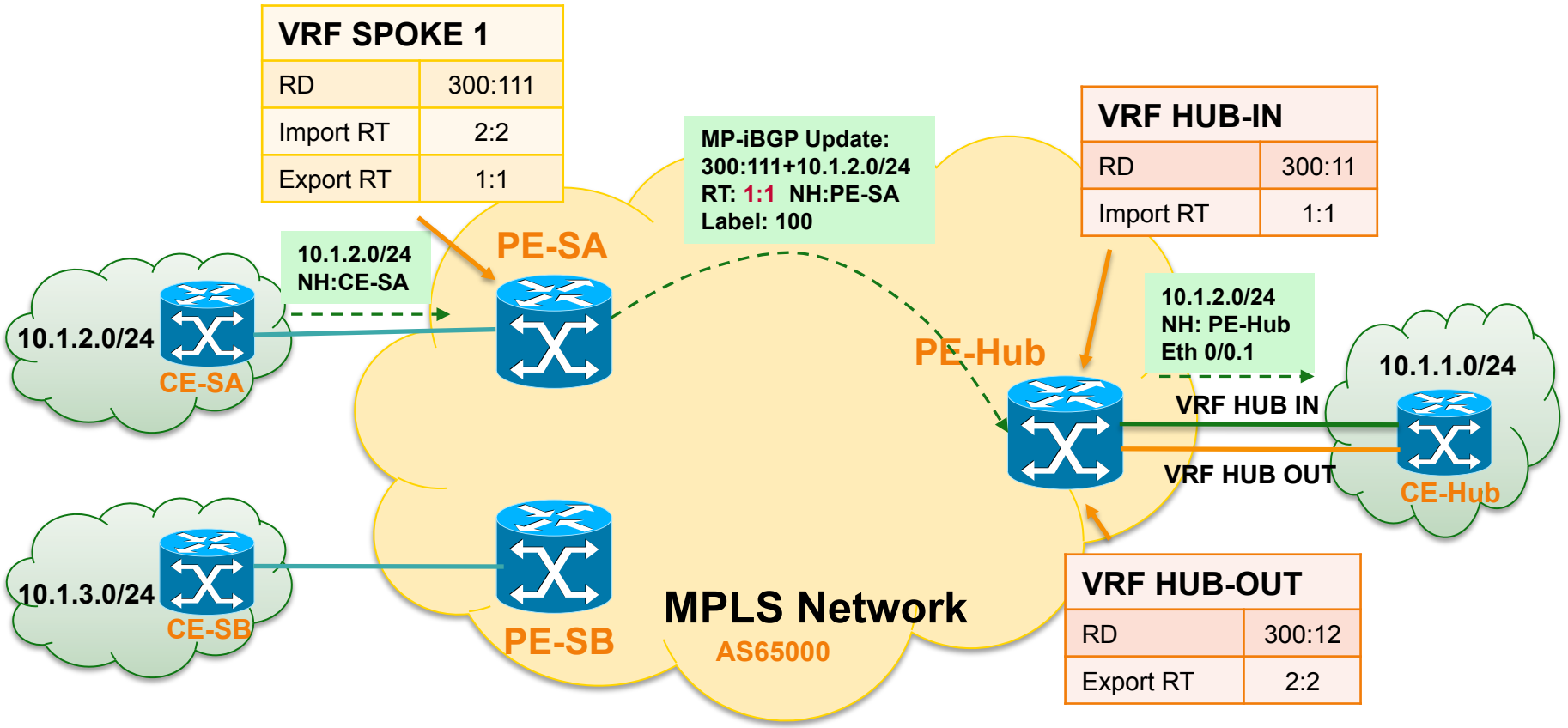


Option 2 – Two Interfaces

If more specific spoke CE routes need to be exchanged between spoke CE routers, option 2 can be selected.

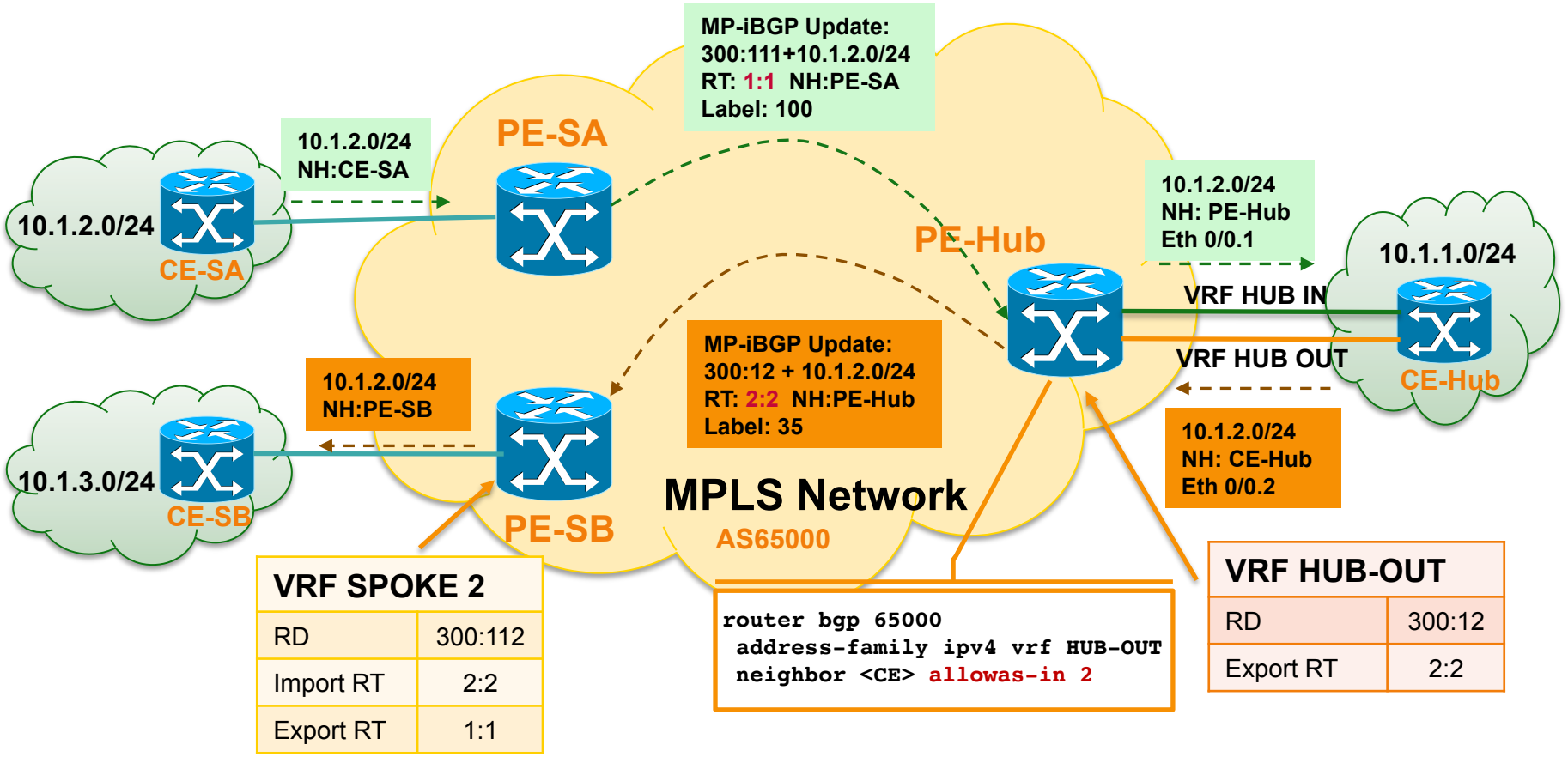


Option 2 – Control Plane (Hub in)

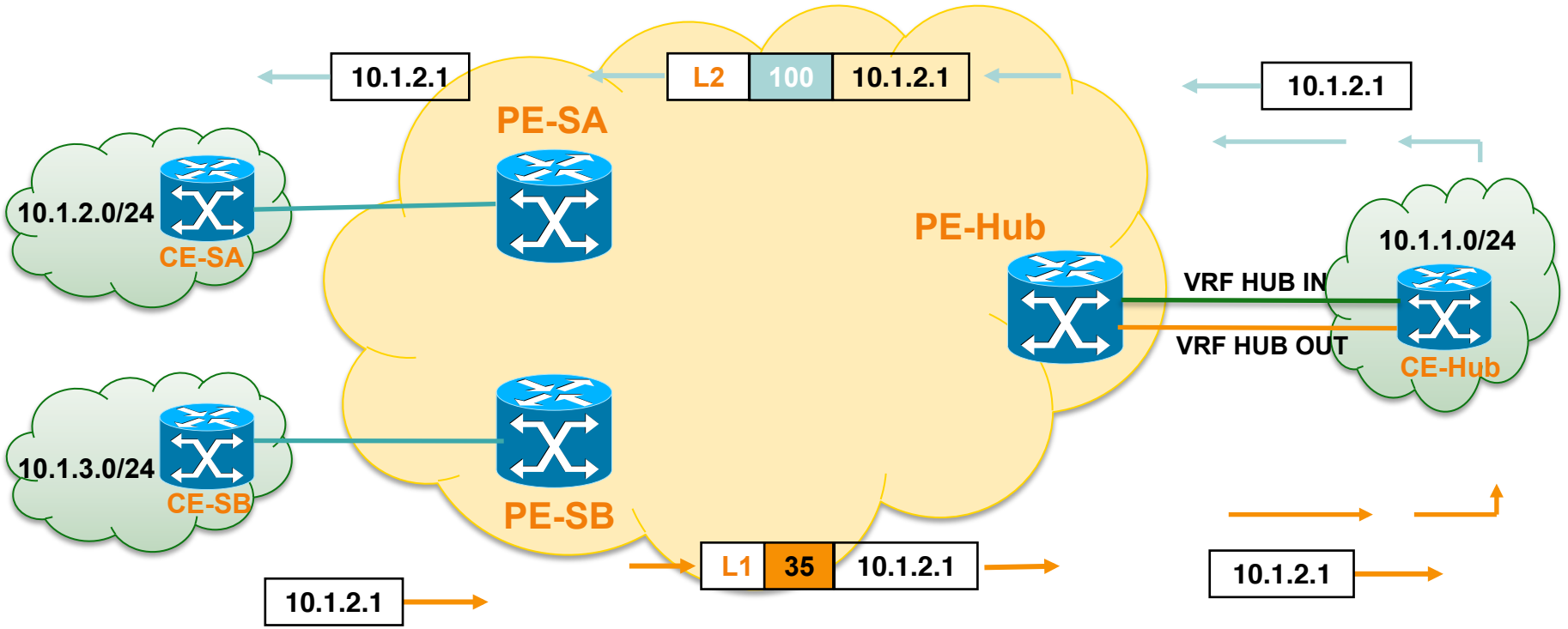


Option 2 – Control Plane (Hub out)

- Deployment of allowas-in feature



Option 2 – Data Plane



L1 Is the Label to Get to PE-Hub

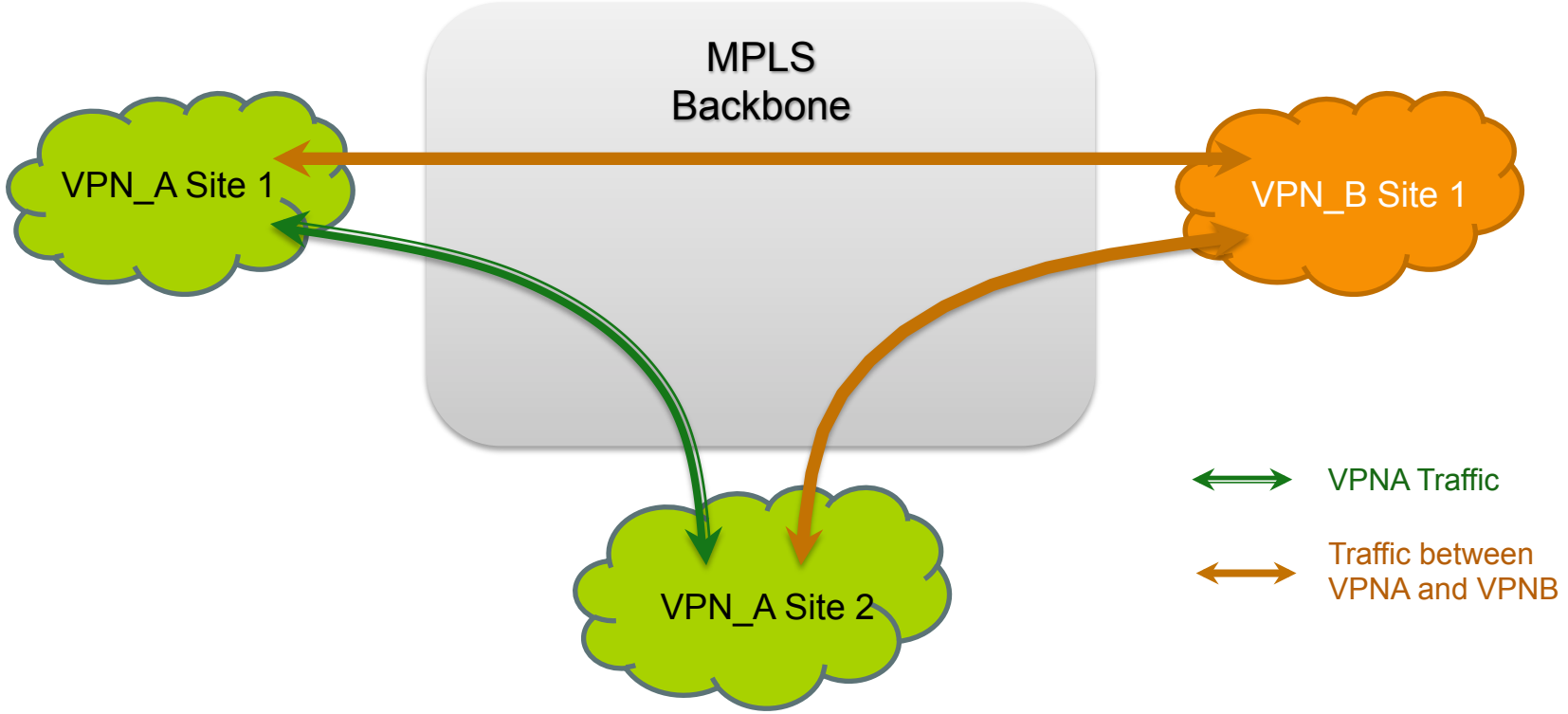
L2 Is the Label to Get to PE-SA

MPLS L3VPN Services

- Multi-homed VPN Sites
- Hub and Spoke Service
- **Extranet Service**
- Internet Access Service

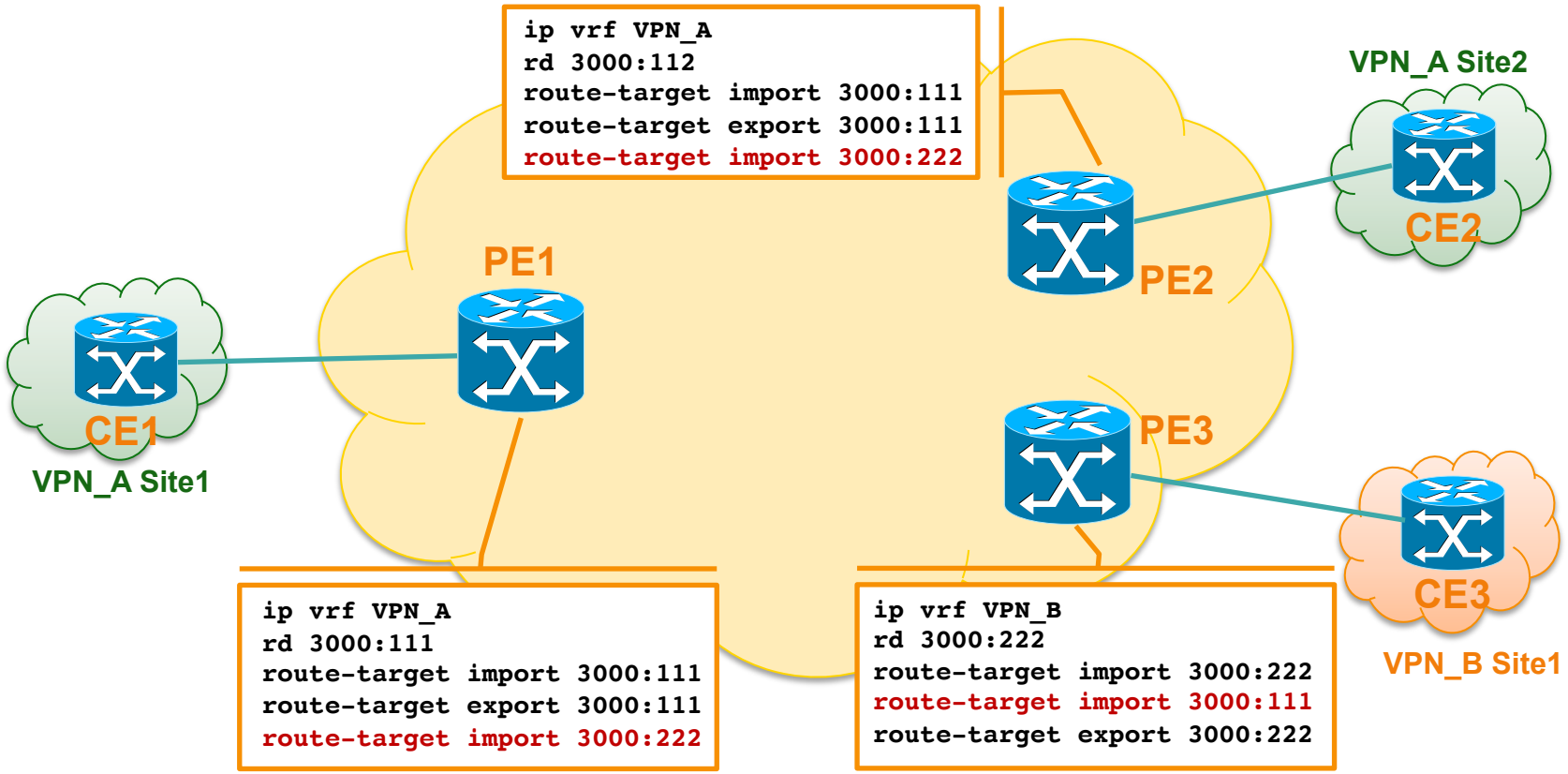
Extranet Service

- Communication between VPNs may be required
  i.e., External intercompany communication (dealers with manufacturer, retailer with wholesale provider, etc.)



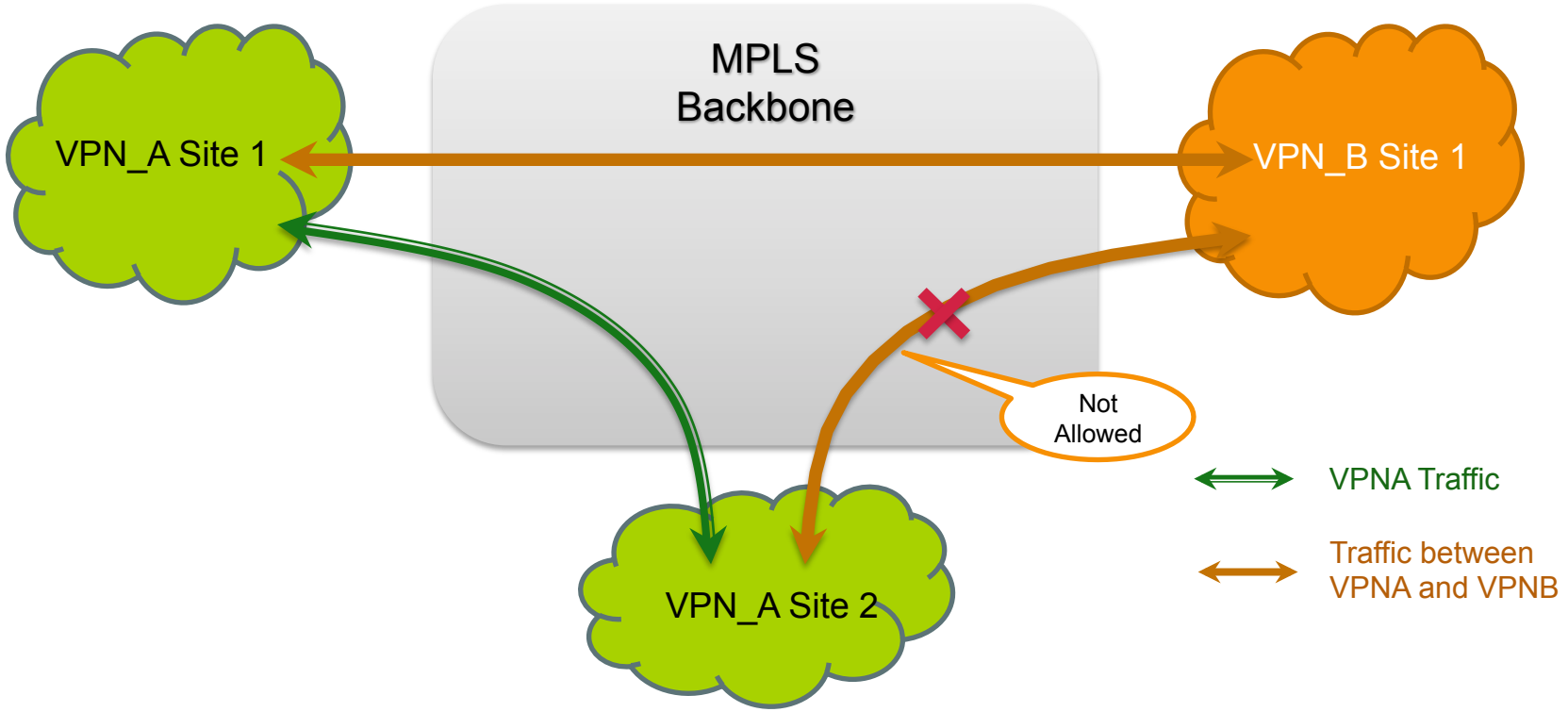
Extranet VPN – Simple Extranet

- Designing RT to implement the communication.

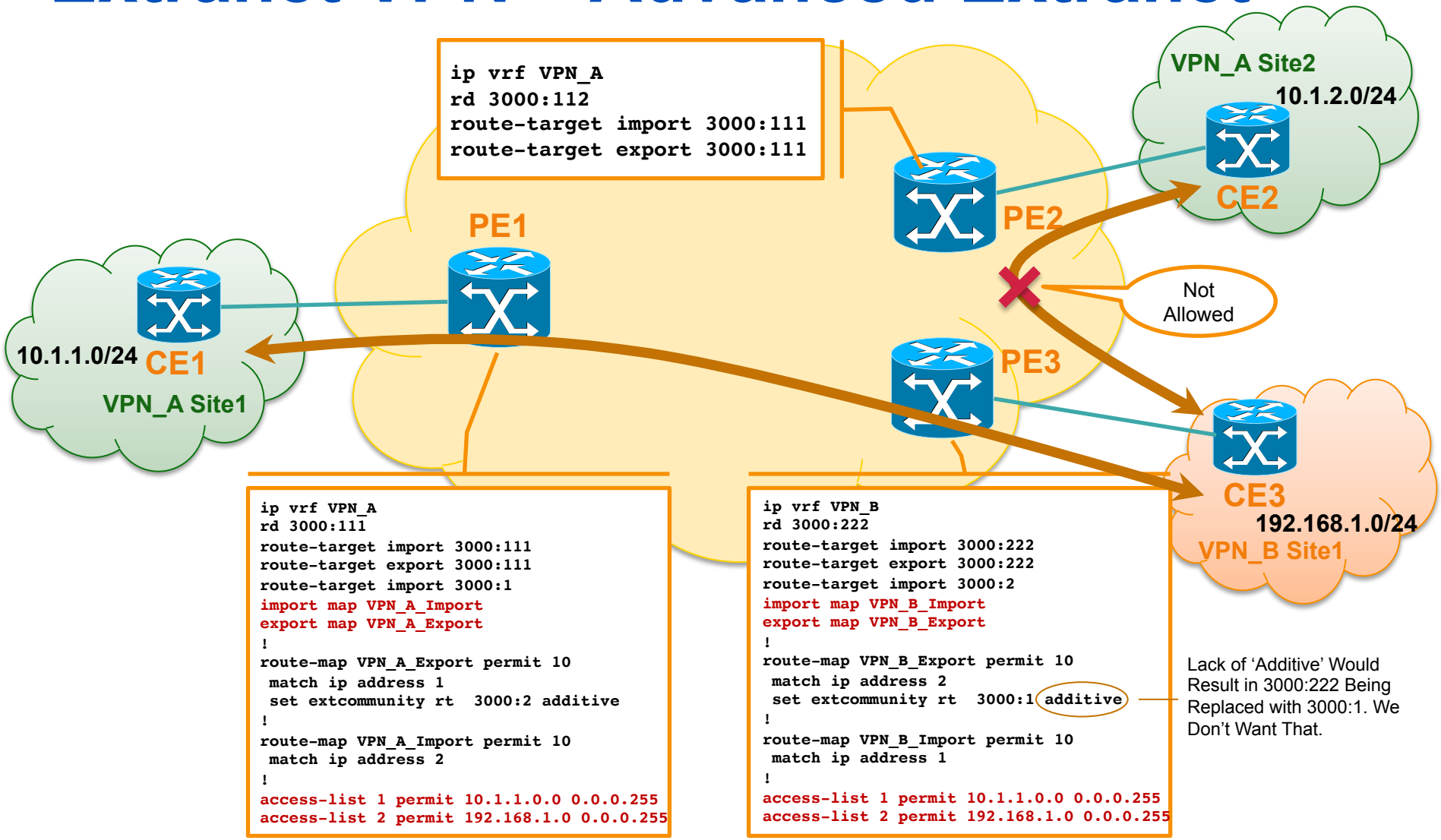


More Complex Scenario

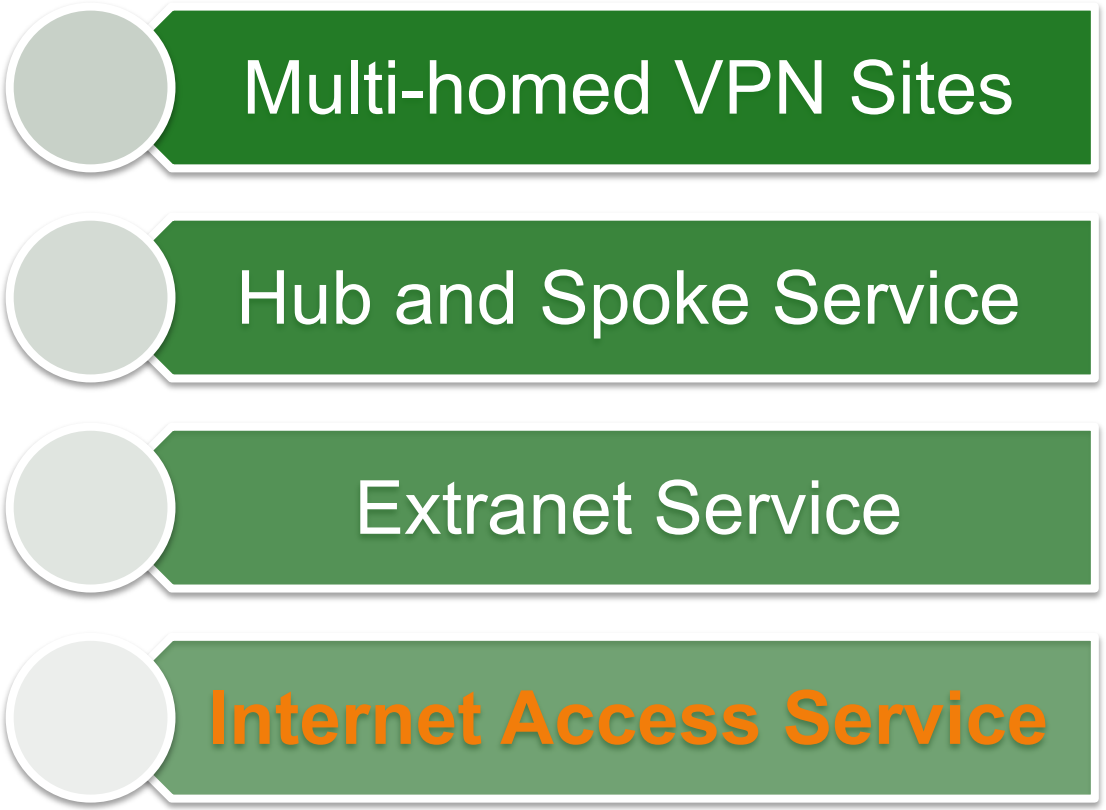
- If only allow VPNB Site1 to communicate with the servers in VPNA Site1.



Extranet VPN – Advanced Extranet



MPLS L3VPN Services



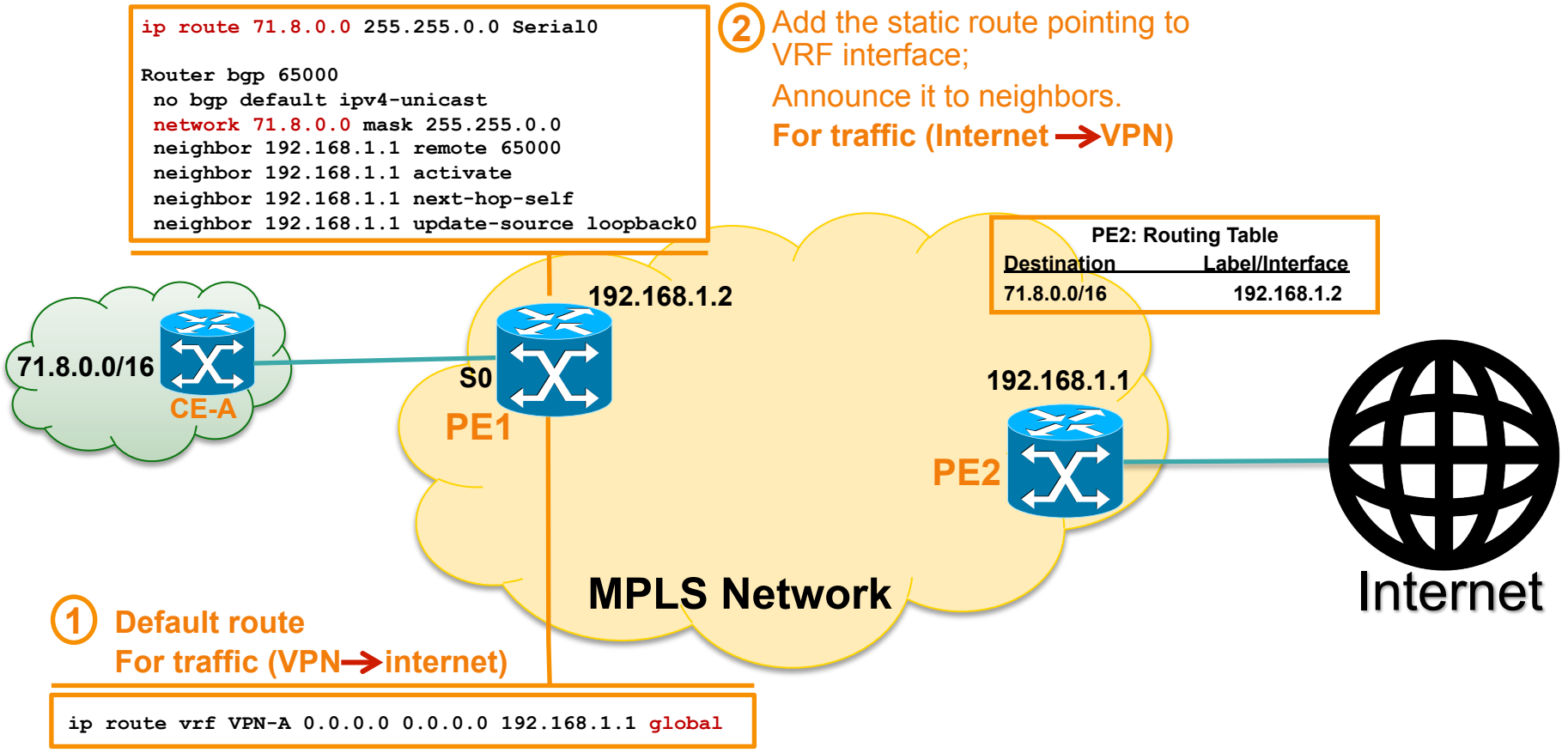
Internet Access Service to VPN Customers

- Internet access service could be provided as another value-added service to VPN customers
- Security mechanism must be in place at both provider network and customer network
  - To protect from the Internet vulnerabilities

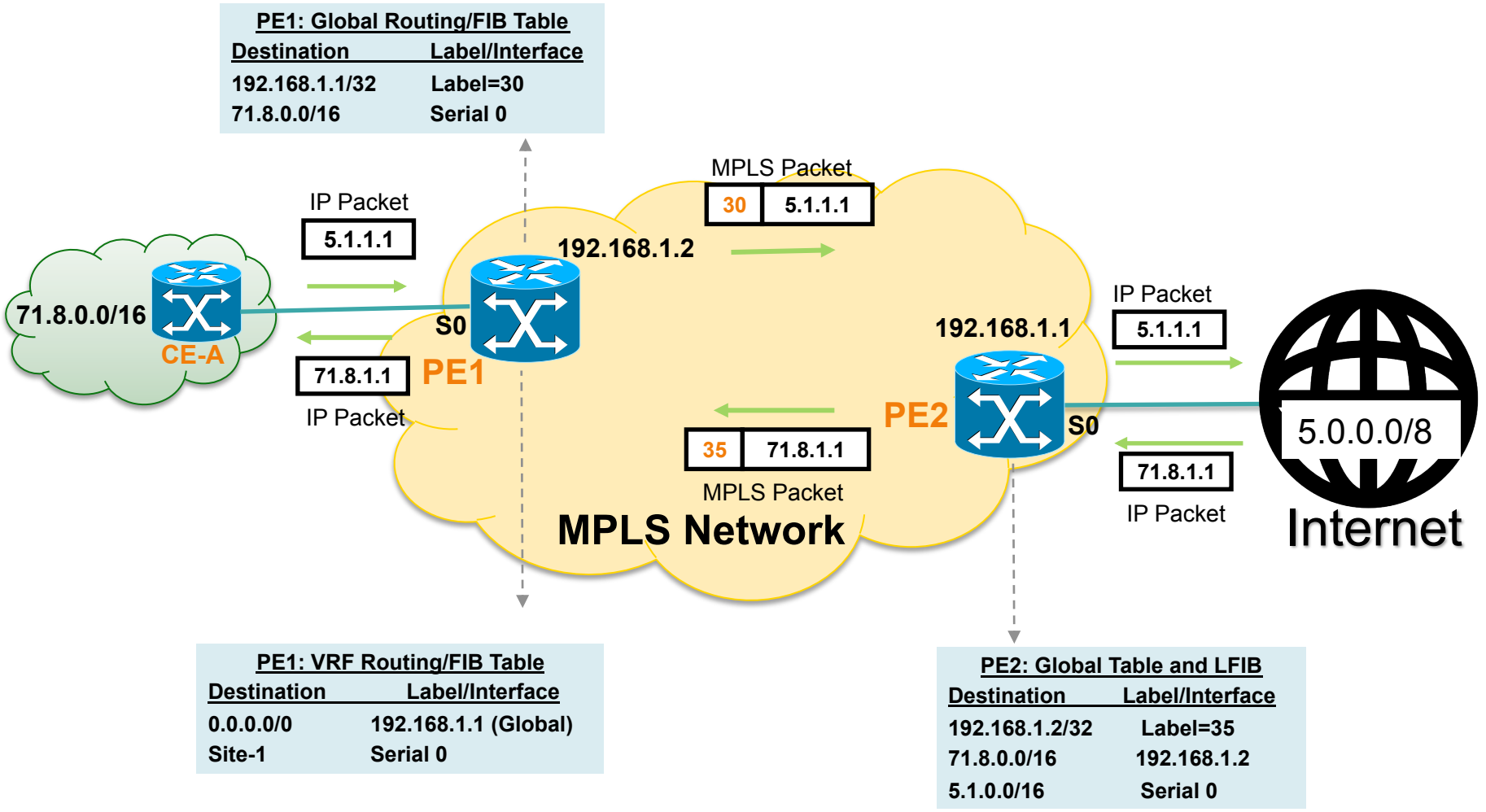
Internet Access: Design Options

1. VRF Specific Default Route
2. Separate PE-CE Sub-interfaces
3. Extranet with Internet-VRF

Option 1: VRF Specific Default Route



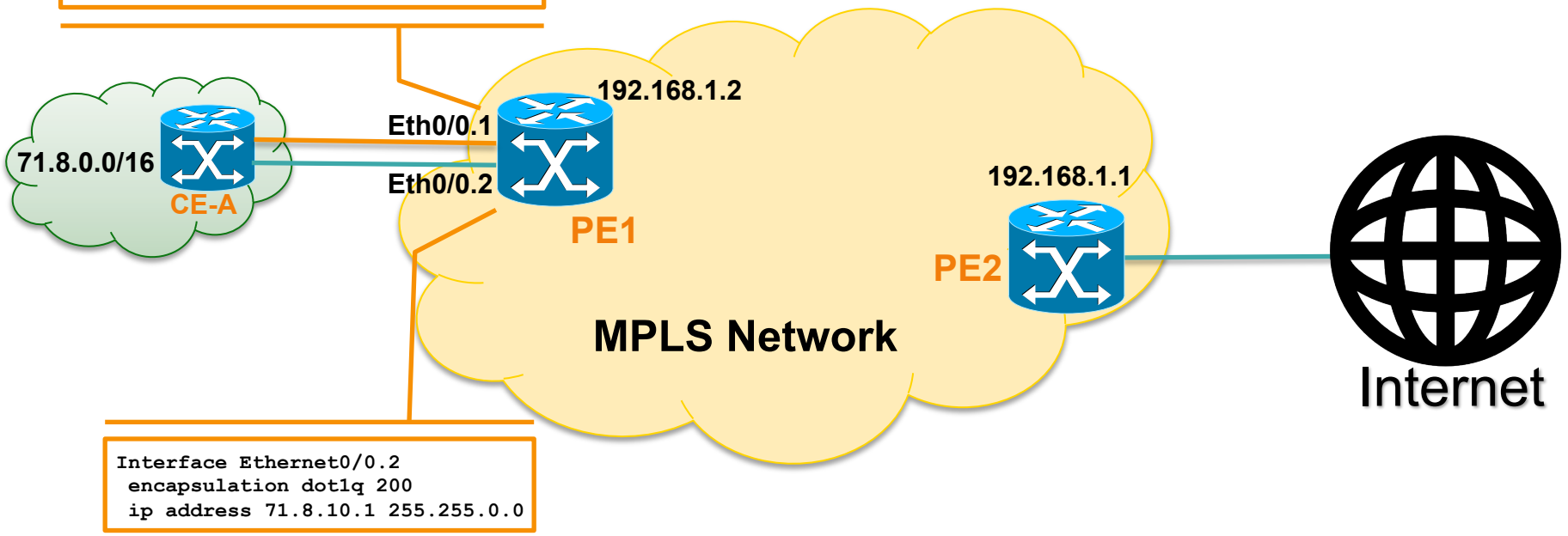
Option 1: Data Plane



Option 2: Separate PE-CE Sub-interfaces

One sub-interface associated to VRF

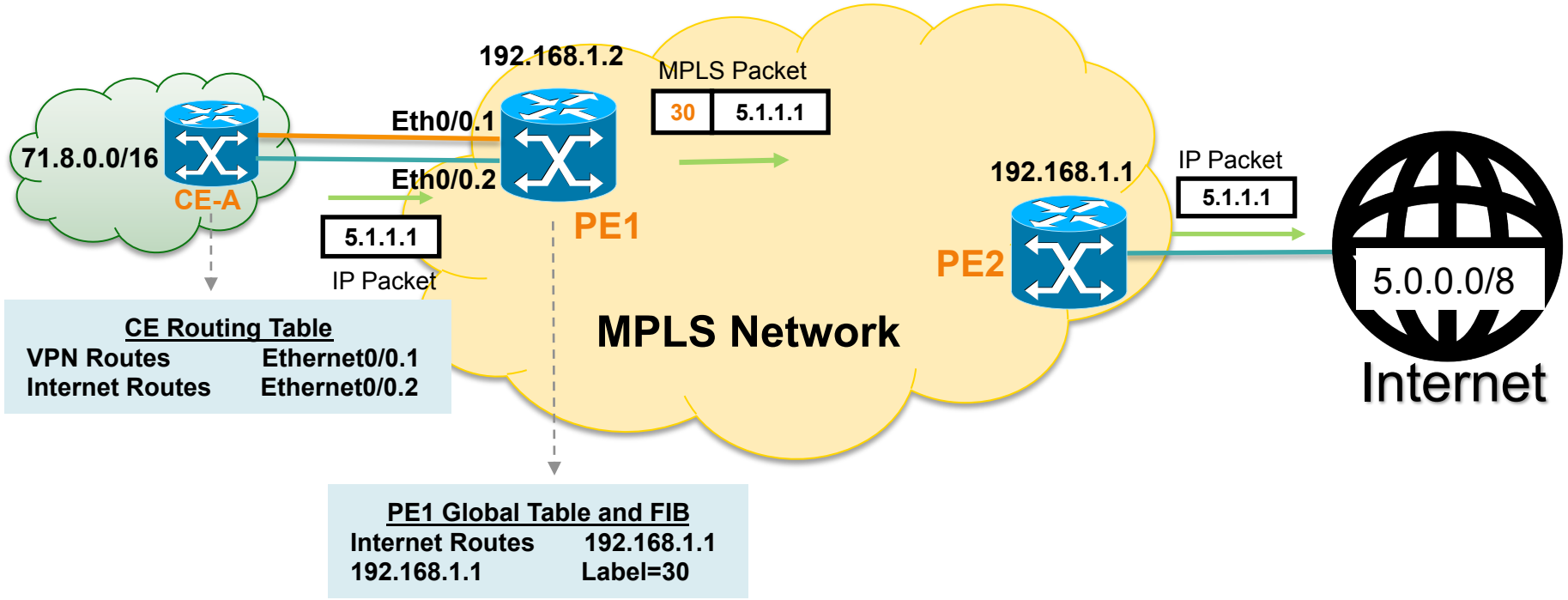
```
Interface Ethernet0/0.1
 encapsulation dot1q 100
 ip vrf forwarding VPN-A
 ip address 192.168.20.1 255.255.255.0
```



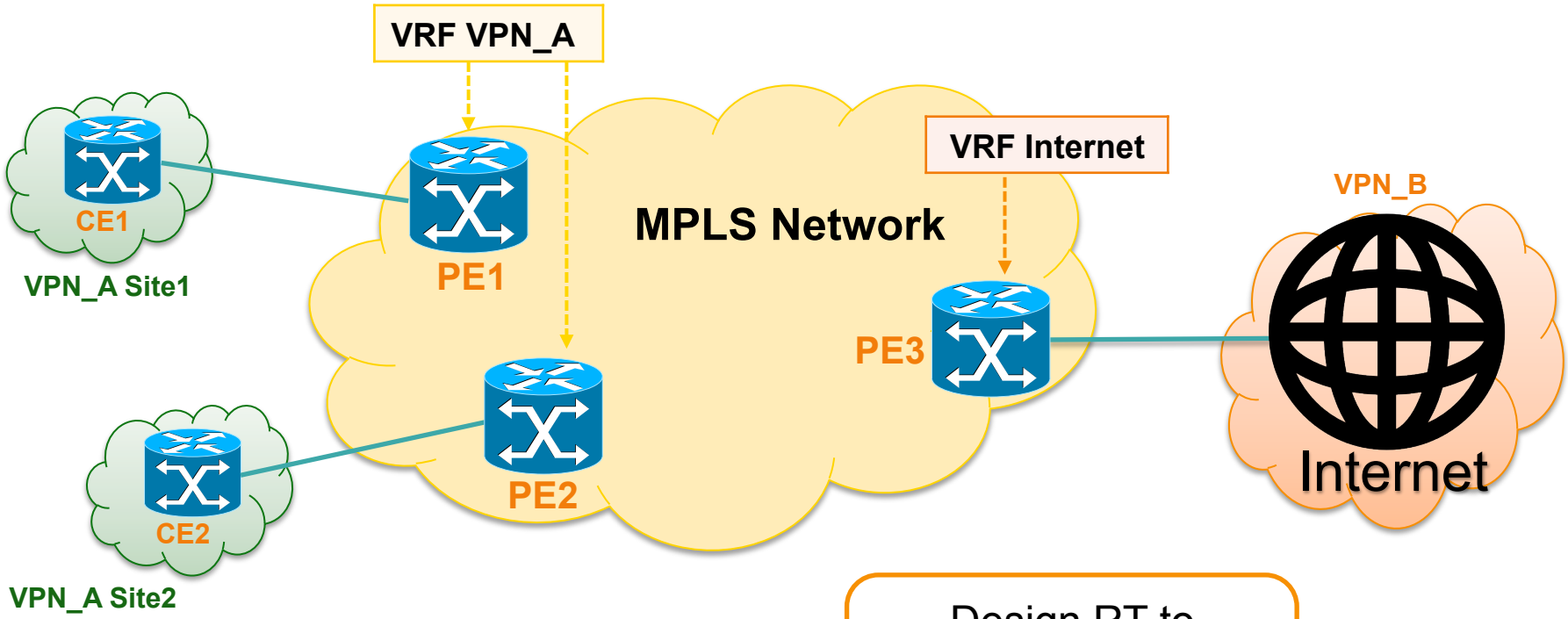

```
Interface Ethernet0/0.2
 encapsulation dot1q 200
 ip address 71.8.10.1 255.255.0.0
```

One sub-interface (global) for Internet routing

Option 2: Data Plane

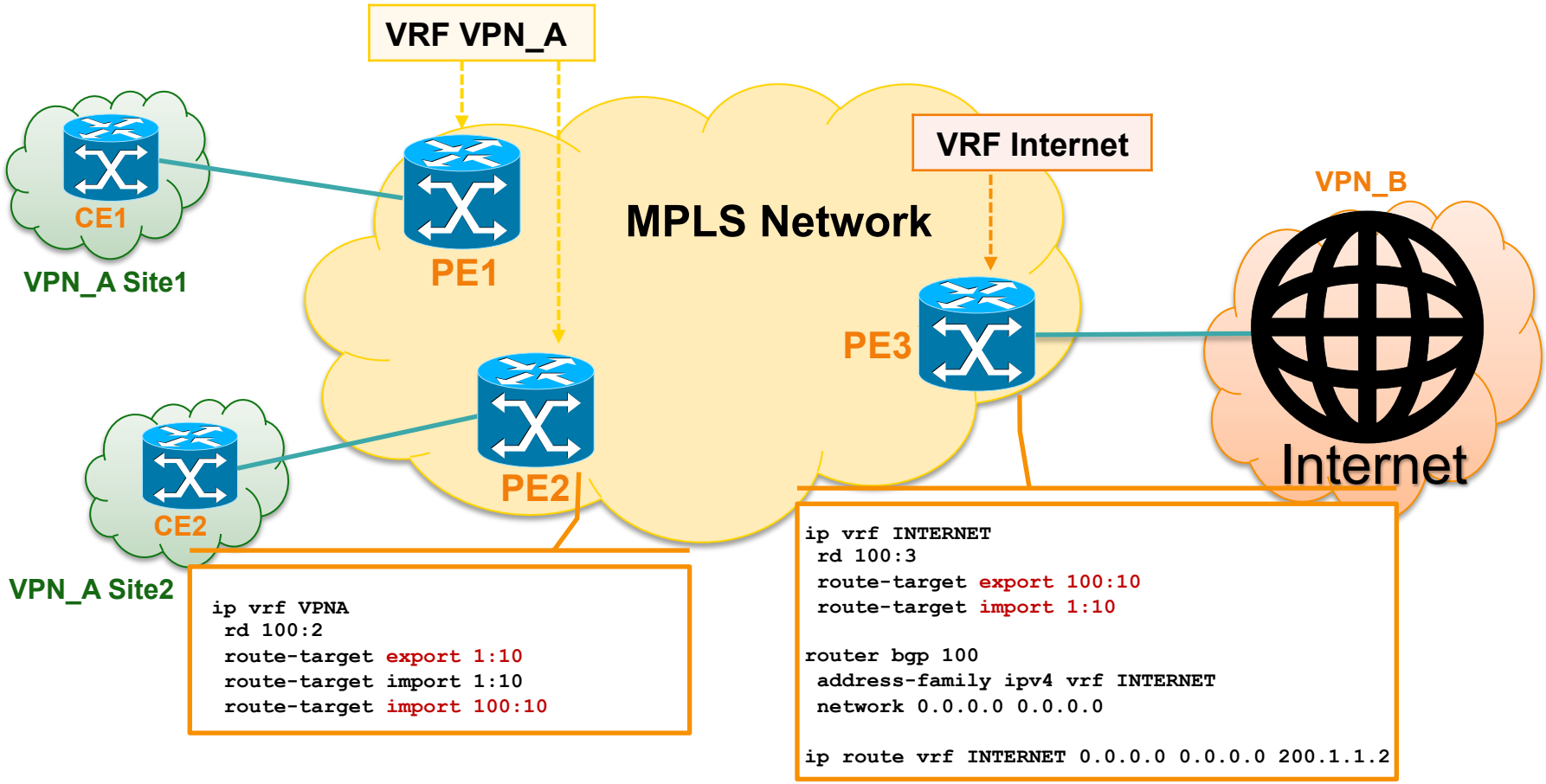


Option 3: Extranet with Internet-VRF



Design RT to implement the VRF communication

Option 3: Extranet with Internet-VRF



Best Practice (1)

1. Use RR to scale BGP; deploy RRs in pair for the redundancy

   Keep RRs out of the forwarding paths and disable CEF (saves memory)

2. Consider unique RD per VRF per PE,

   Helpful for many scenarios such as multi-homing, hub&spoke etc.

3. Utilize SP's public address space for PE-CE IP addressing

   Helps to avoid overlapping; Use /31 subnetting on PE-CE interfaces

Best Practice (2)

4. Limit number of prefixes per-VRF and/or per-neighbor on PE

   Max-prefix within VRF configuration; Suppress the inactive routes

   Max-prefix per neighbor (PE-CE) within OSPF/RIP/BGP VRF af

5. Leverage BGP Prefix Independent Convergence (PIC) for fast convergence <100ms (IPv4 and IPv6):
   - PIC Core
   - PIC Edge
   - Best-external advertisement
   - Next-hop tracking (ON by default)

6. Consider RT-constraint for Route-reflector scalability

7. Consider 'BGP slow peer' for PE or RR – faster BGP convergence

Conclusion

- MPLS based IP/VPN is the most optimal L3VPN technology
  - Any-to-any IPv4 or IPv6 VPN topology
  - Partial-mesh, Hub and Spoke topologies also possible
- Various IP/VPN services for additional value/revenue
- IP/VPN paves the way for virtualization & Cloud Services
  - Benefits whether SP or Enterprise.

Questions?

Issue Date:

Revision:

APNIC

Deploy MPLS VPWS

APNIC

Issue Date: [201609]
Revision: [01]

APNIC

Acknowledgement

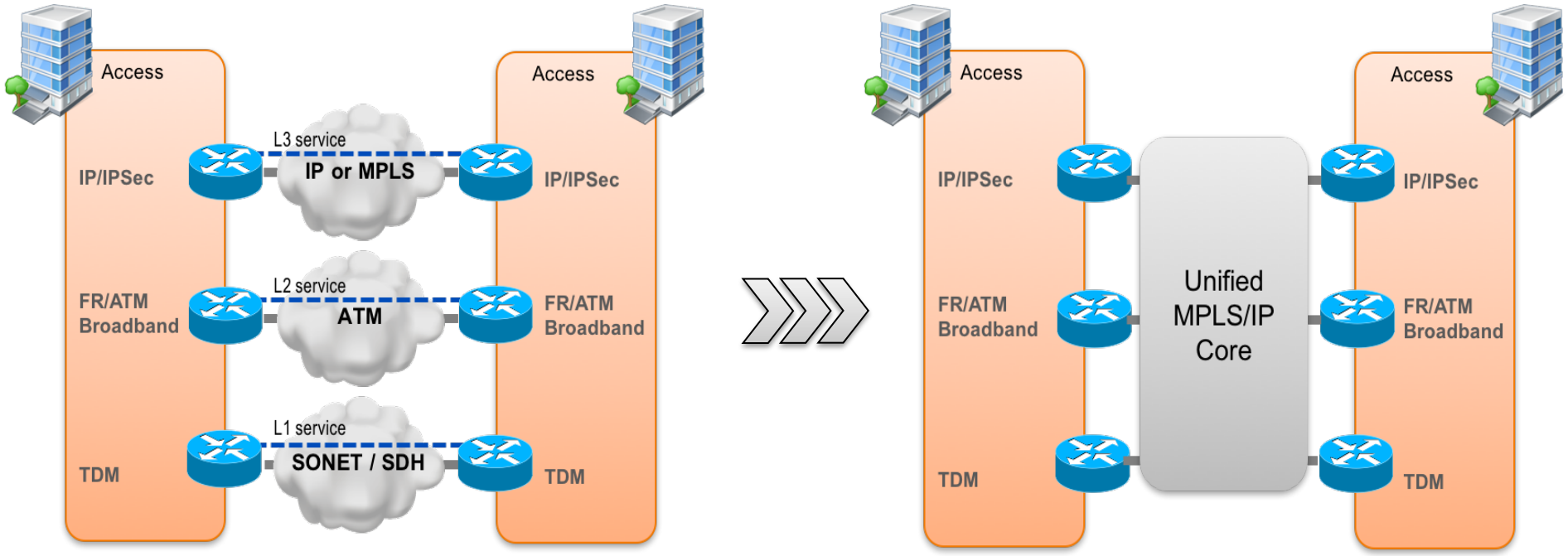
- Cisco Systems

MPLS L2 VPN

APNIC

Motivation for L2VPNs - Consolidation

- Reduced cost—consolidate multiple core technologies into a single packet-based network infrastructure.

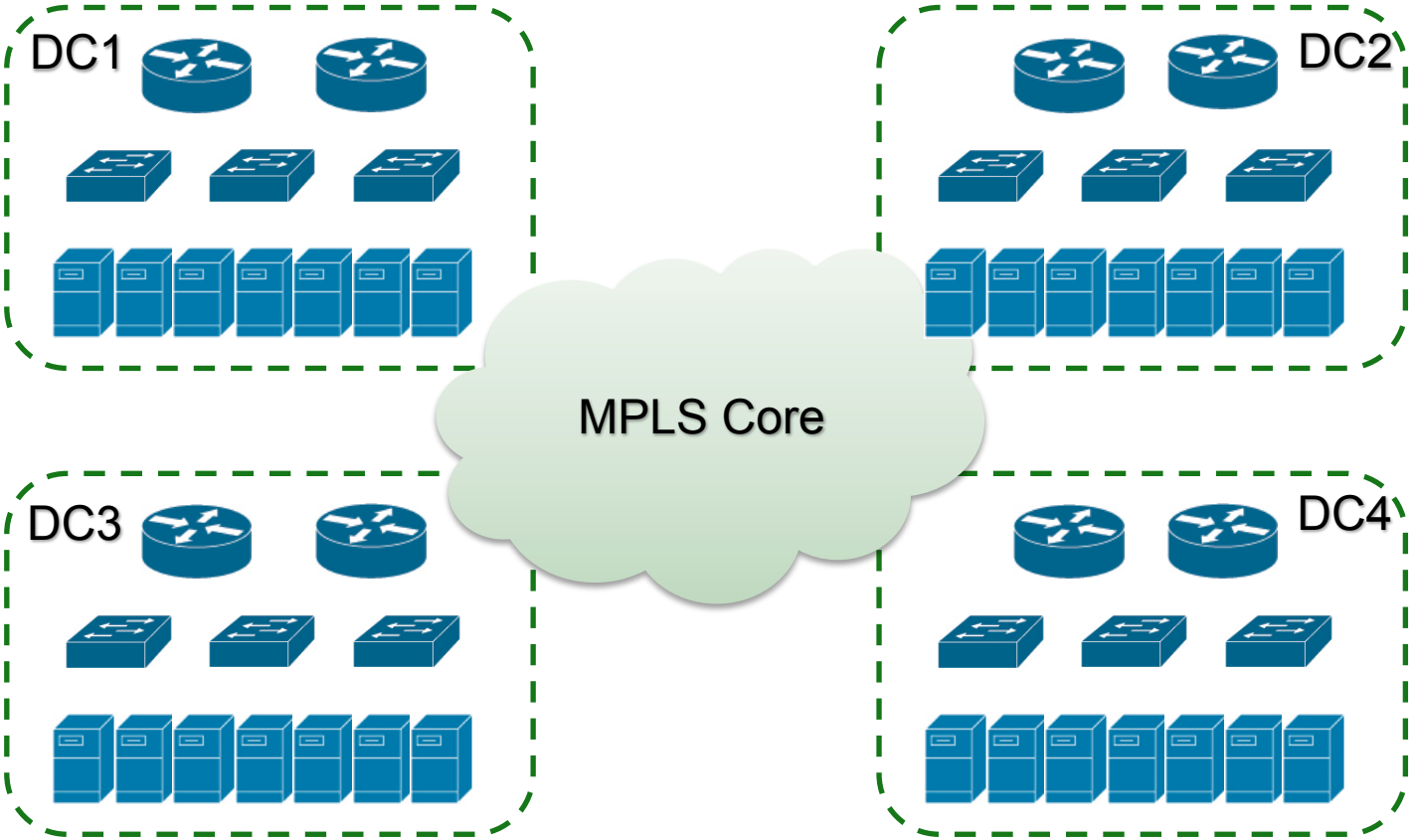


Typical Service Provider

Service Provider with Unified MPLS Core

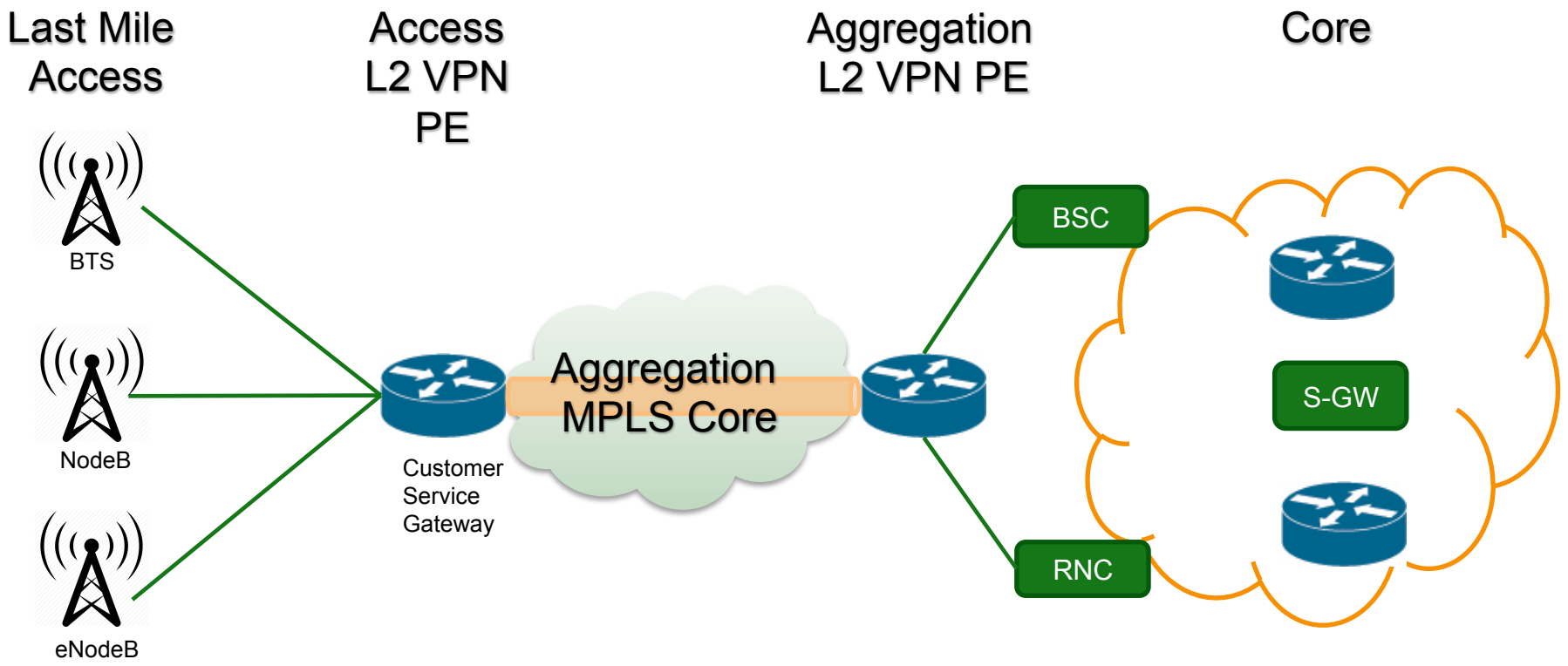
Motivation for L2VPNs - DCI

- Data Center Interconnection - No need to pay for their own WAN infrastructure and flat layer 2 connection.



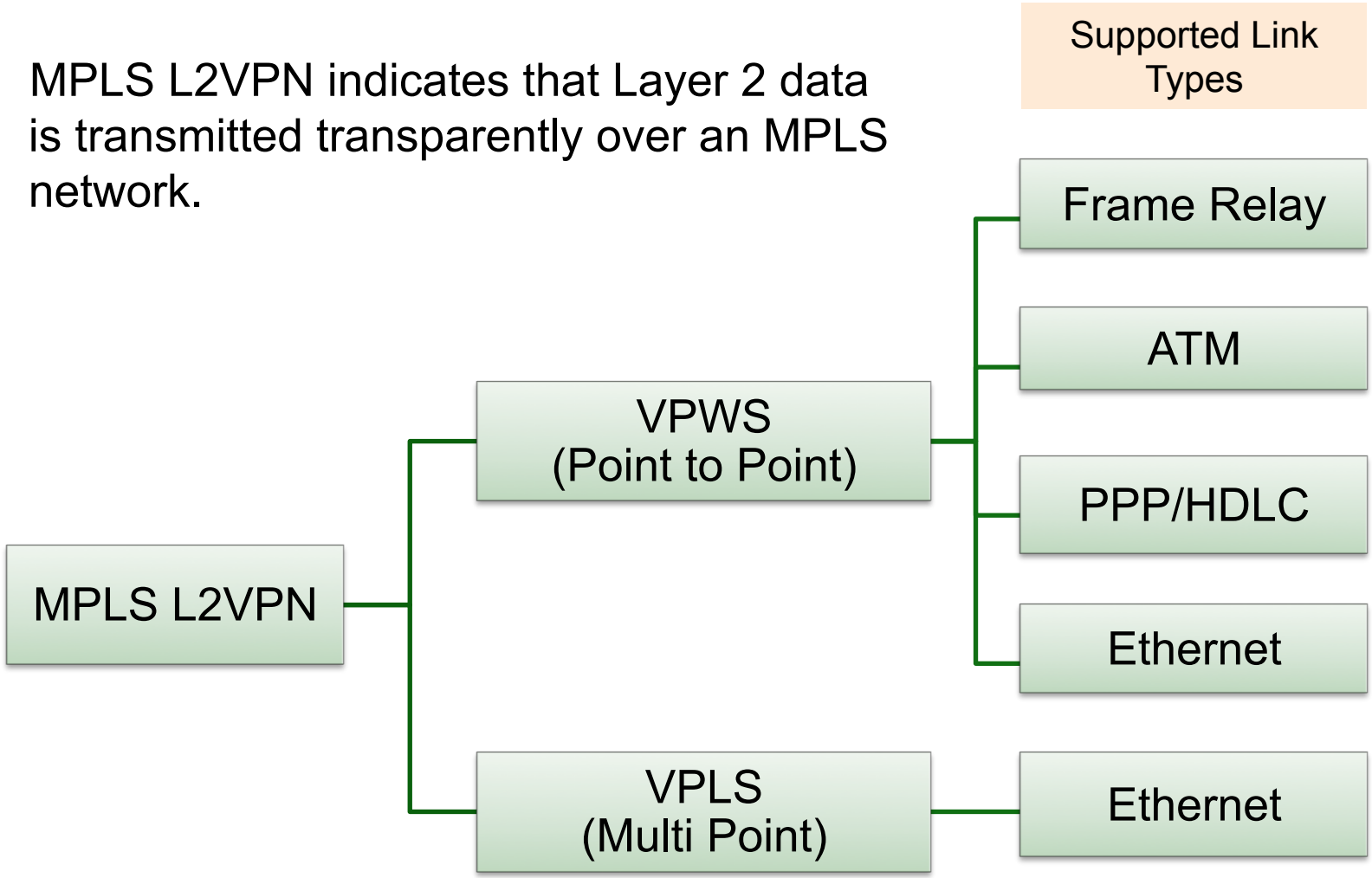
Motivation for L2VPNs - Transport

- Mobile Backhaul Evolution – L2VPN as a transport



MPLS L2VPN Services

- MPLS L2VPN indicates that Layer 2 data is transmitted transparently over an MPLS network.



Advantages of MPLS L2VPN

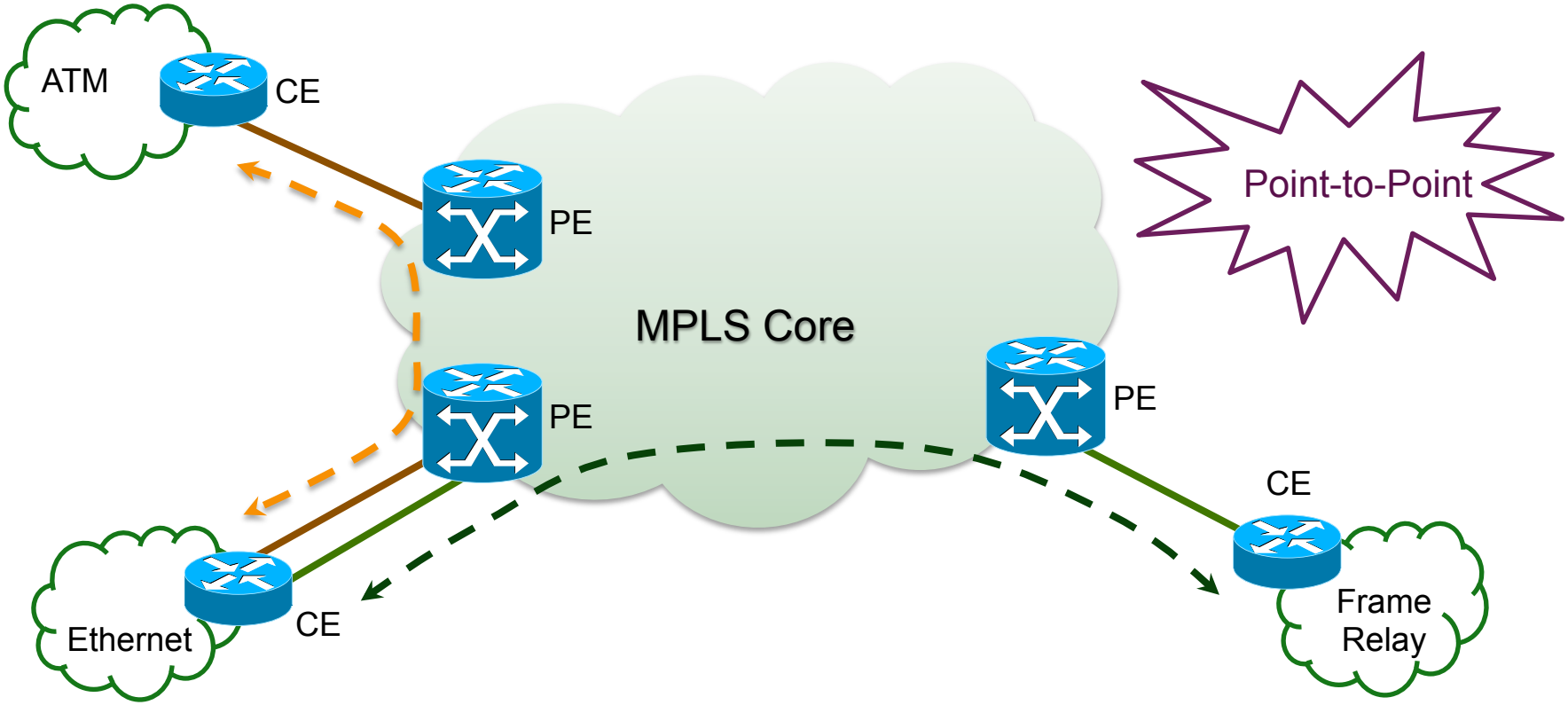
- Extended network functions and service capabilities of operators
- Higher scalability
- Separation of administrative responsibilities
- Privacy of routing and security of user information
- Ease of configuration
- Support for multiple protocols
- Smooth network upgrade

VPWS Overview

APNIC

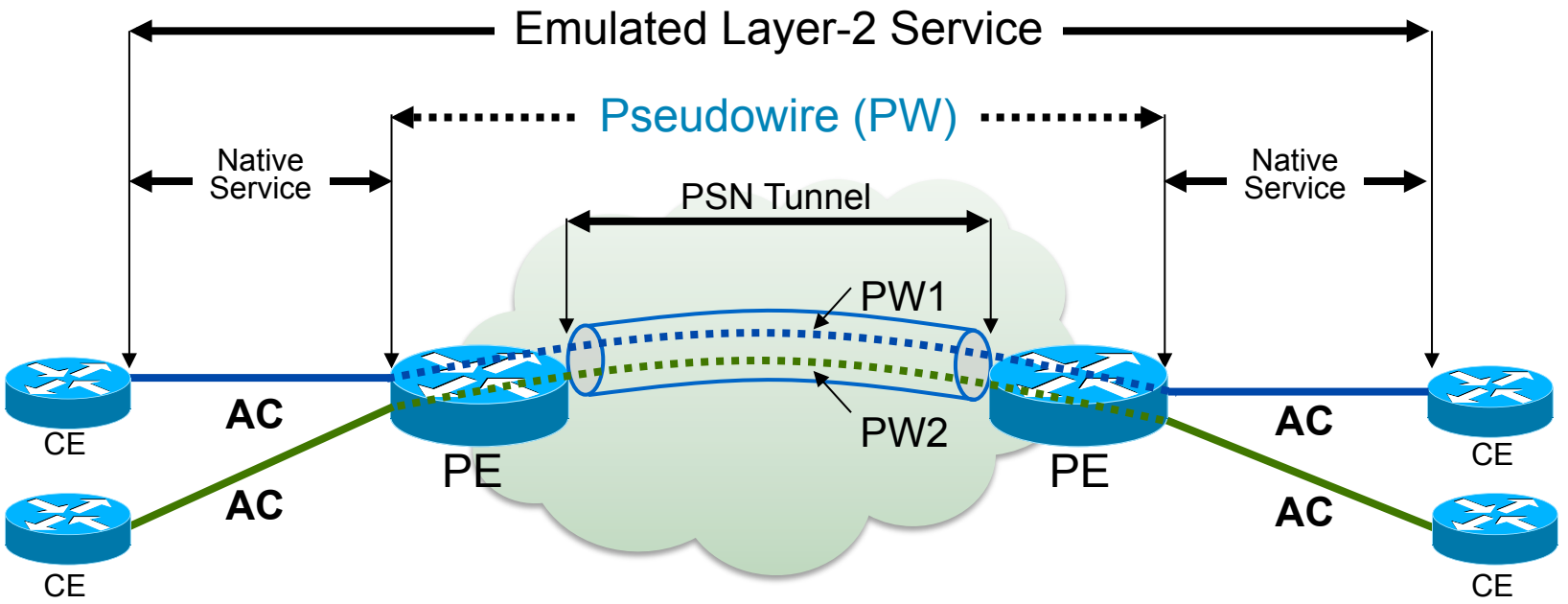
VPWS Reference Model

- VPWS emulates leased lines on an IP network to provide low-cost asymmetrical digital data network service.



Pseudowire

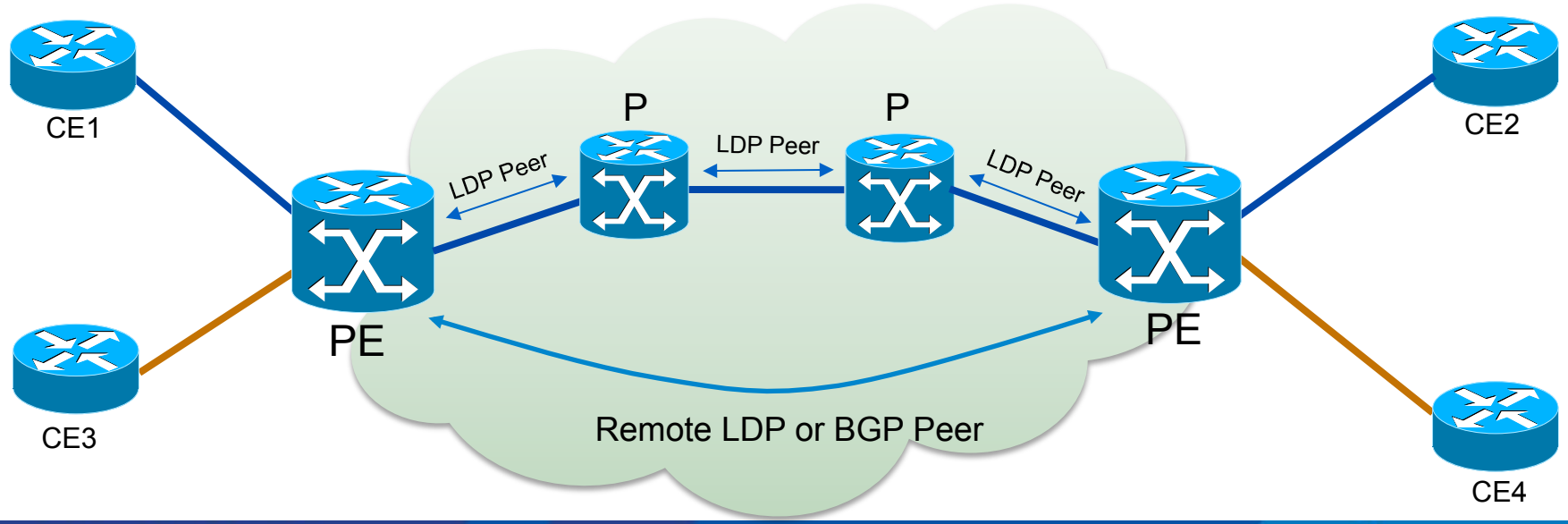
RFC 3985



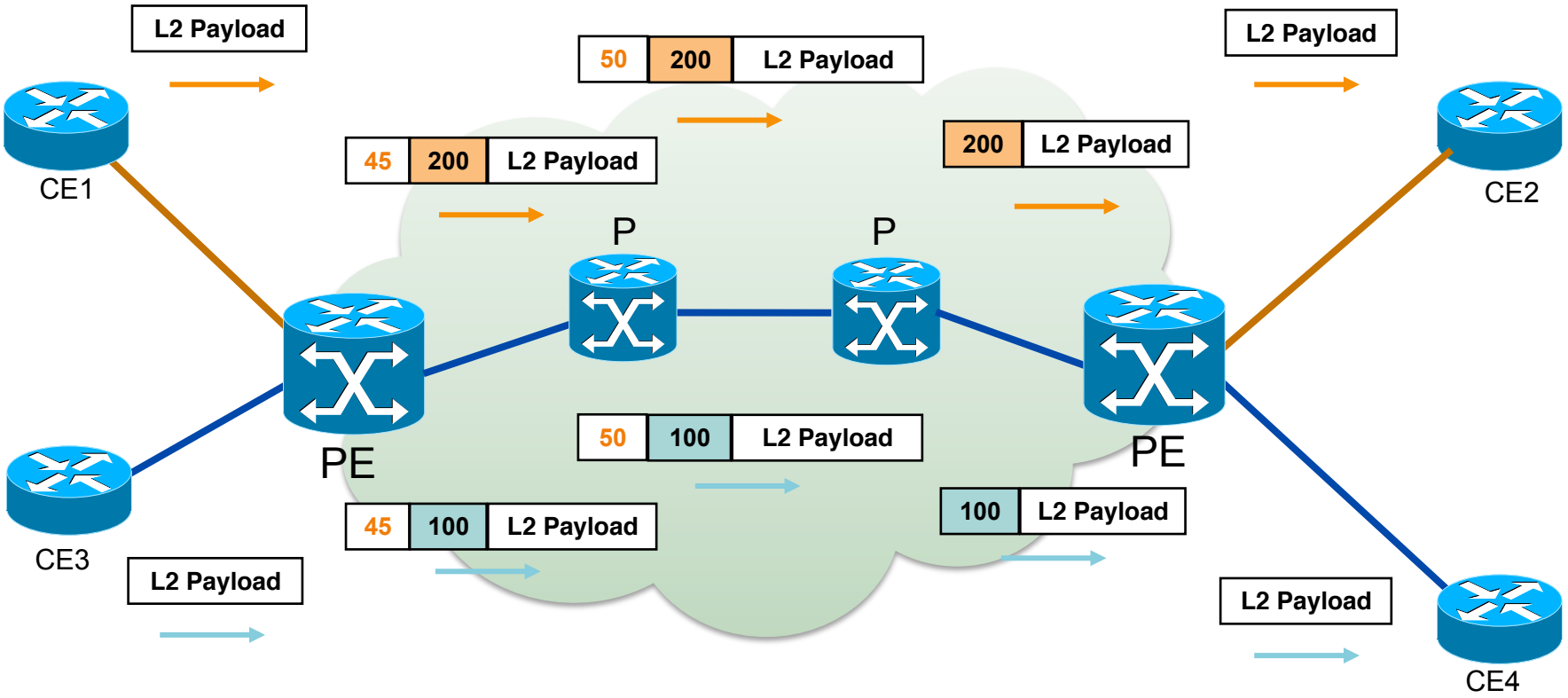
Attachment Circuit(AC)	The physical or virtual circuit attaching a CE to a PE.
Pseudowire(PW)	Pseudowires emulate layer 2 circuits, are used to carry a frame between two PEs

VPWS Control Plane

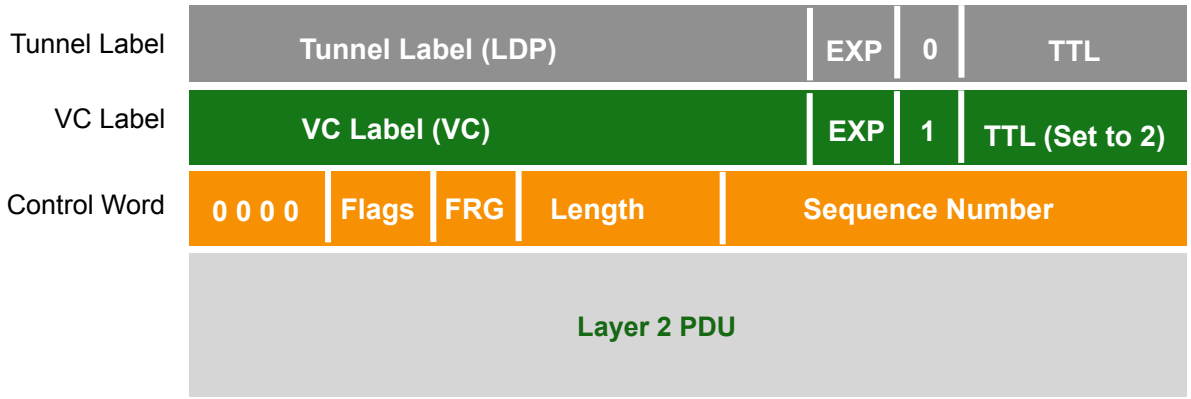
- Tunnel label is distributed by LDP
- VC label is distributed by targeted LDP or BGP
  - LDP Based (also called Martini mode)
  - BGP Based (also called Kompella mode)



Data Plane of VPWS



VPWS Traffic Encapsulation



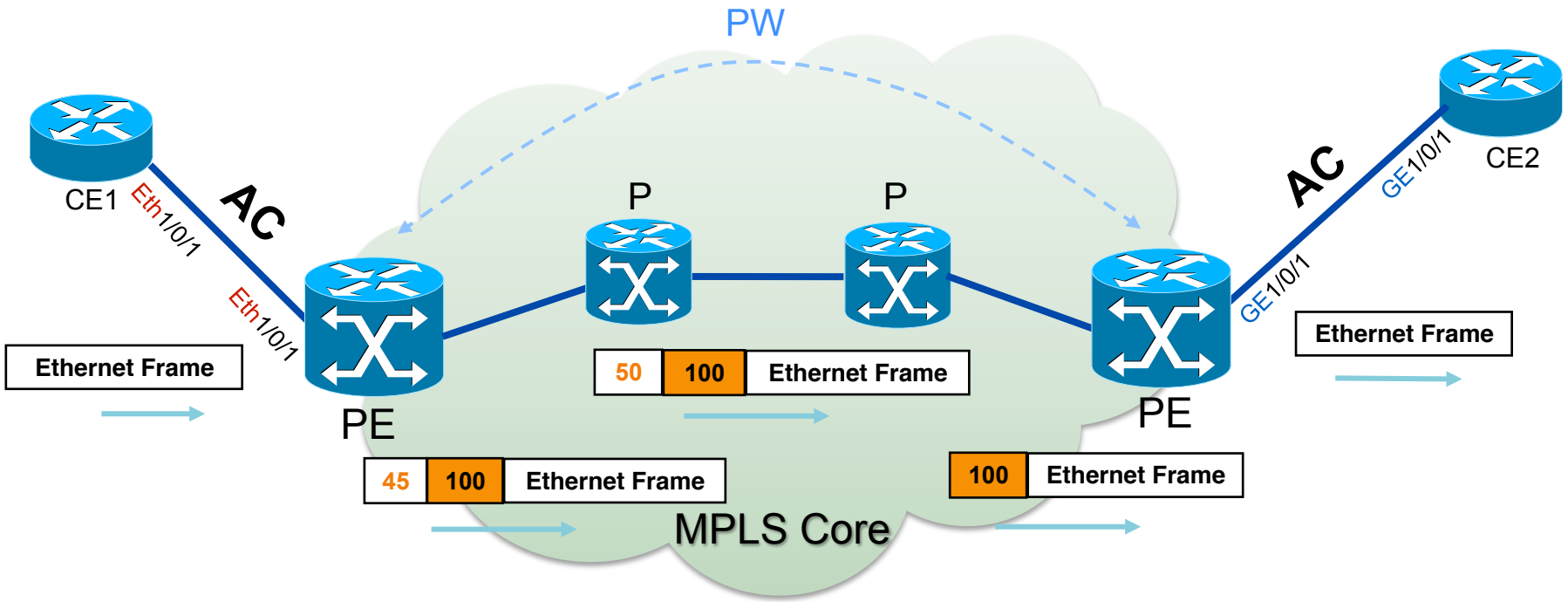
Three-level encapsulation:

1. Packets switched between PEs using Tunnel label
2. VC label identifies PW, VC label signaled between PEs
3. Optional Control Word (CW) carries Layer 2 control bits and enables sequencing

Control Word	
Encap.	**Required**
ATM N:1 Cell Relay	No
ATM AAL5	Yes
Ethernet	No
Frame Relay	Yes
HDLC	No
PPP	No
SAToP	Yes
CESoPSN	Yes

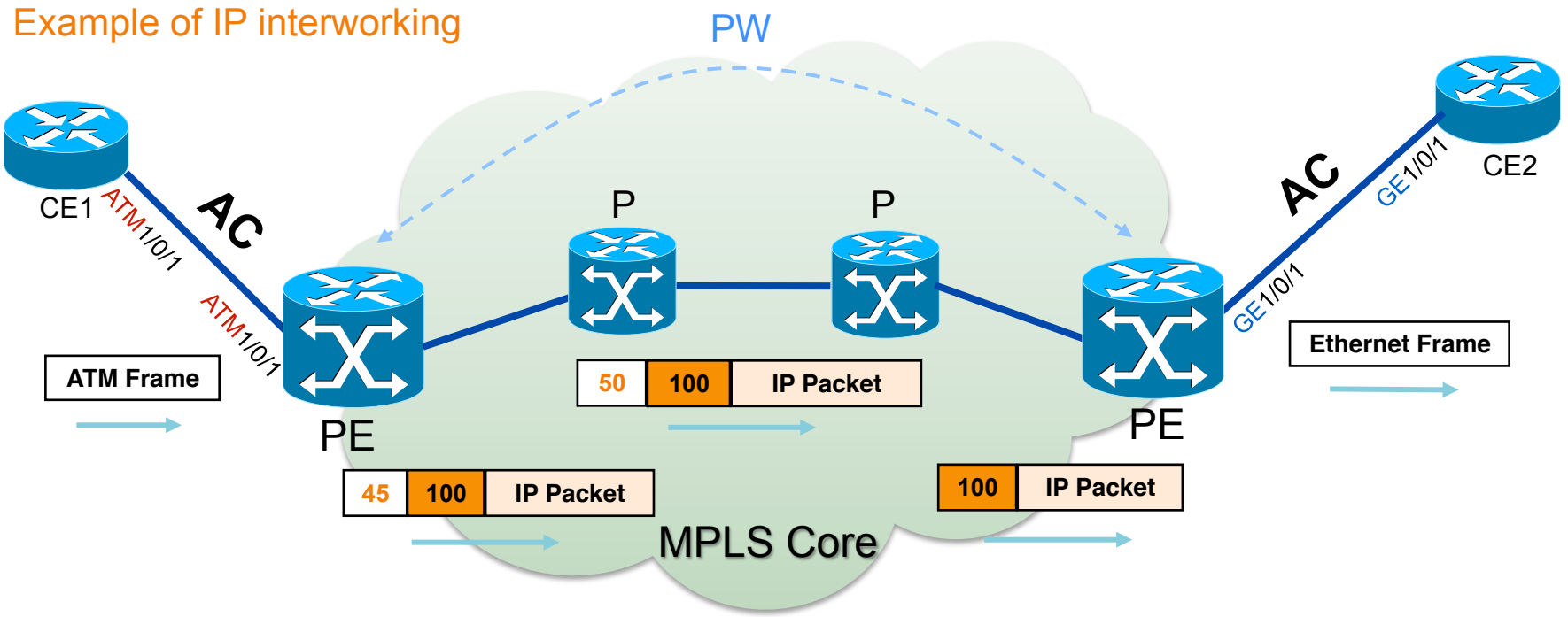
VPWS Service Like-to-Like Transport

- If the link types of CEs on both ends of an L2VPN link are **the same**, for example, both are ethernet, then the whole frames are transferred in the core parts.



VPWS Service Interworking

- If the link types of CEs (such as ATM and Ethernet) on both ends of an L2VPN link are **different**, the L2VPN heterogeneous interworking feature is required.



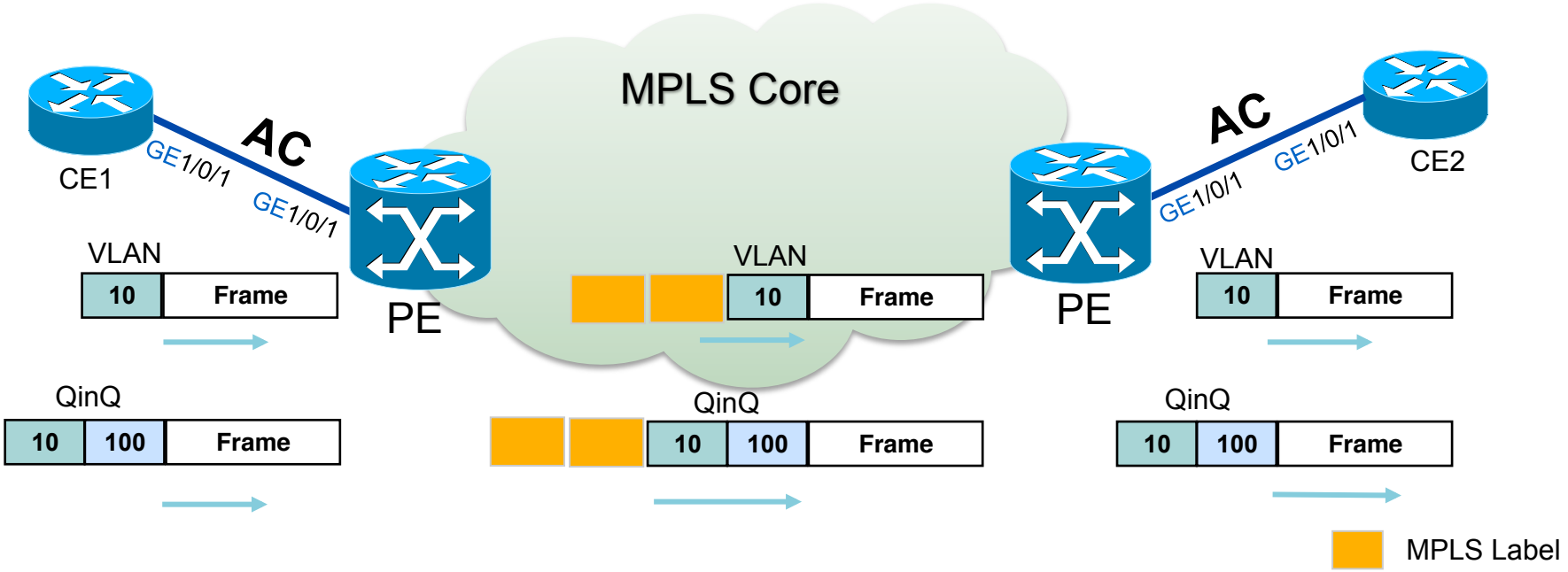
Raw Mode & Tagged Mode

RFC 4448

- Ethernet PW has two modes of operation:
  - Ethernet VLAN / Tagged mode (VC type 0x0004) – Each frame must contain a VLAN tag. The tag value is meaningful to both the ingress and egress PE routers.
  - Ethernet Port / Raw mode (VC type 0x0005) – In raw mode, an Ethernet frame might or might not have a VLAN tag. If the frame does have this tag, the tag is not meaningful to both the ingress and egress PE routers.

VLAN Tag Multiplexing

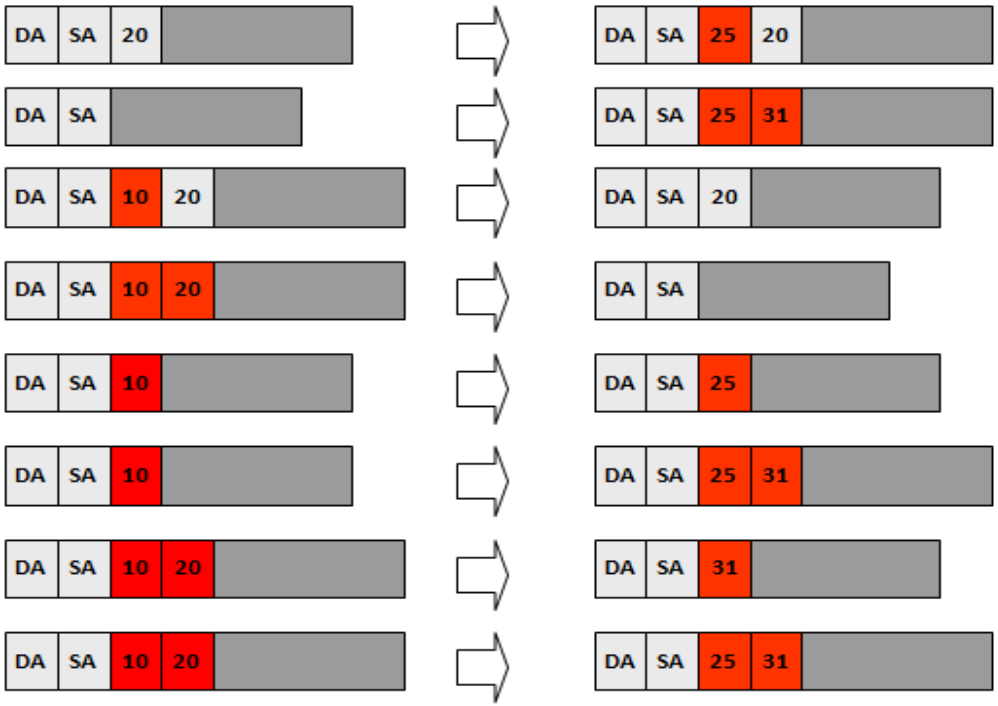
- VLAN tags in the frame can be kept across the whole MPLS domain.



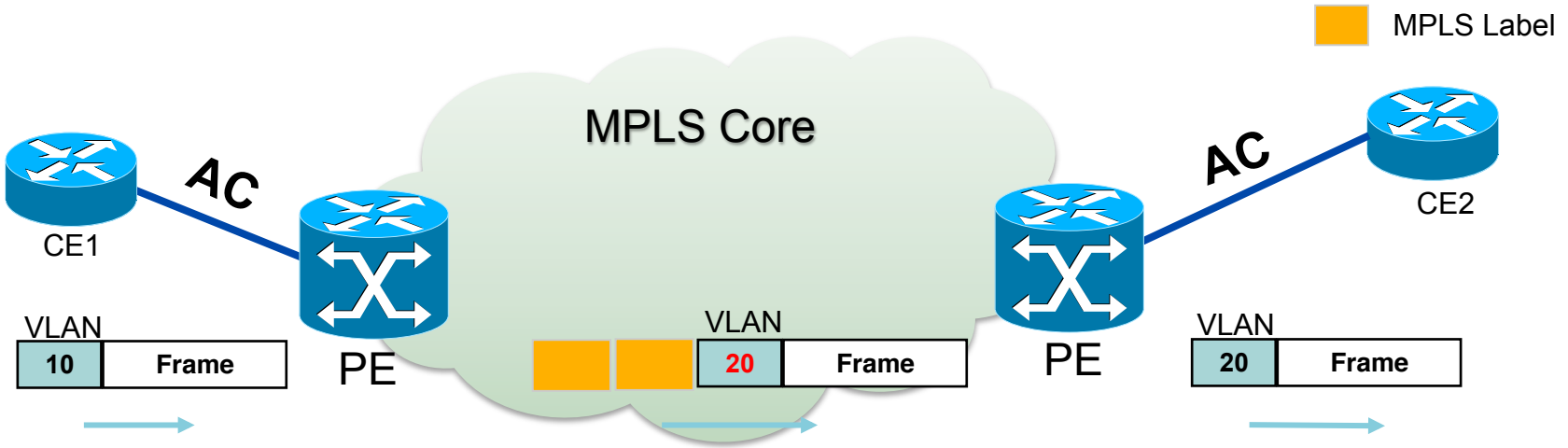
- One VLAN or multiple VLANs can be mapped into one PW.

VLAN Tag Translation and Manipulation

- VLAN tags can be added, removed or translated prior to VC label imposition or after disposition
  - Any VLAN tag(s), if retained, will appear as payload to the VC



VLAN Tag Translation Example



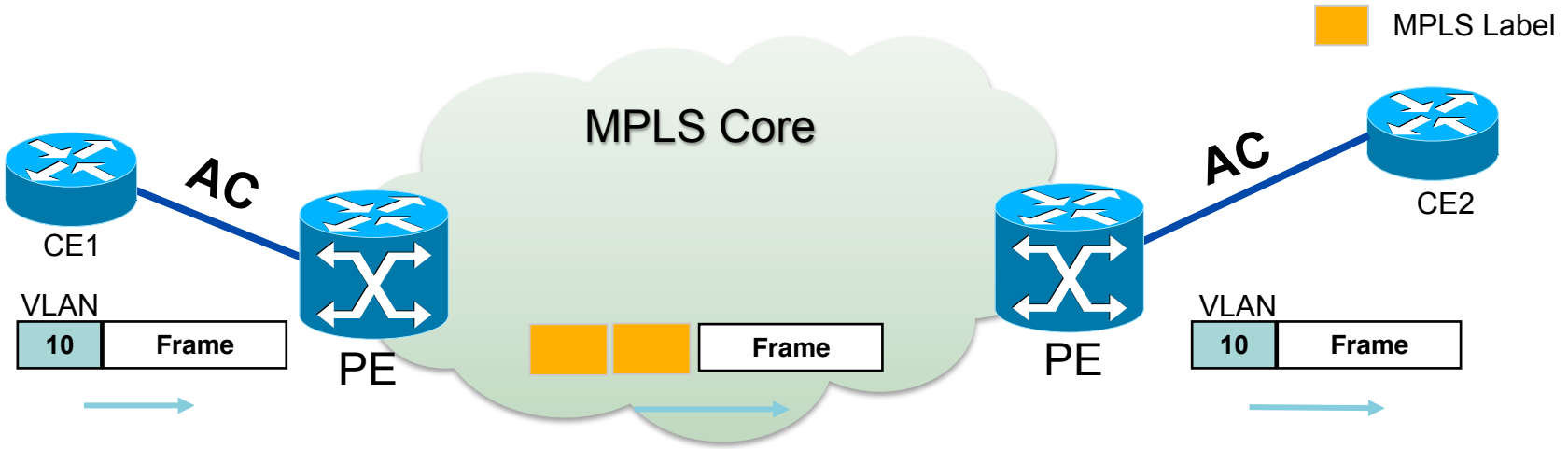
Junos

```
interfaces {
    ge-1/0/3 {
        unit 10 {
            encapsulation vlan-ccc;
            vlan-id 10;
            input-vlan-map {
                swap;
                vlan-id 20;
            }
            output-vlan-map swap;
}}}
```

IOS XR

```
l2vpn
 pw-class class-VC5
  encapsulation mpls
   transport-mode VLAN

interface GigabitEthernet 0/0/0/3.10 l2transport
 encapsulation dot1q 10
 rewrite ingress tag translate 1-to-1 dot1q 20 symmetric
```

VLAN Tag Manipulation Example



Junos

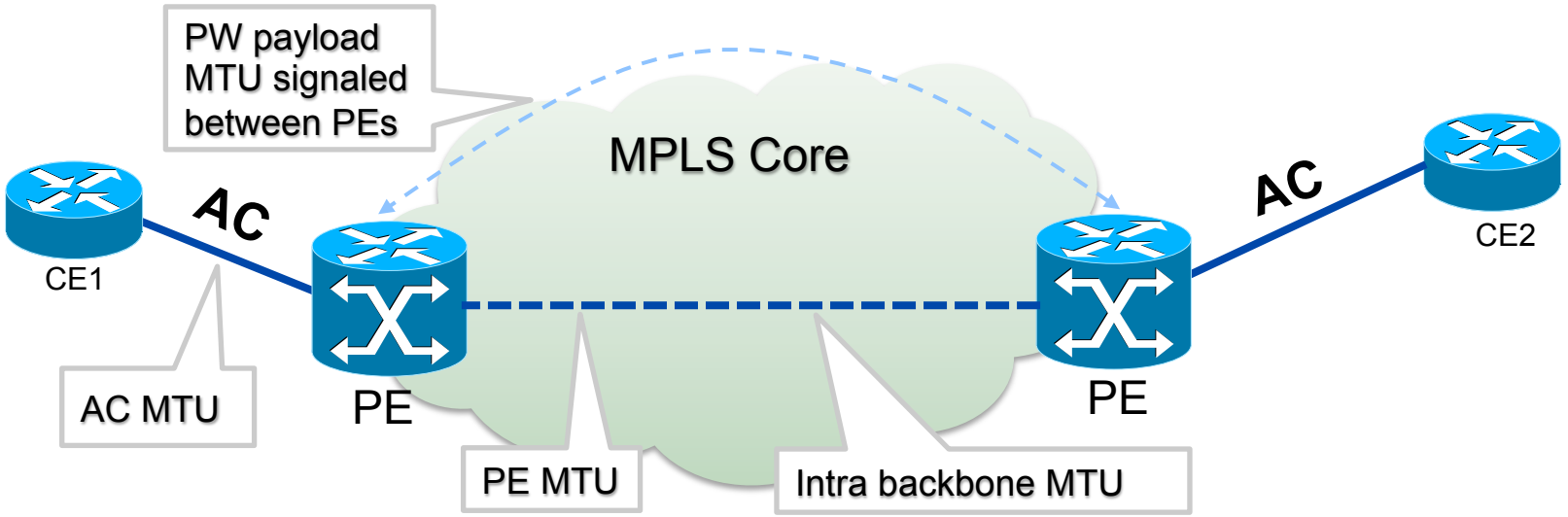
```
interfaces {
         ge-1/0/3 {
             unit 10 {
                 encapsulation vlan-ccc;
                 vlan-id 10;
                 input-vlan-map pop;
                 output-vlan-map push;
    }}}
    routing-instances {
        L2VPN-A {
            protocols {
                l2vpn {
                    encapsulation-type ethernet;
    }}}}
```

IOS XR

```
l2vpn
 pw-class class-VC5
  encapsulation mpls
   transport-mode ethernet

interface GigabitEthernet 0/0/0/3.10 l2transport
 encapsulation dot1q 10
 rewrite ingress tag pop 1 symmetric
```

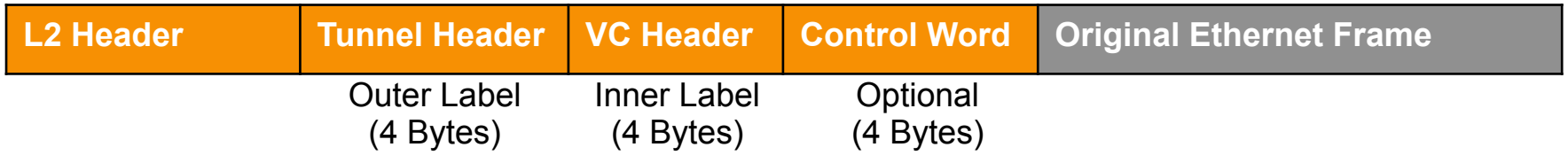
MTU Considerations

- No payload fragmentation. Incoming PDU dropped if MTU exceeds AC MTU
- PEs exchange PW payload MTU as part of PW signaling procedures
  - Both ends must agree to use same value for PW to come UP
  - PW MTU derived from AC MTU
- No mechanism to check Backbone MTU
  - MTU in the backbone must be large enough to carry PW payload and MPLS stack



MTU Calculation for VPWS

Frame encapsulation format



Field	Edge	Transport	Control Word	MPLS	Total
EoMPLS Port Mode	1500	14	4 or 0	8	1526 or 1522
EoMPLS VLAN Mode	1500	18	4 or 0	8	1530 or 1526

How to Modify MTU

- Cisco IOS
- Interface MTU configured as largest ethernet payload size
  - 1500B default
  - Sub-interfaces / Service Instances (EFPs) MTU always inherited from main interface

```
interface GigabitEthernet0/0/4
 description Main interface
 mtu 1600
```

```
R1#show int gigabitEthernet 0/0/4.1000 | include MTU
  MTU 1600 bytes, BW 100000 Kbit/sec, DLY 100 usec,
```

Sub-interface MTU inherited from Main interface

- PW MTU used during PW signaling
  - By default, inherited from attachment circuit MTU
  - Submode configuration CLI allows MTU values to be set per subinterface/EFP in xconnect configuration mode (only for signaling purposes)
  - No MTU adjustments made for EFP rewrite (POP/ PUSH) operations

```
interface GigabitEthernet0/0/4.1000
 encapsulation dot1Q 1000
 xconnect 106.106.106.106 111 encapsulation mpls
  mtu 1500
```

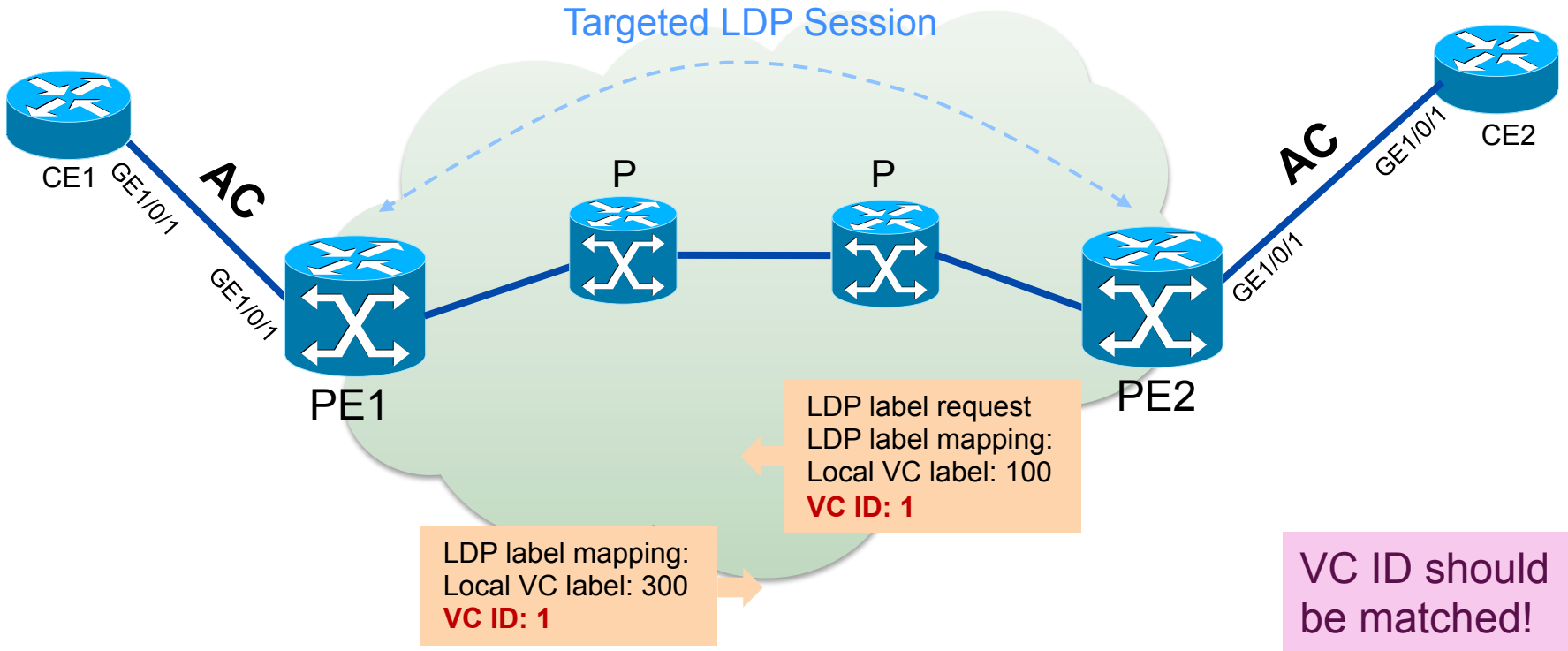
PW MTU used during signaling can be overwritten

VPWS Signaled with LDP

APNIC

VC Signaled with LDP

- Targeted LDP Session has been established between PEs.
- A VC FEC (type 128) has been added to a Label Mapping message to carry VC information during PW establishment.



Configuration Comparison

- Cisco IOS:

```
PE1(config)#pseudowire-class CE1_CE2
PE1(config-pw-class)#encapsulation mpls
PE1(config-pw-class)#interworking ethernet
PE1(config-pw-class)#exit
PE1(config)#interface fastEthernet 0/0
PE1(config-if)#xconnect 10.0.0.4 1315 encapsulation mpls pw-
class CE1_CE2
PE1(config-if)#exit
```

- Huawei VRP:

```
[PE1]mpls ldp remote-peer 10.0.0.4
[PE1-mpls-ldp-remote-10.0.0.4]remote-ip 10.0.0.4
[PE1-mpls-ldp-remote-10.0.0.4]quit
[PE1]interface FastEthernet0/0
[PE1-FastEthernet0/0]mpls l2vc 10.0.0.4 1315
[PE1-FastEthernet0/0]quit
```

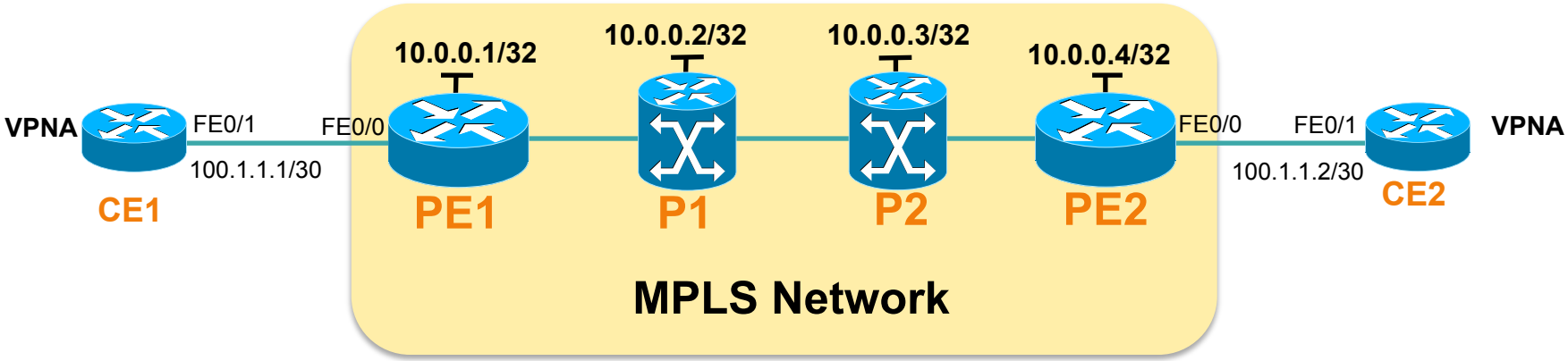
Configuration Comparison

- Juniper Junos

```
interfaces {
          ge-2/0/1 {
             encapsulation ethernet-ccc;
             unit 0;
          }
      }
protocols {
         ldp {
              interface lo0.0;
          }
          l2circuit {
              neighbor 172.16.0.44 {
                  interface ge-2/0/1.0 {
                      virtual-circuit-id 13579;
                      encapsulation-type ethernet;
                      pseudowire-status-tlv;
 }}}}
```

Configuration Example of VPWS Signaled with LDP

- Task: Configure MPLS L2VPN (LDP based)on Cisco IOS (Version 15.2) to make the following CEs communication with each other.
- Prerequisite configuration:
  - 1. IP address configuration on all the routers
  - 2. IGP configuration on PE & P routers
  - 3. LDP configuration on PE & P routers



Configure Pseudowire Class

- Configuration steps:
  - 1. Configure pseudowire class on PE routers

```
PE1(config)#pseudowire-class CE1_CE2
PE1(config-pw-class)#encapsulation mpls
PE1(config-pw-class)#interworking ethernet
PE1(config-pw-class)#exit
```

Specify the tunneling encapsulation

```
PE2(config)#pseudowire-class CE1_CE2
PE2(config-pw-class)#encapsulation mpls
PE2(config-pw-class)#interworking ethernet
PE2(config-pw-class)#exit
```

Bind AC to Pseudowire

- Configuration steps:
  - 2. Bind the attachment circuit to a pseudowire VC

```
PE1(config)#interface fastEthernet 0/0

PE1(config-if)#xconnect 10.0.0.4 1315 encapsulation mpls pw-
class CE1_CE2
```

Under the interface which is connecting to CE

Binds the attachment circuit to a pseudowire VC

```
PE2(config)#interface fastEthernet 0/0
PE2(config-if)#xconnect 10.0.0.1 1315 encapsulation mpls pw-
class CE1_CE2
```

Verification of Targeted LDP

- Verify results:
  - Check targeted LDP session on PE router:

```
PE1#show mpls ldp neighbor
    Peer LDP Ident: 10.0.0.2:0; Local LDP Ident 10.0.0.1:0
          TCP connection: 10.0.0.2.48548 - 10.0.0.1.646
          State: Oper; Msgs sent/rcvd: 41425/41430; Downstream
          Up time: 3w4d
          LDP discovery sources:
            Ethernet1/0, Src IP addr: 10.12.0.2
        Addresses bound to peer LDP Ident:
          10.23.1.1       10.23.2.1       10.12.0.2       10.0.0.2
    Peer LDP Ident: 10.0.0.4:0; Local LDP Ident 10.0.0.1:0
          TCP connection: 10.0.0.4.56428 - 10.0.0.1.646
          State: Oper; Msgs sent/rcvd: 175/176; Downstream
          Up time: 02:23:20
          LDP discovery sources:
            Targeted Hello 10.0.0.1 -> 10.0.0.4, active, passive
        Addresses bound to peer LDP Ident:
          10.34.0.2       10.0.0.4        10.1.1.1
```

Verification of VC

- Verify results:
  - Check VC on PE routers

```
R1#show mpls l2transport binding

  Destination Address: 10.0.0.4,VC ID: 1315
    Local Label:  105          Cbit: 1,    VC Type: Ethernet,    GroupID: 0
        MTU: 1500,   Interface Desc: n/a
        VCCV: CC Type: CW [1], RA [2], TTL [3]
              CV Type: LSPV [2], BFD/Raw [5]
    Remote Label: 405          Cbit: 1,    VC Type: Ethernet,    GroupID: 0
        MTU: 1500,   Interface Desc: n/a
        VCCV: CC Type: CW [1], RA [2], TTL [3]
              CV Type: LSPV [2], BFD/Raw [5]
```

```
R1#show mpls l2transport vc

Local intf     Local circuit              Dest address    VC ID      Status
-------------  -------------------------- --------------- ---------- ----------
Fa0/0          Ethernet                   10.0.0.4        1315       UP
```

Verification of CE Reachability

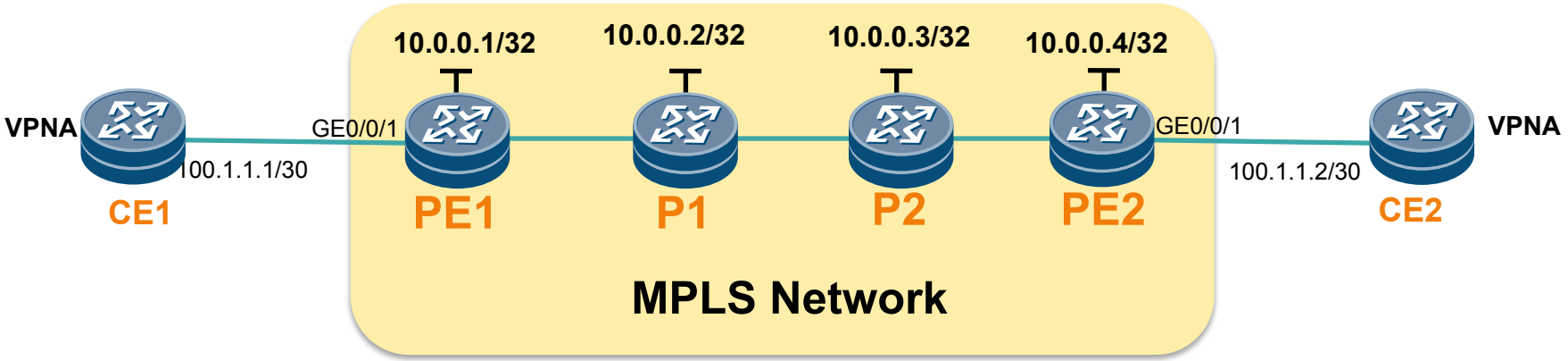
- Verify results:
  - Check the reachability between CEs.

```
CE1# ping 100.1.1.2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 100.1.1.2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 16/24/32 ms
```

```
CE1# traceroute 100.1.1.2
Type escape sequence to abort.
Tracing the route to 100.1.1.2
VRF info: (vrf in name/id, vrf out name/id)
  1 100.1.1.2 16 msec 32 msec *
```

Configuration Example of VPWS Signaled with LDP

- Task: Configure MPLS L2VPN (LDP based)on **HUAWEI VRP5** to make the following CEs communication with each other.
- Prerequisite configuration:
  - 1. IP address configuration on all the routers
  - 2. IGP configuration on PE & P routers
  - 3. LDP configuration on PE & P routers



Configure Remote LDP Session

- Configuration steps:
  - Set up a remote LDP session between PE

```
[PE1]mpls ldp remote-peer PE2
[PE1-mpls-ldp-remote-pe2]remote-ip 10.0.0.4
[PE1-mpls-ldp-remote-pe2]quit
```

```
[PE2]mpls ldp remote-peer PE1
[PE2-mpls-ldp-remote-pe1]remote-ip 10.0.0.1
[PE2-mpls-ldp-remote-pe1]quit
```

Configure VC

- Configuration steps:
  - Enable MPLS L2VPN and create VCs on the PEs.

```
[PE1]mpls l2vpn
[PE1-l2vpn]quit
[PE1]interface GigabitEthernet 0/0/1
[PE1-GigabitEthernet0/0/1]mpls l2vc 10.0.0.4 1315
[PE1-GigabitEthernet0/0/1]quit
```

Binds the attachment circuit to a pseudowire VC
The same VC ID: 1315

```
[PE2]mpls l2vpn
[PE2-l2vpn]quit
[PE2]interface GigabitEthernet 0/0/1
[PE2-GigabitEthernet0/0/1]mpls l2vc 10.0.0.1 1315
[PE2-GigabitEthernet0/0/1]quit
```

Verification of LDP Peers

- Verify results: Check MPLS LDP Peer

```
[PE1]display mpls ldp peer

 LDP Peer Information in Public network
 A '*' before a peer means the peer is being deleted.
 ------------------------------------------------------------------------------
 PeerID                 TransportAddress   DiscoverySource
 ------------------------------------------------------------------------------
 10.0.0.2:0             10.0.0.2           GigabitEthernet0/0/0
 10.0.0.4:0             10.0.0.4           Remote Peer : pe2
 ------------------------------------------------------------------------------
 TOTAL: 2 Peer(s) Found.
```

```
<PE2>display mpls ldp peer

 LDP Peer Information in Public network
 A '*' before a peer means the peer is being deleted.
 ------------------------------------------------------------------------------
 PeerID                 TransportAddress   DiscoverySource
 ------------------------------------------------------------------------------
 10.0.0.1:0             10.0.0.1           Remote Peer : pe1
 10.0.0.3:0             10.0.0.3           GigabitEthernet0/0/0
 ------------------------------------------------------------------------------
 TOTAL: 2 Peer(s) Found.
```

Verification of VC

- Verify results:
  - Check MPLS L2VC

```
<PE1>display mpls l2vc brief
 Total LDP VC : 1     1 up        0 down

 *Client Interface     : GigabitEthernet0/0/1
  Administrator PW     : no
  AC status            : up
  VC State             : up
  Label state          : 0
  Token state          : 0
  VC ID                : 1315
  VC Type              : Ethernet
  session state        : up
  Destination          : 10.0.0.4
  link state           : up
```

Verification of CE Reachability

- Verify results:
  - Check CE reachability.

```
<CE1>ping 100.1.1.2
  PING 100.1.1.2: 56  data bytes, press CTRL_C to break
    Reply from 100.1.1.2: bytes=56 Sequence=1 ttl=255 time=130 ms
    Reply from 100.1.1.2: bytes=56 Sequence=2 ttl=255 time=140 ms
    Reply from 100.1.1.2: bytes=56 Sequence=3 ttl=255 time=130 ms
    Reply from 100.1.1.2: bytes=56 Sequence=4 ttl=255 time=140 ms
    Reply from 100.1.1.2: bytes=56 Sequence=5 ttl=255 time=190 ms

  --- 100.1.1.2 ping statistics ---
    5 packet(s) transmitted
    5 packet(s) received
    0.00% packet loss
    round-trip min/avg/max = 130/146/190 ms
```

VPWS Signaled with BGP

APNIC

VC Signaled with BGP

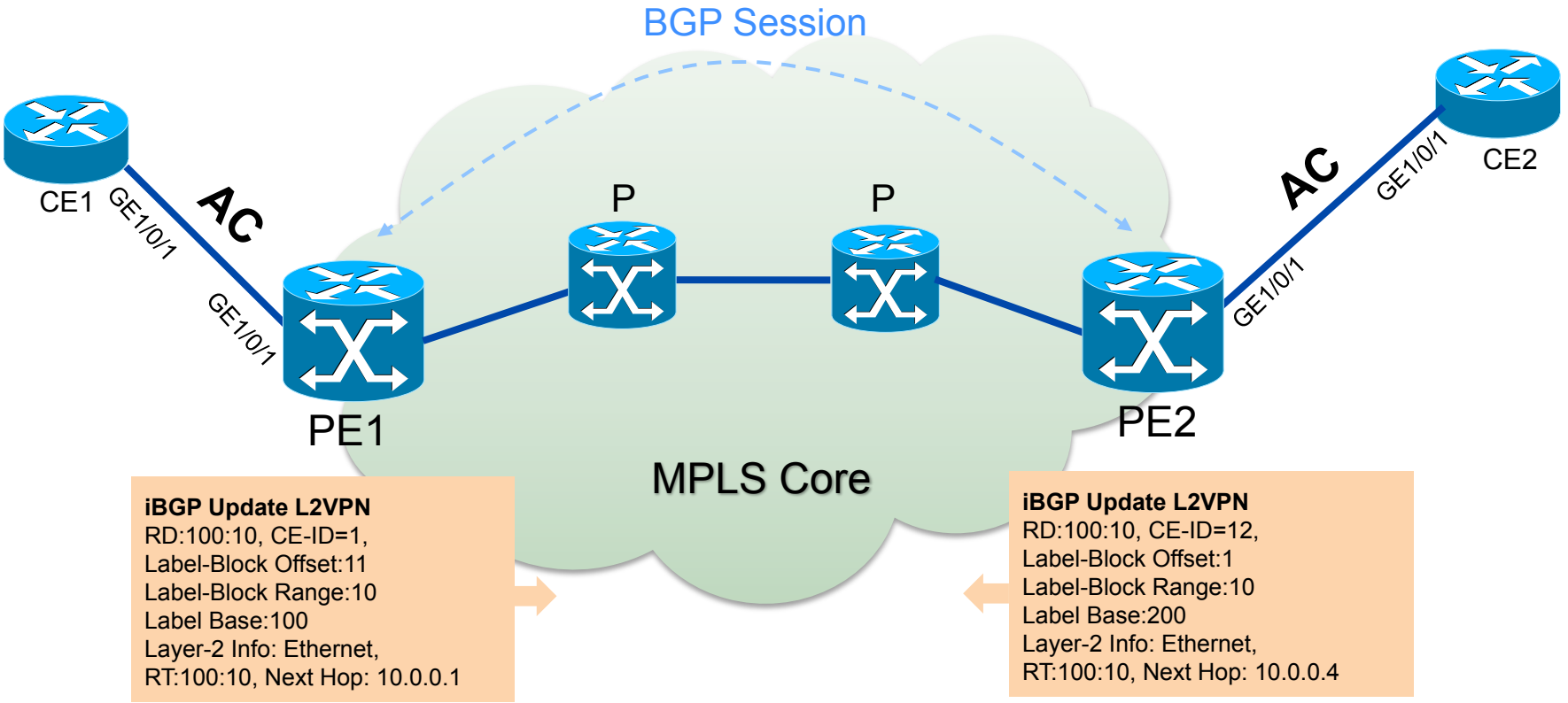
RFC 6624

- BGP is running as the signaling protocol to transmit Layer 2 information and VC labels between PEs.
- BGP was chosen as the means for exchanging L2VPN information for two reasons:
  - It offers mechanisms for both auto-discovery and signaling
  - It allows for operational convergence

VPWS NLRI
Route Distinguisher
CE-ID
Label-block Offset
Label Base
Variable TLVs……

VC Signaled with BGP

- BGP Signaled VPWS uses **VPN targets** to control the receiving and sending of VPN routes, which improves flexibility of the VPN networking.



VC Label in BGP Signaled VPWS

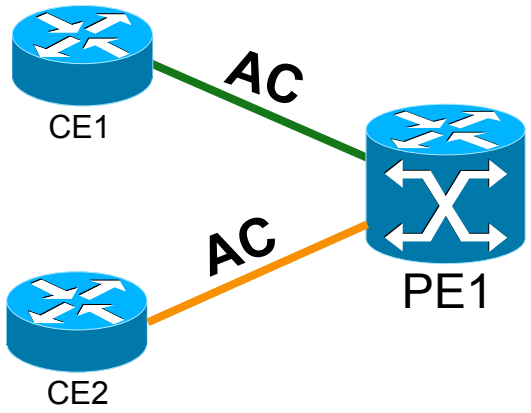
- **VC labels** are assigned through a label block that is pre-allocated for each CE.
- The **size of the label block** determines the number of connections that can be set up between the local CE and other CEs.
- **Additional labels** can be assigned to L2VPNs in the label block for expansion in the future. PEs calculates inner labels according to these label blocks and use the inner labels to transmit packets.

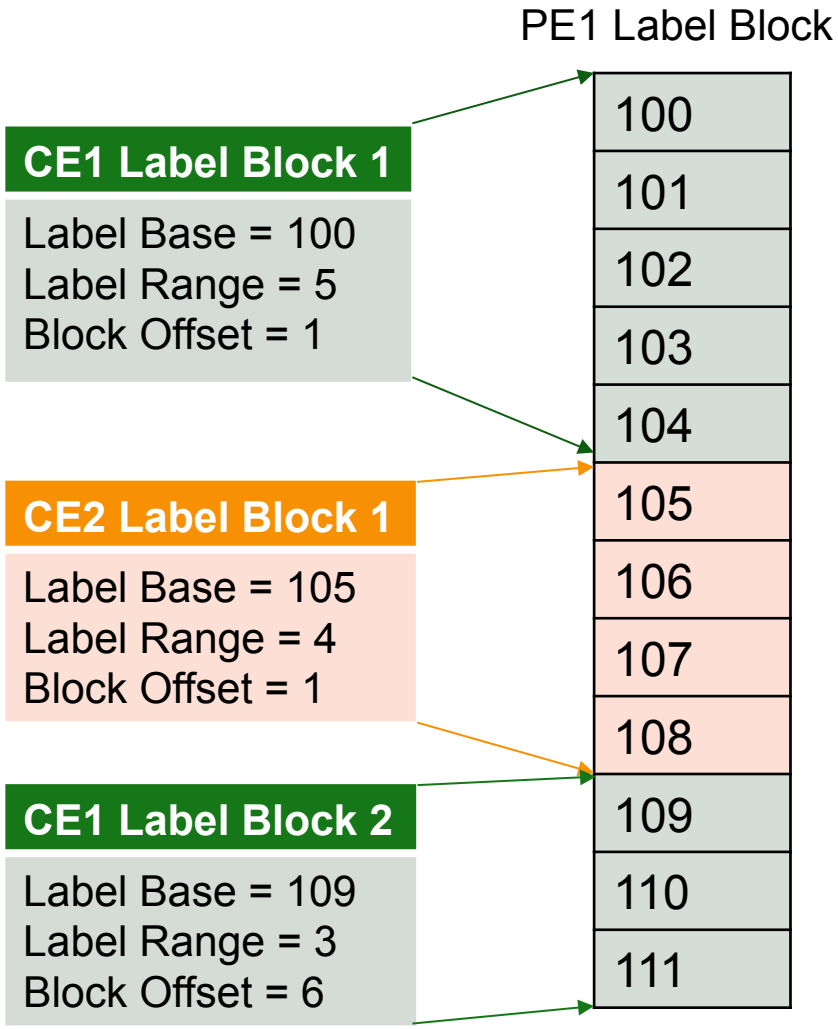
Basic Concepts

Concepts	Explanation
CE ID	A CE ID uniquely identifies a CE in a VPN.
Label Block	A contiguous set of labels.
Label Base	What is the smallest label in one label block?
Label Range	How many labels in one label block?
Block Offset	Value used to identify a label block from which a label value is selected to set up pseudowires for a remote site. Note: In Cisco & Juniper, initial offset is 1. In Huawei, initial offset is 0 by default, can be changed to be 1.

Example of Label Block

- As in the topology, 2 CEs are attached to PE1 to set up L2VPN with other sites.

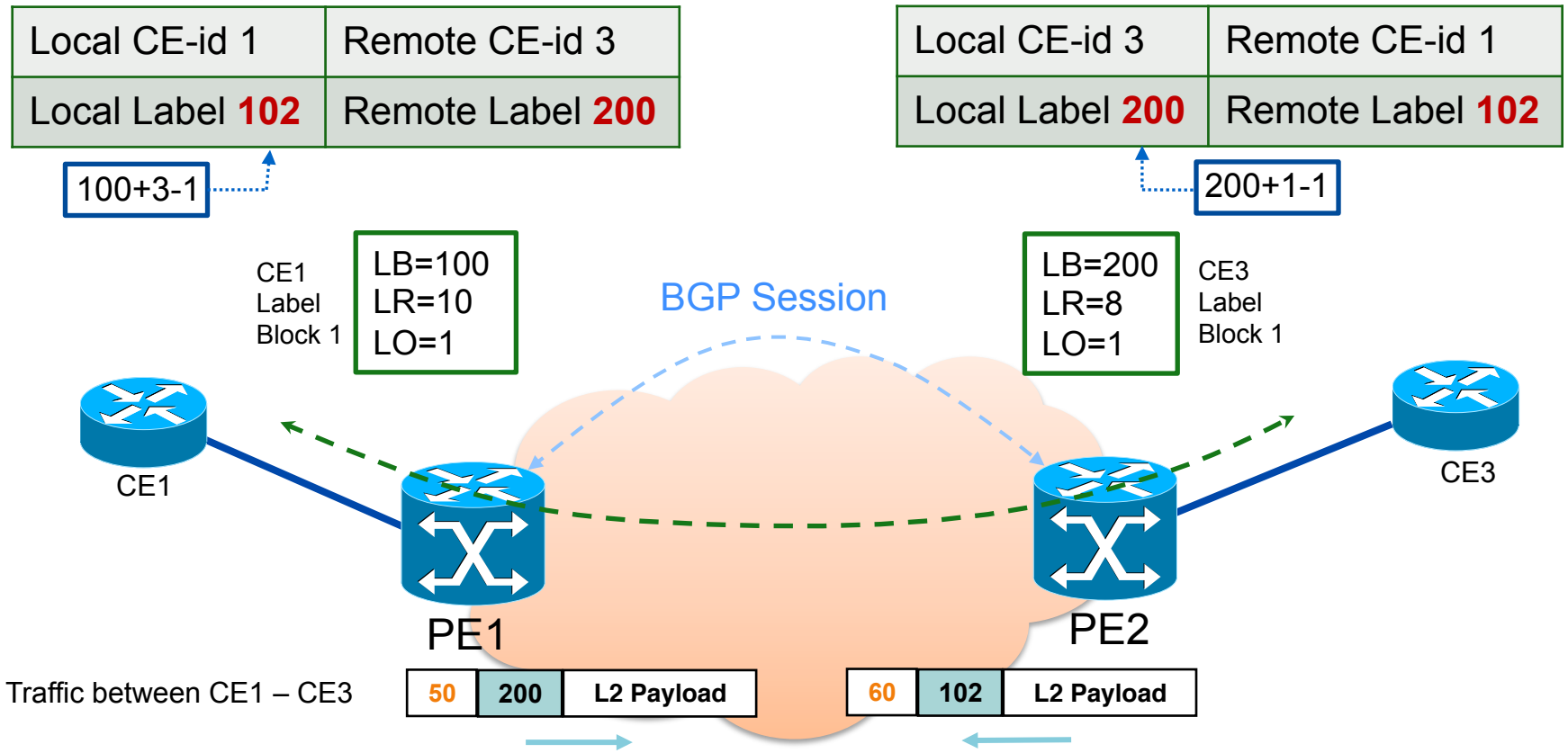




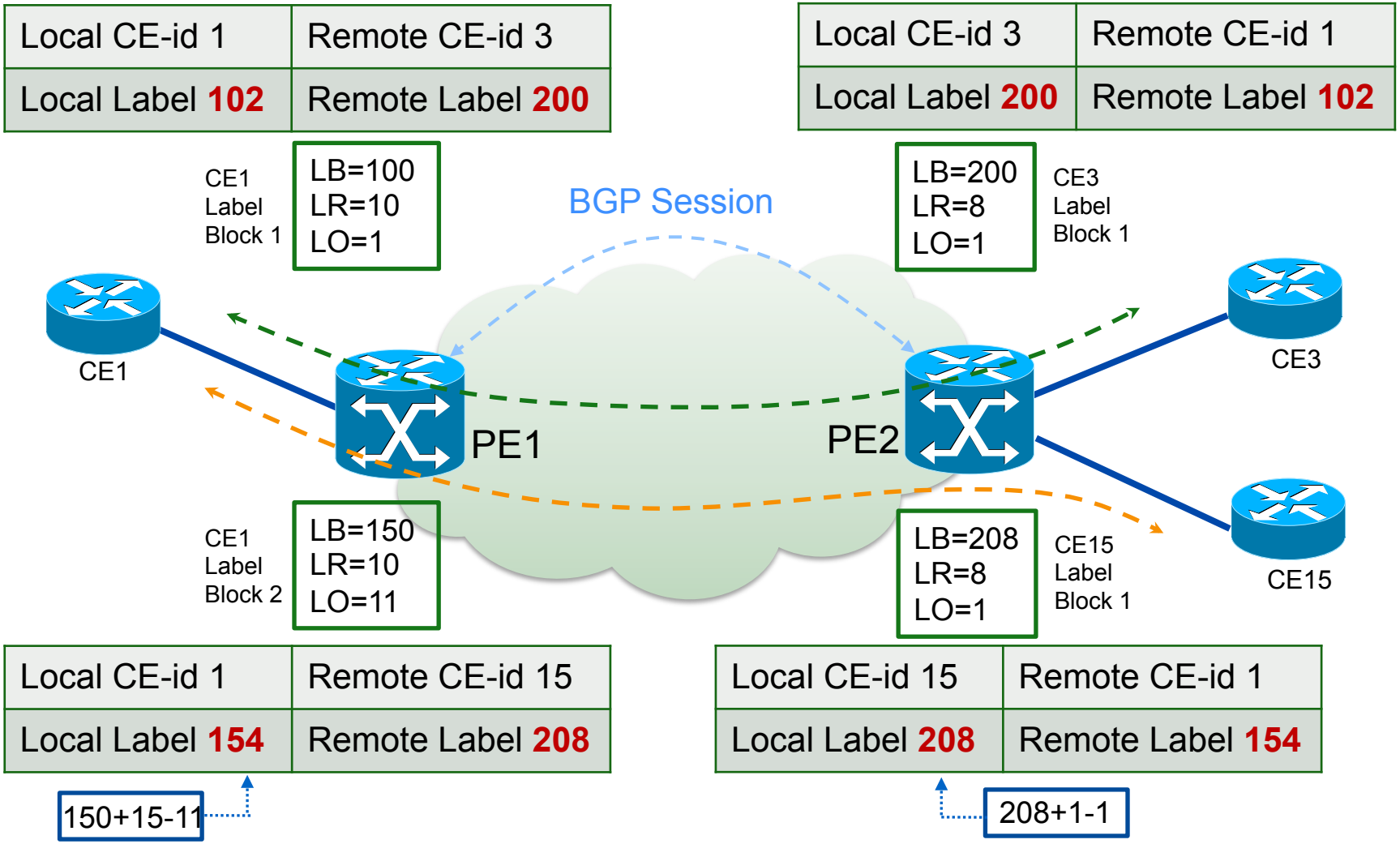
VC Label Calculation

$$Block\ Offset <= Remote\ CE\ ID < Block\ Size + BlockOffset$$

$$Label = Label\ Base + Remote\ CE\ ID - Block\ Offset$$



VC Label Calculation



How to Design CE-id and Label Block

- Label blocks will be generated automatically on the routers by default. Design the CE-id sequentially.
- Cisco IOS XR CLI

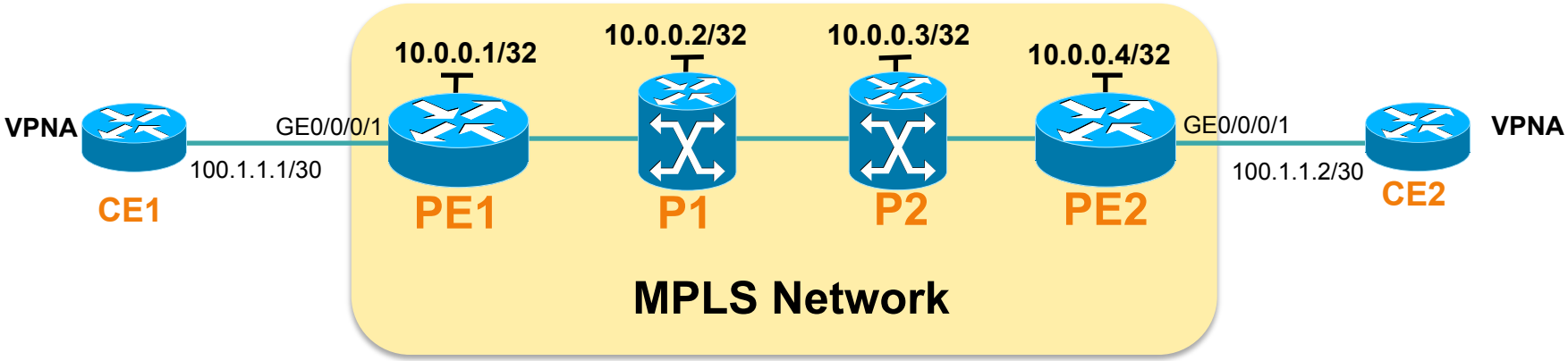
```
……
         signaling-protocol bgp
           ce-id 1
            interface giga0/0/0/1.10 remote-ce-id 4
……
```

- Juniper JunOS CLI

```
……
                  site CE1 {
                      site-identifier 1;
                      interface ge-0/0/1.1 {
                          remote-site-id 4;
……
```

Configuration Example of VPWS Signaled with BGP

- Task: Configure MPLS L2VPN (LDP based)on Cisco **IOS XR** to make the following CEs communication with each other.
- Prerequisite configuration:
  - 1. IP address configuration on all the routers
  - 2. IGP configuration on PE & P routers
  - 3. LDP configuration on PE & P routers



Configure BGP Neighbors

- Configuration steps:
  - 1. Configure BGP neighbors for PE routers in l2vpn address family

On PE1:

```
RP/0/0/CPU0:PE1(config)# router bgp 65000
RP/0/0/CPU0:PE1(config-bgp)# address-family l2vpn vpls-vpws
RP/0/0/CPU0:PE1(config-bgp-af)# exit
RP/0/0/CPU0:PE1(config-bgp)# neighbor 10.0.0.4
RP/0/0/CPU0:PE1(config-bgp-nbr)# remote-as 65000
RP/0/0/CPU0:PE1(config-bgp-nbr)# update-source loopback 0
RP/0/0/CPU0:PE1(config-bgp-nbr)# address-family l2vpn vpls-vpws
RP/0/0/CPU0:PE1(config-bgp-nbr-af)# commit
```

Similar configurations on PE2.

Configure BGP Neighbors (continued)

- Configuration steps:
  - 1. Configure BGP neighbors for PE routers in l2vpn address family

On PE2:

```
RP/0/0/CPU0:PE2(config)# router bgp 65000
RP/0/0/CPU0:PE2(config-bgp)# address-family l2vpn vpls-vpws
RP/0/0/CPU0:PE2(config-bgp-af)# exit
RP/0/0/CPU0:PE2(config-bgp)# neighbor 10.0.0.1
RP/0/0/CPU0:PE2(config-bgp-nbr)# remote-as 65000
RP/0/0/CPU0:PE2(config-bgp-nbr)# update-source loopback 0
RP/0/0/CPU0:PE2(config-bgp-nbr)# address-family l2vpn vpls-vpws
RP/0/0/CPU0:PE2(config-bgp-nbr-af)# commit
```

Enable L2transport

- Configuration steps:
  - 2. Enable L2transport under the interface of PE connecting to CE.

On PE1, GE0/0/0/1.10 connects to CE1:

```
RP/0/0/CPU0:PE1(config)# interface gigabitEthernet 0/0/0/1.10
l2transport
RP/0/0/CPU0:PE1(config-subif)# encapsulation dot1q 10
RP/0/0/CPU0:PE1(config-subif)# rewrite ingress tag pop 1
symmetric
RP/0/0/CPU0:PE1(config-subif)# commit
```

Similar configurations on PE2.

Configure L2VPN xConnect

- Configuration steps:
  - 3. Configuring VPWS with BGP AD & Signaling on PE routers

On PE1:

```
RP/0/0/CPU0:PE1(config)# l2vpn
RP/0/0/CPU0:PE1(config-l2vpn)# xconnect group test1
RP/0/0/CPU0:PE1(config-l2vpn-xc)# mp2mp L2VPN-A
RP/0/0/CPU0:PE1(config-l2vpn-xc-mp2mp)# vpn-id 100
RP/0/0/CPU0:PE1(config-l2vpn-xc-mp2mp)# l2-encapsulation vlan
RP/0/0/CPU0:PE1(config-l2vpn-xc-mp2mp)# autodiscovery bgp
RP/0/0/CPU0:PE1(config-l2vpn-xc-mp2mp-ad)# rd 100:10
RP/0/0/CPU0:PE1(config-l2vpn-xc-mp2mp-ad)# route-target 100:10
RP/0/0/CPU0:PE1(config-l2vpn-xc-mp2mp-ad)# signaling-protocol bgp
RP/0/0/CPU0:PE1(config-l2vpn-xc-mp2mp-ad-sig)# ce-id 1
RP/0/0/CPU0:PE1(config-l2vpn-xc-mp2mp-ad-sig-ce)# interface
giga0/0/0/1.10 remote-ce-id 20
RP/0/0/CPU0:PE1(config-l2vpn-xc-mp2mp-ad-sig-ce)# commit
```

Similar configurations on PE2.

Configure L2VPN xConnect (continued)

- Configuration steps:
  - 3. Configuring VPWS with BGP AD & Signaling on PE routers

On PE2:

```
RP/0/0/CPU0:PE2(config)# l2vpn
RP/0/0/CPU0:PE2(config-l2vpn)# xconnect group test1
RP/0/0/CPU0:PE2(config-l2vpn-xc)# mp2mp L2VPN-A
RP/0/0/CPU0:PE2(config-l2vpn-xc-mp2mp)# vpn-id 100
RP/0/0/CPU0:PE2(config-l2vpn-xc-mp2mp)# l2-encapsulation vlan
RP/0/0/CPU0:PE2(config-l2vpn-xc-mp2mp)# autodiscovery bgp
RP/0/0/CPU0:PE2(config-l2vpn-xc-mp2mp-ad)# rd 100:20
RP/0/0/CPU0:PE2(config-l2vpn-xc-mp2mp-ad)# route-target 100:10
RP/0/0/CPU0:PE2(config-l2vpn-xc-mp2mp-ad)# signaling-protocol bgp
RP/0/0/CPU0:PE2(config-l2vpn-xc-mp2mp-ad-sig)# ce-id 20
RP/0/0/CPU0:PE2(config-l2vpn-xc-mp2mp-ad-sig-ce)# interface
giga0/0/0/1.10 remote-ce-id 1
RP/0/0/CPU0:PE1(config-l2vpn-xc-mp2mp-ad-sig-ce)# commit
```

Verify Status of xConnect

- Check the status of xConnect on PE routers:

```
RP/0/0/CPU0:PE1# show l2vpn xconnect
Thu Jan  5 06:20:48.308 UTC
Legend: ST = State, UP = Up, DN = Down, AD = Admin Down, UR = Unresolved,
        SB = Standby, SR = Standby Ready, (PP) = Partially Programmed


XConnect                    Segment 1                       Segment 2
Group      Name       ST    Description             ST      Description              ST
------------------------    ------------------------------  ------------------------------
test1      L2VPN-A.1:20
                      UP    Gi0/0/0/1.10            UP      10.0.0.4        65556    UP
------------------------------------------------------------------------------------------
```

Verify Status of xConnect

- Check the status of xConnect on PE routers:

```
RP/0/0/CPU0:PE1#show l2vpn discovery xconnect
Thu Jan  5 13:23:13.529 UTC

Service Type: VPWS,  Connected
  List of VPNs (1 VPNs):
  XC Group: test1, MP2MP L2VPN-A, id: 0, signaling protocol: BGP
    List of Local Edges (1 Edges):
      Local Edge ID: 1, Label Blocks (1 Blocks)
        Label base        Offset    Size      Time Created
        ----------        ------    ----      --------------------
        24015             11        10        01/05/2017 07:16:04
        Status Vector: ff bf
    List of Remote Edges (1 Edges):
      Remote Edge ID: 20, NLRIs (1 NLRIs)
        Label base        Offset    Size      Peer ID         Time Created
        ----------        ------    ----      --------------- --------------------
        24000             1         10        10.0.0.4        01/05/2017 07:23:35
        Status Vector: 7f ff
```

VC Label Calculation

$$\text{Label} = \textit{Label Base} + \textit{Remote CE ID} - \textit{Block Offset}$$

- PE1:
  - Local label = 24015+20-11= **24024**

- PE2:
  - Local label = 24000+1-1=**24000**

Check xConnect Detail

- Check the detail of xConnect on PE routers:

```
RP/0/0/CPU0:PE1#show l2vpn xconnect detail
Fri Jan  6 07:03:35.531 UTC
Group test1, XC L2VPN-A.1:20, state is up; Interworking none
  Local CE ID: 1, Remote CE ID: 20, Discovery State: Advertised
  AC: GigabitEthernet0/0/0/1.10, state is up
  ... ...(Omitted)
  PW: neighbor 10.0.0.4, PW ID 65556, state is up ( established )
    PW class not set, XC ID 0xff000001
    Encapsulation MPLS, Auto-discovered (BGP), protocol BGP
    Source address 10.0.0.1
    PW type Ethernet VLAN, control word enabled, interworking none
    PW backup disable delay 0 sec
    Sequencing not set
      MPLS         Local                          Remote
      ------------ ------------------------------ ------------------------------
      Label        24024                          24000
      MTU          1500                           1500
      Control word enabled                        enabled
      PW type      Ethernet VLAN                  Ethernet VLAN
      CE-ID        1                              20
      ------------ ------------------------------ ------------------------------
    MIB cpwVcIndex: 4278190081
```

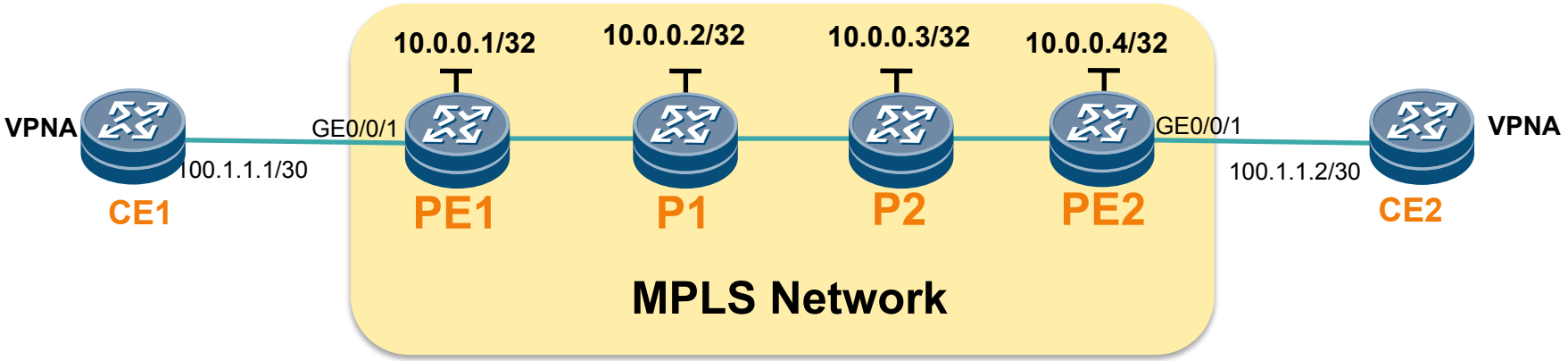
Verify BGP L2VPN VPWS Status

- Check BGP L2VPN VPWS status:

```
RP/0/0/CPU0:PE1# show bgp l2vpn vpws
Thu Jan  5 13:24:55.912 UTC
BGP router identifier 10.0.0.1, local AS number 65000
BGP generic scan interval 60 secs
Non-stop routing is enabled
BGP table state: Active
Table ID: 0x0   RD version: 0
BGP main routing table version 5
BGP NSR Initial initsync version 3 (Reached)
BGP NSR/ISSU Sync-Group versions 0/0
BGP scan interval 60 secs

Status codes: s suppressed, d damped, h history, * valid, > best
              i - internal, r RIB-failure, S stale, N Nexthop-discard
Origin codes: i - IGP, e - EGP, ? - incomplete
   Network            Next Hop        Rcvd Label      Local Label
Route Distinguisher: 100:10 (default for vrf test1:L2VPN-A)
*> 1:11/32            0.0.0.0         nolabel         24015
*>i20:1/32            10.0.0.4        24000           nolabel
Route Distinguisher: 100:20
*>i20:1/32            10.0.0.4        24000           nolabel

Processed 3 prefixes, 3 paths
```

Configuration Example of VPWS Signaled with BGP

- Task: Configure MPLS L2VPN (LDP based)on HUAWEI **VRP5** to make the following CEs communication with each other.
- Prerequisite configuration:
  - 1. IP address configuration on all the routers
  - 2. IGP configuration on PE & P routers
  - 3. LDP configuration on PE & P routers



Configure BGP Neighbors

- Configuration steps:
  - 1. Configure BGP neighbors for PE routers in l2vpn address family

On PE1:

```
[PE1] mpls l2vpn
[PE1-l2vpn] quit
[PE1] bgp 65000
[PE1-bgp] peer 10.0.0.4 as-number 65000
[PE1-bgp] peer 10.0.0.4 connect-interface loopback 0
[PE1-bgp] l2vpn-family
[PE1-bgp-af-l2vpn] peer 10.0.0.4 enable
[PE1-bgp-af-l2vpn] quit
[PE1-bgp] quit
```

Similar configurations required on PE2.

Configure BGP Neighbors (continued)

- Configuration steps:
  - 1. Configure BGP neighbors for PE routers in l2vpn address family

On PE2:

```
[PE2] mpls l2vpn
[PE2-l2vpn] quit
[PE2] bgp 65000
[PE2-bgp] peer 10.0.0.1 as-number 65000
[PE2-bgp] peer 10.0.0.1 connect-interface loopback 0
[PE2-bgp] l2vpn-family
[PE2-bgp-af-l2vpn] peer 10.0.0.1 enable
[PE2-bgp-af-l2vpn] quit
[PE2-bgp] quit
```

Configure VPWS BGP Signaling (1)

- Configuration steps:
  - 2. Configuring VPWS with BGP AD & Signaling on PE routers

On PE1:

```
[PE1] mpls l2vpn vpn1 encapsulation ethernet
[PE1-mpls-l2vpn-vpn1] route-distinguisher 100:10
[PE1-mpls-l2vpn-vpn1] vpn-target 100:10
[PE1-mpls-l2vpn-vpn1] ce ce1 id 1 range 25 default-offset 1
```

Local label range 25

Initial label offset 1

```
[PE1-mpls-l2vpn-ce-vpn1-ce1] connection ce-offset 2 interface
gigabitethernet 0/0/1
[PE1-mpls-l2vpn-ce-vpn1-ce1] quit
[PE1-mpls-l2vpn-vpn1] quit
```

Remote CE ID

Similar configurations required on PE2.

Configure VPWS BGP Signaling (2)

- Configuration steps:
  - 2. Configuring VPWS with BGP AD & Signaling on PE routers

On PE2:

```
[PE2] mpls l2vpn vpn1 encapsulation ethernet
[PE2-mpls-l2vpn-vpn1] route-distinguisher 100:20
[PE2-mpls-l2vpn-vpn1] vpn-target 100:10
[PE2-mpls-l2vpn-vpn1] ce ce1 id 20 range 10 default-offset 1
```

Local label range 25

Initial label offset 1

```
[PE1-mpls-l2vpn-ce-vpn1-ce1] connection ce-offset 2 interface
gigabitethernet 0/0/1
[PE1-mpls-l2vpn-ce-vpn1-ce1] quit
[PE1-mpls-l2vpn-vpn1] quit
```

Remote CE ID

Verify L2VPN Connection

- Verify the results of L2VPN connection:

```
<PE1>display mpls l2vpn connection vpn1
VPN name: vpn1,
1 total connections,
connections: 1 up, 0 down, 0 local, 1 remote, 0 unknown

  CE name: ce1, id: 1,
  Rid type status peer-id         route-distinguisher interface      primary or not
-------------------------------------------------------------------------------
  20  rmt  up     10.0.0.4        100:10              GigabitEthernet0/0/1   primary
```

Verify BGP Neighbor Relationship

- Verify the results of BGP neighbor relationship:

```
<PE1>display bgp l2vpn peer

 BGP local router ID : 10.0.0.1
 Local AS number : 65000
 Total number of peers : 1                 Peers in established state : 1

  Peer            V          AS  MsgRcvd  MsgSent  OutQ  Up/Down       State PrefRcv

  10.0.0.4        4       65000       20       26     0 00:14:44 Established       0
```

Verify BGP L2VPN

- Verify the results of BGP L2VPN:

```
<PE1>display bgp l2vpn route-distinguisher 100:20 ce-id 20

 BGP Local router ID : 10.0.0.1, local AS number : 65000
 Origin codes:i - IGP, e - EGP, ? - incomplete
 CE ID    Label Offset    Label Base     nexthop          pref    as-path
 20       1               100006         10.0.0.4         100
```

Information received from remote site:
Remote CE ID, Remote Label Offset, Remote Label Base

```
<PE2>display bgp l2vpn route-distinguisher 100:10 ce-id 1

 BGP Local router ID : 10.0.0.4, local AS number : 65000
 Origin codes:i - IGP, e - EGP, ? - incomplete
 CE ID    Label Offset    Label Base     nexthop          pref    as-path
 1        1               100001         10.0.0.1         100
```

VC Label Calculation

$$\text{Label} = \textit{Label Base} + \textit{Remote CE ID} - \textit{Block Offset}$$

- PE1:

```
<PE1>display mpls l2vpn vpn1 local-ce
ce-name                 ce-id     range     conn-num  CEBase/LBBase/Offset/Range
ce1                     1         25        1         0/100001/1/25
```

  – Local label = 100001+20-1= **100020**

- PE2:

```
<PE2>display mpls l2vpn vpn1 local-ce | include ce20
ce-name                 ce-id     range     conn-num  CEBase/LBBase/Offset/Range
ce20                    20        10        1         0/100006/1/10
```

  – Local label = 100006+1-1=**100006**

Verify Detail of L2VPN Connection

- Check the detail of L2VPN connection:

```
<PE1> display mpls l2vpn connection vpn1 verbose
VPN name: vpn1,
1 total connections,
connections: 1 up, 0 down, 0 local, 1 remote, 0 unknown
conn-type: remote
     local vc state:              up
     remote vc state:             up
     local ce-id:                 1
     local ce name:               ce1
     remote ce-id:                20
     intf(state,encap):           GigabitEthernet0/0/1(up,ethernet)
     peer id:                     10.0.0.4
     route-distinguisher:         100:20
     local vc label:              100020
     remote vc label:             100006
     tunnel policy:               default
     CKey:                        18
     NKey:                        17
     primary or secondary:        primary
     forward entry exist or not: true
     forward entry active or not:true
     manual fault set or not:     not set
     AC OAM state:                up
```

Verification of CE Reachability

- Check the reachability between CEs.

```
<CE2>ping 100.1.1.1
  PING 100.1.1.1: 56  data bytes, press CTRL_C to break
    Reply from 100.1.1.1: bytes=56 Sequence=1 ttl=255 time=90 ms
    Reply from 100.1.1.1: bytes=56 Sequence=2 ttl=255 time=150 ms
    Reply from 100.1.1.1: bytes=56 Sequence=3 ttl=255 time=140 ms
    Reply from 100.1.1.1: bytes=56 Sequence=4 ttl=255 time=110 ms
    Reply from 100.1.1.1: bytes=56 Sequence=5 ttl=255 time=140 ms

  --- 100.1.1.1 ping statistics ---
    5 packet(s) transmitted
    5 packet(s) received
    0.00% packet loss
    round-trip min/avg/max = 90/126/150 ms
```

```
<CE2>tracert 100.1.1.1

 traceroute to  100.1.1.1(100.1.1.1), max hops: 30 ,packet length:
40,press CTRL_C to break

 1 100.1.1.1 180 ms  130 ms  140 ms
```

Questions?

Issue Date:

Revision:

APNIC

Deploy VPLS

APNIC

Issue Date: [201609]
Revision: [01]

APNIC

Acknowledgement

- Cisco Systems

VPLS Overview

APNIC

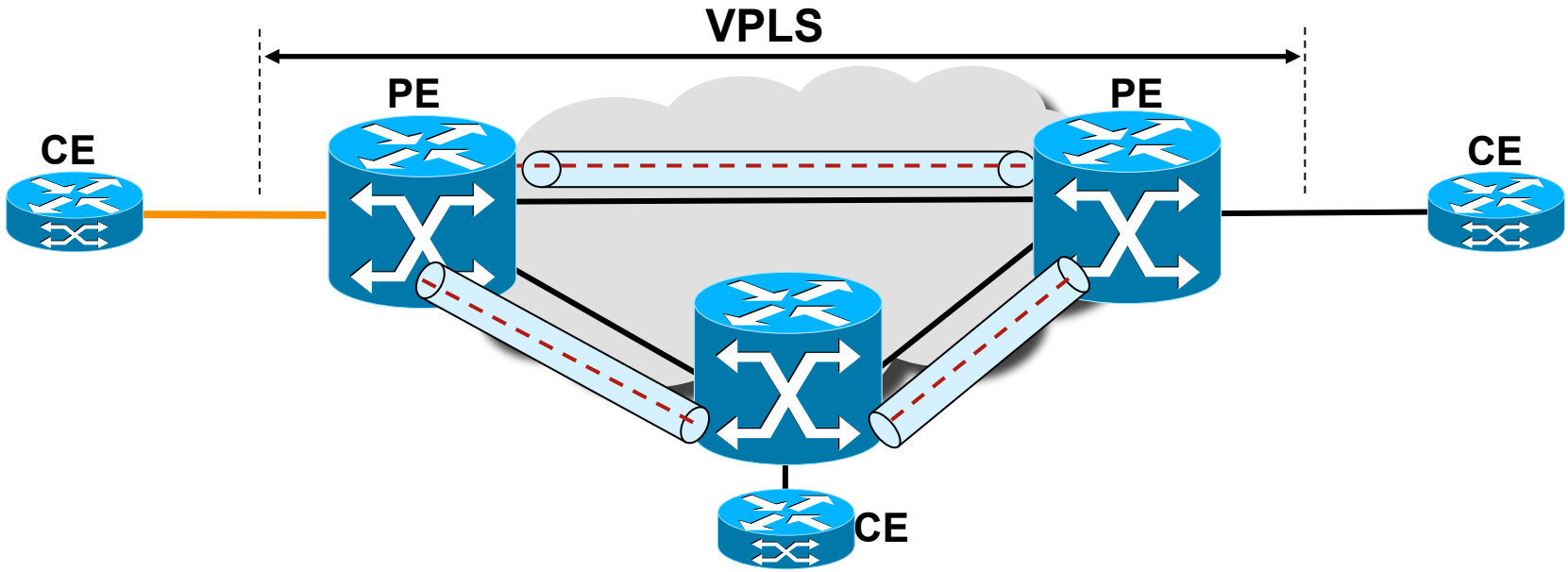
Virtual Private LAN Service

- End-to-end architecture that allows MPLS networks to provide Multipoint Ethernet services

Virtual	• Multiple instances of this service share the same physical infrastructure
Private	• Each instance of the service is independent and isolated from one another
LAN Service	• It emulates Layer 2 multipoint connectivity between subscribers

Virtual Private LAN Service (VPLS)

- VPLS defines an architecture allows MPLS networks offer Layer 2 multipoint Ethernet Services
- SP emulates an IEEE Ethernet bridge network (virtual)
- Virtual Bridges linked with MPLS Pseudo Wires
  - Data Plane used is same as VPWS(point-to-point)

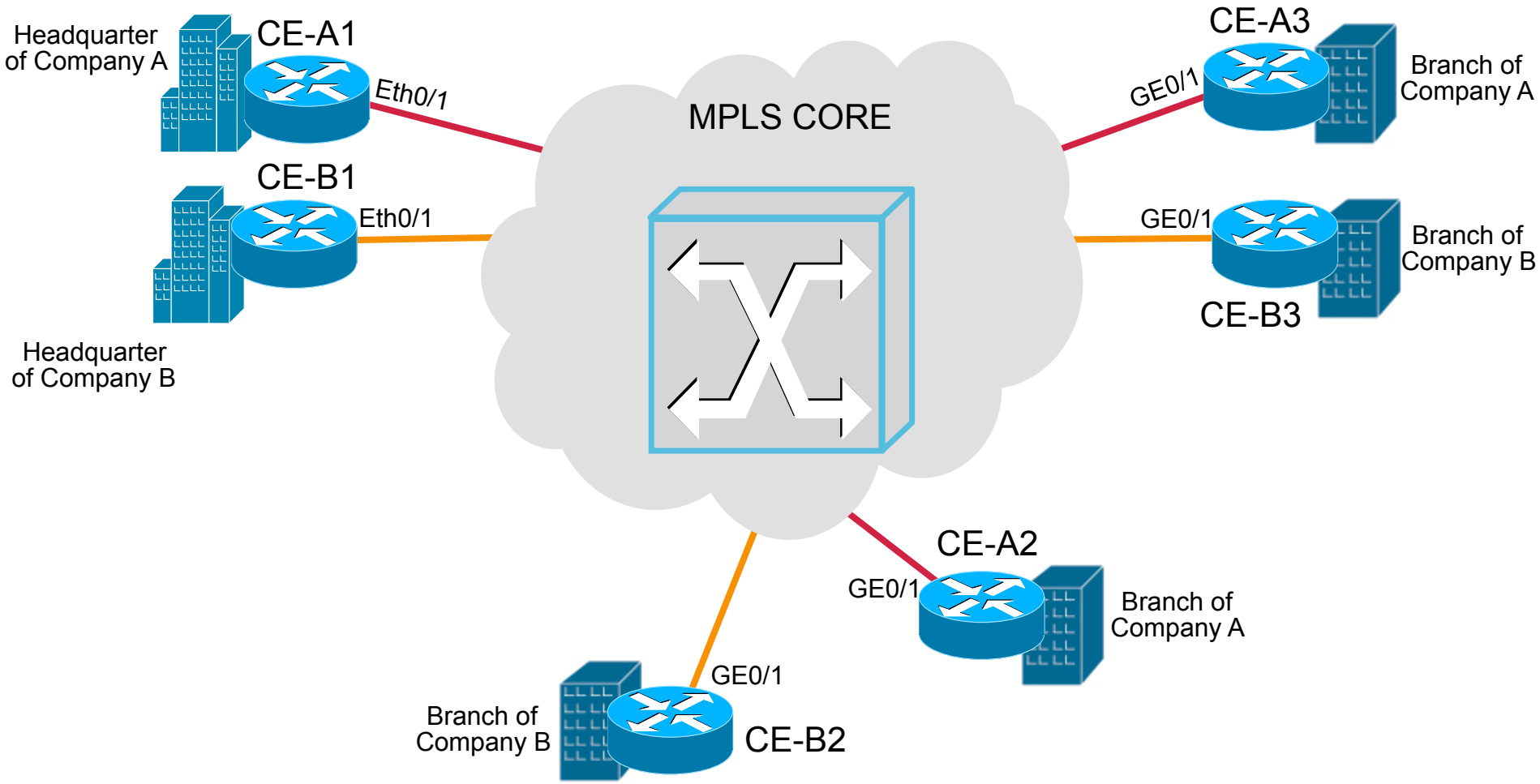


Ethernet Advantage

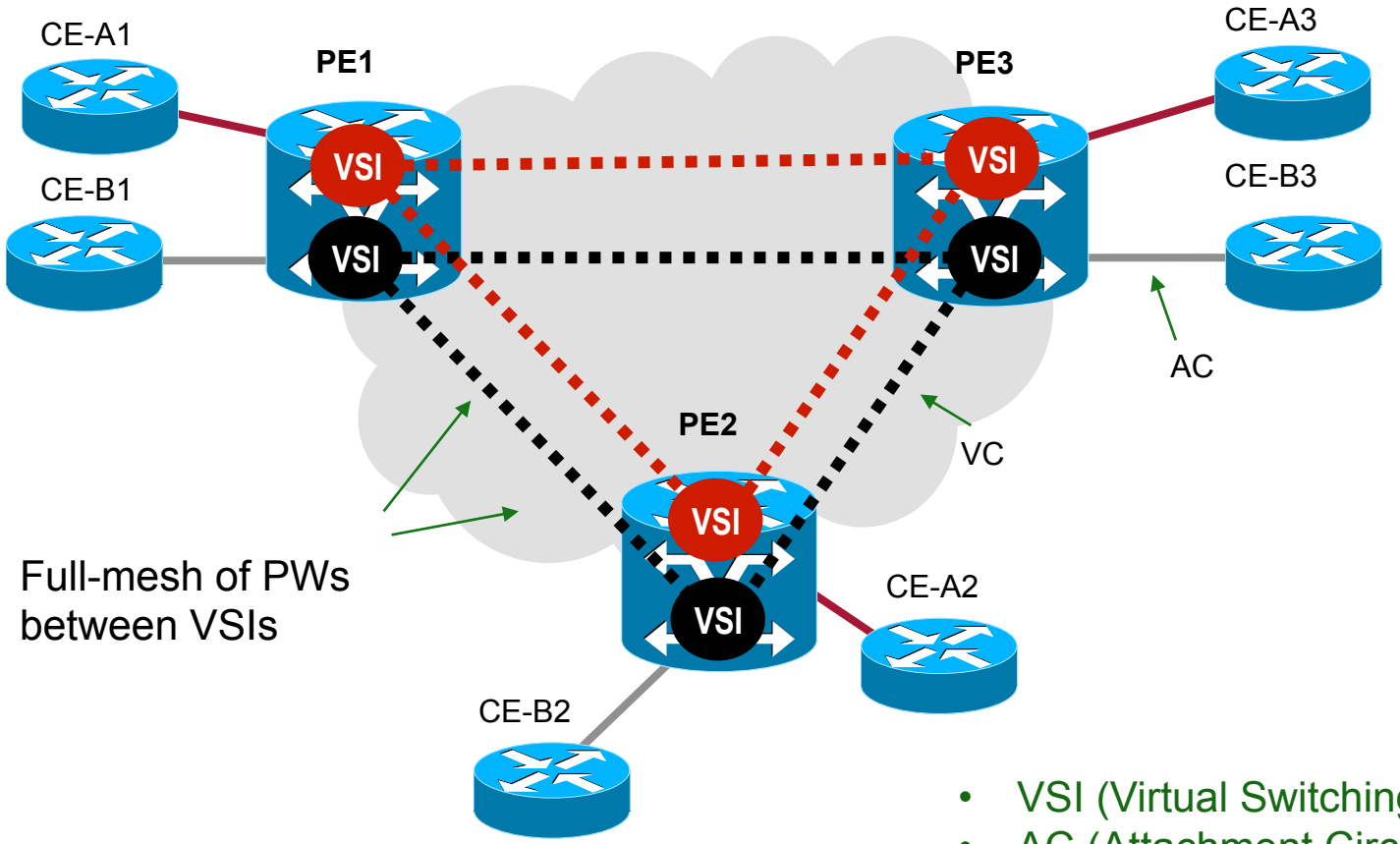
- Flexible logical interface definitions based on VLANs
- Flexible bandwidth provisioning
- Ubiquitous, low-cost interface technology
- Compatibility with technology currently deployed in enterprise LAN networks
- Outstanding bandwidth-to-cost ratio
- Simplified operational support requirements

http://www.cisco.com/en/US/products/hw/routers/ps368/products_white_paper09186a00801df1df.shtml

VPLS Topology

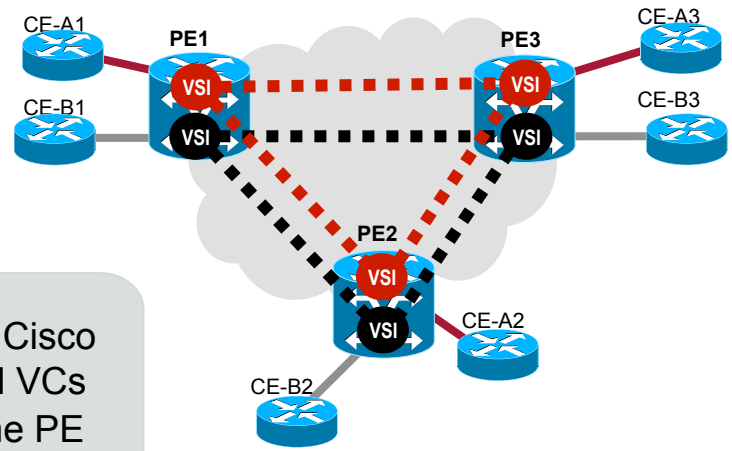


VPLS Basic Concepts



- VSI (Virtual Switching Instance)
- AC (Attachment Circuit)
- VC (Virtual Circuit)

VPLS Basic Concepts



VSI

- Also called VFI (Virtual Forwarding Instance) in Cisco
- Emulates L2 broadcast domain among ACs and VCs
- Unique per service. Multiple VSIs can exist same PE

AC

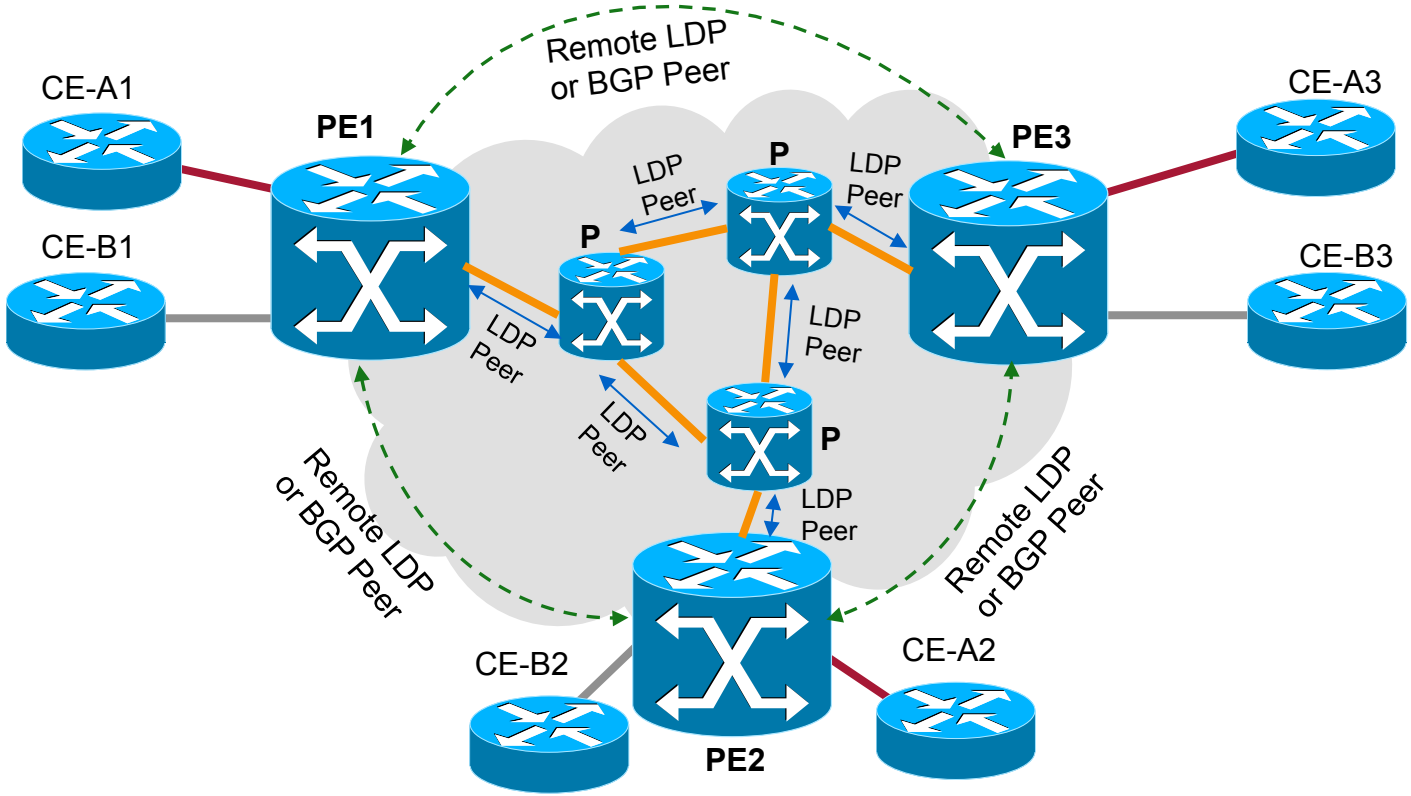
- Connect to CE device, it could be Ethernet physical or logical port
- One or multiple ACs can belong to same VSI

VC

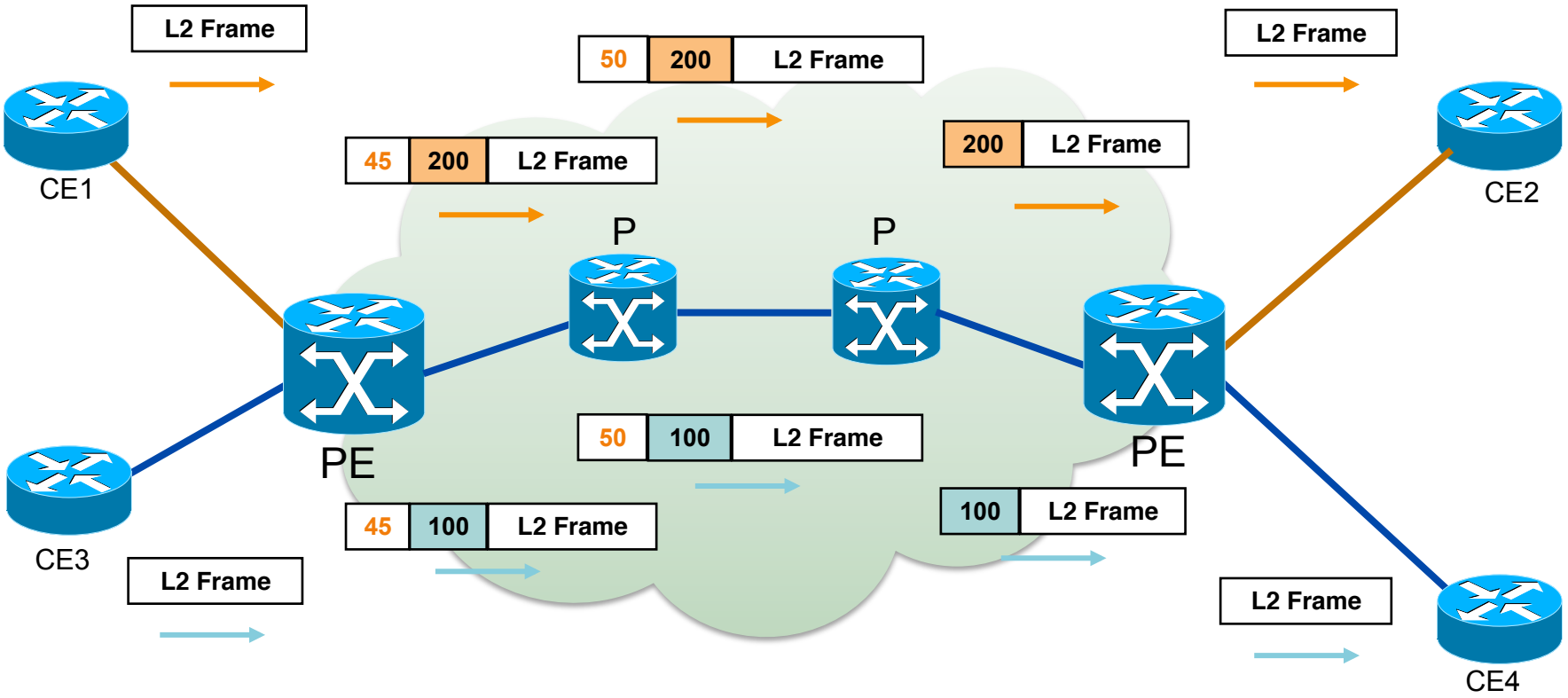
- EoMPLS data encapsulation, tunnel label used to reach remote PE, VC label used to identify VSI
- One or multiple VCs can belong to same VSI
- PEs must have a full-mesh of PWs in the VPLS core

VPLS Control Plane

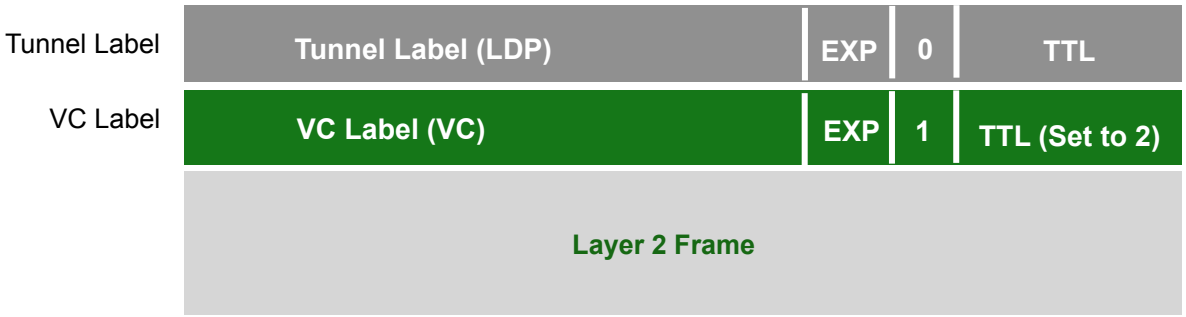
- Tunnel label is distributed by LDP
- VC label is distributed by targeted LDP or BGP



Data Plane of VPLS



VPLS Traffic Encapsulation



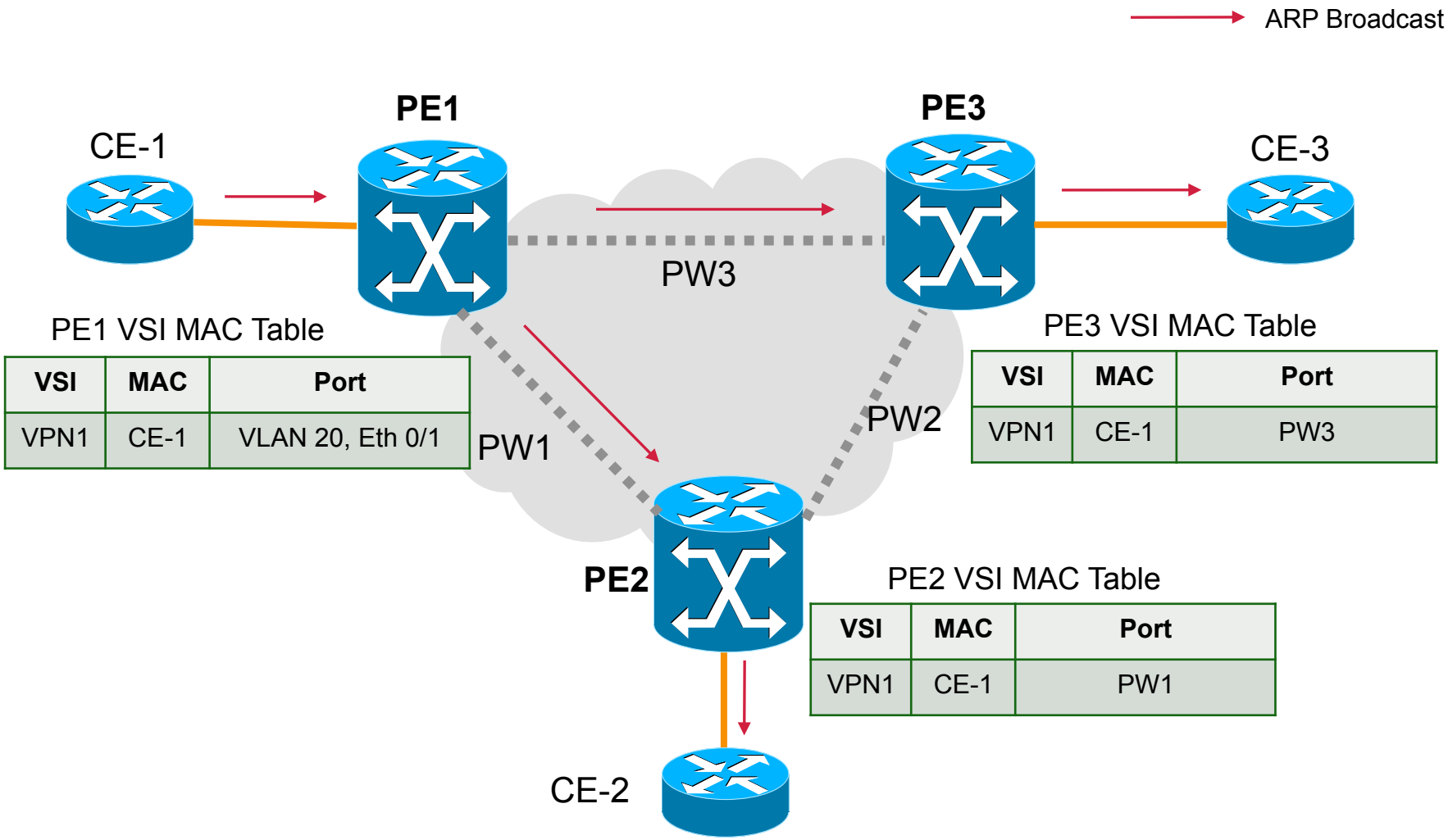
Three-level encapsulation:

1. Packets switched between PEs using Tunnel label
2. VC label identifies PW, VC label signaled between PEs

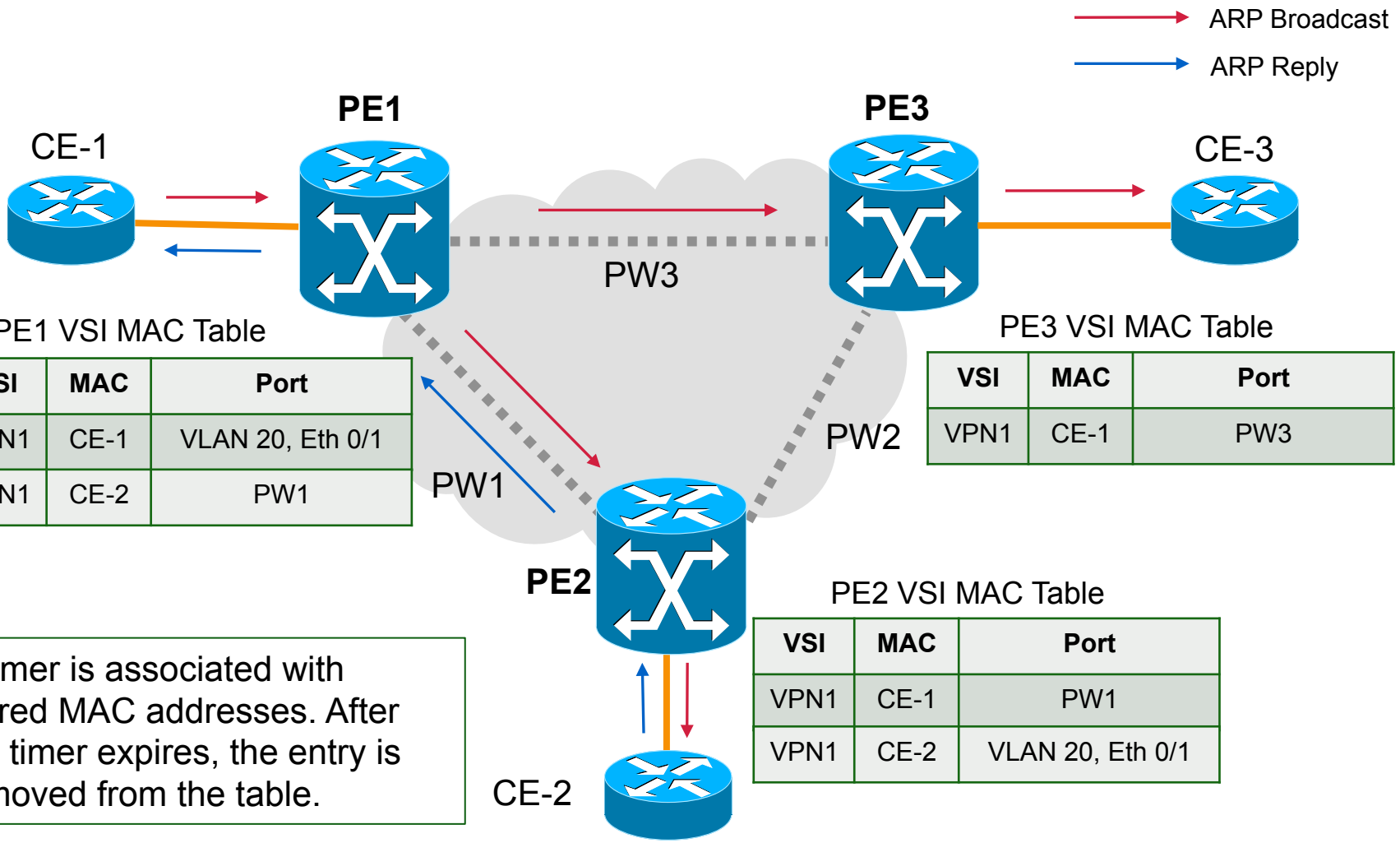
Virtual Switch

- A Virtual Switch MUST operate like a conventional Layer2 switch
- Flooding / Forwarding;
  - Unicast forwarding if destination MAC address is learn before, otherwise flood all (Broadcast/ Multicast/ Unknown Unicast frame)
  - MAC table instances per customer and per customer VLAN for each PE
  - VSI will participate in learning, forwarding process
- Address Learning / Aging:
  - Self Learn/data plane help learning source MAC address to Port
  - Refresh MAC timers with incoming frames
- Loop Prevention:
  - Use “split horizon” concepts instead of STP to prevent loops

VPLS MAC Address Learning (1)



VPLS MAC Address Learning (2)



PE1 VSI MAC Table

VSI	MAC	Port
VPN1	CE-1	VLAN 20, Eth 0/1
VPN1	CE-2	PW1

PE3 VSI MAC Table

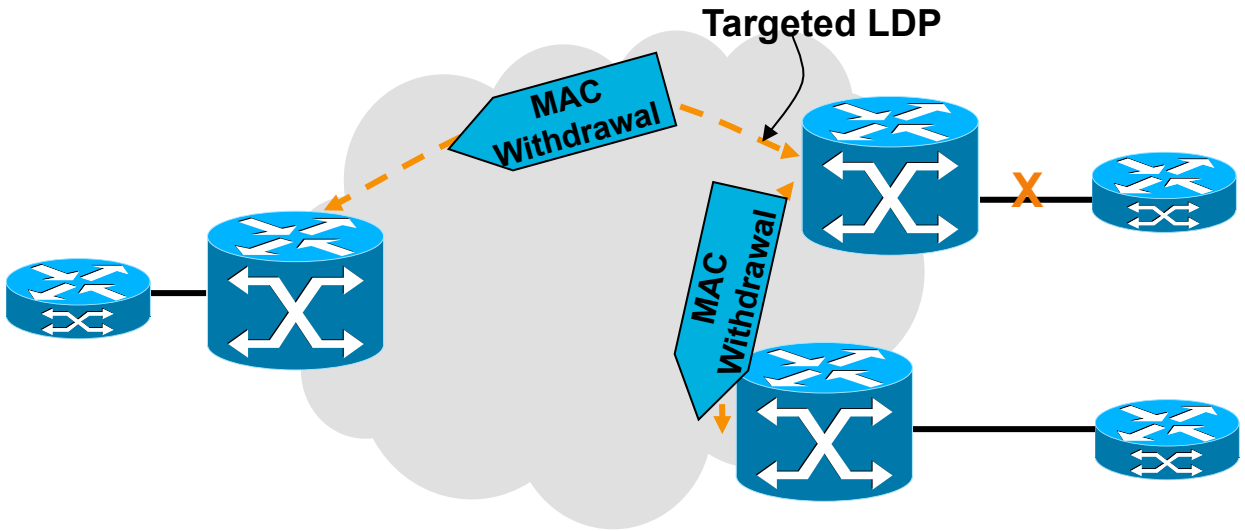
VSI	MAC	Port
VPN1	CE-1	PW3

PE2 VSI MAC Table

VSI	MAC	Port
VPN1	CE-1	PW1
VPN1	CE-2	VLAN 20, Eth 0/1

A timer is associated with stored MAC addresses. After the timer expires, the entry is removed from the table.

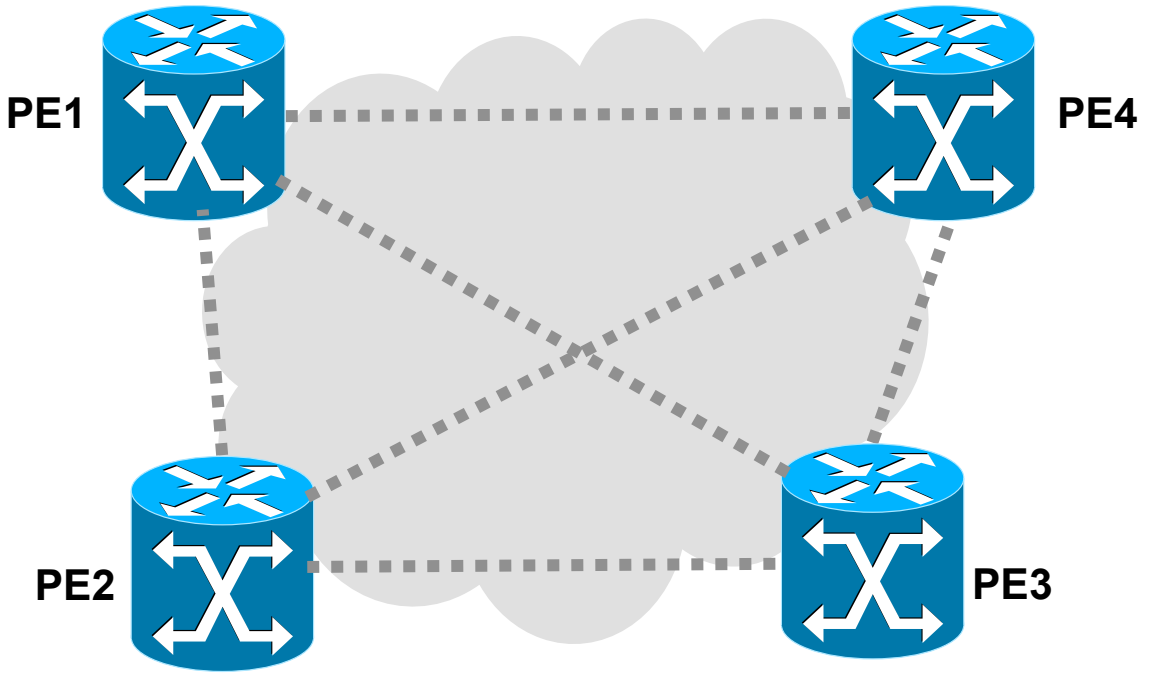
MAC Address Withdrawal Message (LDP)



- Message speeds up convergence process
  - Otherwise PE relies on MAC Address Aging Timer
- Upon failure PE removes locally learned MAC addresses
- Send LDP Address Withdraw (RFC3036) to remote PEs in VPLS (using the Directed LDP session)
- New MAC List TLV is used to withdraw addresses

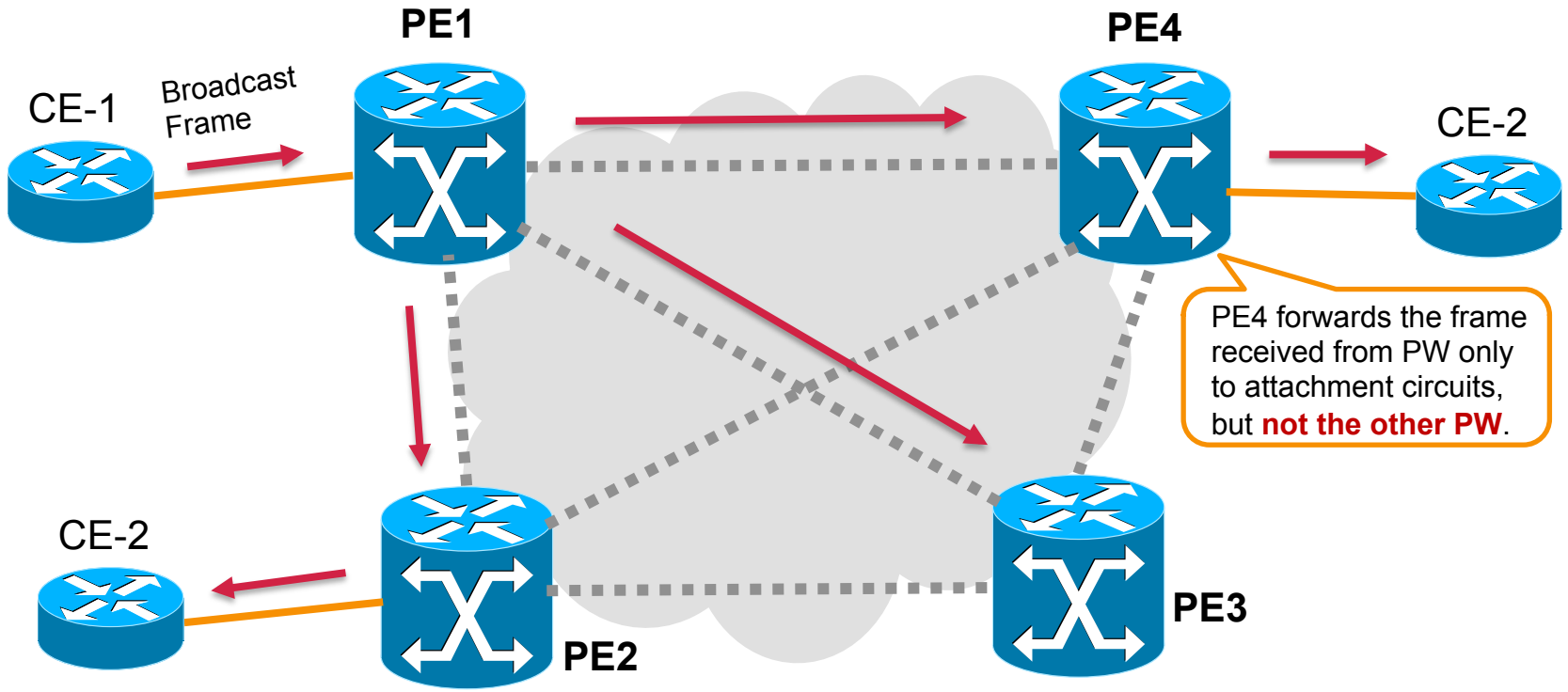
Full Mesh between PEs

- The full mesh between PEs ensure that each host can receive traffic from all other hosts.
- PW signal plane can be LDP or BGP.



Split Horizon

- The split horizon between PEs ensures loop-free in VPLS forwarding.



Abstraction of VPLS

Provisioning Model

- What information needs to be configured and in what entities
- Semantic structure of the endpoint identifiers (e.g. VPN ID)

Discovery

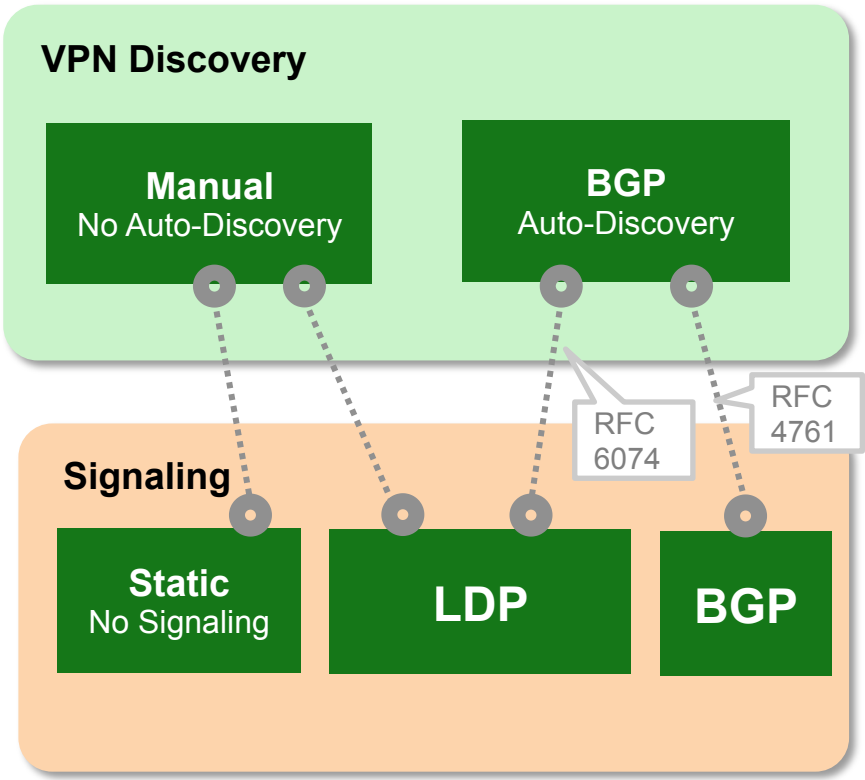
- Provisioning information is distributed by a "discovery process“
- Distribution of endpoint identifiers

Signaling

- When the discovery process is complete, a signaling protocol is automatically invoked to set up pseudowires (PWs)

Discovery and Signaling Alternatives

- VPLS Signaling
  - LDP-based (RFC 4762)
  - BGP-based (RFC 4761)
- VPLS with LDP-signaling and No auto-discovery
  - Operational complexity for larger deployments
- BGP-based Auto-Discovery (BGP-AD) (RFC 6074)
  - Enables discovery of PE devices in a VPLS instance
- BGP Signaling (RFC 4761)

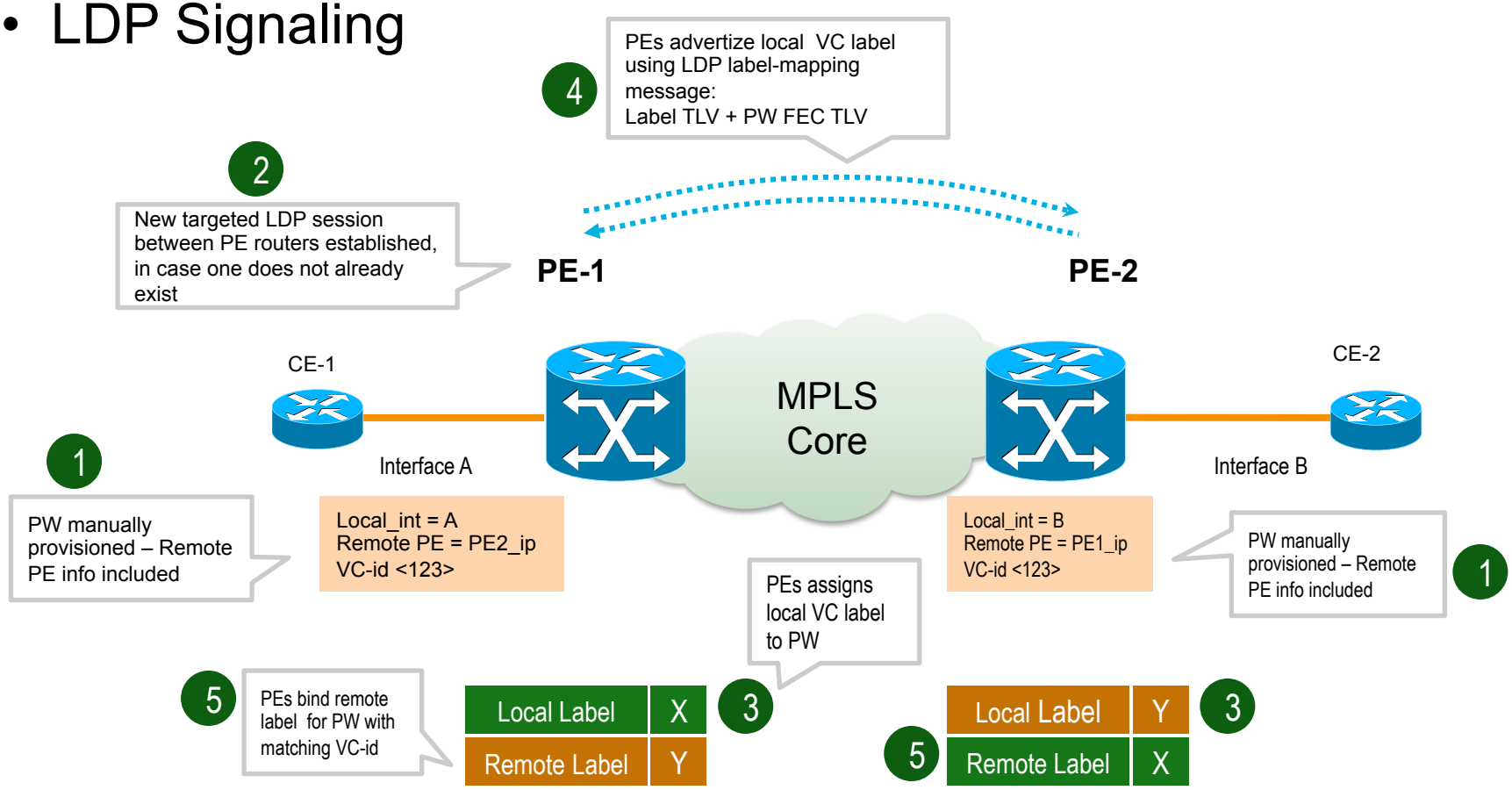


VPLS Signaled with LDP

APNIC

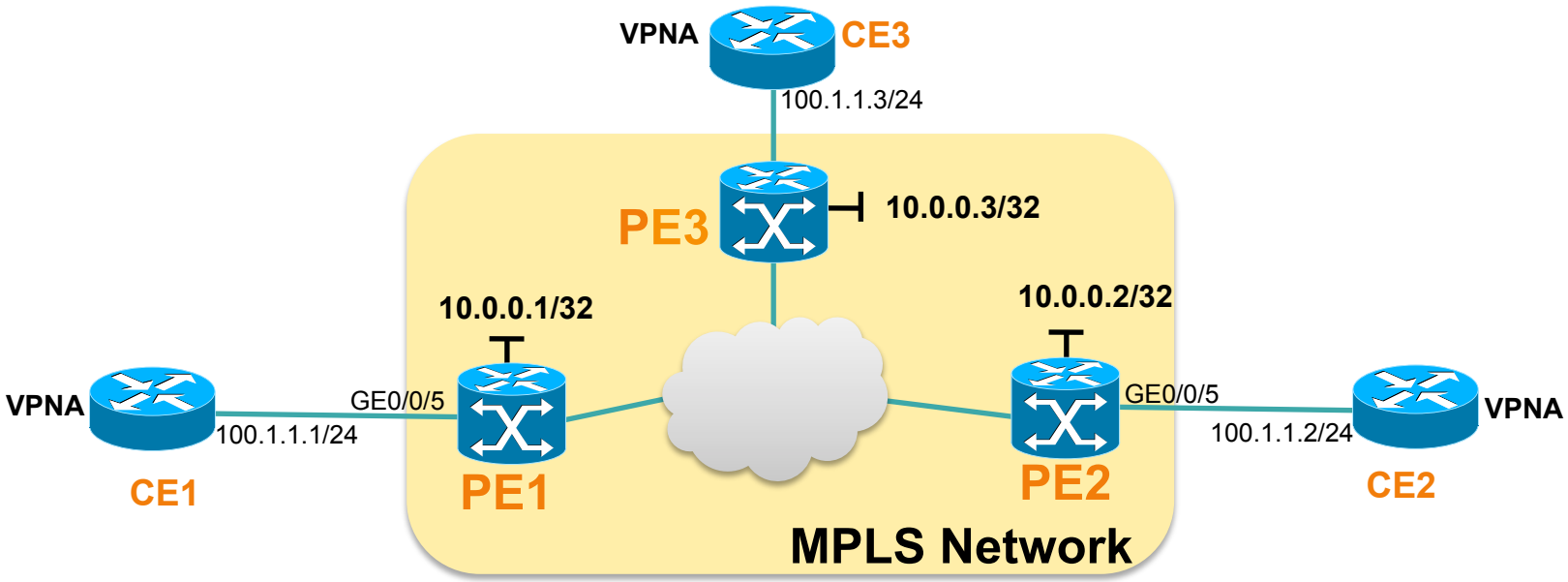
PW Control Plane Operation

- LDP Signaling



Configuration Example of VPLS Signaled with LDP (Manually)

- Task: Configure MPLS VPLS (LDP based)on Cisco IOS XE (Version 3.16) to make the following CEs communication with each other.
- Prerequisite configuration:
  - 1. IP address configuration on all the routers (Including PE & CE)
  - 2. IGP configuration on PE & P routers
  - 3. LDP configuration on PE & P routers



Configure L2 VFI

- Configuration steps:
  - 1. Configure l2 vfi on all the PEs

On PE1:

```
l2 vfi VPLS-CUST1-ETHERNET manual
vpn id 1
bridge-domain 1
neighbor 10.0.0.2 encapsulation mpls
neighbor 10.0.0.3 encapsulation mpls
```

Configure the neighbors manually.

On PE2:

```
l2 vfi VPLS-CUST1-ETHERNET manual
vpn id 1
bridge-domain 1
neighbor 10.0.0.1 encapsulation mpls
neighbor 10.0.0.3 encapsulation mpls
```

Configure L2 VFI (continued)

- Configuration steps:
  - 1. Configure l2 vfi on all the PEs

On PE3:

```
l2 vfi VPLS-CUST1-ETHERNET manual
vpn id 1
bridge-domain 1
neighbor 10.0.0.1 encapsulation mpls
neighbor 10.0.0.2 encapsulation mpls
```

Bridge domain
—A set of logical ports that share the same flooding or broadcast characteristics

Configure Bridge Domain

- Configuration steps:
  - 2. Configure bridge domain under the interface on PE connecting to CE

    Following is the configuration on PE1, similar configurations on the other PEs.

```
interface GigabitEthernet0/0/5
 no ip address
 negotiation auto
 no cdp enable
 service instance 10 ethernet
  encapsulation untagged
  bridge-domain 1
```

Specifies the service instance ID.

Binds a service instance to a bridge domain instance.

Service instance could be considered as a way through which you can use a single port as a combination of layer 2 and layer 3 ports.
Multiple service instances can be created under one physical interface.

Verify LDP Targeted Peers

- After the configuration, verify the results:
  - 1. Check the LDP targeted peers on PEs

```
PE1#show mpls ldp discovery
 Local LDP Identifier:
    10.0.0.1:0

    ....(omitted)

    Targeted Hellos:
           10.0.0.1 -> 10.0.0.2 (ldp): active/passive, xmit/recv
               LDP Id: 10.0.0.2:0
           10.0.0.1 -> 10.0.0.3 (ldp): active/passive, xmit/recv
               LDP Id: 10.0.0.3:0
```

Verify the VC Status

– 2. Check the VC status on PEs

```
PE1#show mpls l2transport vc 1
Local intf     Local circuit                 Dest address    VC ID      Status
-------------  ----------------------------- --------------- ---------- ----------
VFI VPLS-CUST1-ETHERNET  \
               vfi                           10.0.0.2        1          UP
VFI VPLS-CUST1-ETHERNET  \
               vfi                           10.0.0.3        1          UP
```

Verify VFI Information

– 3. Check the vfi information:

```
PE1#show vfi name VPLS-CUST1-ETHERNET
Legend: RT=Route-target, S=Split-horizon, Y=Yes, N=No

VFI name: VPLS-CUST1-ETHERNET, state: up, type: multipoint, signaling: LDP
  VPN ID: 1
  Bridge-Domain 1 attachment circuits:
  Neighbors connected via pseudowires:
  Peer Address     VC ID         S
  10.0.0.2         1             Y
  10.0.0.3         1             Y
```

Verify L2transport Bindings

– 4. Check the l2transport bindings:

```
PE1#show mpls l2transport binding

  Destination Address: 10.0.0.2,VC ID: 1
    Local Label:  1000
        Cbit: 1,    VC Type: Ethernet,    GroupID: n/a
        MTU: 1500,   Interface Desc: n/a
        VCCV: CC Type: CW [1], RA [2]
              CV Type: LSPV [2]
    Remote Label: 702
        Cbit: 1,    VC Type: Ethernet,    GroupID: n/a
        MTU: 1500,   Interface Desc: n/a
        VCCV: CC Type: CW [1], RA [2]
              CV Type: LSPV [2]
  Destination Address: 10.0.0.3,VC ID: 1
    Local Label:  1019
        Cbit: 1,    VC Type: Ethernet,    GroupID: n/a
        MTU: 1500,   Interface Desc: n/a
        VCCV: CC Type: CW [1], RA [2]
              CV Type: LSPV [2]
    Remote Label: 904
        Cbit: 1,    VC Type: Ethernet,    GroupID: n/a
        MTU: 1500,   Interface Desc: n/a
        VCCV: CC Type: CW [1], RA [2]
              CV Type: LSPV [2]
```

Verification of MAC Address Table

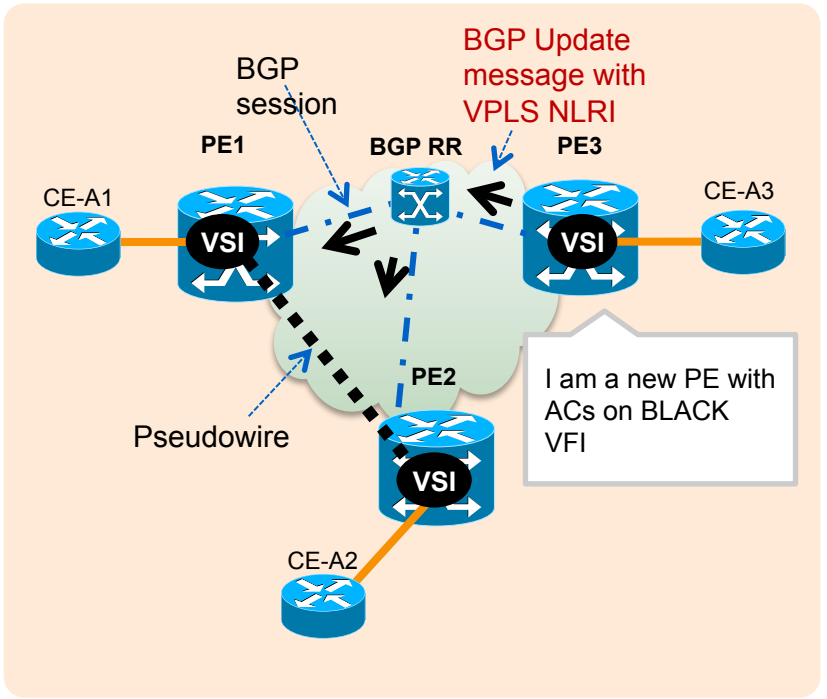
– 5. Check the MAC address table on both PE and CE

```
PE1#show bridge-domain
Bridge-domain 1 (3 ports in all)
State: UP                    Mac learning: Enabled
Aging-Timer: 300 second(s)
Maximum address limit: 16000
    GigabitEthernet0/0/5 service instance 10
    vfi VPLS-CUST1-ETHERNET neighbor 10.0.0.2 1
    vfi VPLS-CUST1-ETHERNET neighbor 10.0.0.3 1
   Nile Mac Address Entries
   BD    mac addr        type     ports
   ------------------------------------------------------------------------------
   1     0042.6856.3805  DYNAMIC  Gi0/0/5.Efp10
   1     0078.88f7.1405  DYNAMIC  10.0.0.2, 1
   1     0078.88f8.fb85  DYNAMIC  10.0.0.3, 1
```

```
CE1#show arp
Protocol  Address          Age (min)  Hardware Addr   Type   Interface
Internet  100.1.1.1              -   0042.6856.3805  ARPA   GigabitEthernet0/0/5
Internet  100.1.1.2              0   0078.88f7.1405  ARPA   GigabitEthernet0/0/5
Internet  100.1.1.3              0   0078.88f8.fb85  ARPA   GigabitEthernet0/0/5
```

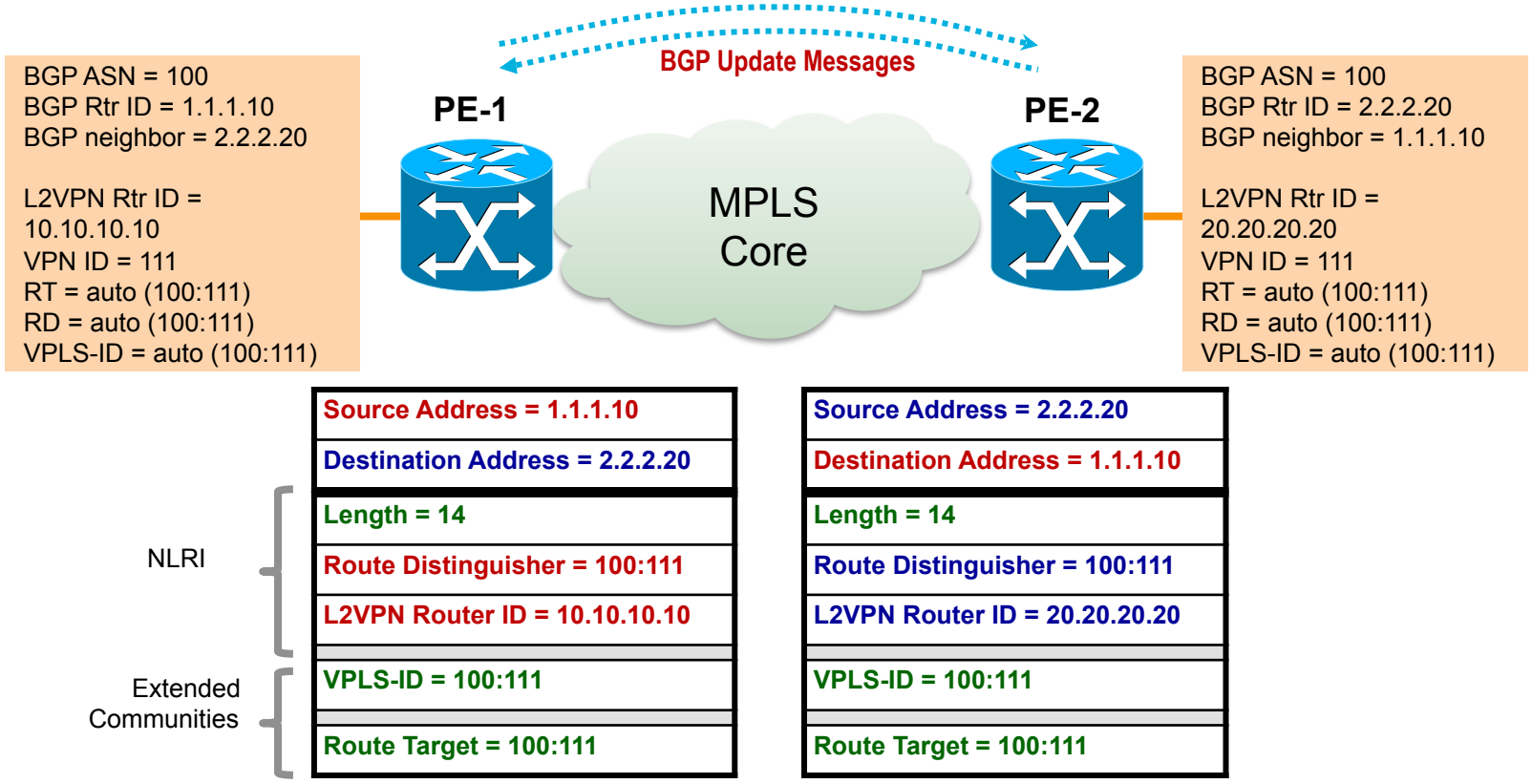
BGP Auto-Discovery (BGP-AD)

- Eliminates need to manually provision VPLS neighbors
- Automatically detects when new PEs are added / removed from the VPLS domain
- Uses BGP Update messages to advertize PE/VFI mapping (VPLS NLRI)
- Typically used in conjunction with BGP Route Reflectors to minimize iBGP full-mesh peering requirements
- Two (2) RFCs define use of BGP for VPLS AD[1]
  - RFC 6074 – when LDP used for PW signaling
  - RFC 4761 – when BGP used for PW signaling



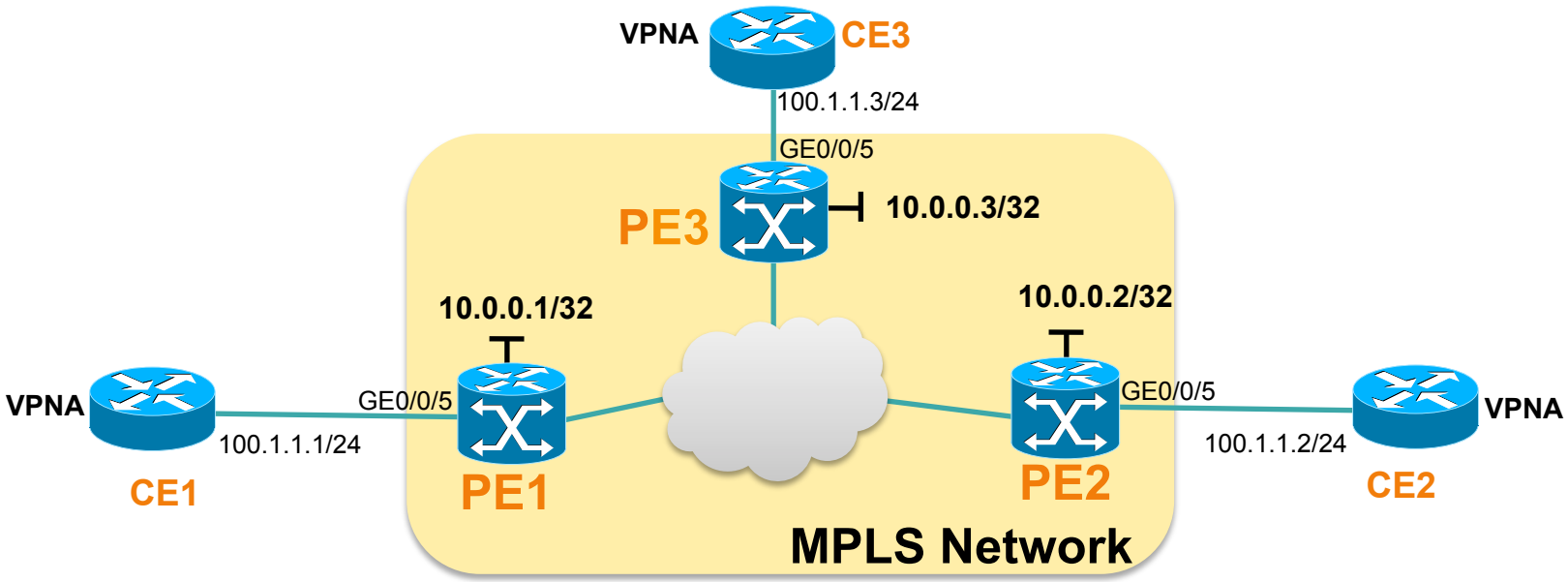
(1) VPLS BGP NLRIs from RFC 6074 and 4761 are different in format and thus not compatible, even though they share same AFI / SAFI values

What is Discovered? NLRI + Extended Communities



Configuration Example of VPLS Signaled with LDP (AD)

- Task: Configure MPLS VPLS (LDP based Autodiscovery)on Cisco IOS XE (Version 3.16) to make the following CEs communication with each other.
- Prerequisite configuration:
  - 1. IP address configuration on all the routers
  - 2. IGP configuration on PE & P routers
  - 3. LDP configuration on PE & P routers



Configure L2 VFI

- Configuration steps:
  - 1. Configure l2 vfi on all the PEs

On PE1(Similar configurations on the other PEs):

```
l2 vfi VPLS-CUST1-ETHERNET autodiscovery
 vpn id 1
 bridge-domain 1
 vpls-id 100:10
 rd 100:10
 route-target export 100:10
 route-target import 100:10
```

Optional commands. VPLS Autodiscovery automatically generates a VPLS ID, an RD, and RT.

Configure L2 VFI (continued)

- Configuration steps:
  - 1. Configure l2 vfi on all the PEs

On PE2:

```
l2 vfi VPLS-CUST1-ETHERNET autodiscovery
 vpn id 1
 bridge-domain 1
 vpls-id 100:10
 rd 100:20
 route-target export 100:10
 route-target import 100:10
```

On PE3:

```
l2 vfi VPLS-CUST1-ETHERNET autodiscovery
 vpn id 1
 bridge-domain 1
 vpls-id 100:10
 rd 100:30
 route-target export 100:10
 route-target import 100:10
```

Configure BGP Neighbors in VPLS

- Configuration steps:
  - 2. Configure BGP neighbors in VPLS on PEs

On PE1, similar configurations on the other PEs:

```
router bgp 100
 neighbor 10.0.0.2 remote-as 100
 neighbor 10.0.0.2 update-source loopback 0
 neighbor 10.0.0.3 remote-as 100
 neighbor 10.0.0.3 update-source loopback 0
 address-family l2vpn vpls
  neighbor 10.0.0.2 activate
  neighbor 10.0.0.2 send-community both
  neighbor 10.0.0.3 activate
  neighbor 10.0.0.3 send-community both
```

Configure Interface in Bridge Domain

- Configuration steps:
  - 3. Configure bridge domain under the interface on PE connecting to CE

On PE1:

```
interface GigabitEthernet0/0/5
 no ip address
 service instance 10 ethernet
 encapsulation untagged
 negotiation auto
 no cdp enable
 bridge-domain 1
```

Specifies the service instance ID.

Binds a service instance to a bridge domain instance.

Verify LDP Targeted Peers

- After the configuration, verify the results:
  - 1. Check the LDP targeted peers on PEs

```
PE1#show mpls ldp discovery

 Local LDP Identifier:
    10.0.0.1:0
    Discovery Sources:
    ......
    Targeted Hellos:  10.0.0.1 -> 10.0.0.2 (ldp): active/passive, xmit/recv
                LDP Id: 10.0.0.2:0
                      10.0.0.1 -> 10.0.0.3 (ldp): active/passive, xmit/recv
                LDP Id: 10.0.0.3:0
```

Verify VC Status

– 2. Check the VC status on PEs

```
PE1#show mpls l2transport vc 1
Local intf     Local circuit                Dest address    VC ID      Status
-------------  ---------------------------- --------------- ---------- ----------
VFI VPLS-CUST1-ETHERNET  \
               vfi                          10.0.0.2        1          UP
VFI VPLS-CUST1-ETHERNET  \
               vfi                          10.0.0.3        1          UP
```

```
PE1#show vfi name VPLS-CUST1-ETHERNET
Legend: RT=Route-target, S=Split-horizon, Y=Yes, N=No

VFI name: VPLS-CUST1-ETHERNET, state: up, type: multipoint, signaling: LDP
  VPN ID: 1, VPLS-ID: 100:10
  RD: 100:10, RT: 100:1,100:10,
  Bridge-Domain 1 attachment circuits:
  Neighbors connected via pseudowires:
  Peer Address     VC ID        Discovered Router ID    S
  10.0.0.2         1            10.0.0.2                Y
  10.0.0.3         1            10.0.0.3                Y
```

Verify BGP VPLS

– 3. Check the BGP VPLS status on PEs

```
PE1#show bgp l2vpn vpls all
BGP table version is 44, local router ID is 10.0.0.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incompleteRPKI validation codes: V valid, I
invalid, N Not found
     Network          Next Hop            Metric LocPrf Weight Path
Route Distinguisher: 100:10
 *>  100:10:10.0.0.1/96
                      0.0.0.0                              32768 ?
Route Distinguisher: 100:20
 *>i 100:20:10.0.0.2/96
                      10.0.0.2                 0    100      0 ?
Route Distinguisher: 100:30
 *>i 100:30:10.0.0.3/96
                      10.0.0.3                 0    100      0 ?
```

Verification – LDP bindings

– 4. Check the l2transport bindings:

```
PE1#show mpls l2transport binding

  Destination Address: 10.0.0.2,VC ID: 1
    Local Label:  911
        Cbit: 1,    VC Type: Ethernet,    GroupID: n/a
        MTU: 1500,   Interface Desc: n/a
        VCCV: CC Type: CW [1], RA [2]
              CV Type: LSPV [2]
    Remote Label: 712
        Cbit: 1,    VC Type: Ethernet,    GroupID: n/a
        MTU: 1500,   Interface Desc: n/a
        VCCV: CC Type: CW [1], RA [2]
              CV Type: LSPV [2]
  Destination Address: 10.0.0.3,VC ID: 1
    Local Label:  906
        Cbit: 1,    VC Type: Ethernet,    GroupID: n/a
       MTU: 1500,   Interface Desc: n/a
        VCCV: CC Type: CW [1], RA [2]
              CV Type: LSPV [2]
    Remote Label: 1239
        Cbit: 1,    VC Type: Ethernet,    GroupID: n/a
        MTU: 1500,   Interface Desc: n/a
        VCCV: CC Type: CW [1], RA [2]
              CV Type: LSPV [2]
```

Verification – MAC Address Table

– 5. Check the MAC address table on both PE and CE

```
PE1#show mac-address-table dynamic bdomain 1
   Nile Mac Address Entries

   BD    mac addr        type     ports
   ------------------------------------------------------------------------------
   1     0078.88f7.1405  DYNAMIC  10.0.0.2, 1
   1     0078.88f8.fb85  DYNAMIC  10.0.0.3, 1
   1     0042.6856.3805  DYNAMIC  Gi0/0/5.Efp10
```

```
CE1#show arp
Protocol  Address          Age (min)  Hardware Addr   Type   Interface
Internet  100.1.1.1              -   0042.6856.3805  ARPA   GigabitEthernet0/0/5
Internet  100.1.1.2              0   0078.88f7.1405  ARPA   GigabitEthernet0/0/5
Internet  100.1.1.3              0   0078.88f8.fb85  ARPA   GigabitEthernet0/0/5
```

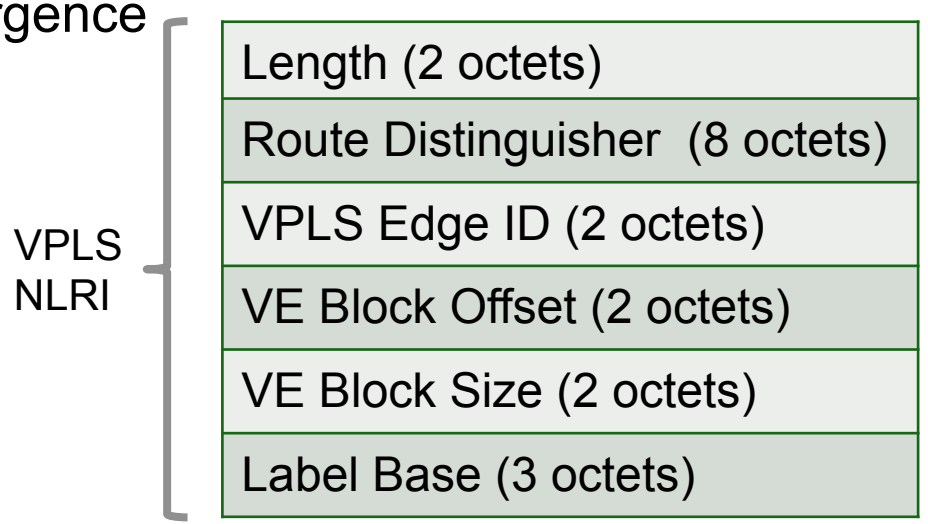
VPLS Signaled with BGP

APNIC

VC Signaled with BGP

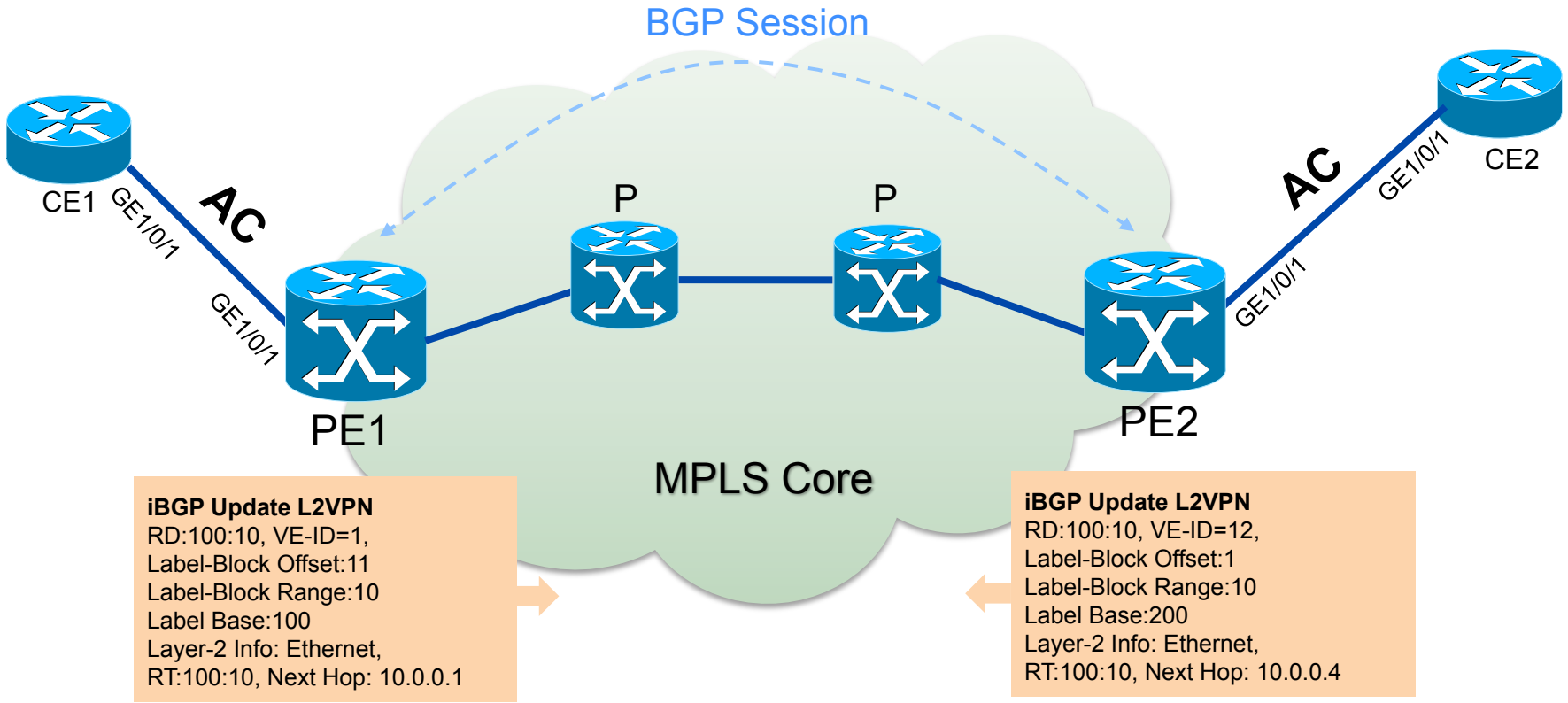
RFC 4761

- BGP is running as the signaling protocol to transmit Layer 2 information and VE labels between PEs.
- BGP was chosen as the means for exchanging L2VPN information for two reasons:
  - It offers mechanisms for both auto-discovery and signaling
  - It allows for operational convergence



VPLS Signaled with BGP

- BGP Signaled VPWS uses **VPN targets** to control the receiving and sending of VPN routes, which improves flexibility of the VPN networking.



VE Label in BGP Signaled VPLS

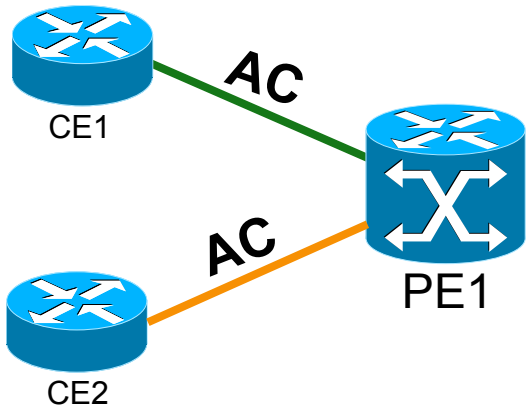
- **VE labels** are assigned through a label block that is pre-allocated for each CE.
- The **size of the label block** determines the number of connections that can be set up between the local CE and other CEs.
- **Additional labels** can be assigned to L2VPNs in the label block for expansion in the future. PEs calculates inner labels according to these label blocks and use the inner labels to transmit packets.

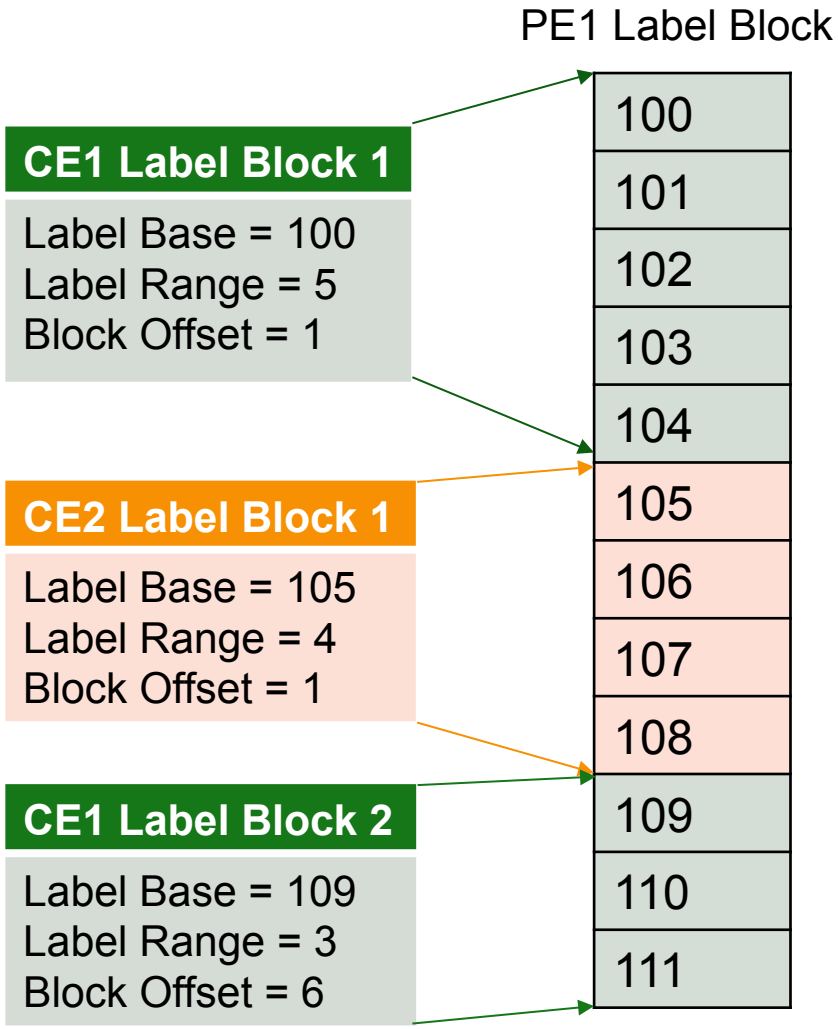
Basic Concepts

Concepts	Explanation
VE ID	A VE ID uniquely identifies a CE in a VPN.
Label Block	A contiguous set of labels.
Label Base	What is the smallest label in one label block?
Label Range	How many labels in one label block?
Block Offset	Value used to identify a label block from which a label value is selected to set up pseudowires for a remote site. Note: In Cisco & Juniper, initial offset is 1. In Huawei, initial offset is 0 by default, can be changed to be 1.

Example of Label Block

- As in the topology, 2 CEs are attached to PE1 to set up L2VPN with other sites.

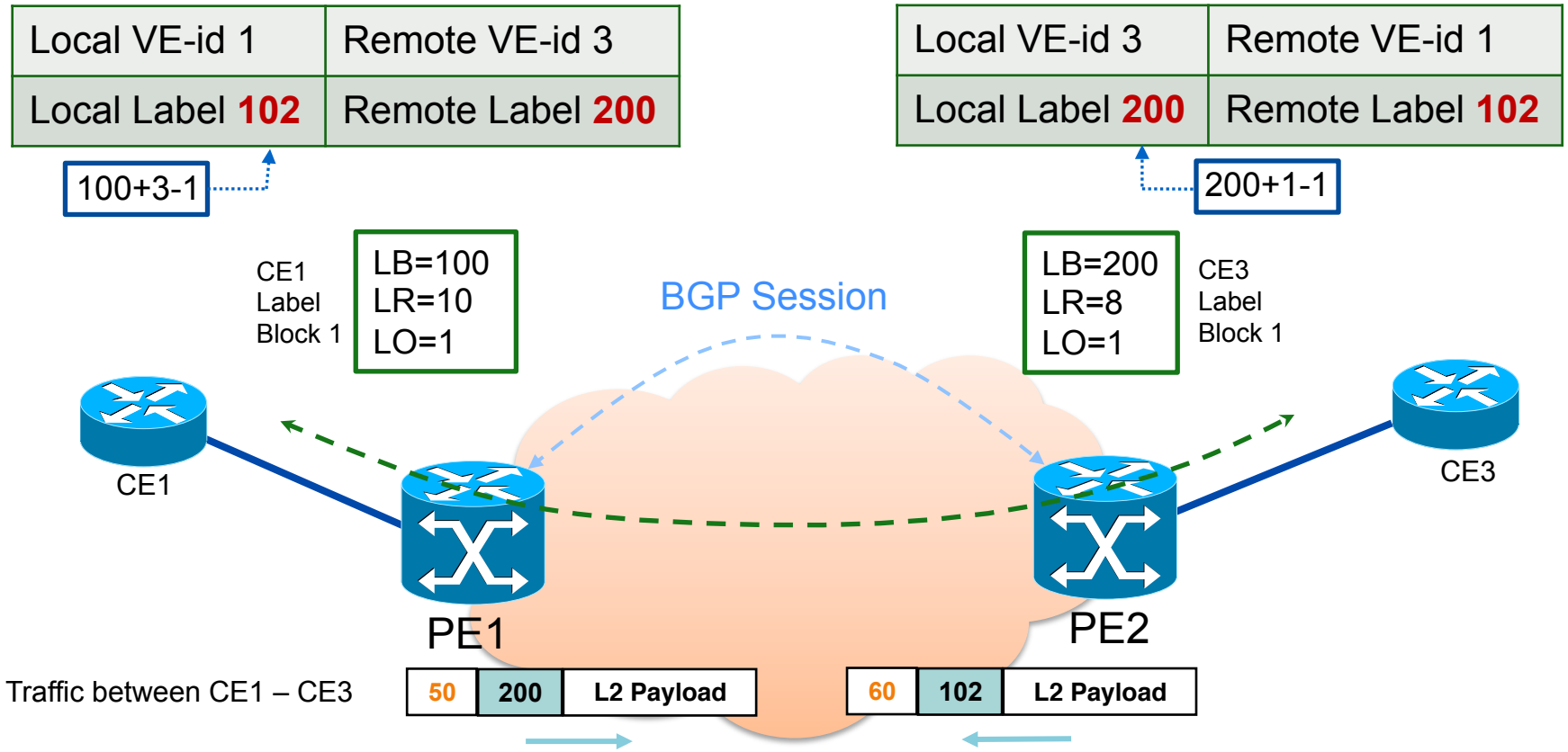




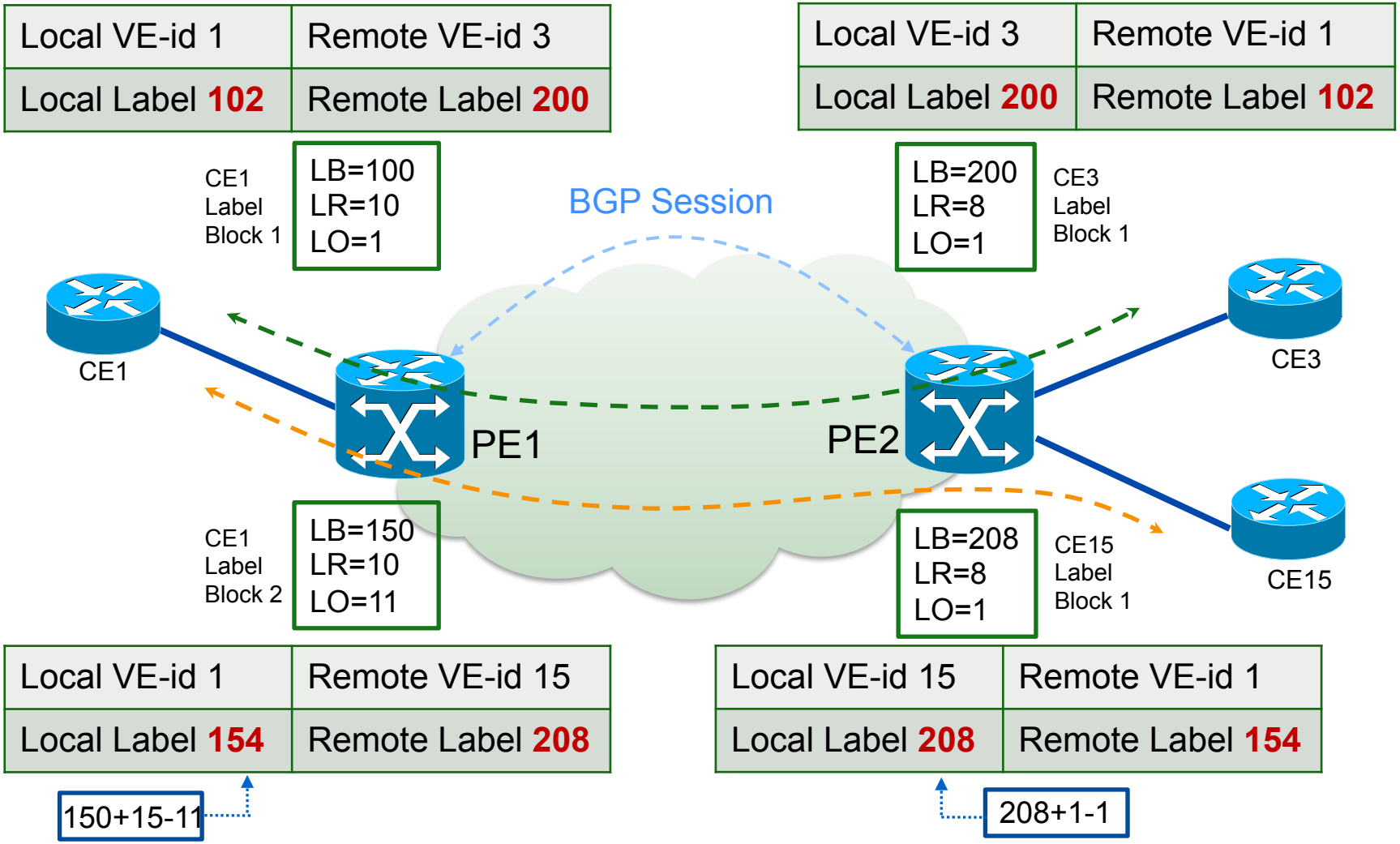
VC Label Calculation

$$Block\ Offset <= Remote\ VE\ ID < Block\ Size + Block Offset$$

$$Label = Label\ Base + Remote\ VE\ ID - Block\ Offset$$

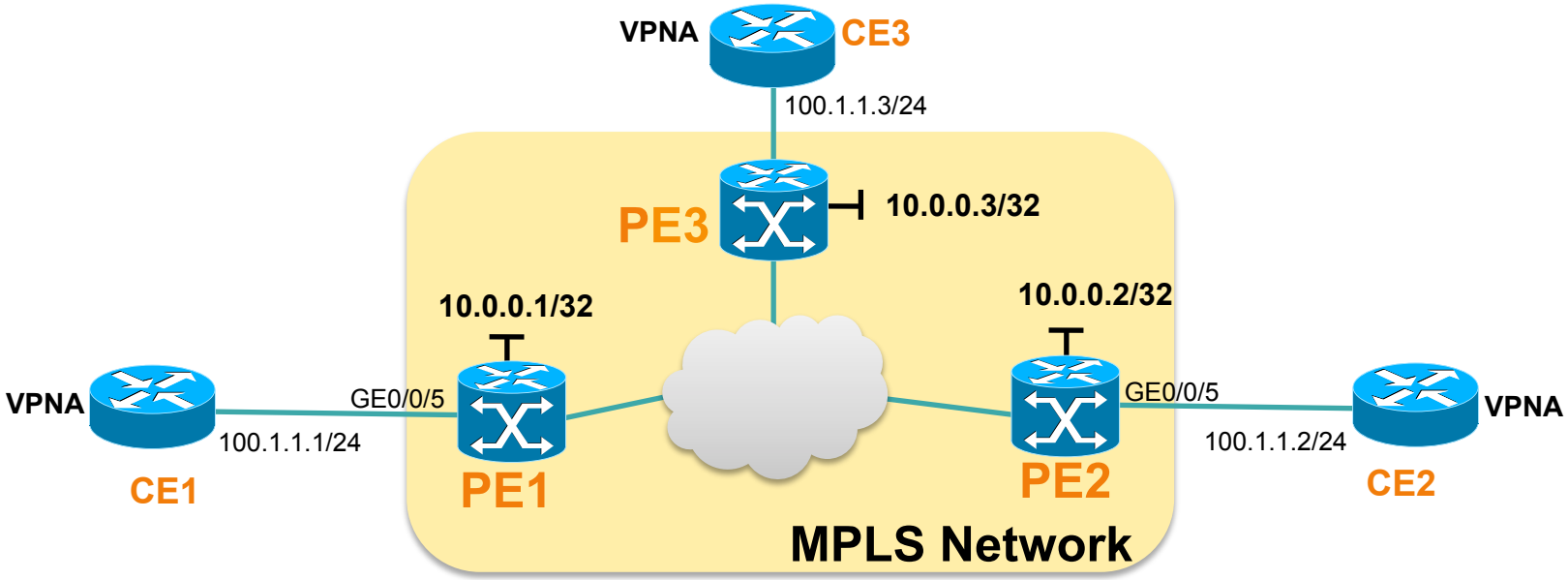


VC Label Calculation



Configuration Example of VPLS Signaled with BGP

- Task: Configure MPLS VPLS (BGP based)on Cisco IOS XE (Version 3.16) to make the following CEs communication with each other.
- Prerequisite configuration:
  - 1. IP address configuration on all the routers
  - 2. IGP configuration on PE & P routers
  - 3. LDP configuration on PE & P routers



Configure L2 VFI

- Configuration steps:
  - 1. Configure l2 vfi on all the PEs

On PE1, similar configurations on other PEs:

```
l2vpn vfi context VPLS-CUST1
 vpn id 200
 autodiscovery bgp signaling bgp
  ve id 1010
  ve range 50
  route-target export 100:20
  route-target import 100:20
```

Specifies VPLS Endpoint Device ID.

Specifies the VE device ID range value, it is label block size.

Specifies RT. Can be generated automatically

Configure L2 VFI (continued)

- Configuration steps:
  - 1. Configure l2 vfi on all the PEs

On PE2

```
l2vpn vfi context VPLS-CUST1
 vpn id 200
 autodiscovery bgp signaling bgp
  ve id 1030
  ve range 50
  route-target export 100:20
  route-target import 100:20
```

On PE3

```
l2vpn vfi context VPLS-CUST1
 vpn id 200
 autodiscovery bgp signaling bgp
  ve id 1040
  ve range 50
  route-target export 100:20
  route-target import 100:20
```

Configure BGP Neighbors in VPLS

- Configuration steps:
  - 2. Configure BGP neighbors in VPLS on PEs

On PE1:

```
router bgp 100
 neighbor 10.0.0.2 remote-as 100
 neighbor 10.0.0.2 update-source loopback 0
 neighbor 10.0.0.3 remote-as 100
 neighbor 10.0.0.3 update-source loopback 0
 address-family l2vpn vpls
  neighbor 10.0.0.2 activate
  neighbor 10.0.0.2 send-community both
  neighbor 10.0.0.2 suppress-signaling-protocol ldp
  neighbor 10.0.0.3 activate
  neighbor 10.0.0.3 send-community both
  neighbor 10.0.0.3 suppress-signaling-protocol ldp
```

Specifies that a communities attribute should be sent to a BGP neighbor.

Suppresses LDP signaling and enables BGP signaling

Configure Interface in Bridge Domain

- Configuration steps:
  - 3. Configure bridge domain under the interface on PE connecting to CE

On PE1:

```
bridge-domain 1
 member Ethernet0/0 service-instance 100
 member vfi VPLS-CUST1

interface GigabitEthernet0/0/5
 no ip address
 service instance 100 ethernet
 encapsulation untagged
 negotiation auto
 no cdp enable
```

Create bridge-domain and add service intance & VFI.

Specifies the service instance ID.

Verify VPLS BGP Signaling

- After the configuration, verify the results:
  - 1. Check the BGP signaling:

```
PE1#show bgp l2vpn vpls all
BGP table version is 68, local router ID is 10.0.0.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found
     Network          Next Hop            Metric LocPrf Weight Path
Route Distinguisher: 100:200
 *>  100:200:VEID-1010:Blk-1000/136
                      0.0.0.0                            32768 ?
 *>i 100:200:VEID-1030:Blk-1000/136
                      10.0.0.2                 0    100      0 ?
 *>i 100:200:VEID-1040:Blk-1050/136
                      10.0.0.3                 0    100      0 ?
```

RD, generated automatically with AS number and VPN ID.

Specifies VEID.

Specifies label block offset.

View Prefixes in Detail

– 2. View the prefix in detail:

```
PE1#show bgp l2vpn vpls rd 100:200 ve-id 1030 block-offset 1000
BGP routing table entry for 100:200:VEID-1030:Blk-1000/136, version 81
Paths: (1 available, best #1, table L2VPN-VPLS-BGP-Table)
  Not advertised to any peer
  Refresh Epoch 2
  Local
    10.0.0.2 (metric 3) from 10.0.0.2 (10.0.0.2)
      Origin incomplete, metric 0, localpref 100, valid, internal, best
      AGI version(0), VE Block Size(50) Label Base(1075)
      Extended Community: RT:100:20 RT:100:200 L2VPN L2:0x0:MTU-1500
      mpls labels in/out exp-null/1075
      rx pathid: 0, tx pathid: 0x0

PE1#show bgp l2vpn vpls rd 100:200 ve-id 1040 block-offset 1000
BGP routing table entry for 100:200:VEID-1040:Blk-1000/136, version 82
Paths: (1 available, best #1, table L2VPN-VPLS-BGP-Table)
  Not advertised to any peer  Refresh Epoch 2
  Local
    10.0.0.3 (metric 1) from 10.0.0.3 (10.0.0.3)
      Origin incomplete, metric 0, localpref 100, valid, internal, best
      AGI version(0), VE Block Size(50) Label Base(925)
      Extended Community: RT:100:20 RT:100:200 L2VPN L2:0x0:MTU-1500
      mpls labels in/out exp-null/925
      rx pathid: 0, tx pathid: 0x0
```

Verify the VFI State

– 3. Check the VFI state:

```
PE3#show bgp l2vpn vpls rd 100:200 ve-id 1010 block-offset 1000
BGP routing table entry for 100:200:VEID-1010:Blk-1000/136, version 74
Paths: (1 available, best #1, table L2VPN-VPLS-BGP-Table)
  Not advertised to any peer  Refresh Epoch 4
  Local
    10.0.0.1 (metric 1) from 10.0.0.1 (10.0.0.1)
      Origin incomplete, metric 0, localpref 100, valid, internal, best
      AGI version(0), VE Block Size(50) Label Base(775)
      Extended Community: RT:100:20 RT:100:200 L2VPN L2:0x0:MTU-1500
      mpls labels in/out exp-null/775
      rx pathid: 0, tx pathid: 0x0
```

```
PE1#show l2vpn vfi name VPLS-CUST1
Legend: RT=Route-target, S=Split-horizon, Y=Yes, N=No
VFI name: VPLS-CUST1, state: up, type: multipoint, signaling: BGP
  VPN ID: 200, VE-ID: 1010, VE-SIZE: 50
  RD: 100:200, RT: 100:200, 100:20,
  Bridge-Domain 2 attachment circuits:
  Pseudo-port interface: pseudowire100024
  Interface          Peer Address    VE-ID  Local Label  Remote Label    S
  pseudowire100035   10.0.0.2        1030   805          1085            Y
  pseudowire100039   10.0.0.3        1040   815          935             Y
```

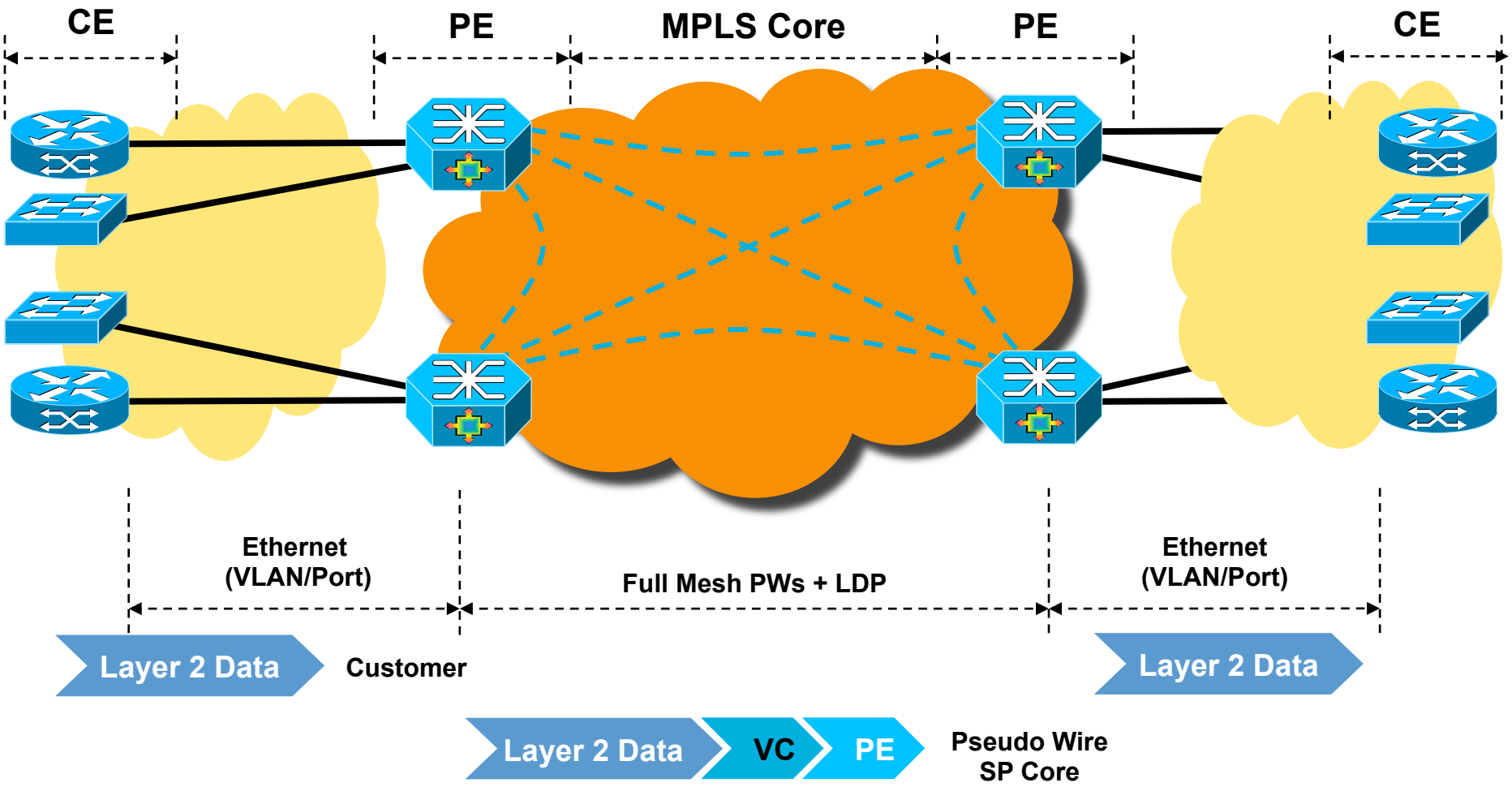
Calculate the Labels

- On PE1:
  - Local label for VE-id 1030 (on PE2):
  - Local label = 775 (Label base) + 1030 (remote VE-id) – 1000 (offset)
    = 805
- On PE1:
  - Local label for VE-id 1040 (on PE3):
  - Local label = 775 (Label base) + 1040 (remote VE-id) – 1000 (offset)
    = 815
- On PE2:
  - Local label for VE-id 1010 (on PE1):
  - Local label = 1075 (Label base) + 1010 (remote VE-id) –1000 (offset)
    = 1085

H-VPLS

APNIC

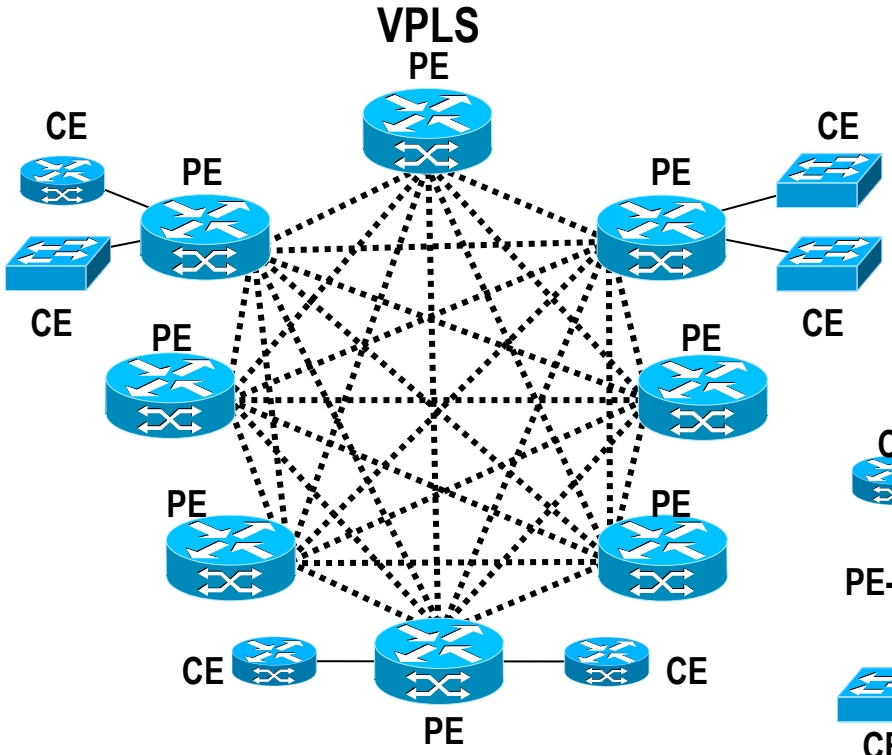
VPLS (Flat Architecture)

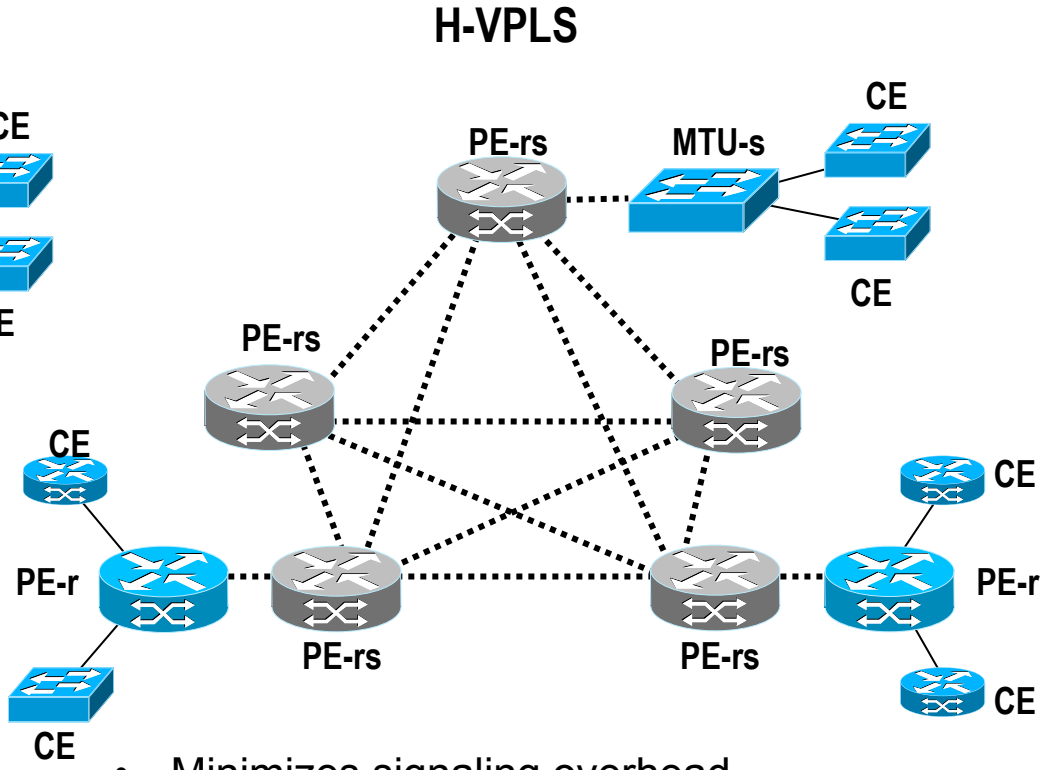


Characteristics of Flat Architecture

- Suitable for simple/small scale network
- Full mesh of directed LDP sessions required
  - N*(N-1)/2 Pseudo Wires required
  - Scalability issue a number of PE routers grows
- No hierarchical scalability
- VLAN and Port level support
- Potential signaling and packet replication overhead
  - Large amount of multicast replication over same physical
  - CPU overhead for replication

Why H-VPLS?





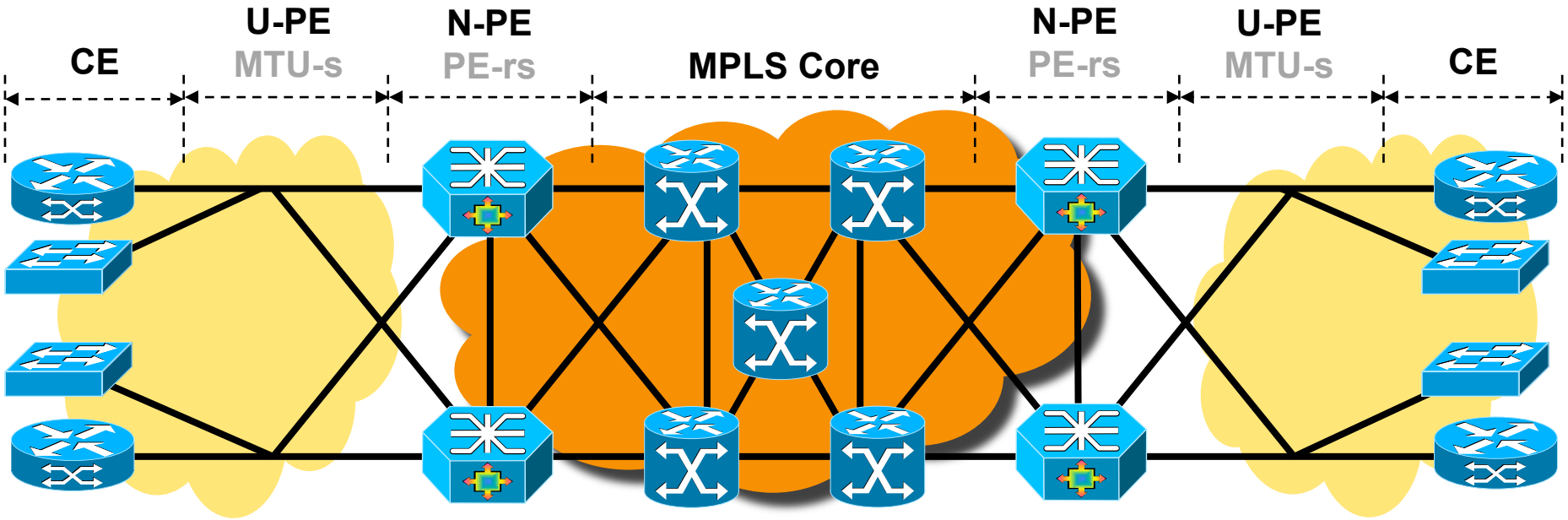
- Potential signaling overhead
- Full PW mesh from the Edge
- Packet replication done at the Edge
- Node Discovery and Provisioning extends end to end

- Minimizes signaling overhead
- Full PW mesh among Core devices
- Packet replication done the Core
- Partitions Node Discovery process

Hierarchical VPLS (H-VPLS)

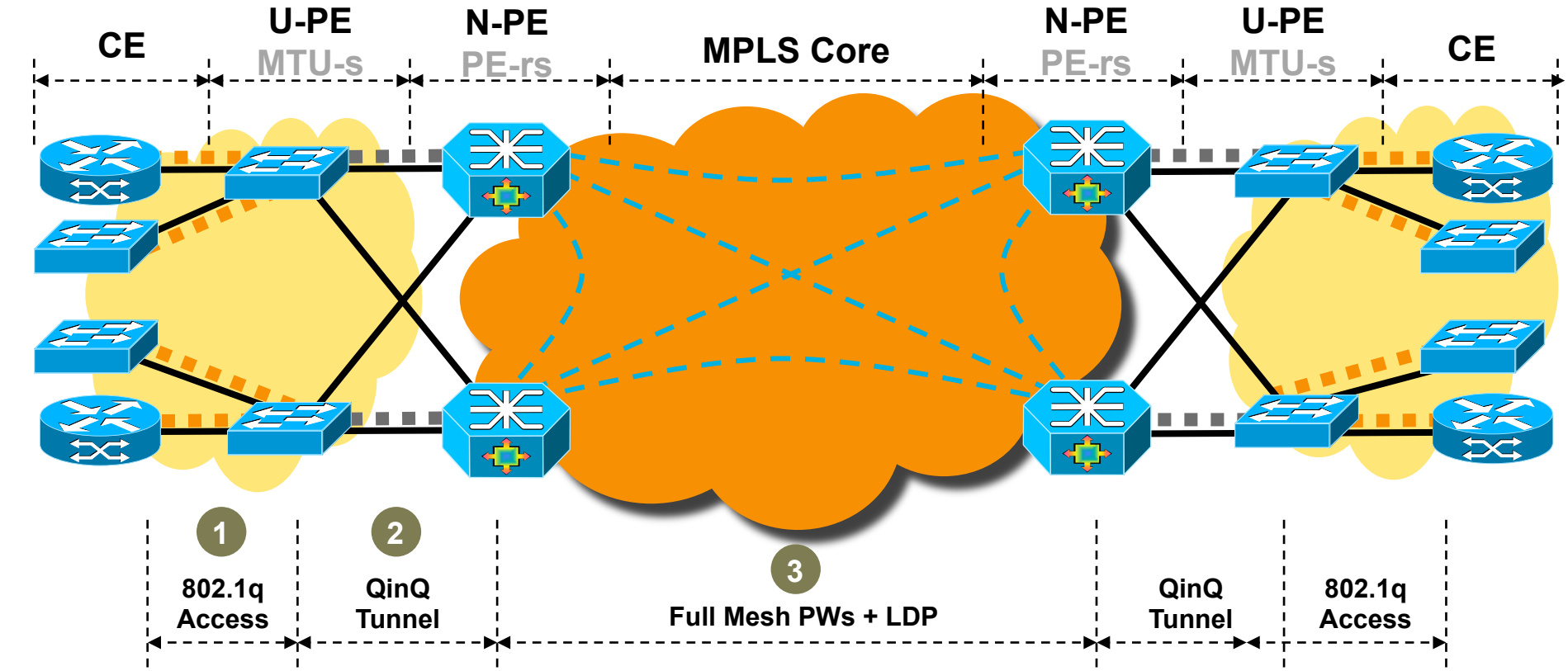
- Best for larger scale deployment
- Reduction in packet replication and signaling overhead
- Consists of two levels in a Hub and Spoke topology
  - Hub consists of full mesh VPLS Pseudo Wires in MPLS core
  - Spokes consist of L2/L3 tunnels connecting to VPLS (Hub) PEs
    - Q-in-Q (L2), MPLS (L3), L2TPv3 (L3)
- Some additional H-VPLS terms
  - **MTU-s** Multi-Tenant Unit Switch capable of bridging (U-PE)
  - **PE-r** Non bridging PE router
  - **PE-rs** Bridging and Routing capable PE

HVPLS Functional Components

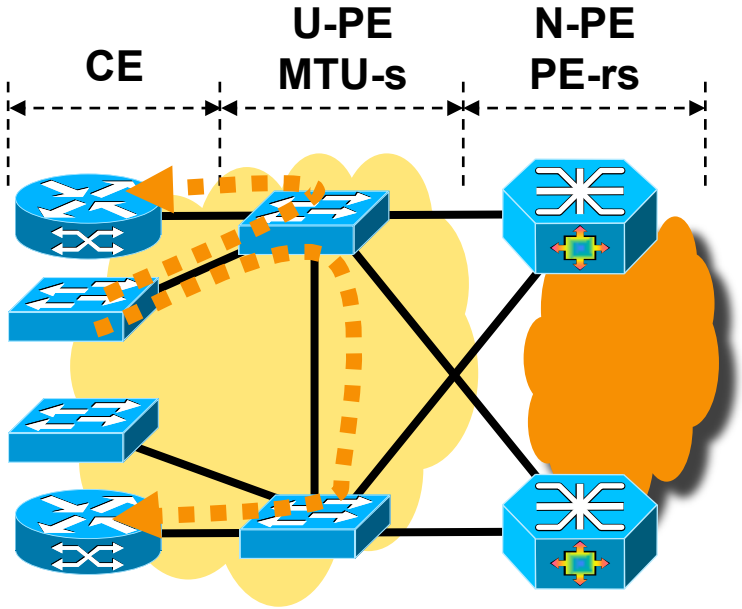


- N-PE: acts as a gateway between MPLS core and edge domain
- U-PE: is used to connect Customer Edge (CE) devices to the service
- CE is the customer device

Ethernet Edge H-VPLS (EE-H-VPLS)

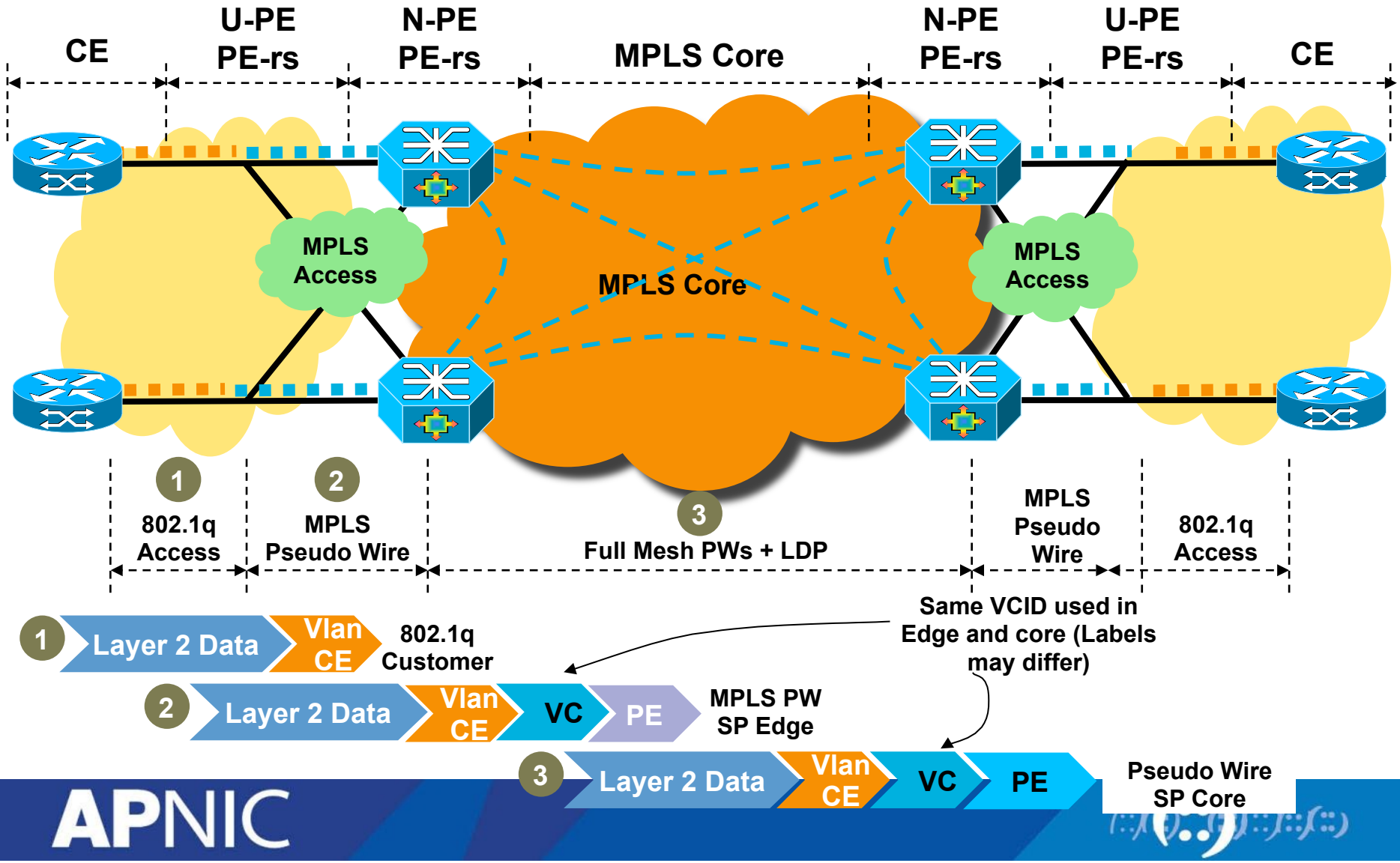


Bridge Capability in EE-H-VPLS

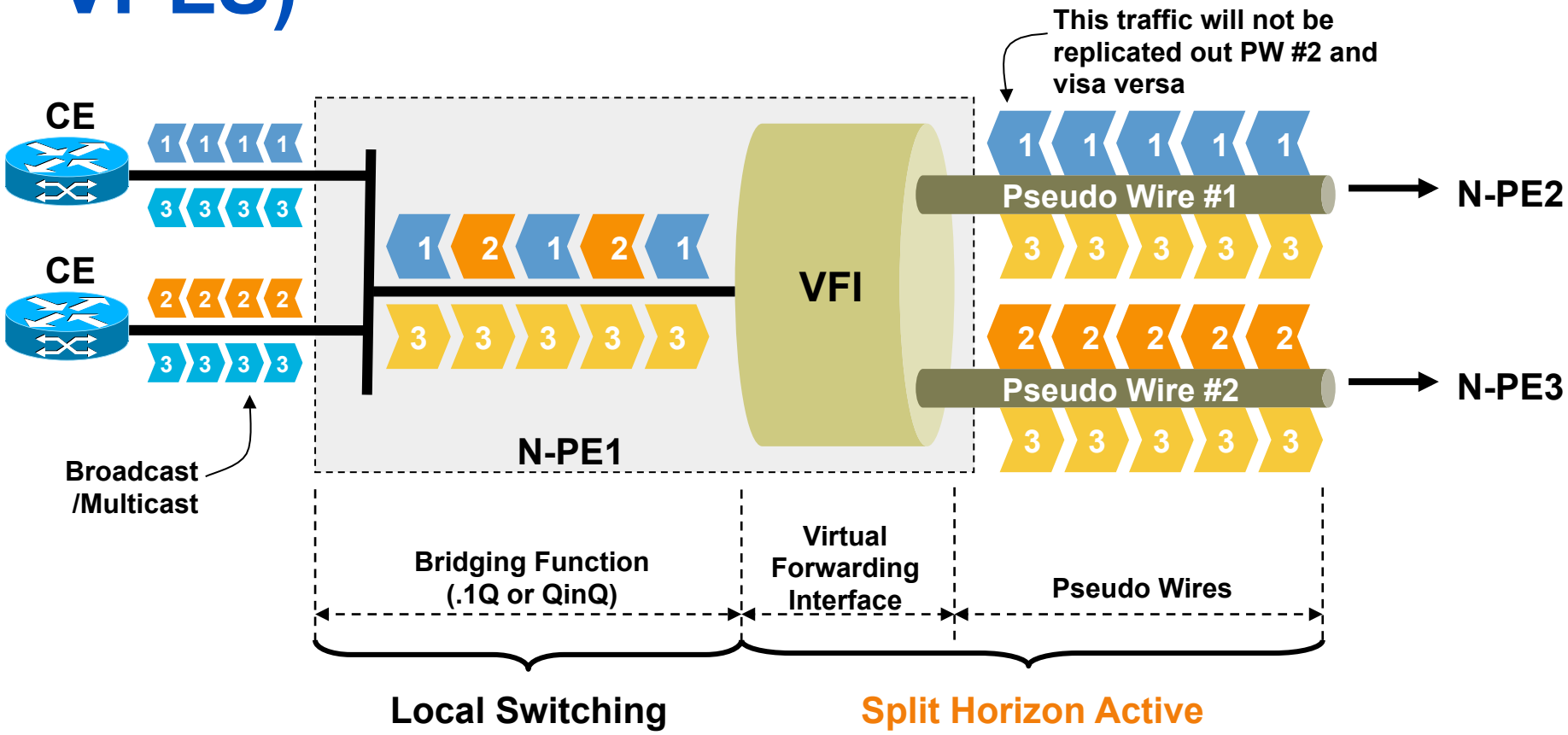


- Local edge traffic does not have to traverse N-PE
  - MTU-s can switch traffic locally
  - Saves bandwidth capacity on circuits to N-PE

MPLS Edge H-VPLS

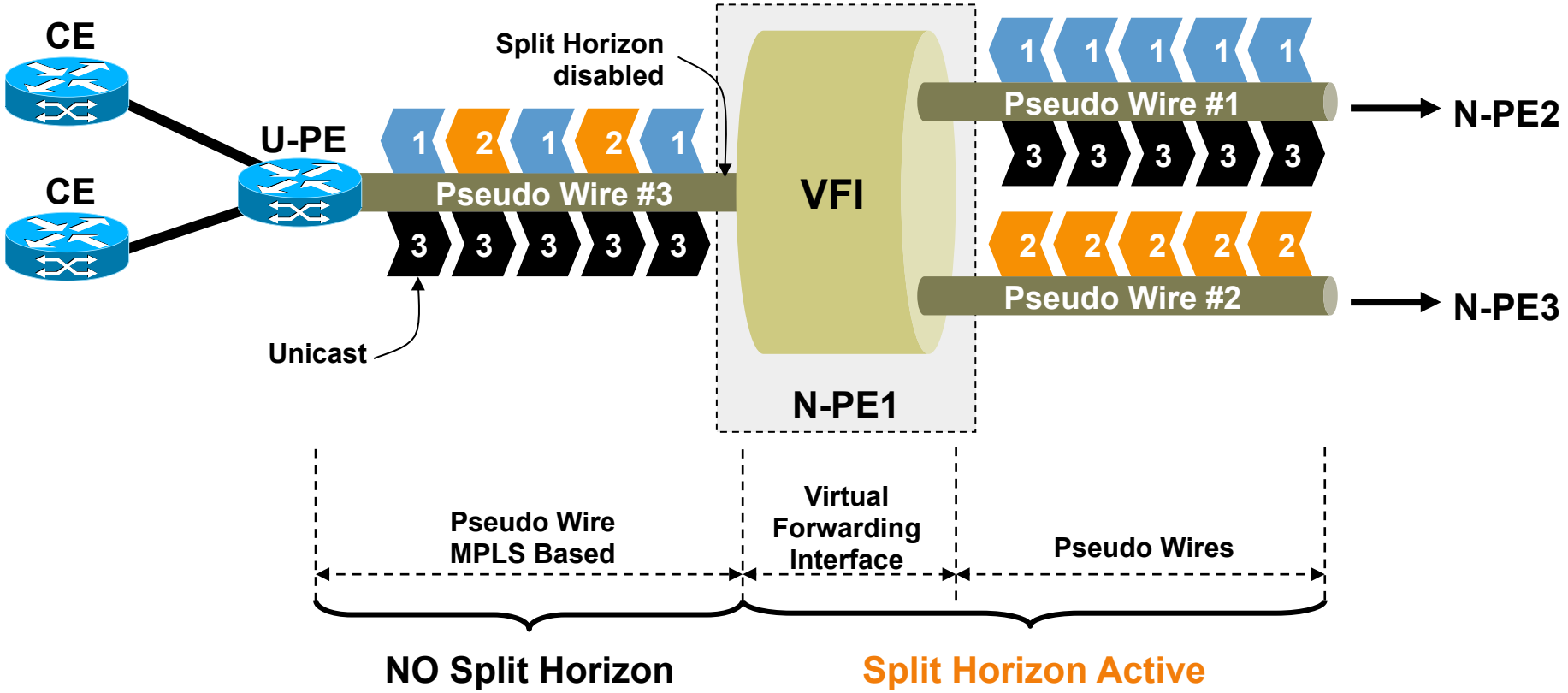


VFI and Split Horizon (VPLS, EE-H-VPLS)



- Virtual Forwarding Interface is the VSI representation in IOS
  - Single interface terminates all PWs for that VPLS instance
  - This model applicable in direct attach and H-VPLS with Ethernet Edge

VFI and NO Split Horizon (ME-H-VPLS)



- This model applicable H-VPLS with MPLS Edge
  - PW #1, PW #2 will forward traffic to PW #3 (non split horizon port)

Questions?

Issue Date:

Revision:

APNIC

Deploy MPLS Traffic Engineering

APNIC

Acknowledgement

- Cisco Systems

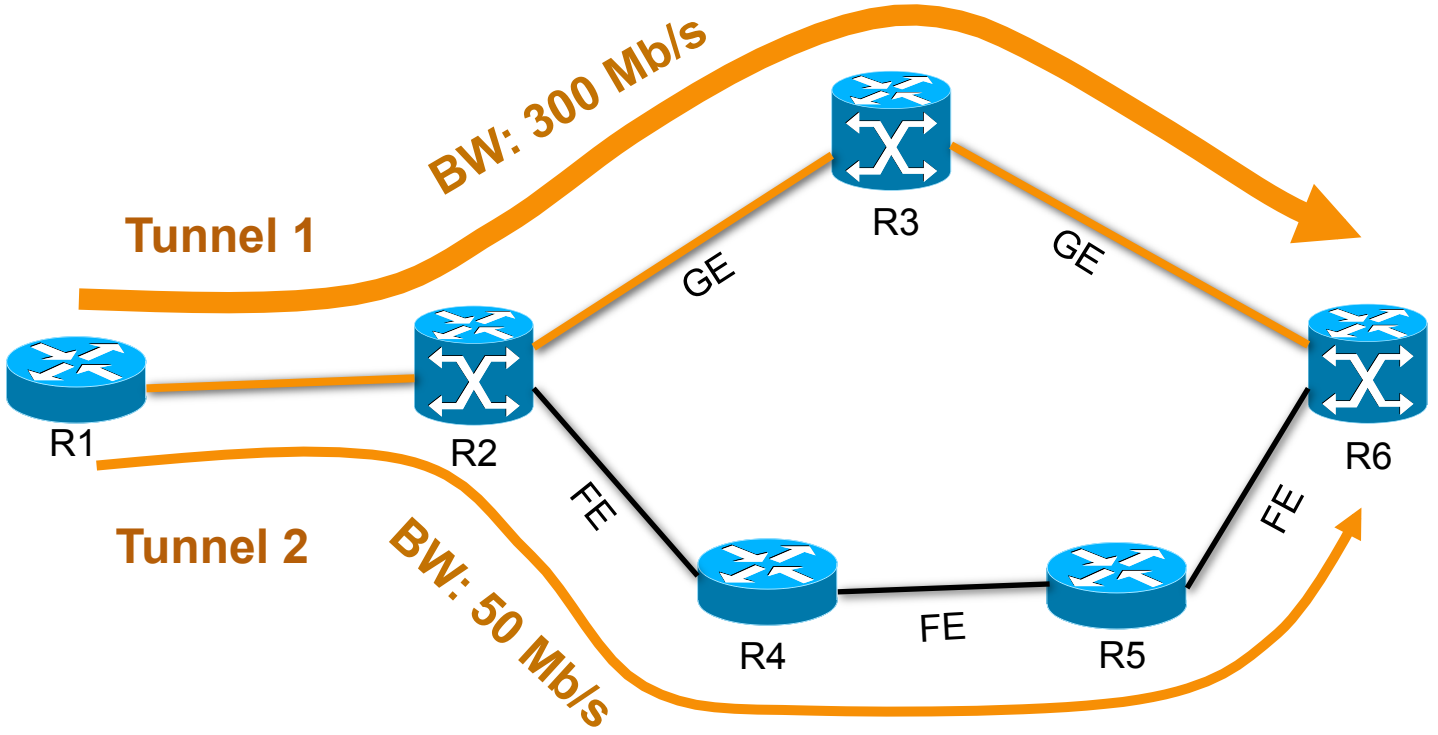
Overview of MPLS TE

APNIC

Why MPLS Traffic Engineering?

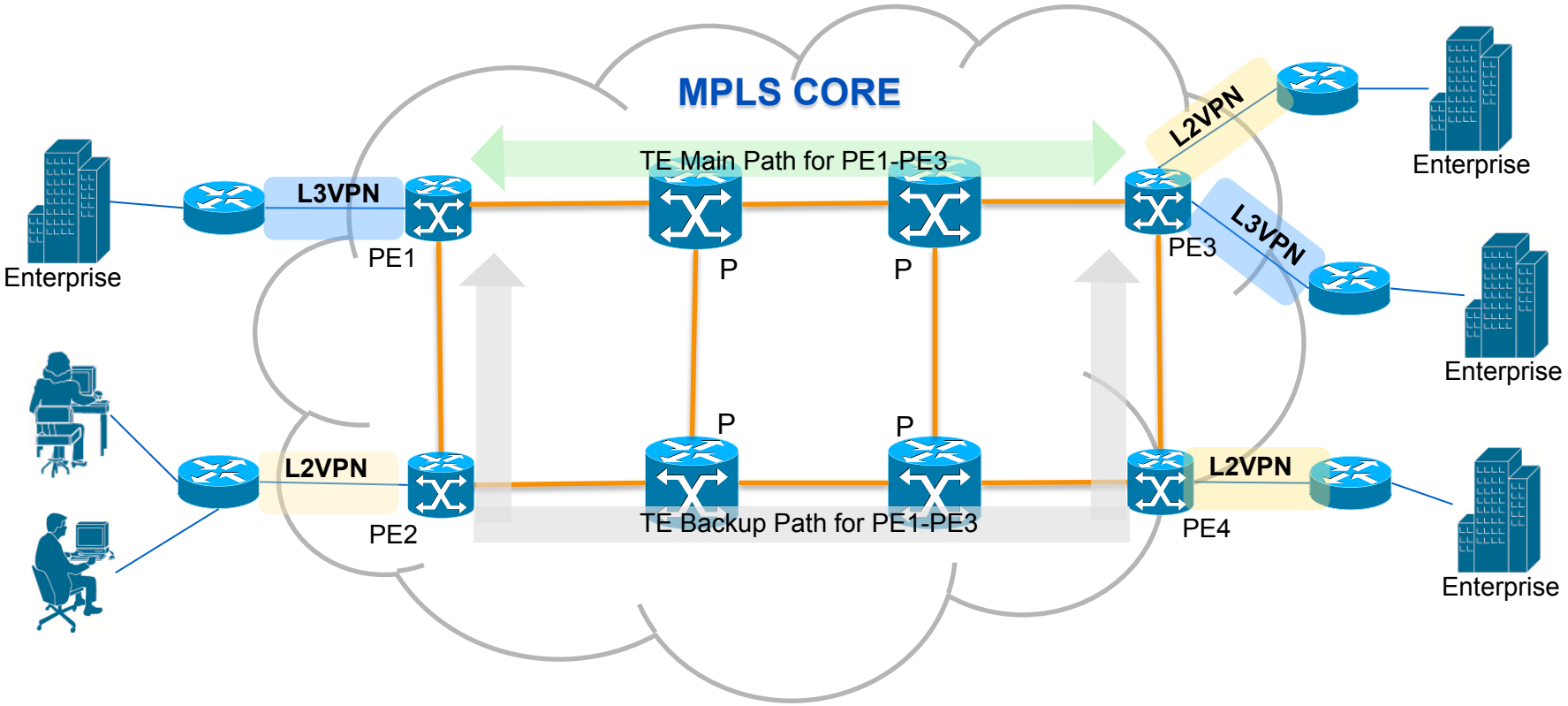
- Handling unexpected congestion
- Better utilization of available bandwidth
- Route around failed links/nodes
- Capacity planning

Optimal Traffic Engineering



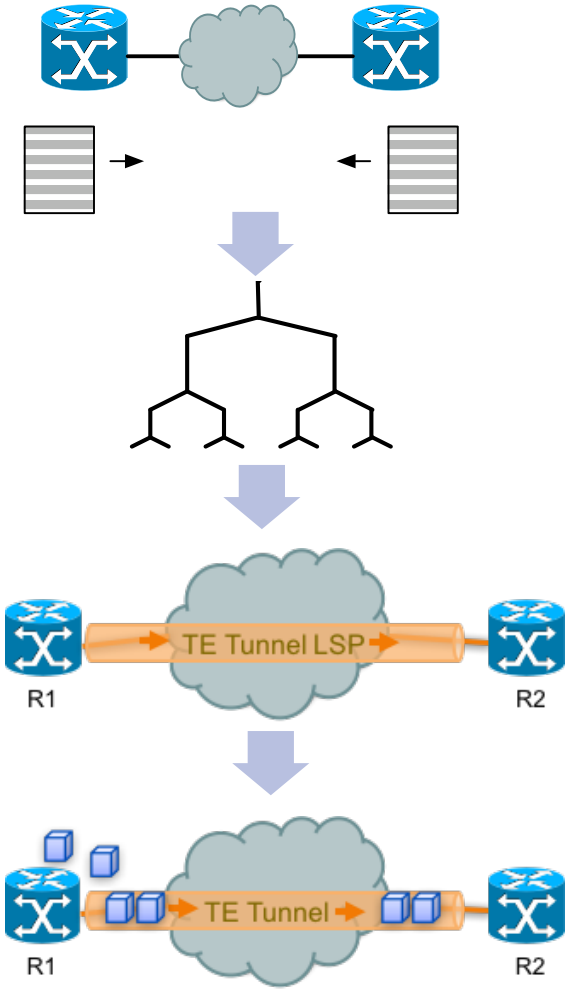
IP TE	MPLS TE
Shortest path	Determines the path at the source based on additional parameters (available resources and constraints, etc.)
Equal cost load balancing	Load sharing across unequal paths can be achieved.

MPLS Application Scenario

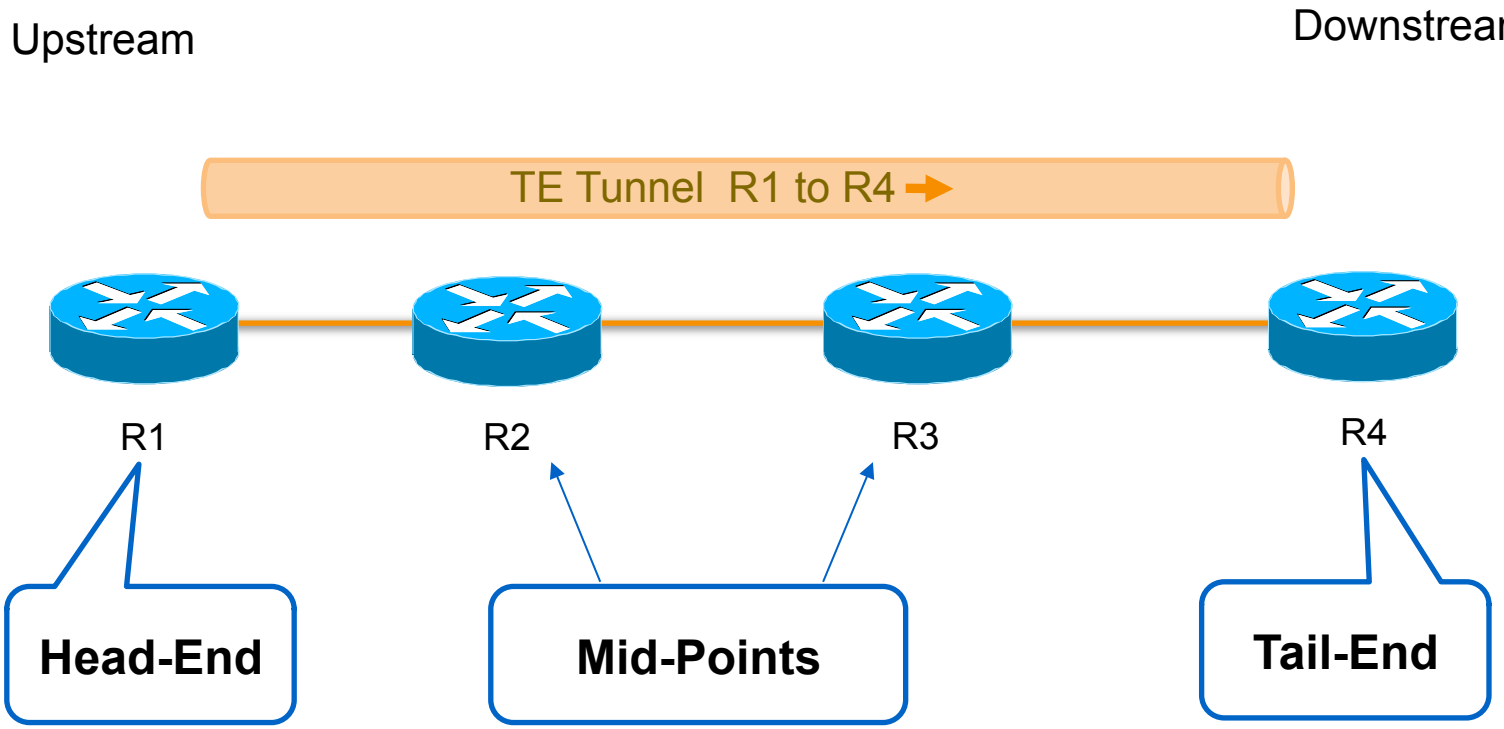


How MPLS TE Works

- **Information Distribution**
  - What is the information?
- **Path Calculation**
  - Dynamically
  - Manually
- **Path Setup**
  - RSVP-TE
  - (CR-LDP)
- **Forward the Traffic Down to the Tunnel**
  - Autoroute
  - Static
  - Policy



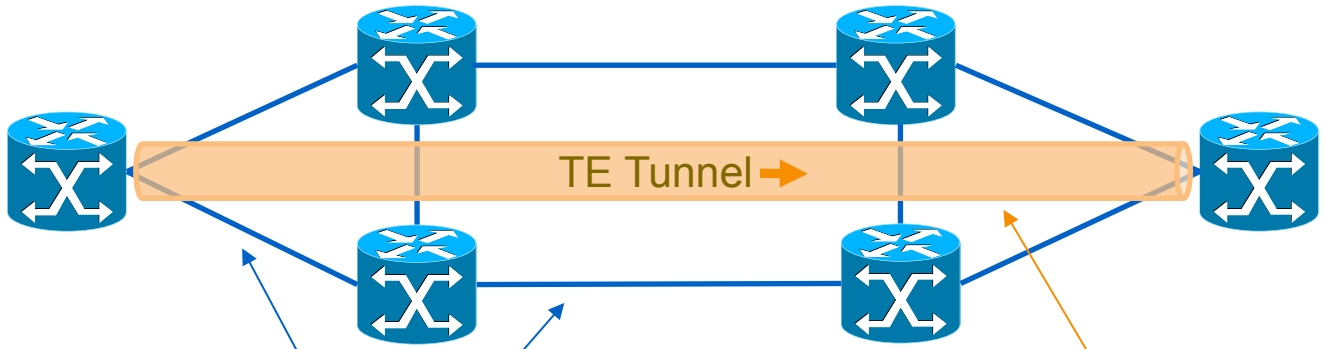
Terminology—Head, Tail, LSP



Information Distribution

APNIC

Attributes



Link Attributes

- Available Bandwidth
- Attribute flags (Link Affinity)
- Administrative weight (TE-specific link metric)

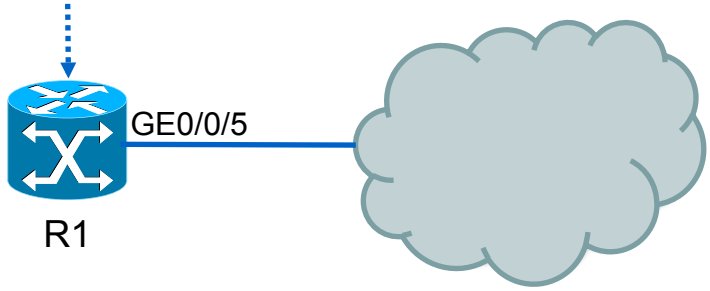
Tunnel Attributes

- Tunnel Required Bandwidth
- Tunnel Affinity & Mask
- Priority

Bandwidth on Physical Link

- Bandwidth – the amount of **reservable bandwidth** on a physical link

```
R1(Config)# interface gigabitethernet 0/0/5
R1(config-if)# mpls traffic-eng tunnels
R1(config-if)# ip rsvp bandwidth 512
```



Reserved bandwidth is 512 kbps

Bandwidth Required by Tunnel

- **Bandwidth required** by the tunnel across the network

```
R1(config)# interface tunnel 1
R1(config-if)# tunnel mpls traffic-eng bandwidth 100
```

- Not a mandatory command. If not configured, tunnel is requested with zero bandwidth.

Priority

- Each tunnel has 2 priorities:
  - Setup priority
  - Holding priority
- Value: 0~7, 0 indicated the highest priority, 7 indicates the lowest priority.

TE Tunnel 1 R1 to R4 →

BW=200 kbps Priority: S_1=3 **H_1=3**

$S_2 < H_1$

TE Tunnel 2 R1 to R3 →

BW=200 kbps Priority: **S_2=2** H_2=2

Priority

- Each tunnel has 2 priorities:
  - Setup priority
  - Holding priority
- Value: 0~7, 0 indicated the highest priority, 7 indicates the lowest priority.

TE Tunnel 1 R1 to R4 → Torn down

BW=200 kbps Priority: S_1=3 **H_1=3**

TE Tunnel 2 R1 to R3 → Set up

BW=200 kbps Priority: **S_2=2** H_2=2

$S_2 < H_1$
Tunnel 2 preempts the BW, tunnel 1 has been torn down.

Configure Priority

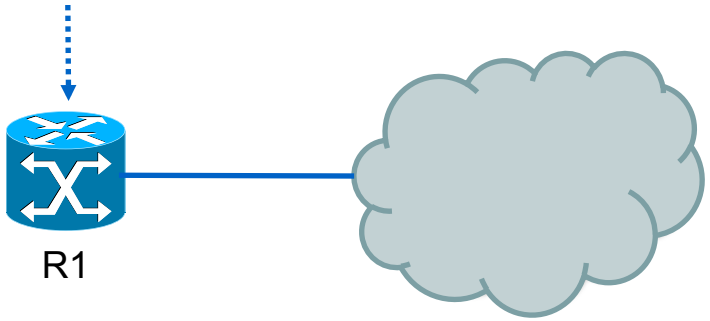
```
R1(config)# interface tunnel 1
R1(config-if)# tunnel mpls traffic-eng priority 3 3
```

- Recommended that priority S=H; if a tunnel can setup at priority "X", then it should be able to hold at priority "X" too!
- Configuring S > H is illegal; tunnel will most likely be preempted
- Default is S = 7, H = 7

Attribute Flags

- An **attribute flag** is a 32-bit bitmap on a link that indicate the existence of up to 32 separate properties on that link, also called link affinity or administrative group.

```
R1(config)# interface ethernet 0/1
R1(config-if)# mpls traffic-eng attribute-flags 0x8
```



Tunnel Affinity & Mask

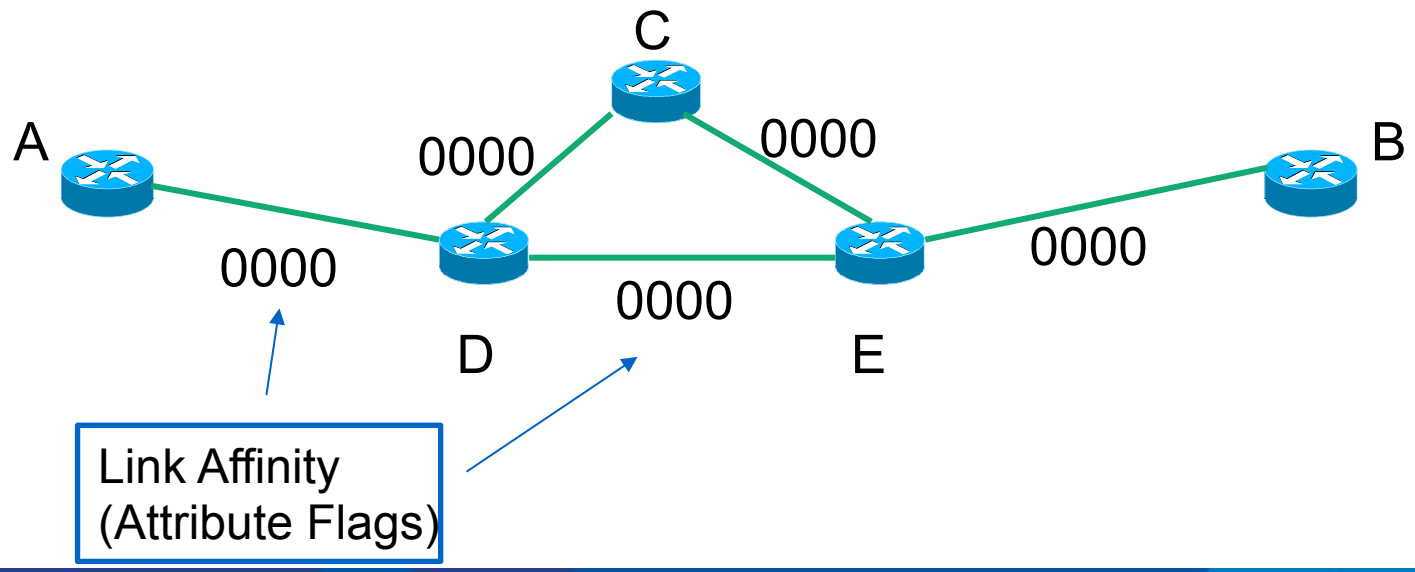
- **Tunnel affinity** helps select which tunnels will go over which links.
- For example: a network with OC-12 and Satellite links will use affinities to prevent tunnels with VoIP traffic from taking the satellite links.

Tunnel can only go over a link if
(Tunnel Mask) && (Attribute Flags) == Tunnel Affinity

```
R1(config)# interface tunnel 1
R1(config-if)# tunnel mpls traffic-eng affinity 0x80 mask 0x80
```

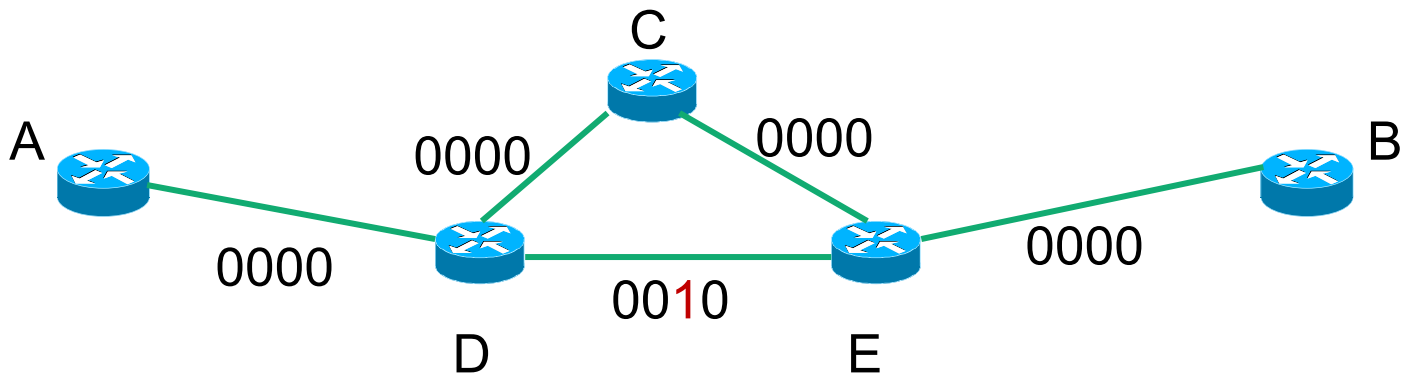
Example0: 4-bit string, default

- Traffic from A to B:
  - tunnel affinity= 0000, t-mask = 0011
- ADEB and ADCEB are possible



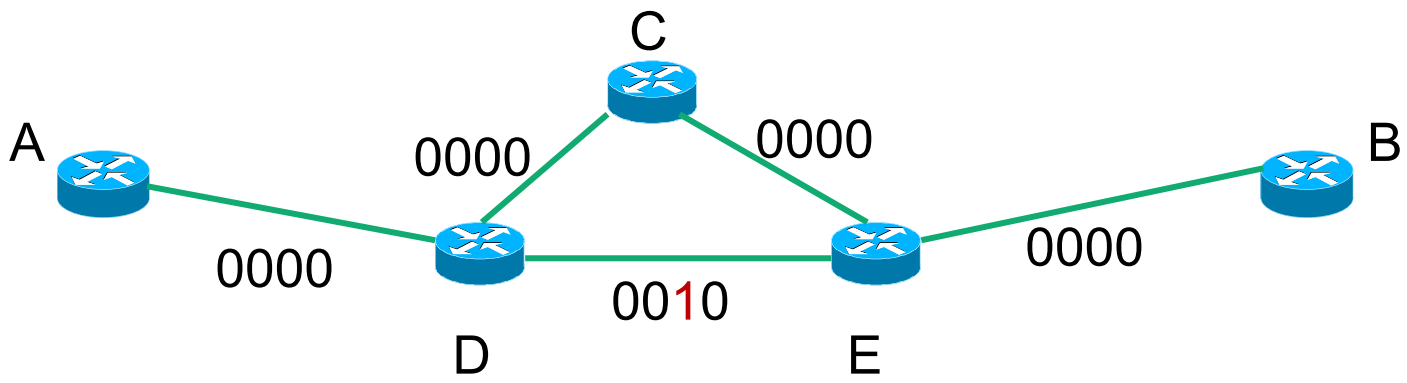
Example1a: 4-bit string

- Setting a link bit in the lower half drives all tunnels off the link, except those specially configured
- Traffic from A to B:
  - tunnel = 0000, t-mask = 0011
- Only ADCEB is possible



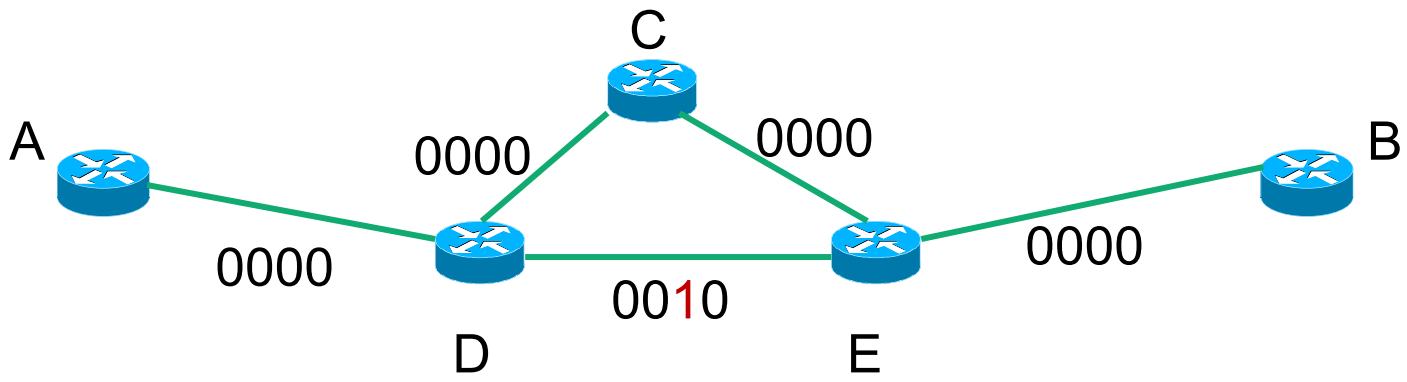
Example1b: 4-bit string

- A specific tunnel can then be configured to allow such links by clearing the bit in its affinity attribute mask
- Traffic from A to B:
  - tunnel = 0000, t-mask = 0001
- Again, ADEB and ADCEB are possible



Example1c: 4-bit string

- A specific tunnel can be restricted to only such links by instead turning on the bit in its affinity attribute bits
- Traffic from A to B:
  - tunnel = 0010, t-mask = 0011
- No path is possible



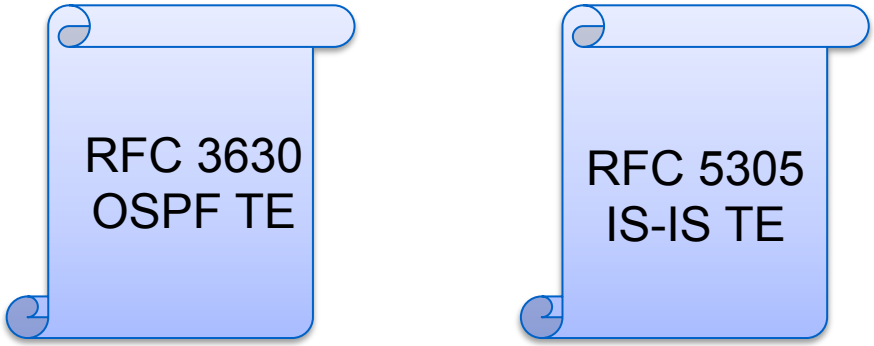
Administrative Weight (TE Metric)

- Two costs are associated with a link
  - TE cost (Administrative weight)
  - IGP cost
- The default TE cost on a link is the same as the IGP cost. We can also configure as following:

```
Router(config)# interface ethernet 0/1
Router(config-if)# mpls traffic-eng administrative-weight 20
```

Link-State Protocol Extensions/ IGP Flooding

- TE finds paths other than shortest-cost. To do this, TE must have more info than just per-link cost
- OSPF and IS-IS have been extended to carry additional information
  - Physical bandwidth
  - RSVP configured bandwidth
  - RSVP available bandwidth
  - Link TE metric
  - Link affinity



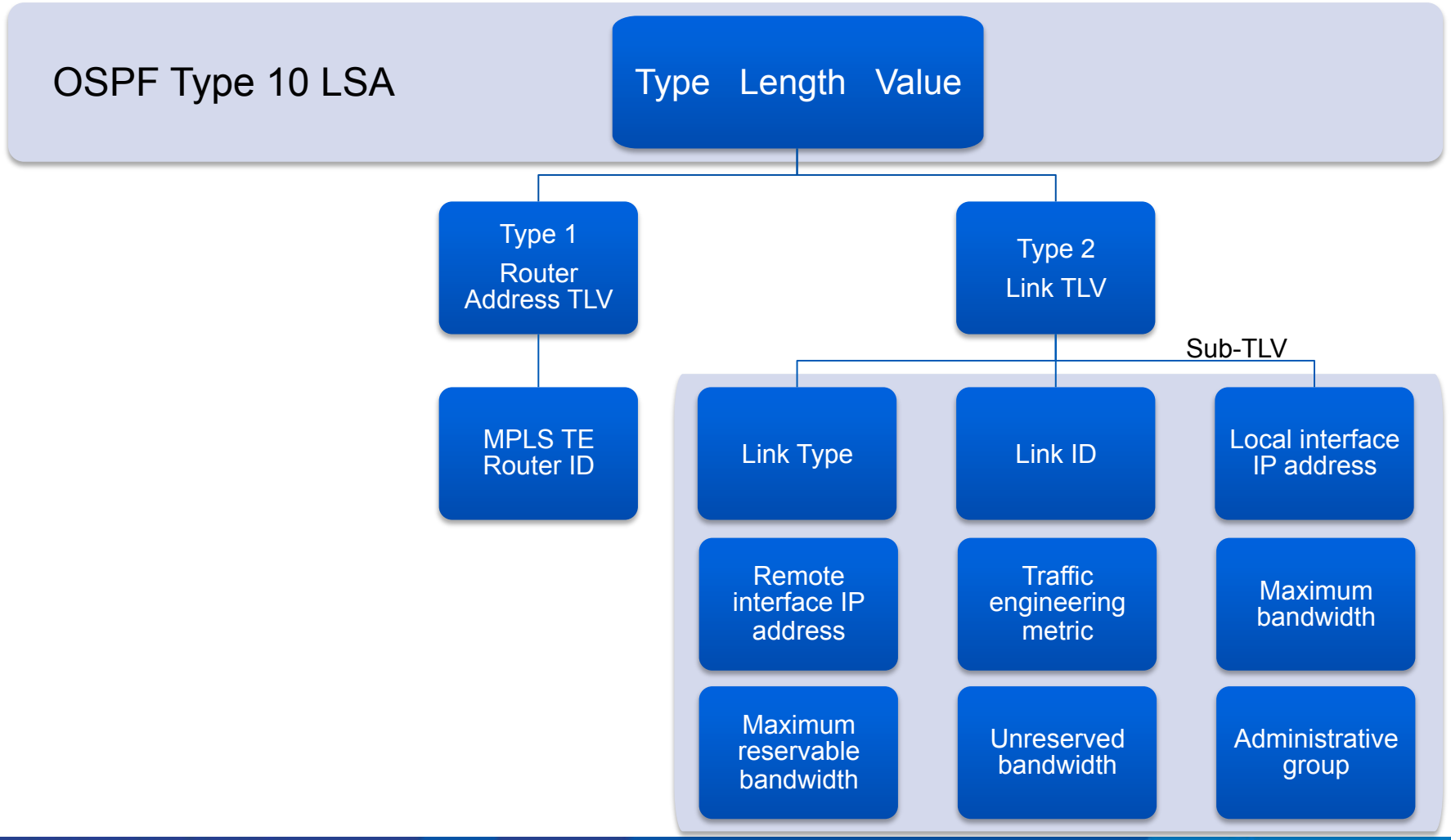
OSPF Extensions

- OSPF
  - Uses Type 10 (Opaque Area-Local) LSAs

- Enable OSPF TE:

```
Router(config)# router ospf 100
Router(config-router)# mpls traffic-eng router-id loopback 0
Router(config-router)# mpls traffic-eng area 0
```

OSPF Type 10 LSA



IS-IS Extensions

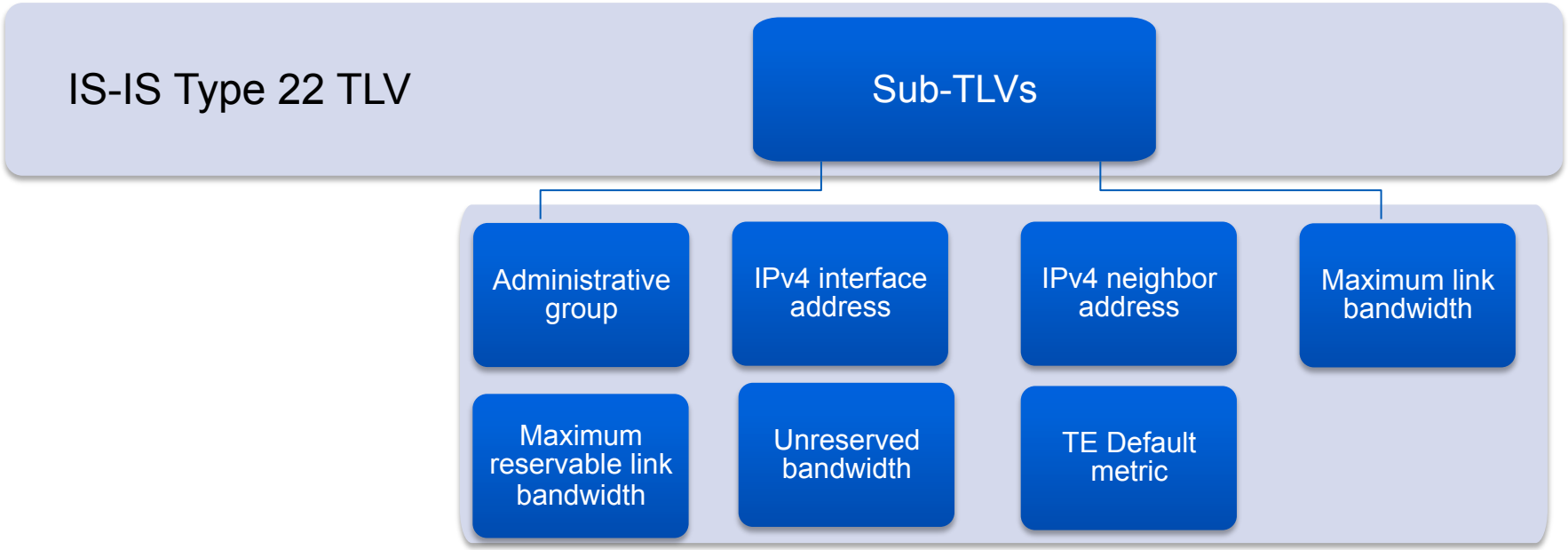
- IS-IS
  - Uses Type 22 TLVs to carrry MPLS TE link information
  - Also uses Type 134 and Type 135 TLV

- Enable IS-IS TE

```
Router(config)# router isis
Router(config-router)# mpls traffic-eng level-2
Router(config-router)# mpls traffic-eng router-id loopback0
Router(config-router)# metric-style wide
```

- Support for wide metrics must be enabled.

IS-IS Extensions for TE



IS-IS Type 134 TLV Advertise a 32-bit router ID

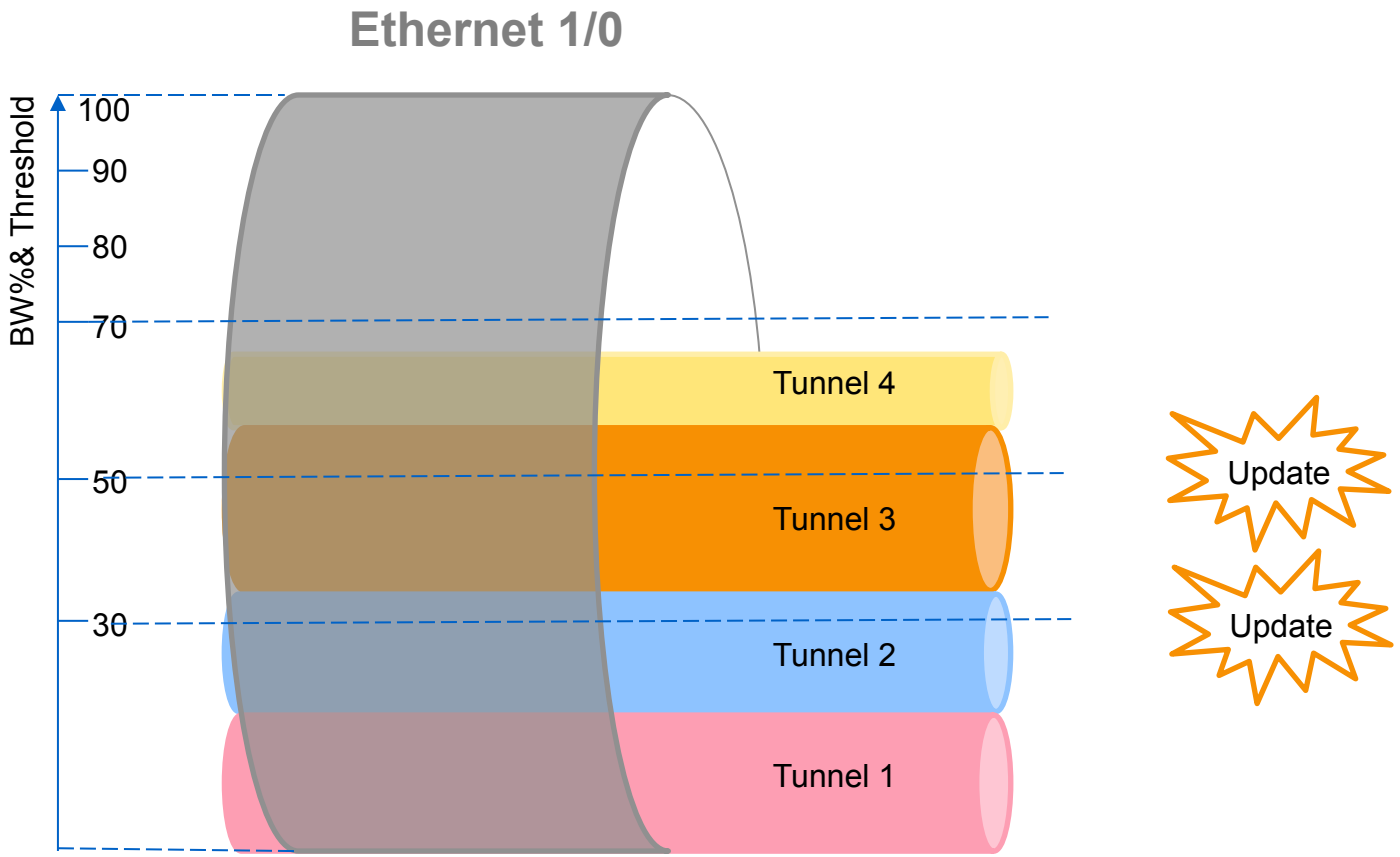
IS-IS Type 135 TLV Expand the link metrics and path metric

When to Flood the Information

- When a link goes up or down
- When a link's configuration is changed
- Periodically reflood the router's IGP information
- When link bandwidth changes **significantly**

Bandwidth Significant Change

- Each time a threshold is crossed, an update message is sent.



Default Threshold

- We can view the current threshold using the command `show mpls traffic-eng link-management badwidth-allocation`

```
Router1#show mpls traffic-eng link bandwidth-allocation ethernet 1/0
...(Omitted)
      Up Thresholds:        15 30 45 60 75 80 85 90 95 96 97 98 99 100 (default)
      Down Thresholds:      100 99 98 97 96 95 90 85 80 75 60 45 30 15 (default)
...(Omitted)
```

  - Denser population as utilization increases
  - Different thresholds for Up and Down

Path Calculation and Setup

APNIC

Tunnel Path Selection

- Tunnel has two path options
  1. Dynamic
  2. Explicit
- Path is a set of next-hop addresses (physical or loopbacks) to destination
- This set of next-hops is called Explicit Route Object (ERO)

Dynamic Path Option

- Dynamic = router calculates path using TE topology database
- Router will take best IGP path that meets BW requirements, also called CSPF algorithm.

```
R1(config)# interface tunnel 1
R1(config-if)# tunnel mpls traffic-eng path-option 10 dynamic
```

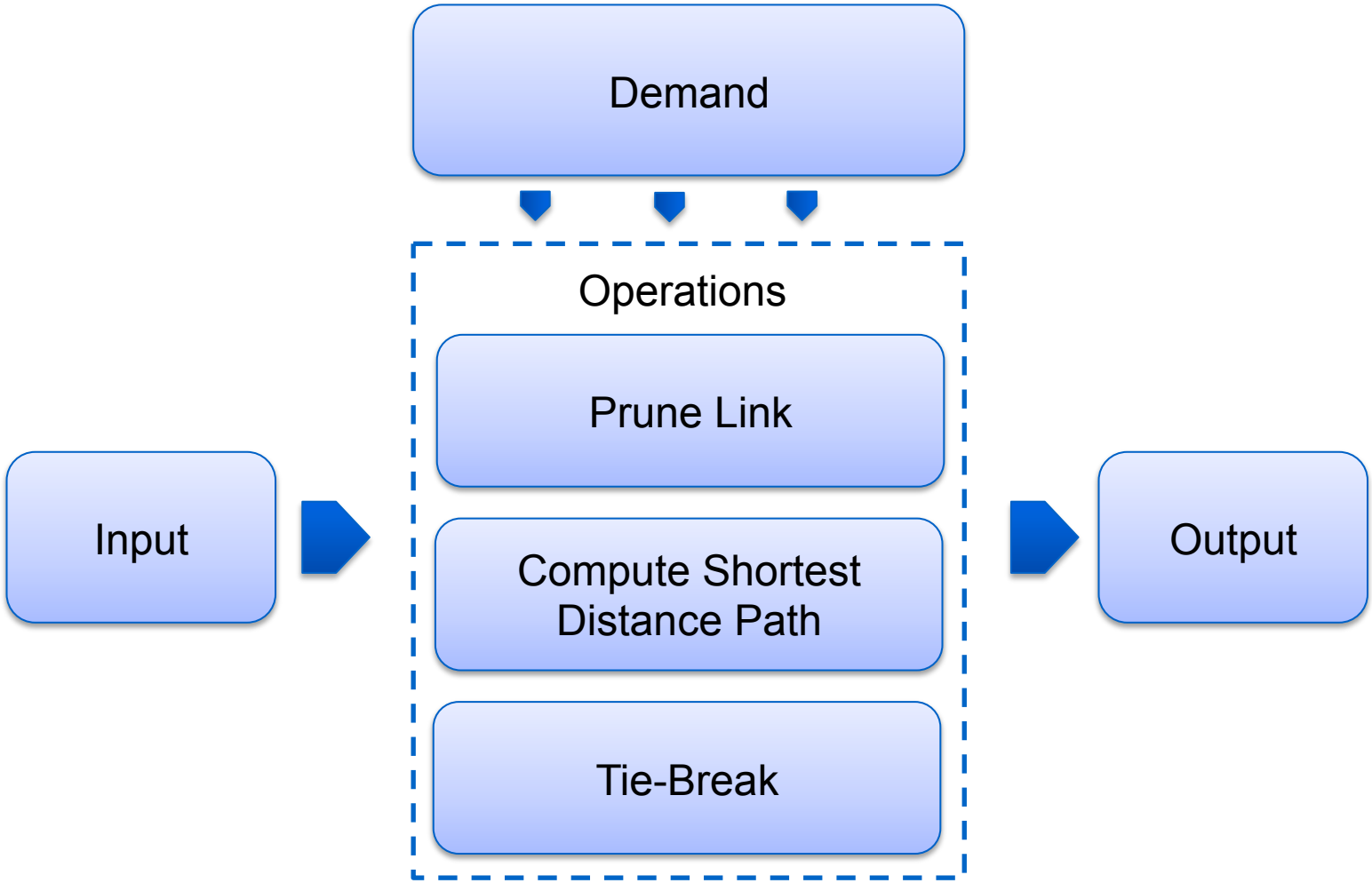
Path Calculation

- Modified Dijkstra
- Often referred to as CSPF
  - Constrained SPF
- …or PCALC (path calculation)
- Final result is explicit route meeting desired constraint

C-SPF

- Shortest-cost path is found that meets administrative constraints
- These constraints can be
  - bandwidth
  - link attribute (aka color, resource group)
  - priority
- The addition of constraints is what allows MPLS-TE to use paths other than *just* the shortest one

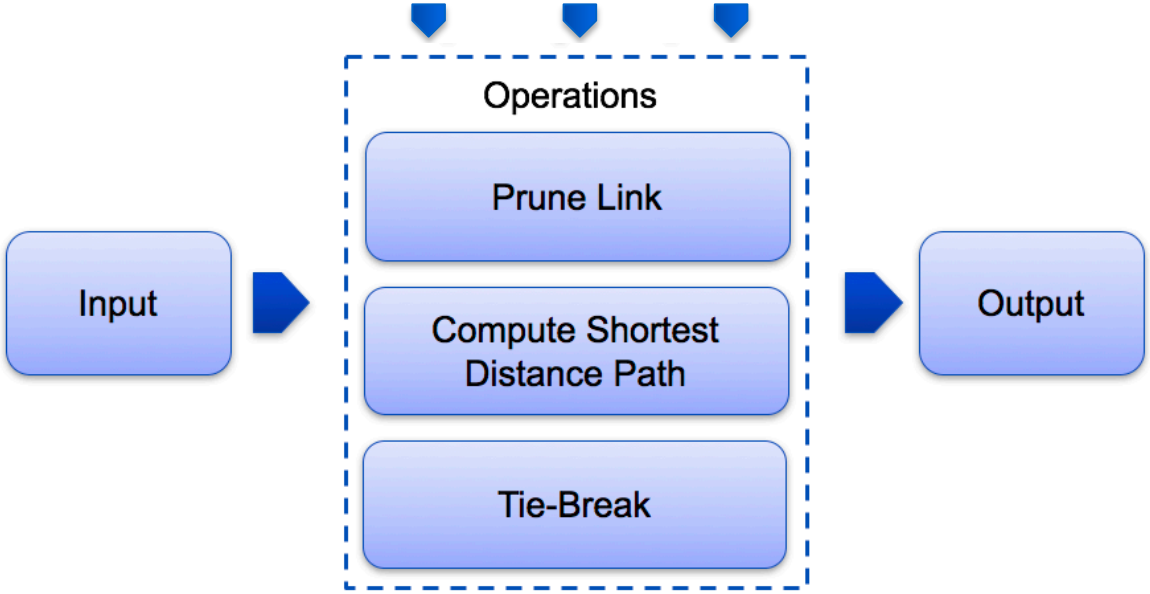
Path Computation



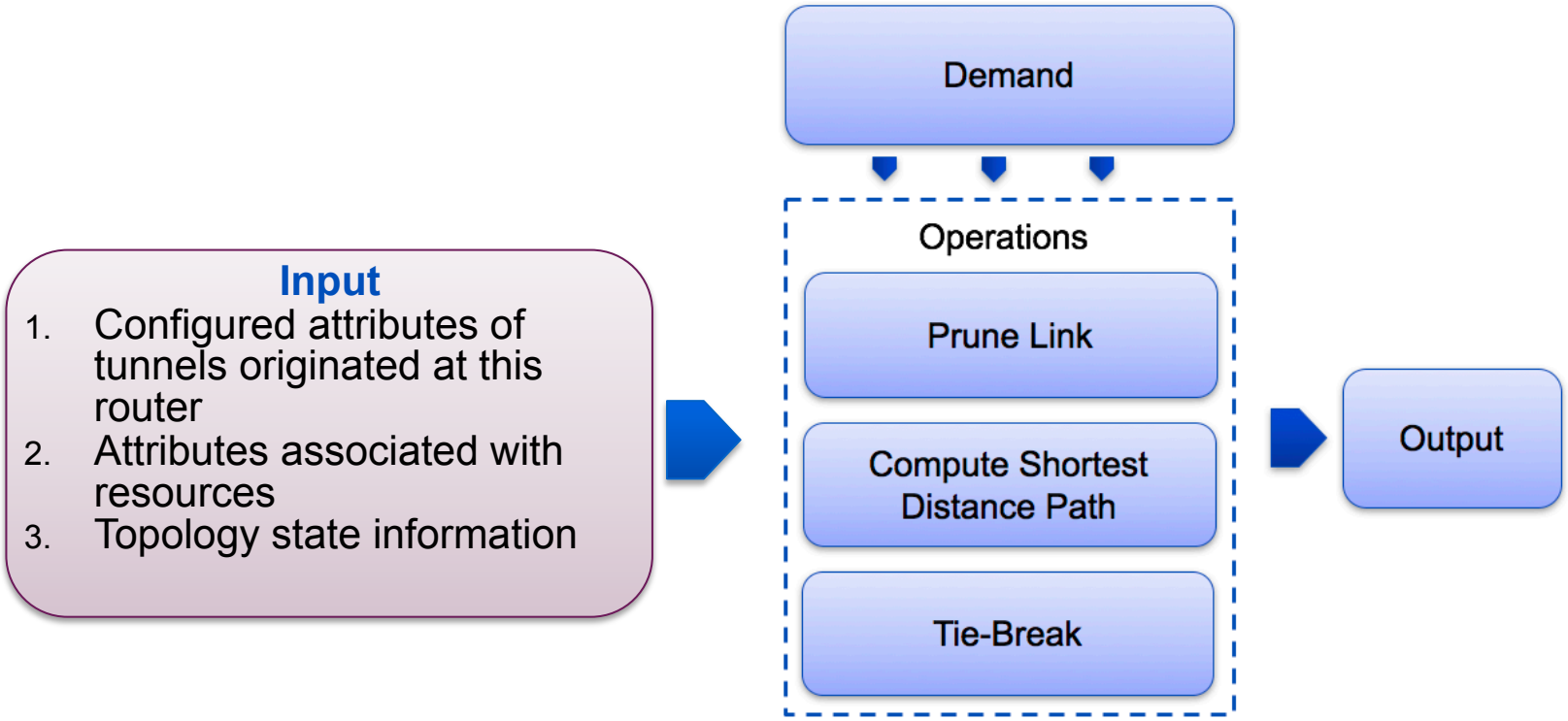
Path Computation - Demand

"On demand" by the tunnel's head-end:

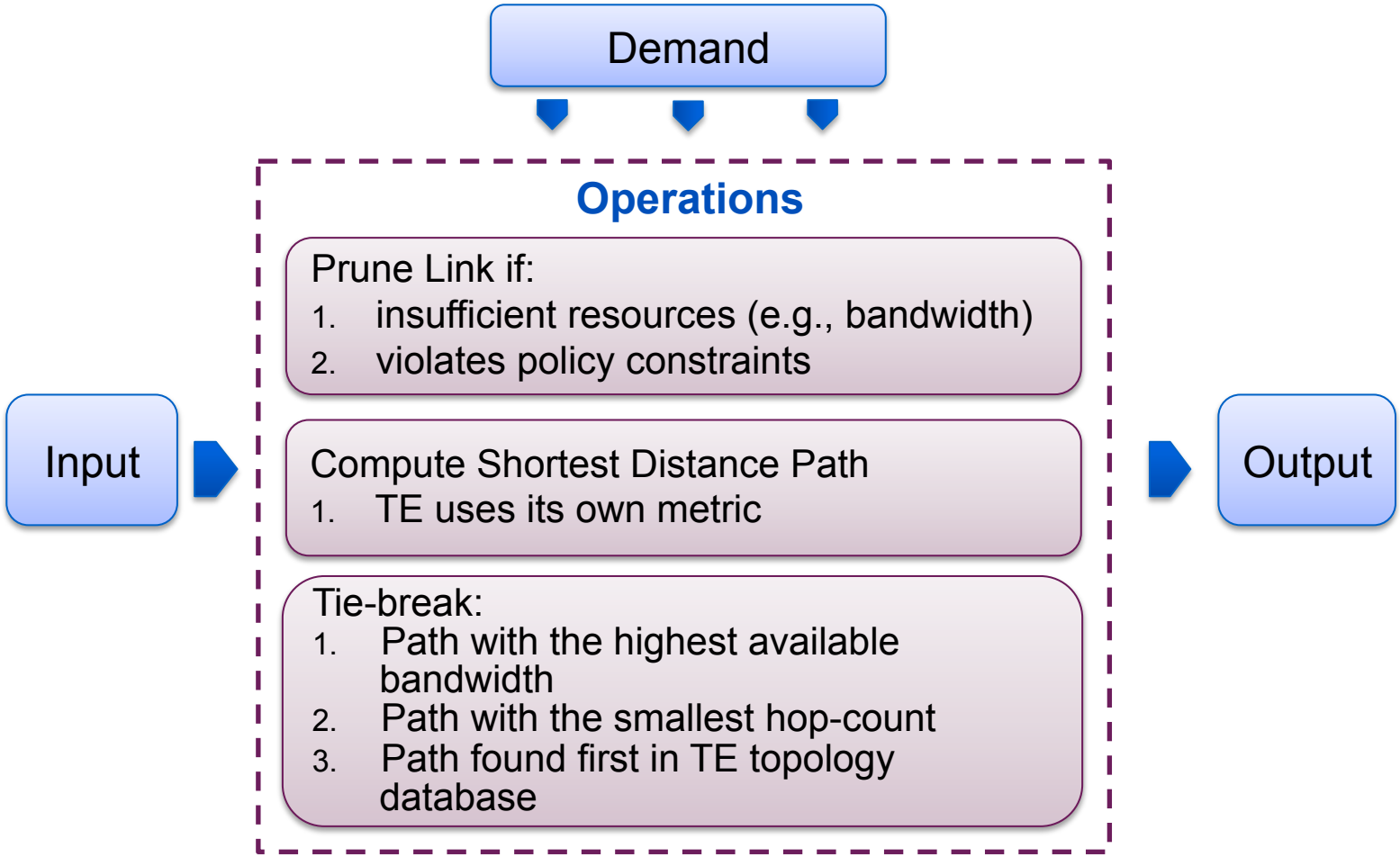
1. for a new tunnel
2. for an existing tunnel whose (current) LSP failed
3. for an existing tunnel when doing re-optimization



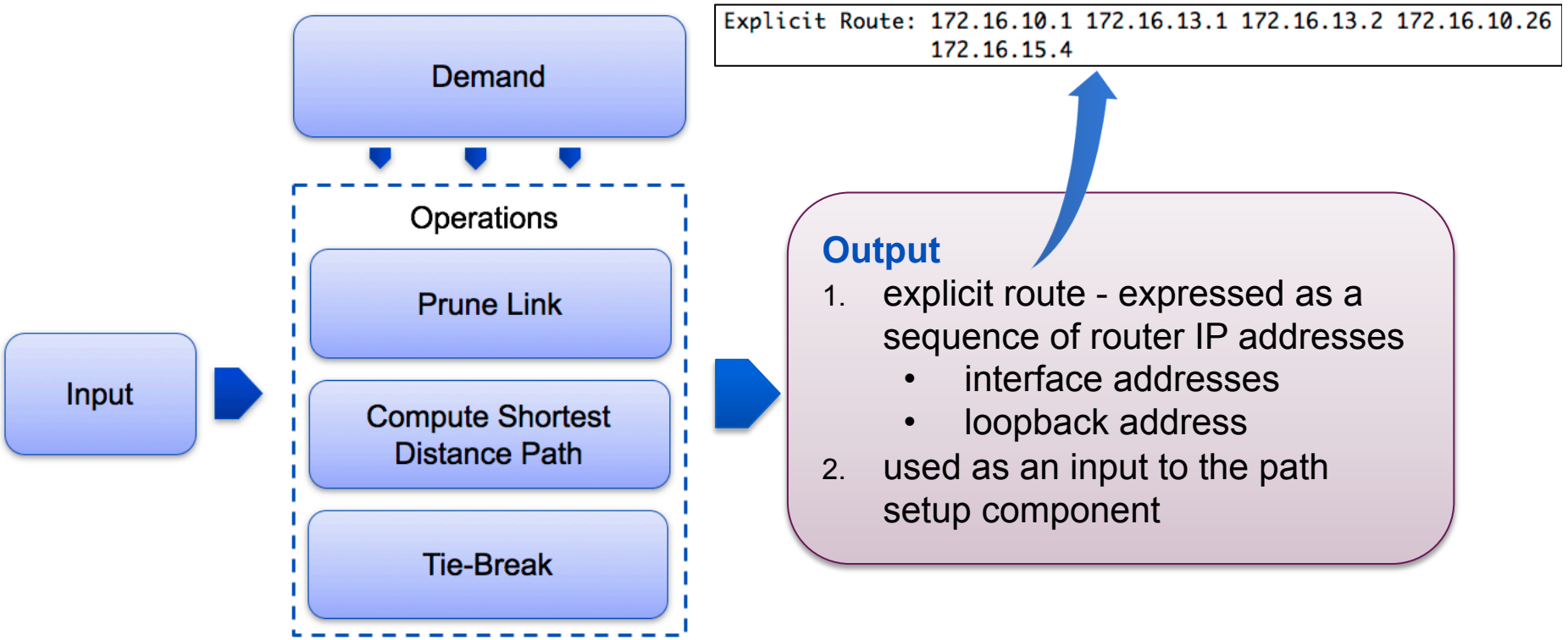
Path Computation - Input



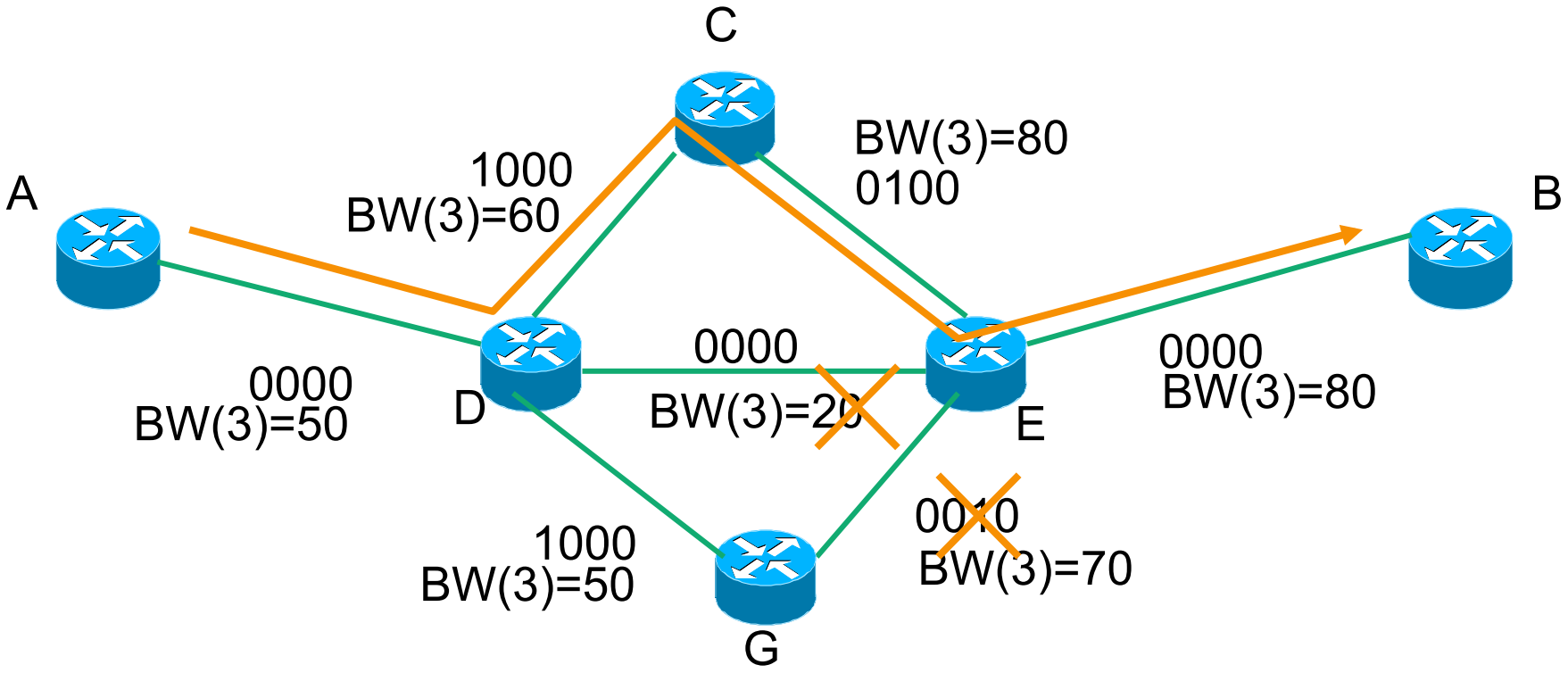
Path Computation - Operation



Path Computation - Output



BW/Policy Example

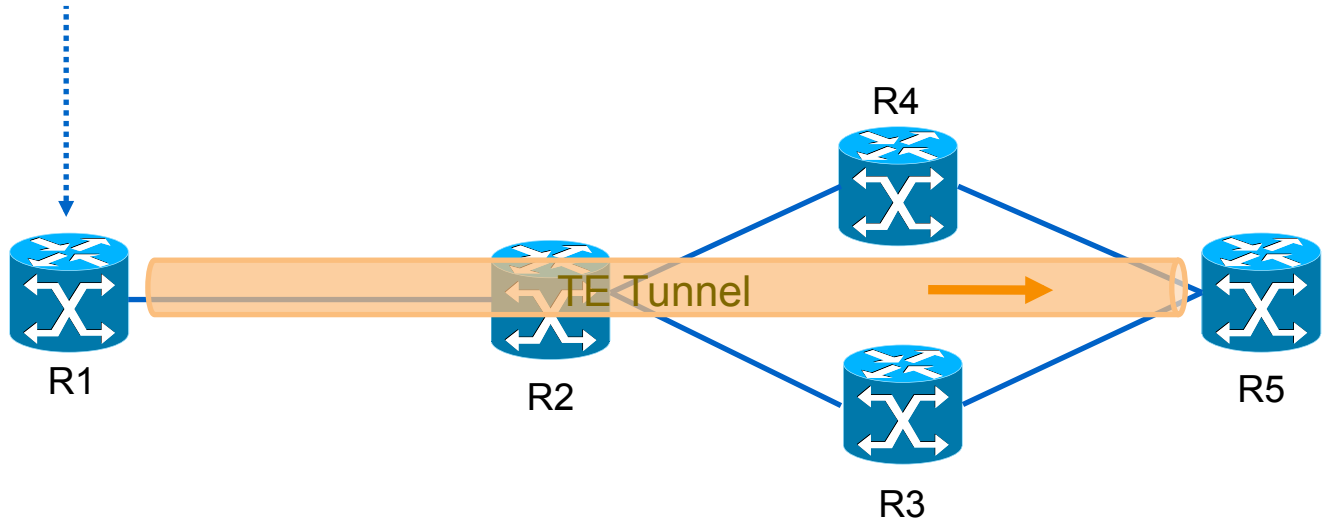


- Tunnel's request:
  - Priority 3, BW = 30 units,
  - Policy string: 0000, mask: 0011

Explicit Path Option

- explicit = take specified path.
- Router sets up path you specify.

```
R1(config)# interface tunnel 1
R1(config-if)# tunnel mpls traffic-eng path-option 10 explicit
name R1toR5
```

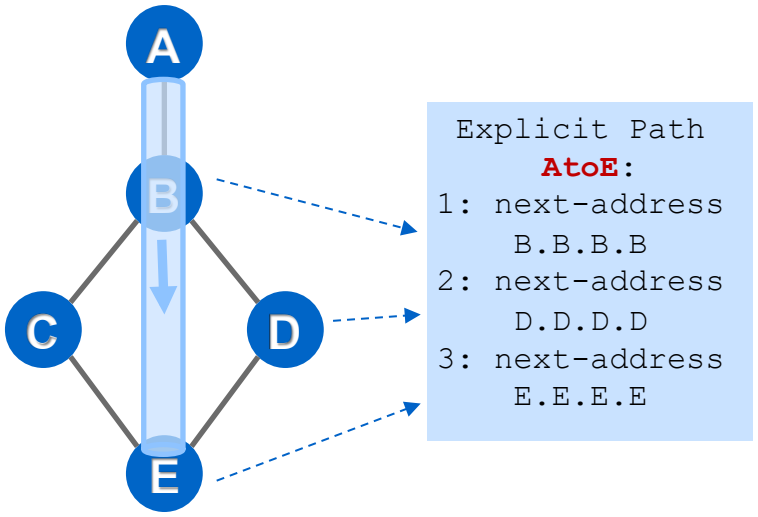


Strict and Loose Path

- Paths are configured manually. Each hop is a physical interface or loopback.

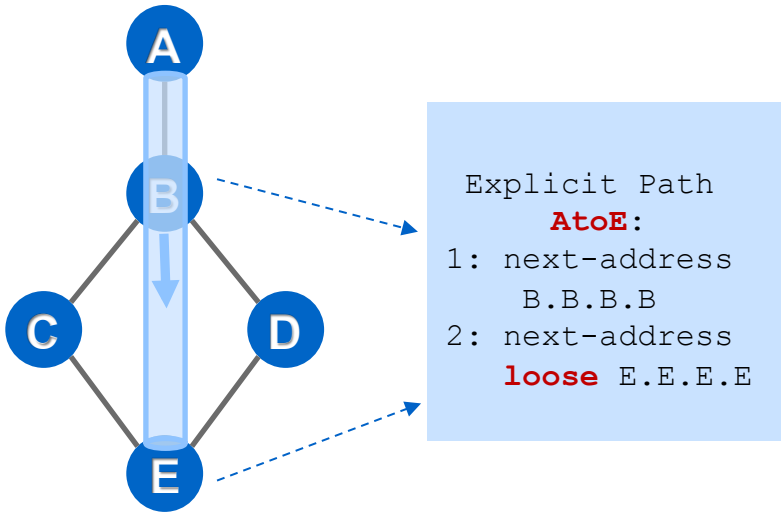
Strict Path

A network node and its preceding node in the path must be adjacent and directly connected.

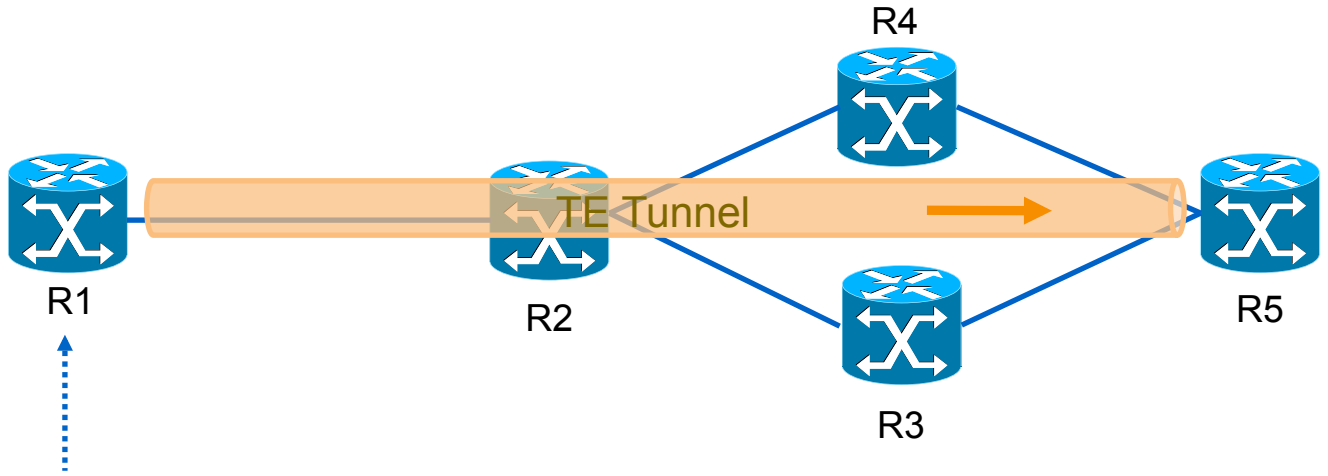


Loose Path

A network node must be in the path but is not required to be directly connected to its preceding node.

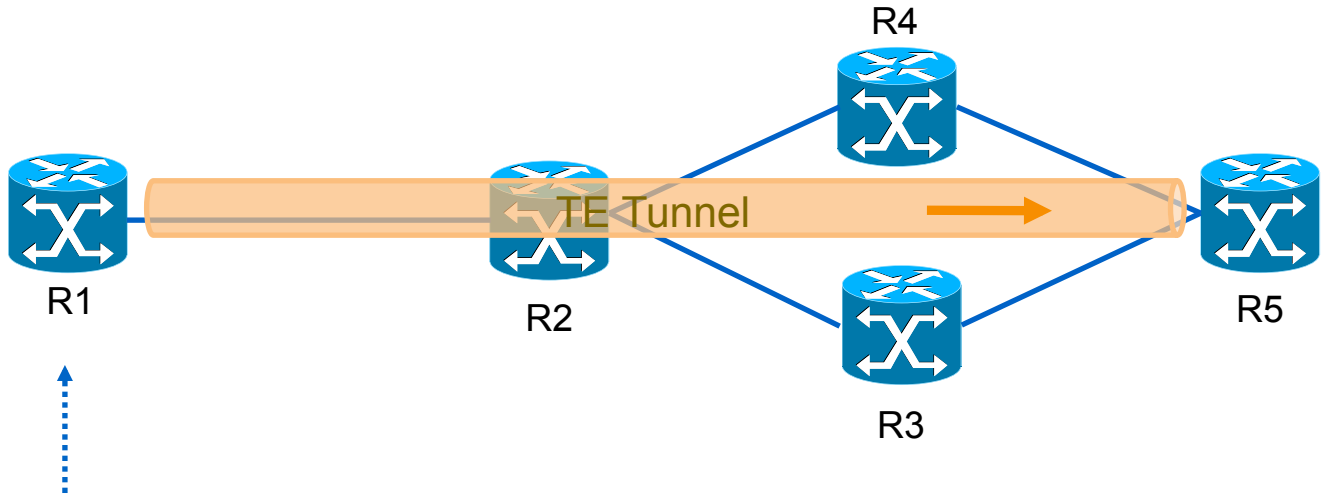


Configure Strict Explicit Path




```
R1(config)# ip explicit-path name R1toR5
R1(cfg-ip-expl-path)# next-address strict 2.2.2.2
Explicit Path name R1toR5:
    1: next-address 2.2.2.2
R1(cfg-ip-expl-path)# next-address strict 4.4.4.4
Explicit Path name R1toR5:
    1: next-address 2.2.2.2
    2: next-address 4.4.4.4
R1(cfg-ip-expl-path)# next-address strict 5.5.5.5
Explicit Path name R1toR5:
    1: next-address 2.2.2.2
    2: next-address 4.4.4.4
    3: next-address 5.5.5.5
```

Configure Loose Explicit Path




```
R1(config)# ip explicit-path name R1toR5
R1(cfg-ip-expl-path)# next-address 2.2.2.2
Explicit Path name R1toR5:
    1: next-address 2.2.2.2
R1(cfg-ip-expl-path)# next-address loose 5.5.5.5
Explicit Path name R1toR5:
    1: next-address 2.2.2.2
    2: next-address loose 5.5.5.5
```

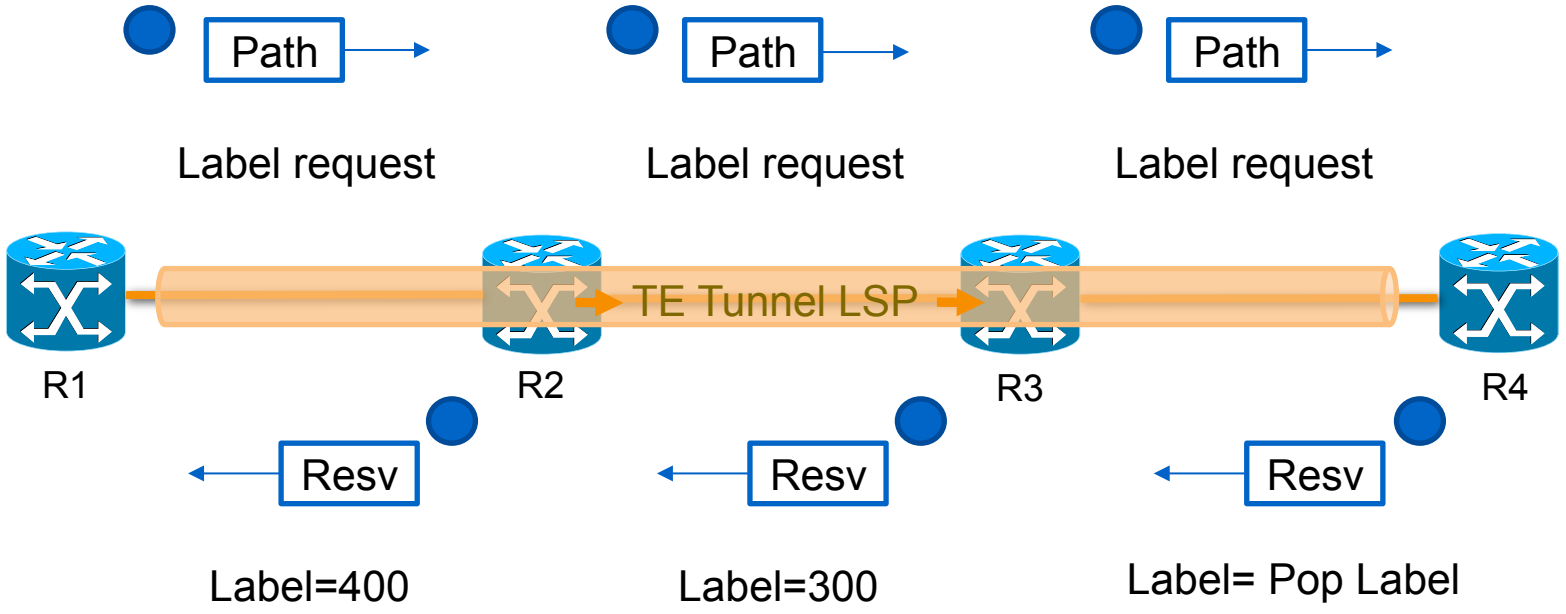
By default, it is strict.

RSVP-TE

- After calculating the path, tunnel will be set up by using RSVP-TE.
- RSVP has three basic functions:
  - Path setup and maintenance
  - Path teardown
  - Error signalling

Setup of TE LSP

- In following topo, R1 will set up a TE tunnel from R1 to R4:



Check RSVP Status

- show ip rsvp interface

```
Router2# show ip rsvp interface
interface     rsvp       allocated  i/f max   flow max sub max   VRF
Fa0/0         ena        0          256K      256K     0
Fa0/1         ena        100K       256K      256K     0
Et1/0         ena        100K       256K      256K     0
Et1/1         ena        0          256K      256K     0
```

- show ip rsvp neighbor

```
Router2# show ip rsvp neighbor
Neighbor         Encapsulation  Time since msg rcvd/sent
172.16.10.2      Raw IP         00:00:11   00:00:06
172.16.13.2      Raw IP         00:00:08   00:00:14
```

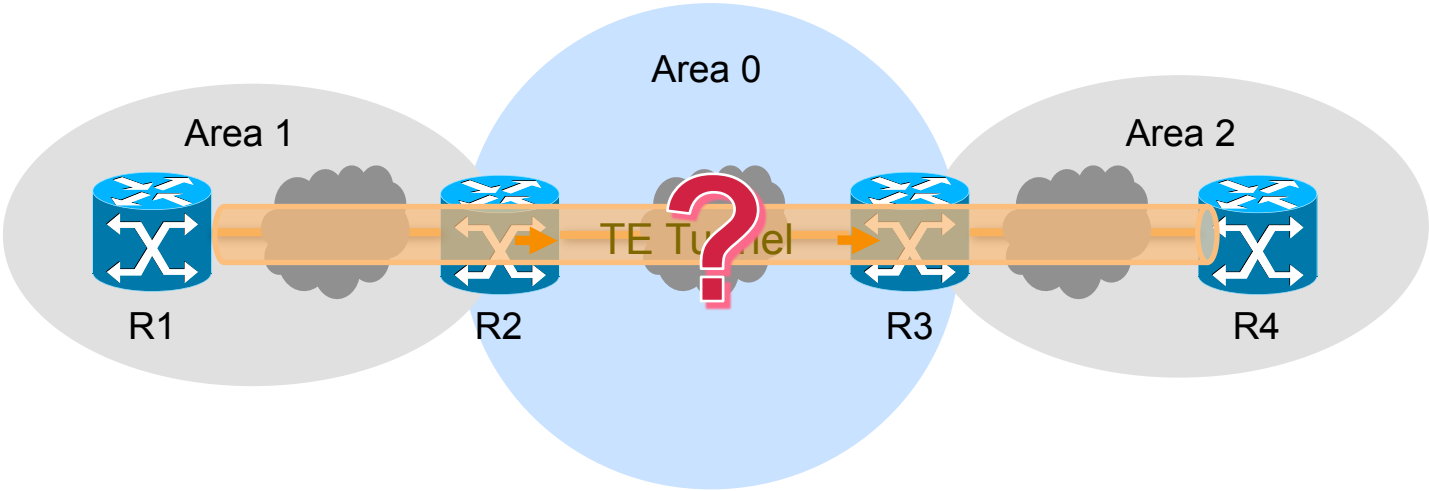
Check RSVP Reservation

- show ip rsvp reservation detail

```
Router2# show ip rsvp reservation detail
Reservation:
  Tun Dest:   172.16.15.1  Tun ID: 10  Ext Tun ID: 172.16.15.4
  Tun Sender: 172.16.15.4  LSP ID: 3
  Next Hop: 172.16.10.2 on Ethernet1/0
  Label: 3 (outgoing)
  Reservation Style is Shared-Explicit, QoS Service is Controlled-Load
  Resv ID handle: 02000413.
  Created: 20:27:28 AEST Thu Dec 22 2016
  Average Bitrate is 100K bits/sec, Maximum Burst is 1K bytes
  Min Policed Unit: 0 bytes, Max Pkt Size: 1500 bytes
  Status:  Policy: Accepted. Policy source(s): MPLS/TE
Reservation:
  Tun Dest:   172.16.15.4  Tun ID: 10  Ext Tun ID: 172.16.15.1
  Tun Sender: 172.16.15.1  LSP ID: 814
  Next Hop: 172.16.13.2 on FastEthernet0/1
  Label: 16 (outgoing)
  Reservation Style is Shared-Explicit, QoS Service is Controlled-Load
  Resv ID handle: 02000408.
  Created: 19:56:58 AEST Thu Dec 22 2016
  Average Bitrate is 100K bits/sec, Maximum Burst is 1K bytes
  Min Policed Unit: 0 bytes, Max Pkt Size: 1500 bytes
  Status:  Policy: Accepted. Policy source(s): MPLS/TE
```

Interarea Tunnel

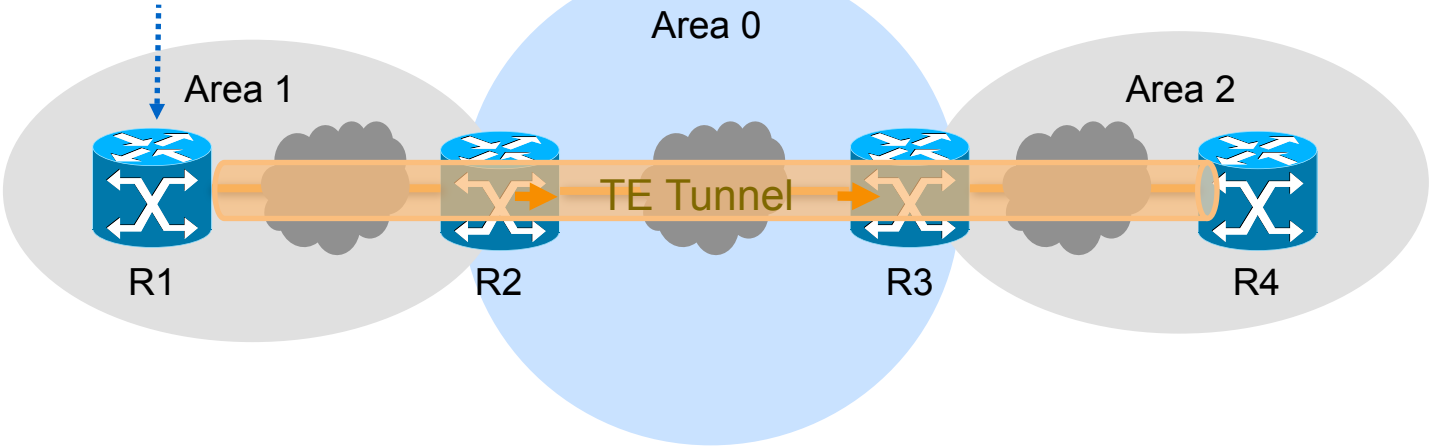
- Remember?
  - OSPF Type 10 LSA's flooding scope is local area.
- The TE LSP path is computed for single IGP area only due to lack of visibility of the topology in other IGP areas.
- How can we build tunnels between areas?



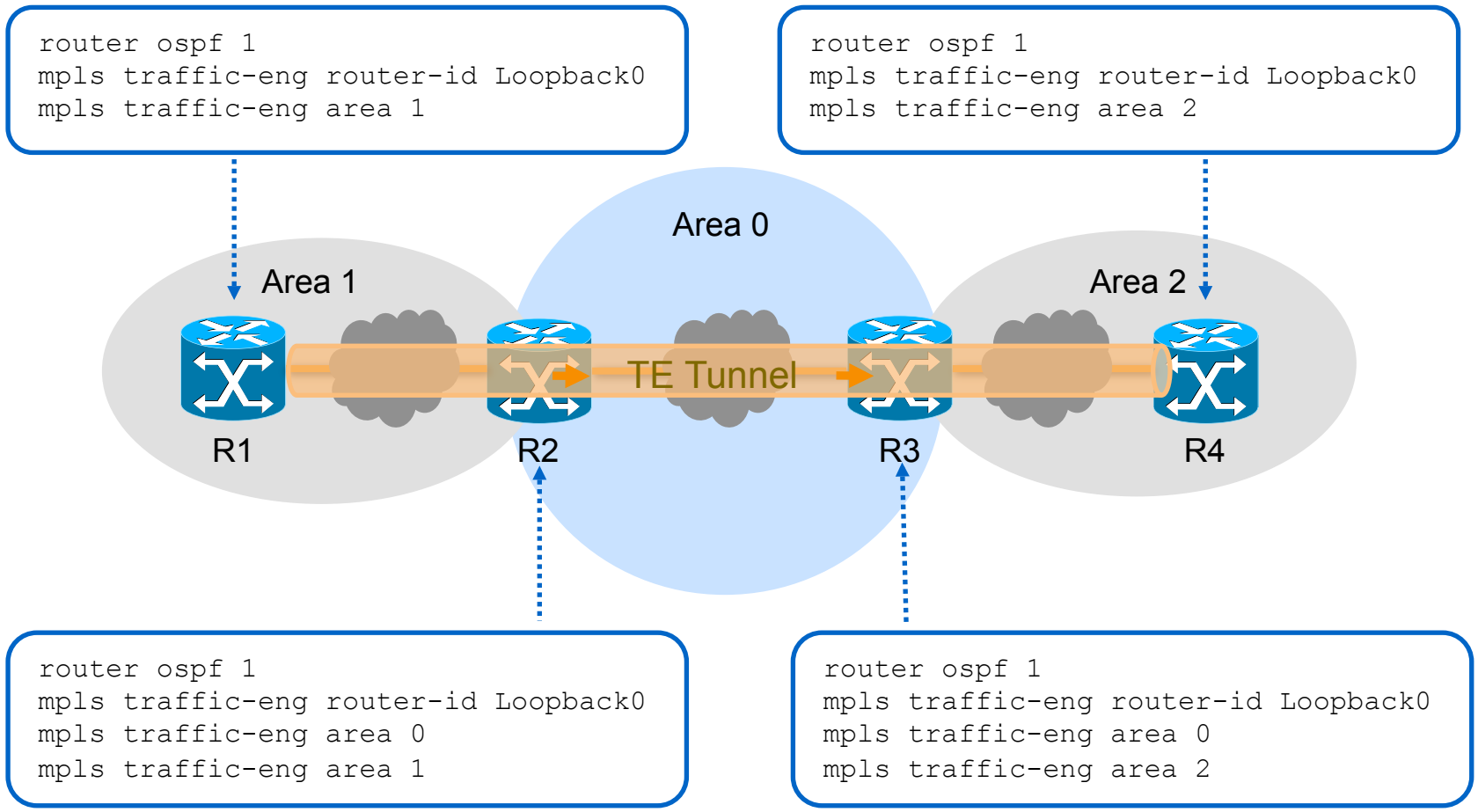
Define Contiguous Path

- An **explicit path** identifying the **ABR**s is required.
- The head-end router and the ABRs along the specified explicit path expand the loose hops, each computing the path segment to the next ABR or tunnel destination.

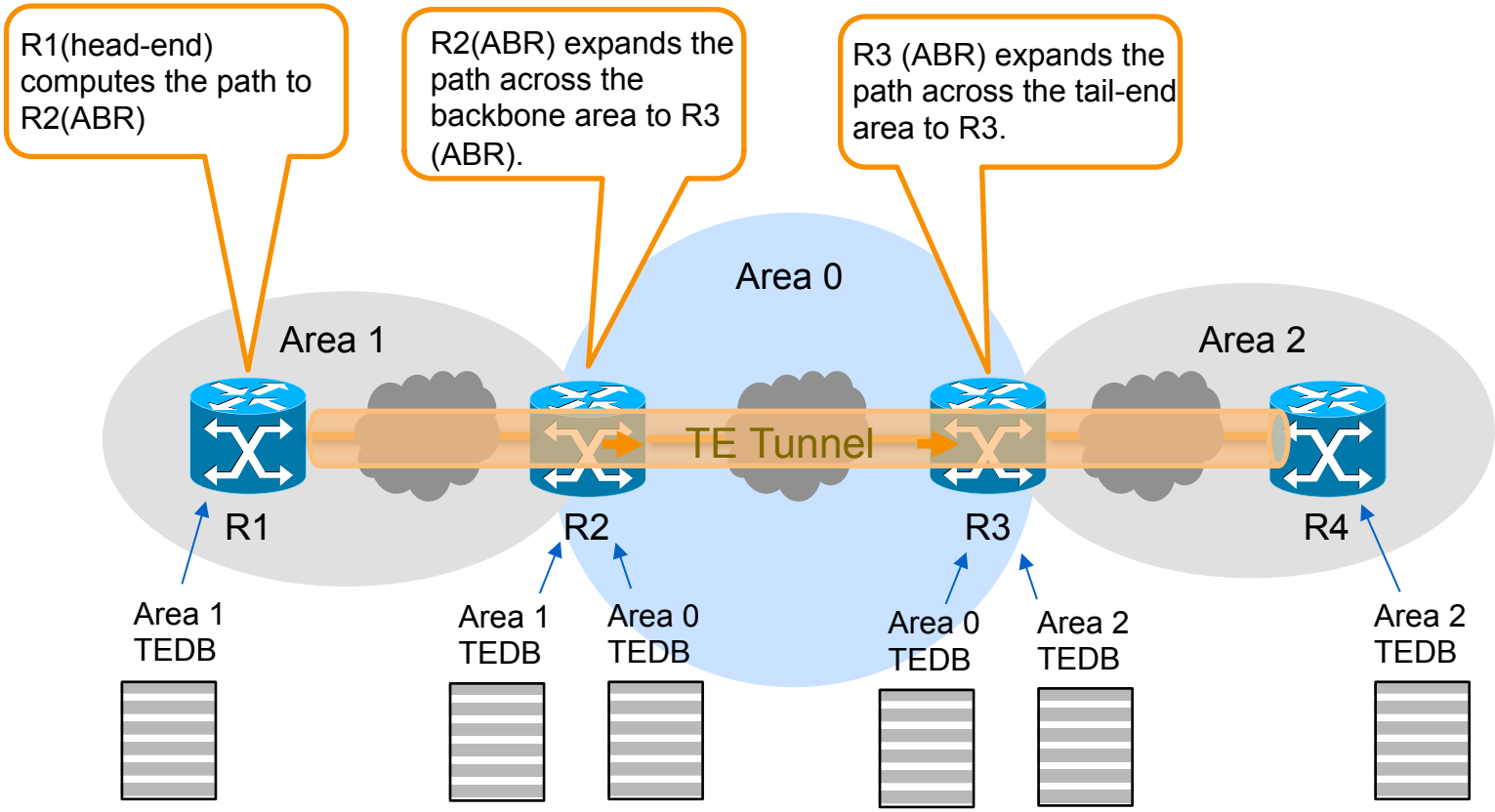
```
ip explicit path name R1toR4
  next-address loose 2.2.2.2
  next-address loose 3.3.3.3
  next-address loose 4.4.4.4
```



Enable MPLS TE for Areas



Setup Interarea Tunnel



ABR is maintaining multiple TEDB.
One TEDB for one area.

Forwarding Traffic Down Tunnels

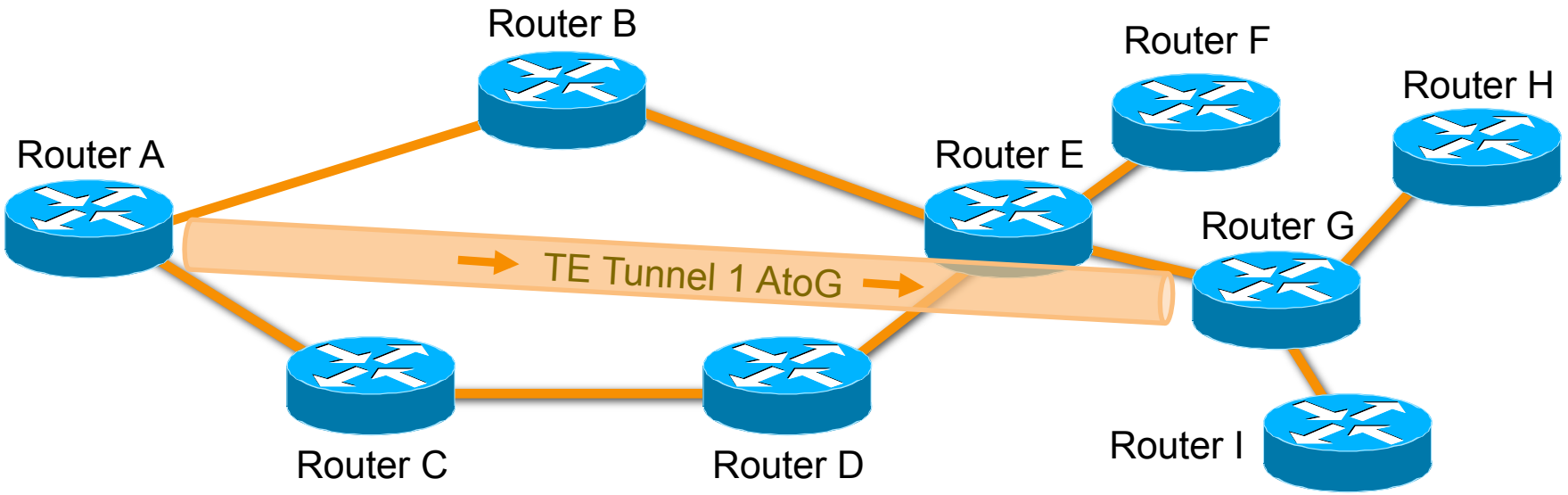
APNIC

Routing Traffic Down a Tunnel

- Once the tunnel is established and operational, it's ready to forward data traffic.
- However, no traffic will enter the tunnel unless the IP routing tables and FIB tables are modified.
- How to get traffic down the tunnel?
  1. Static route
  2. Auto route
  3. Policy routing

Example Topology

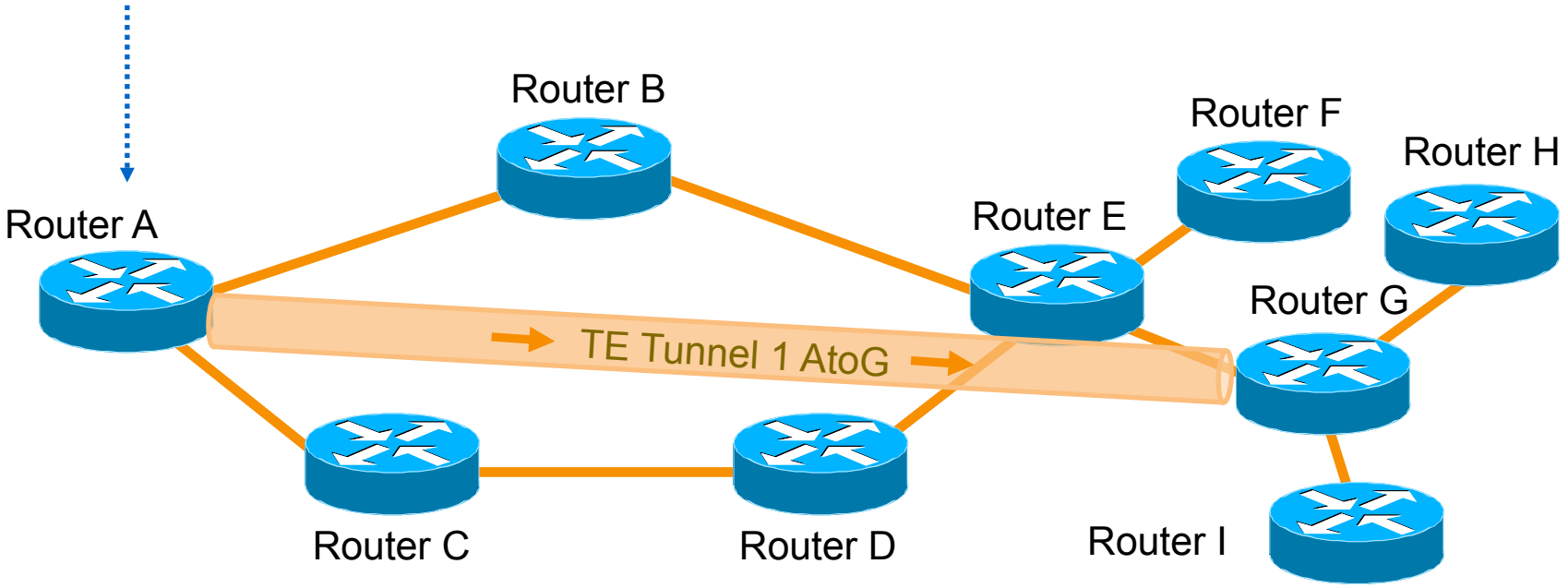
- In the following topology, cost=10 for each link.
- Tunnel 1 has been created on Router A from A to G.



Static Route

- Configure a static route pointing down the tunnel interface.

```
RtrA(config)# ip route H.H.H.H 255.255.255.255 Tunnel1
```

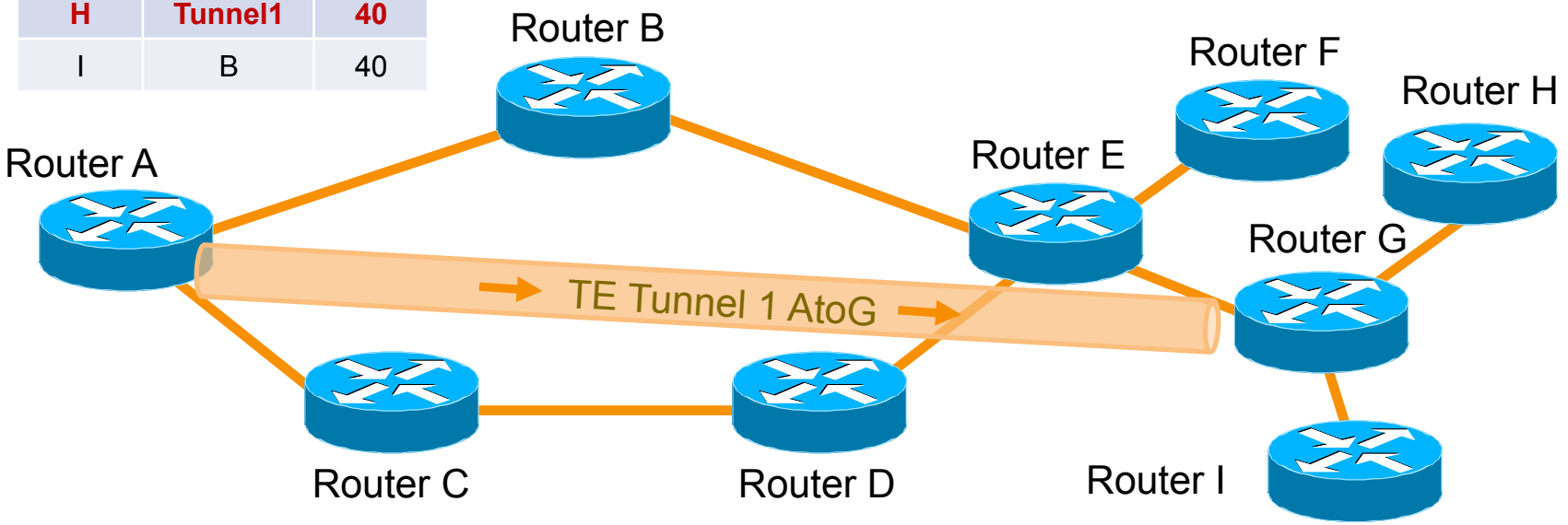


Static Route

Routing Table of RouterA

Node	Next-Hop	Cost
B	B	10
C	C	10
D	C	20
E	B	20
F	B	30
G	**B**	**30**
H	**Tunnel1**	**40**
I	B	40

- Router H is known via the tunnel
- Router G is **not** routed to over the tunnel, even though it's the tunnel tail!



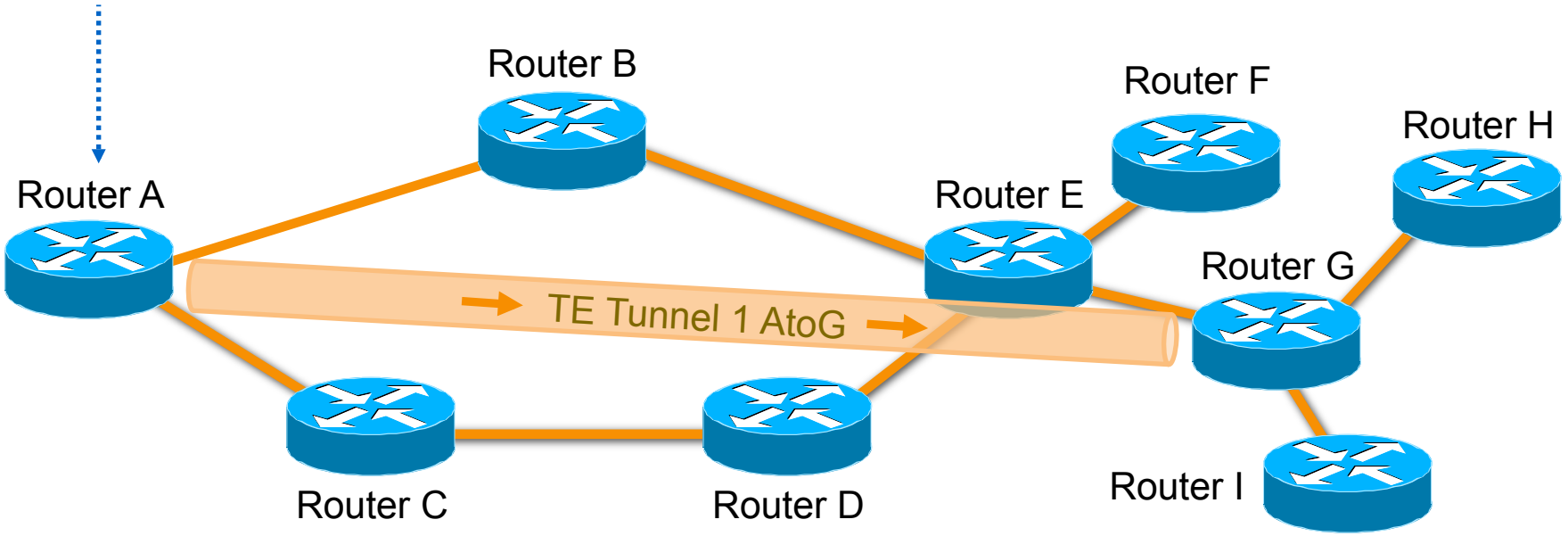
Auto Route

- Auto route allows a TE tunnel to participate in IGP route calculations as a logical link. The tunnel interface is used as the outbound interface of the route.
  - **IGP Shortcut**: The command is "**autoroute announce**" in Cisco IOS. Tunnel interface will be included in SPF tree on the **Head-end** router.
  - **Forwarding adjacency**: allows **all nodes** in IGP domains to see the TE tunnels and use them in the SPF calculation.

IGP Shortcut

- In the tunnel interface, configure *autoroute announce*

```
RtrA(config)# interface Tunnel1
RtrA(config-if)# tunnel mpls traffic-eng autoroute announce
```

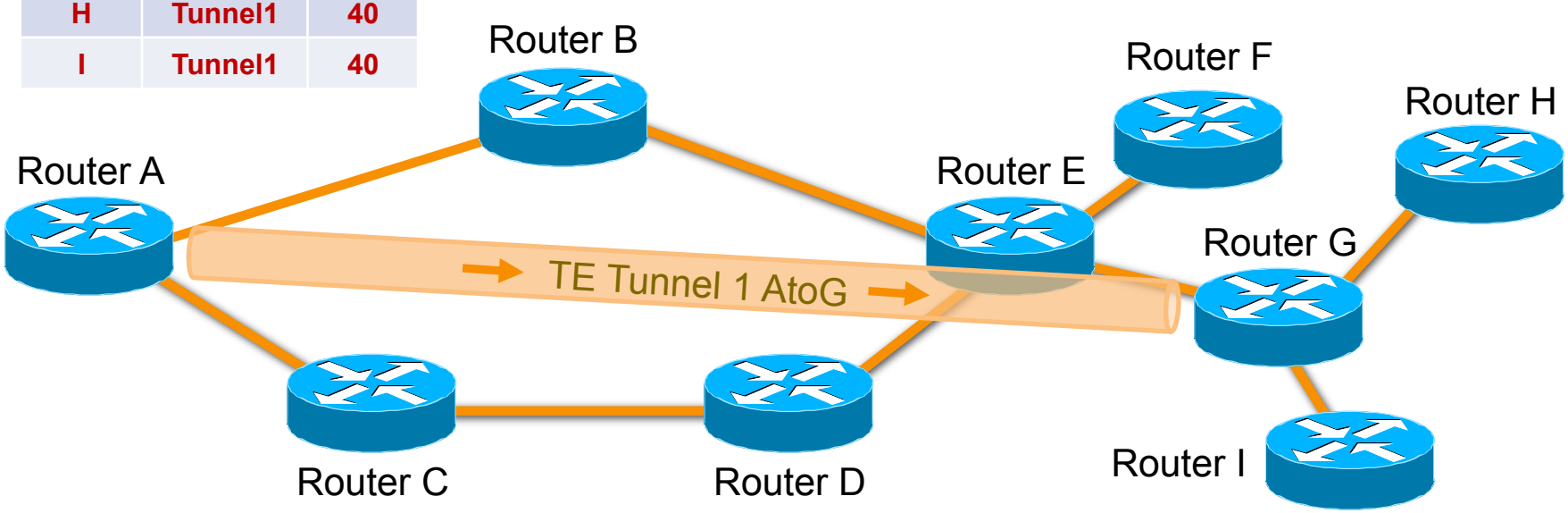


IGP Shortcut

Routing Table of RouterA

Node	Next-Hop	Cost
B	B	10
C	C	10
D	C	20
E	B	20
F	B	30
G	**Tunnel1**	**30**
H	**Tunnel1**	**40**
I	**Tunnel1**	**40**

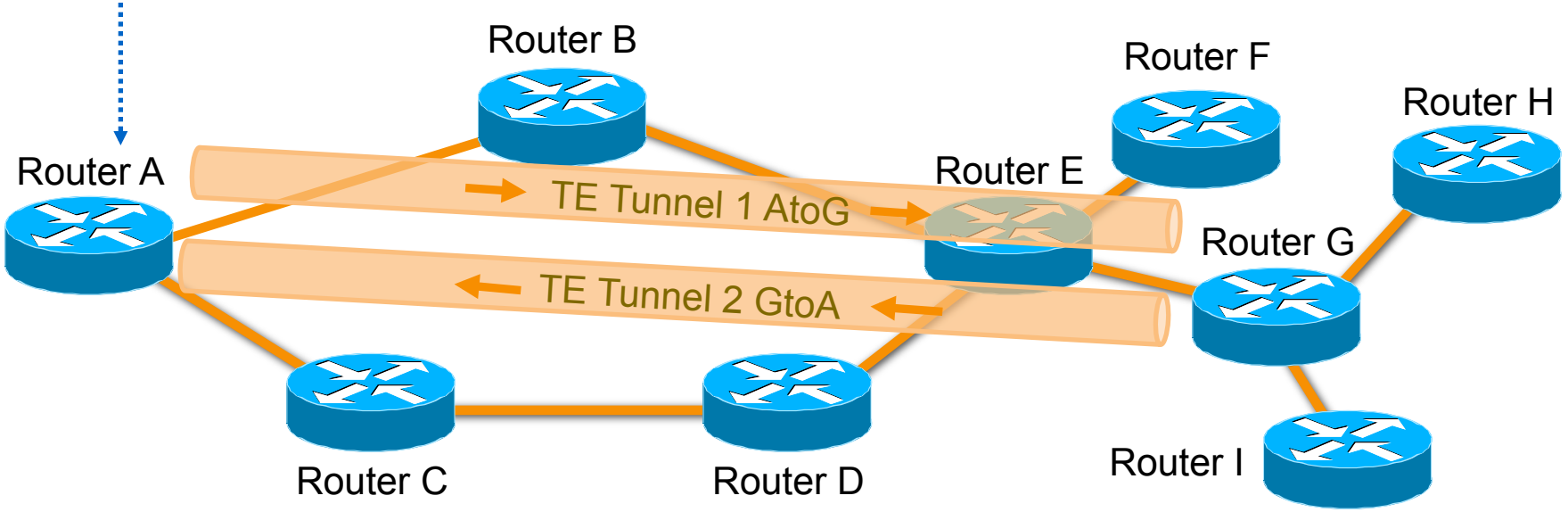
- Everything “behind” the tunnel is routed via the tunnel



Forwarding Adjacency

- Bidirectional tunnels: tunnel 1 AtoG, tunnel 2 GtoA
- Configure following commands on **both Router A and G**

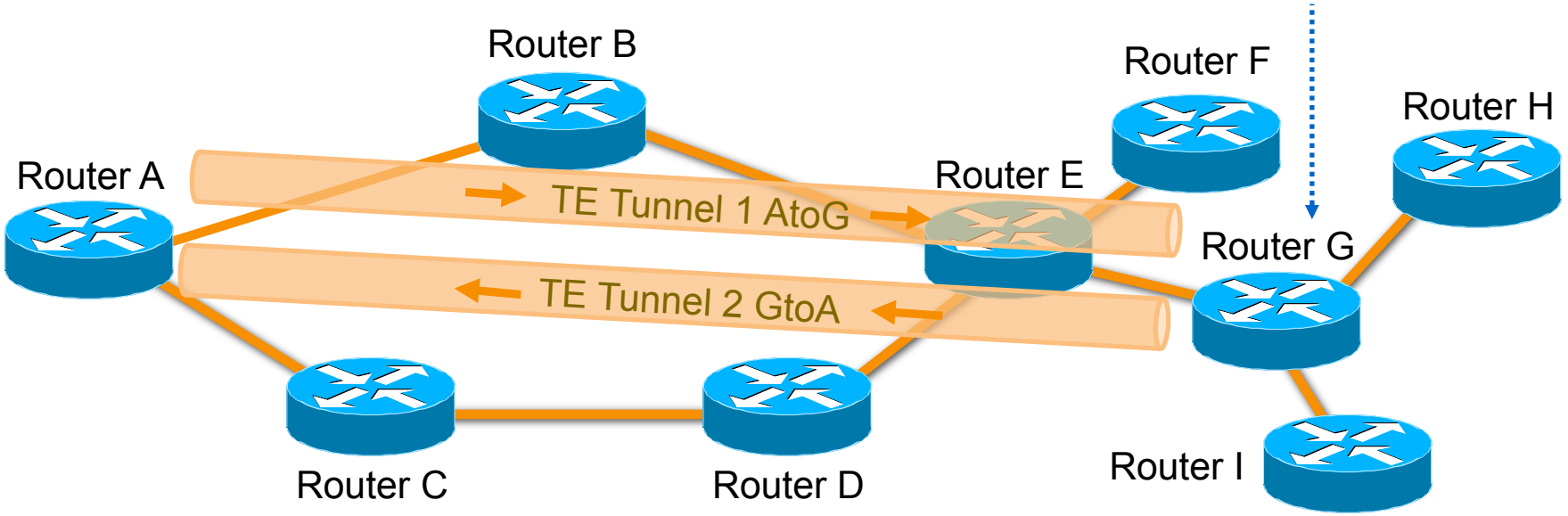
```
RtrA(config)# interface Tunnel1
RtrA(config-if)# tunnel mpls traffic-eng forwarding-adjacency
RtrA(config-if)# ip ospf cost 5
```



Forwarding Adjacency

- Similar commands on **Router G**

```
RtrG(config)# interface Tunnel1
RtrG(config-if)# tunnel mpls traffic-eng forwarding-adjacency
RtrG(config-if)# ip ospf cost 5
```

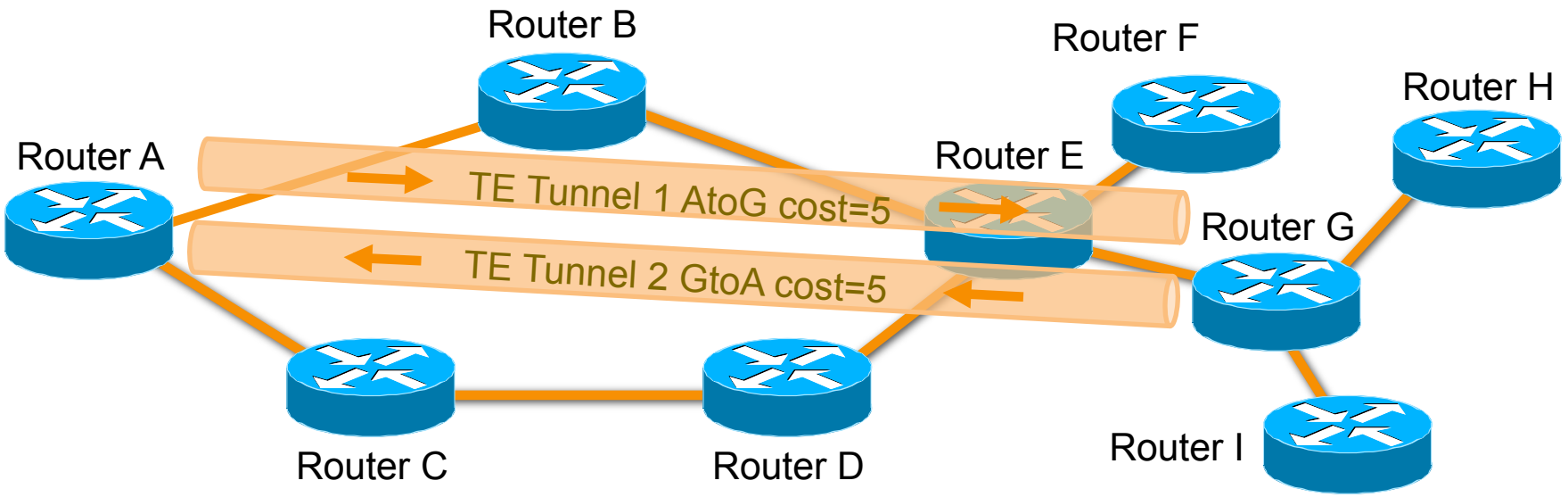


Before Configuring Forwarding Adjacency

- Assume cost of the tunnels is 5.
- Before FA configured, Router H does not include TE tunnel into its SPF calculation.

Routing Table of RouterH

Node	Next-Hop	Cost
A	G	40
…	…	...

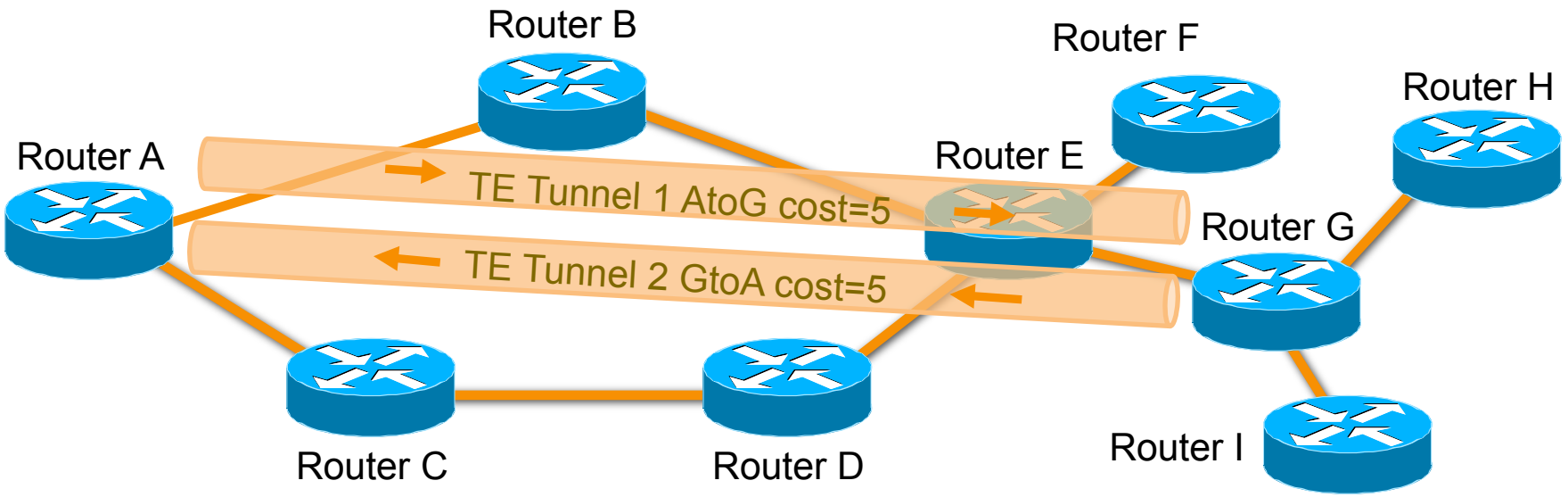


After Configuring Forwarding Adjacency

- After FA configured, Router G advertised as a link in an IGP network with the link's cost associated with it.
- Router H can see the TE tunnel and use it to compute the shortest path for routing traffic throughout the network.

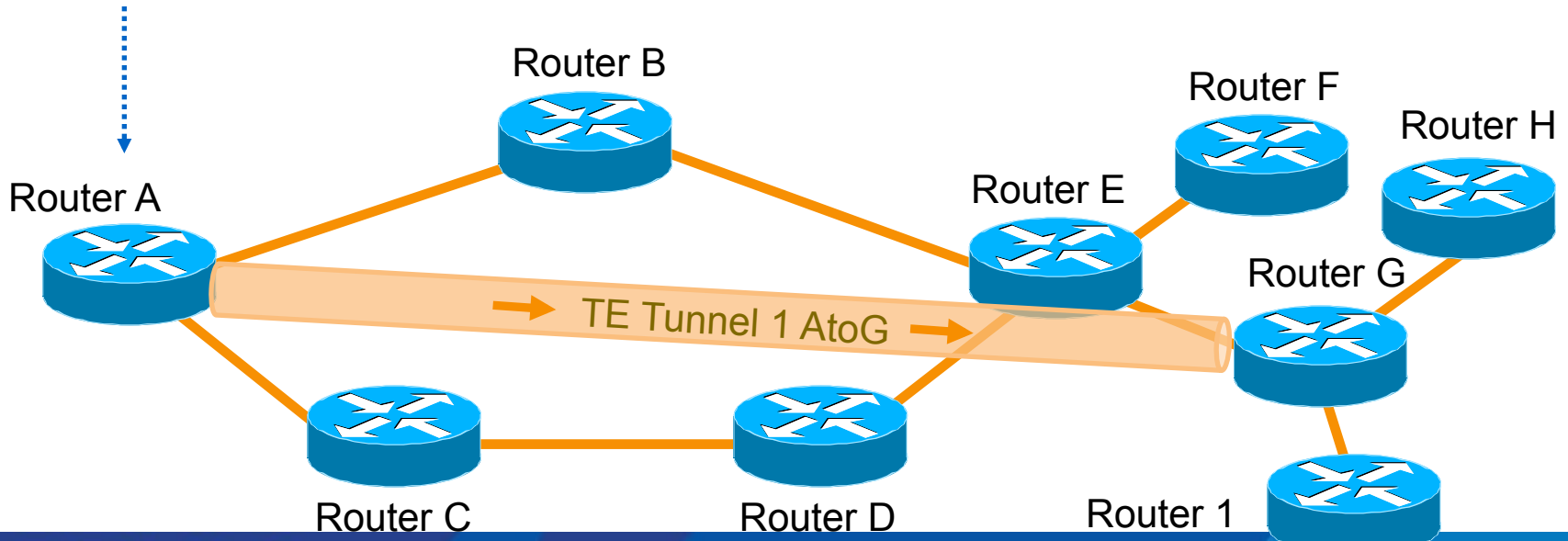
Routing Table of RouterH

Node	Next-Hop	Cost
A	G	15
…	…	…



Policy Routing

```
RtrA(config)# interface ethernet 1/0
RtrA(config-if)# ip policy route-map set-tunnel
RtrA(config)# route-map set-tunnel
RtrA(config-route-map)# match ip address 101
RtrA(config-route-map)# set interface Tunnel1
```

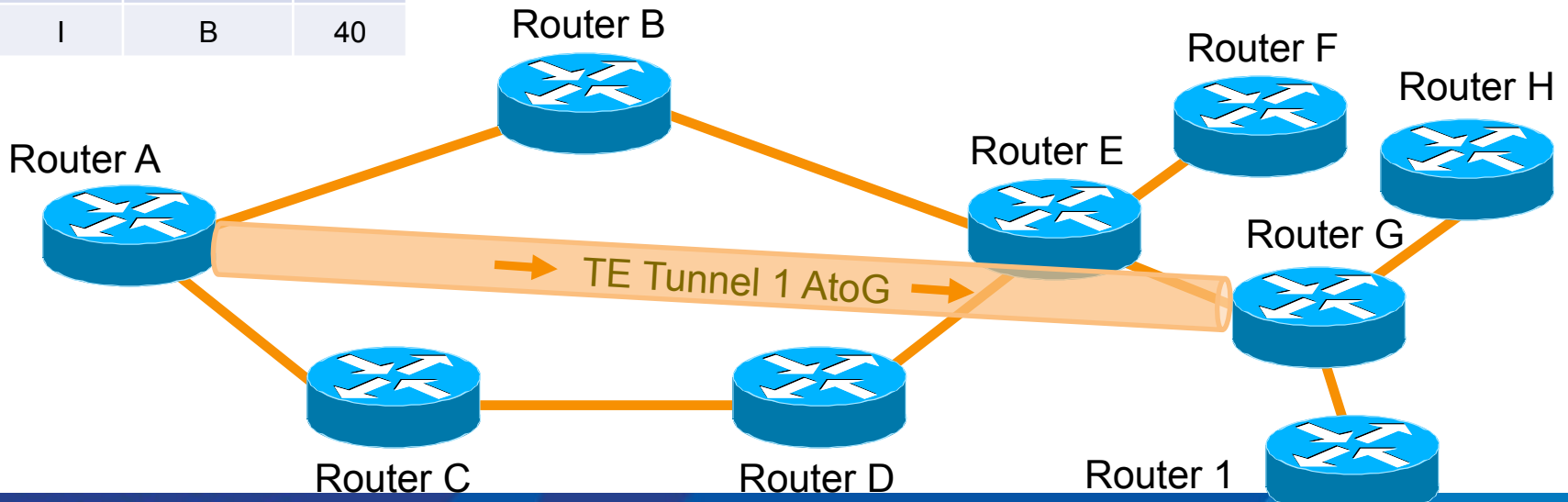


Policy Routing

Routing Table of RouterA

Node	Next-Hop	Cost
B	B	10
C	C	10
D	C	20
E	B	20
F	B	30
G	B	30
H	B	40
I	B	40

- Routing table isn't affected by policy routing
- Cisco IOS Need (12.0(23)S or 12.2T) or higher for 'set interface tunnel' to work

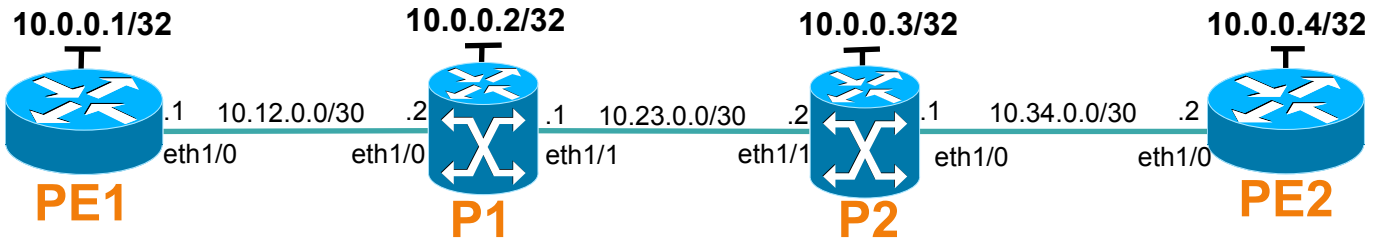


MPLS TE Configuration Example

APNIC

Configuration Example

- Task: Bidirectional tunnels need to be set up between PE1 to PE2 (BW=500kbps), path will be selected automatically.
- Prerequisite configuration:
  - 1. IP address configuration on PE & P routers
  - 2. IGP configuration on PE & P routers
    - Make sure all the routers in public network can reach each other.
  - 3. Enable MPLS on the router and interfaces



Enable MPLS TE

- Configuration steps:
  - 1. Configure MPLS TE on all the routers and the reservable bandwidth on the router interface

```
mpls traffic-eng tunnels

interface ethernet1/0
mpls traffic-eng tunnels
ip rsvp bandwidth 2000

interface ethernet1/1
mpls traffic-eng tunnels
ip rsvp bandwidth 2000
```

Enable OSPF to Support TE

– 2. Enable OSPF to support MPLS TE on every router:

```
router ospf 1
mpls traffic-eng router-id loopback 0
mpls traffic-eng area 0
```

Configure Tunnel Interface on Head-end Routers

– 3. Configure tunnel interface on both PE1 and PE2

```
interface tunnel 10
ip unnumbered loopback 0
mpls ip
tunnel mode mpls traffic-eng
tunnel destination 10.0.0.4
tunnel mpls traffic-eng path-option 10 dynamic
tunnel mpls traffic-eng bandwidth 500
tunnel mpls traffic-eng autoroute announce
tunnel mpls traffic-eng priority 0 0
```

Verify MPLS TE and RSVP Bandwidth on Interfaces

- Check MPLS TE configured on interfaces:

```
P1#show mpls interfaces
Interface              IP             Tunnel   BGP Static Operational
Ethernet1/1            Yes (ldp)      Yes      No  No     Yes
Ethernet1/0            Yes (ldp)      Yes      No  No     Yes
```

- Check RSVP bandwidth status for interfaces:

```
PE1#show ip rsvp interface
interface    rsvp       allocated  i/f max  flow max sub max  VRF
Et1/0        ena        500K       2000K     2000K     0
```

Verification of MPLS TE Support

- Check MPLS TE support on each router:

```
PE1#show mpls traffic-eng tunnels summary
Signalling  Summary:
    LSP Tunnels Process:            running
    Passive LSP Listener:           running
    RSVP Process:                   running
    Forwarding:                     enabled
    Periodic reoptimization:        every 3600 seconds, next in 2747 seconds
    Periodic FRR Promotion:         Not Running
    Periodic auto-bw collection:    disabled
    P2P:
      Head: 1 interfaces,   1 active signalling attempts, 1 established
            2 activations,  1 deactivations
            822 failed activations
            0 SSO recovery attempts, 0 SSO recovered
      Midpoints: 0, Tails: 1
    P2MP:
      Head: 0 interfaces,   0 active signalling attempts, 0 established
            0 sub-LSP activations,  0 sub-LSP deactivations
            0 LSP successful activations,  0 LSP deactivations
            0 SSO recovery attempts, LSP recovered: 0 full, 0 partial, 0 fail
      Midpoints: 0, Tails: 0
```

Verify Tunnel Interface Status

- Check tunnel interfaces on head-end routers:

```
PE1#show ip interface brief
Interface              IP-Address      OK? Method Status                Protocol
Ethernet1/0            10.12.0.1       YES NVRAM  up                    up
Tunnel10               10.0.0.1        YES TFTP   up                    up
```

```
PE1#show mpls traffic-eng tunnels brief
Signalling  Summary:
    LSP Tunnels Process:            running
    Passive LSP Listener:           running
    RSVP Process:                   running
    Forwarding:                     enabled
    Periodic reoptimization:        every 3600 seconds, next in 7 seconds
    Periodic FRR Promotion:         Not Running
    Periodic auto-bw collection:    disabled
P2P TUNNELS/LSPs:
TUNNEL NAME                     DESTINATION     UP IF      DOWN IF    STATE/PROT
Router1_t10                     10.0.0.4        -          Et1/0      up/up
Router4_t10                     10.0.0.1        Et1/0      -          up/up
Displayed 1 (of 1) heads, 0 (of 0) midpoints, 1 (of 1) tails
```

Verify Reachability

- Check ip routing table

```
PE1#show ip route
C        10.0.0.1/32 is directly connected, Loopback0
O        10.0.0.2/32 [110/11] via 10.12.0.2, 1d08h, Ethernet1/0
O        10.0.0.3/32 [110/21] via 10.12.0.2, 09:08:57, Ethernet1/0
O        10.0.0.4/32 [110/31] via 10.0.0.4, 00:15:34, Tunnel10
```

- Check LSP

```
PE1#traceroute mpls traffic-eng tunnel 10
Tracing MPLS TE Label Switched Path on Tunnel10, timeout is 2 seconds
Type escape sequence to abort.
  0 10.12.0.1 MRU 1500 [Labels: 16 Exp: 0]
L 1 10.12.0.2 MRU 1500 [Labels: 16 Exp: 0] 16 ms
L 2 10.23.0.2 MRU 1504 [Labels: implicit-null Exp: 0] 20 ms
! 3 10.34.0.2 20 ms
```

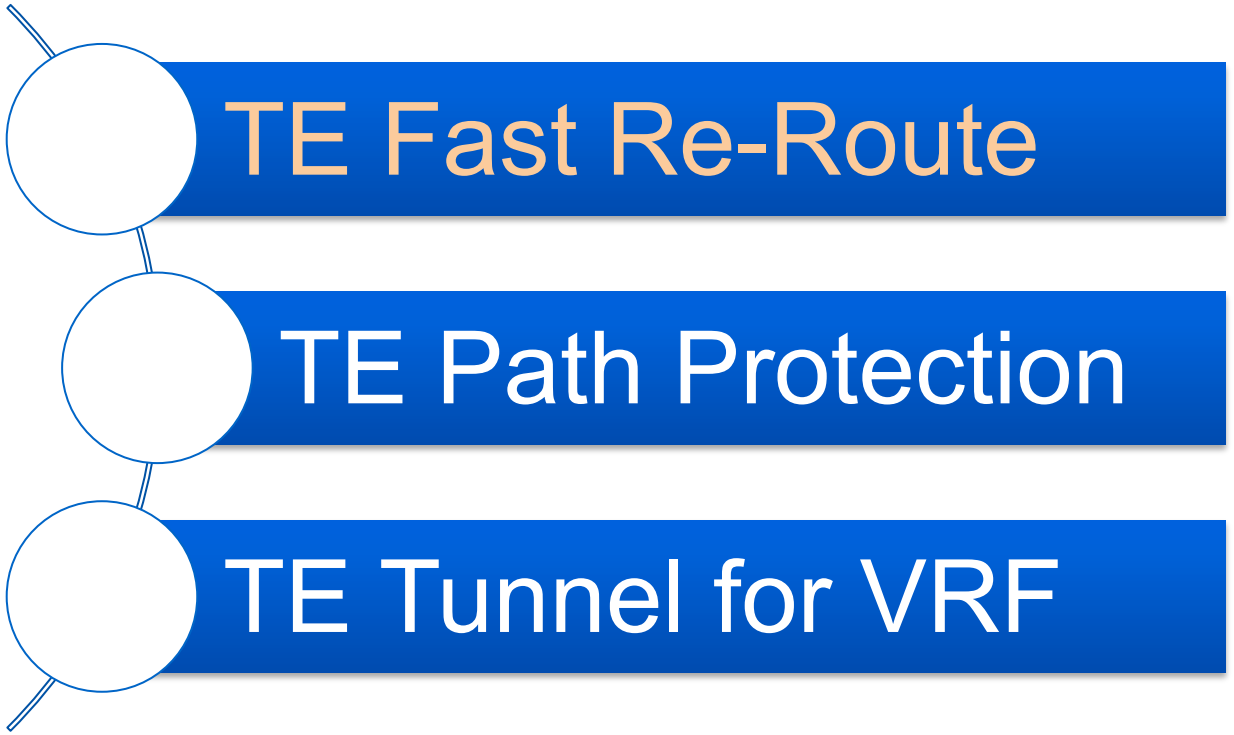
Questions?

APNIC

MPLS TE Services

APNIC

MPLS TE Services

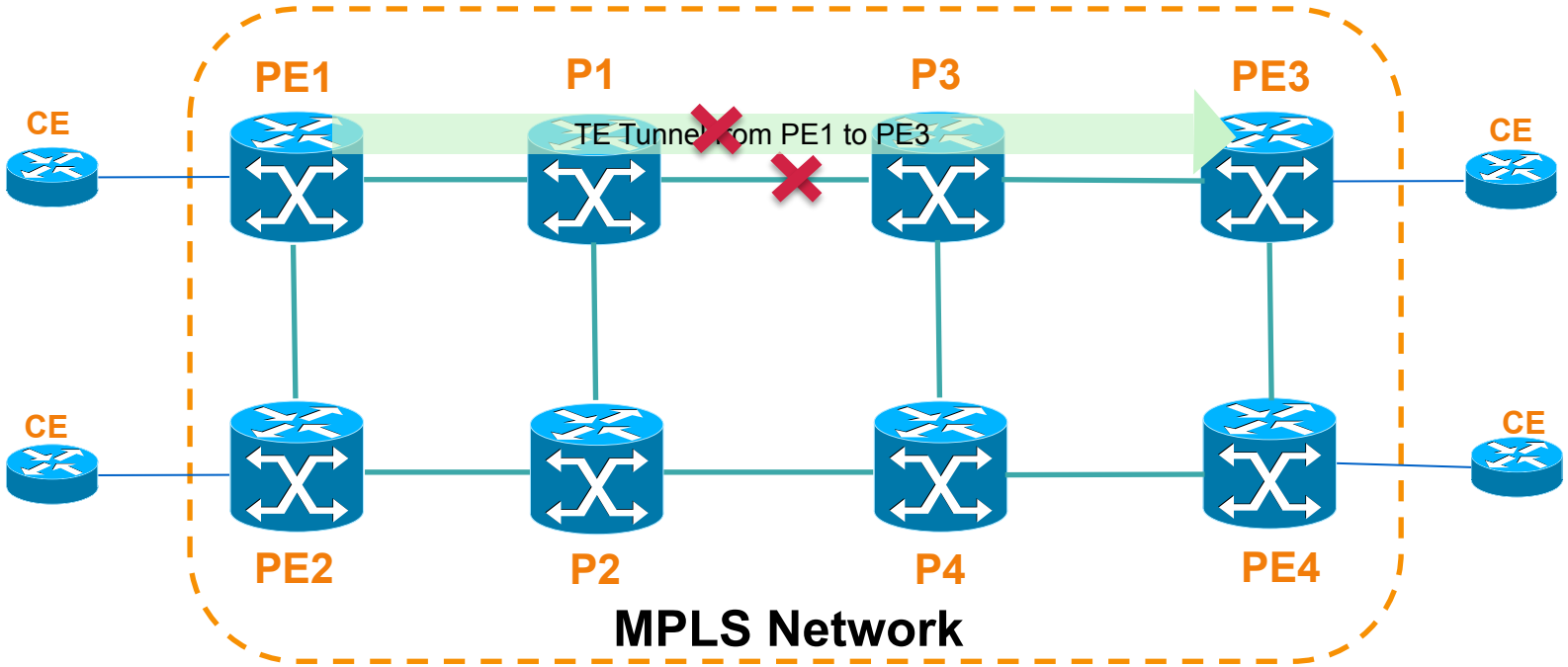


MPLS TE Fast Re-Route

- Traffic engineering fast reroute (TE FRR) provides protection for MPLS TE tunnels.
- If a link or a node fails, FRR locally repairs the protected LSPs by rerouting them over backup tunnels that bypass failed links or node, minimizing traffic loss.
  1. Link Protection
  2. Node Protection

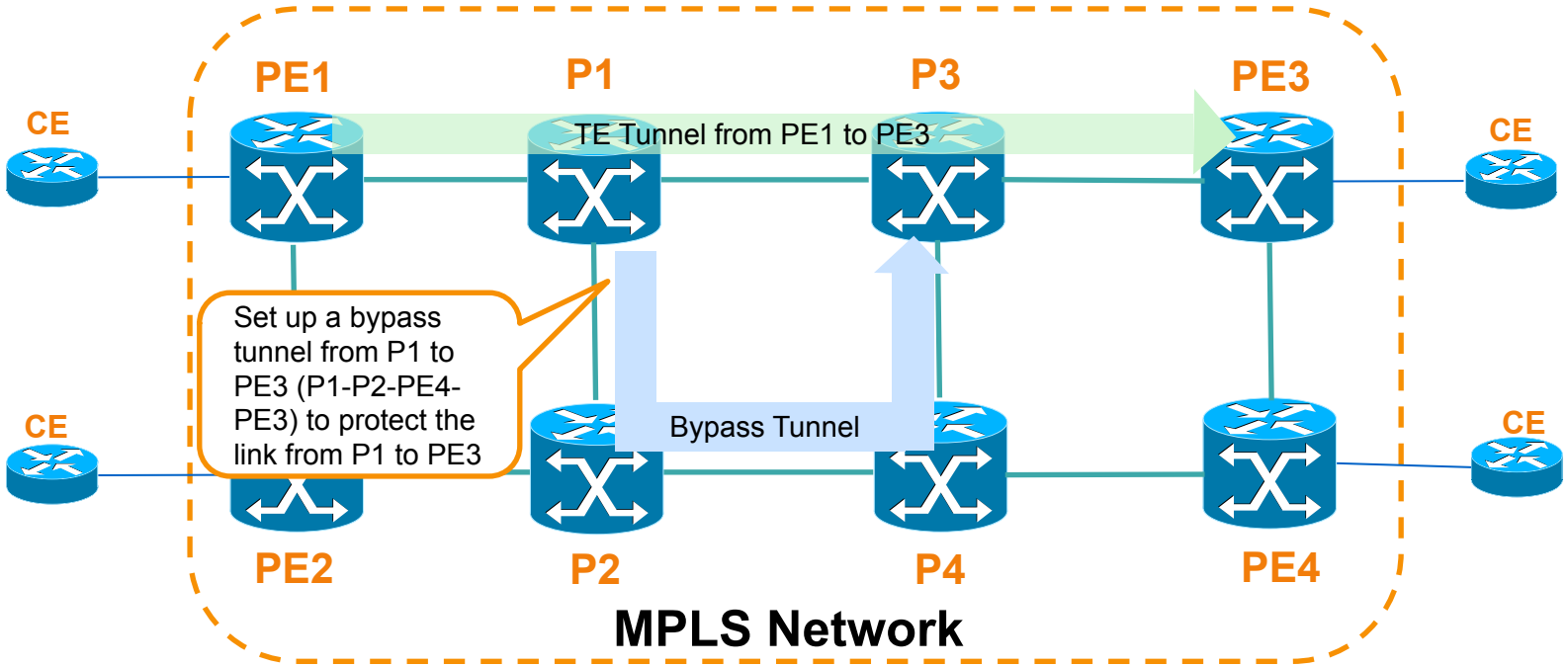
Link Failure

- If the link between P1 to P3 is down, the TE tunnel LSP will be torn down and take time to recompute a new path if autoroute.



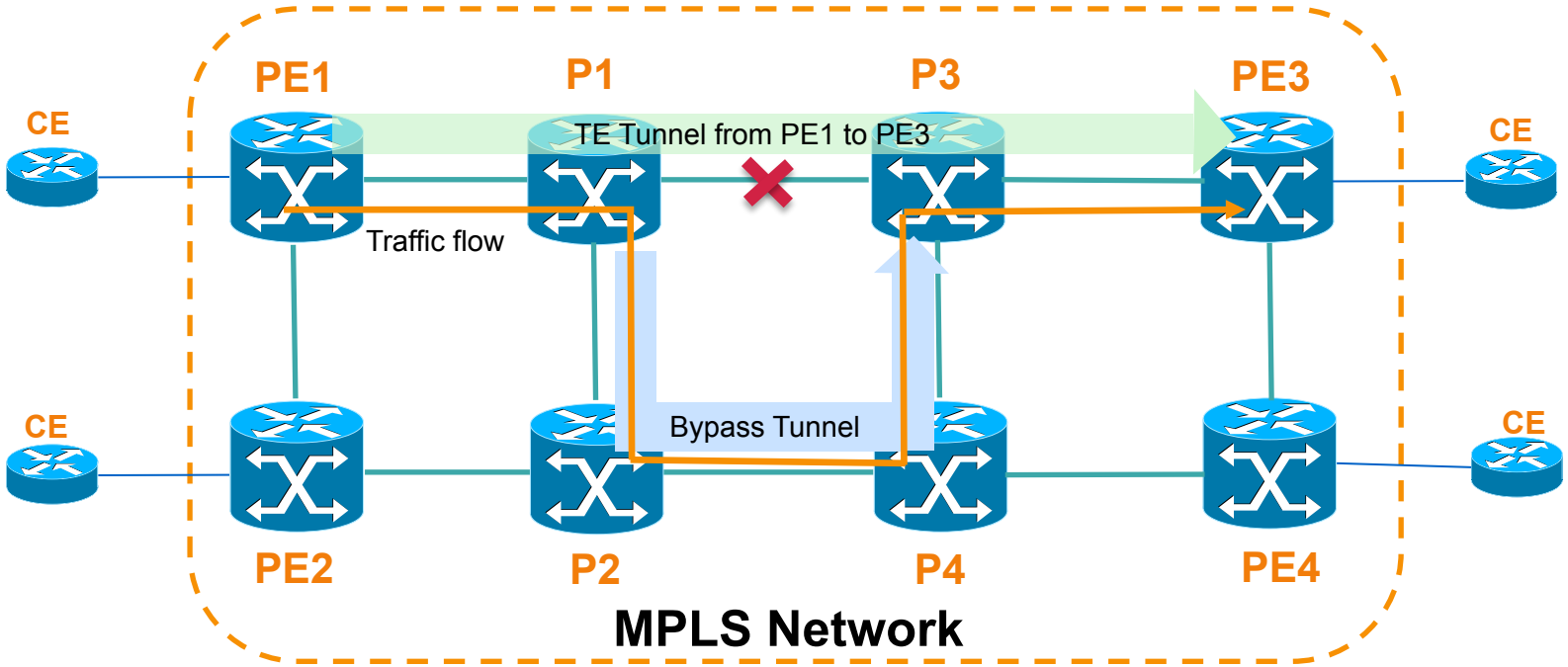
Fast Re-Route ---- Link Protection

- In link protection, the bypass tunnel has been set up to bypass only a single link of the LSP's path.



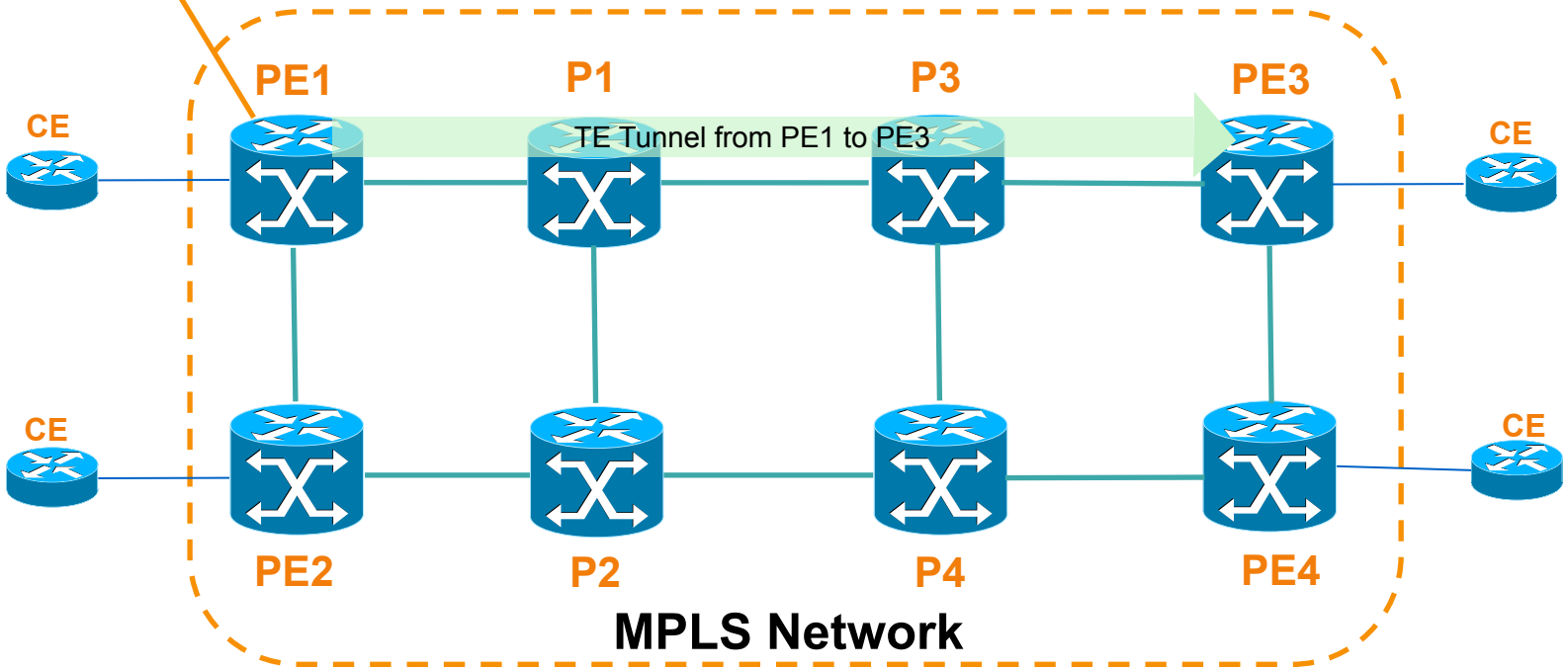
Fast Re-Route ---- Link Protection

- The bypass tunnel protects LSP if the protected link along their path fails by rerouting the LSP's traffic to the next hop (bypassing the failed link).



Configuration of Link Protection(1) - Primary Tunnel

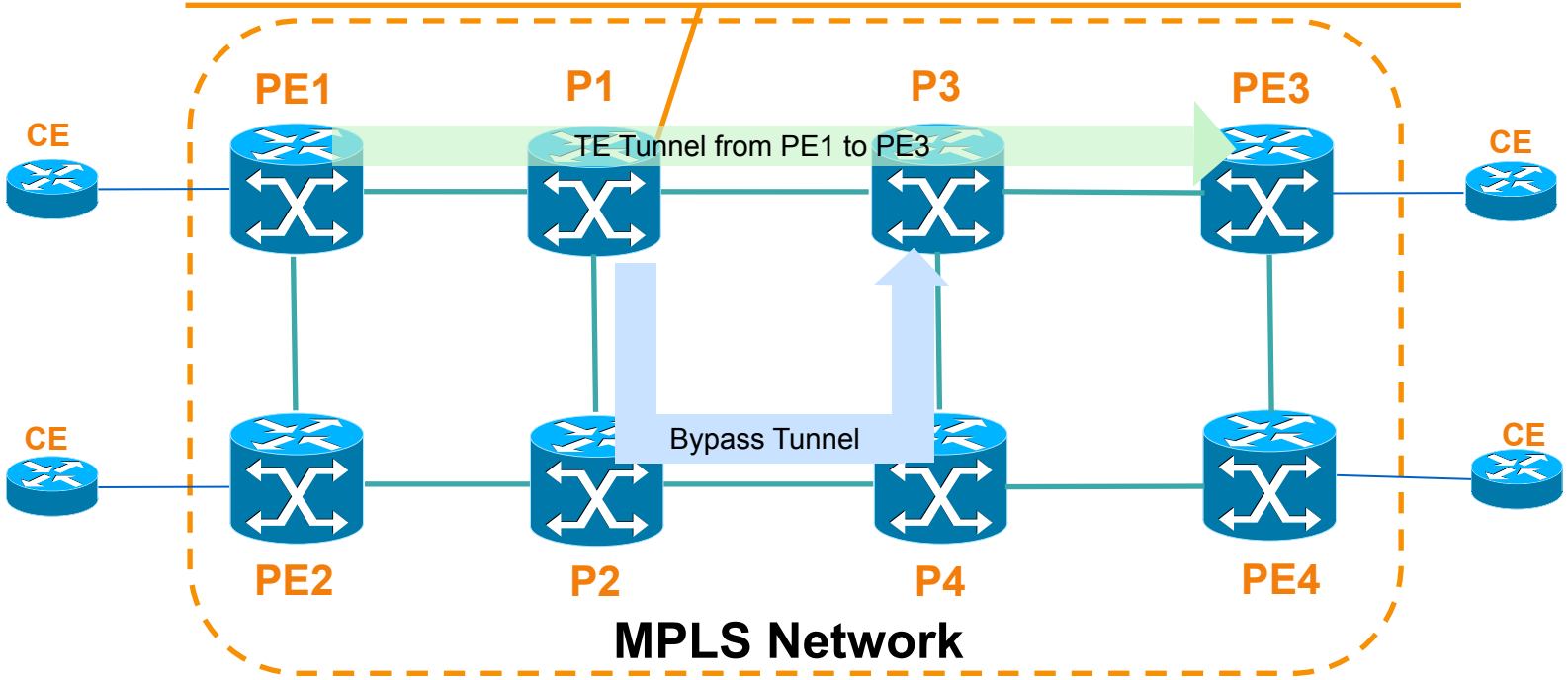
```
interface tunnel 100
ip unnumbered loopback0
tunnel mode mpls traffic-eng
tunnel destination 3.3.3.3
tunnel mpls traffic-eng bandwidth 256
tunnel mpls traffic-eng path-option 1 explicit name PRIMARY
tunnel mpls traffic-eng autoroute announce
```



Cisco IOS

Configuration of Link Protection(2) ---- Backup Tunnel

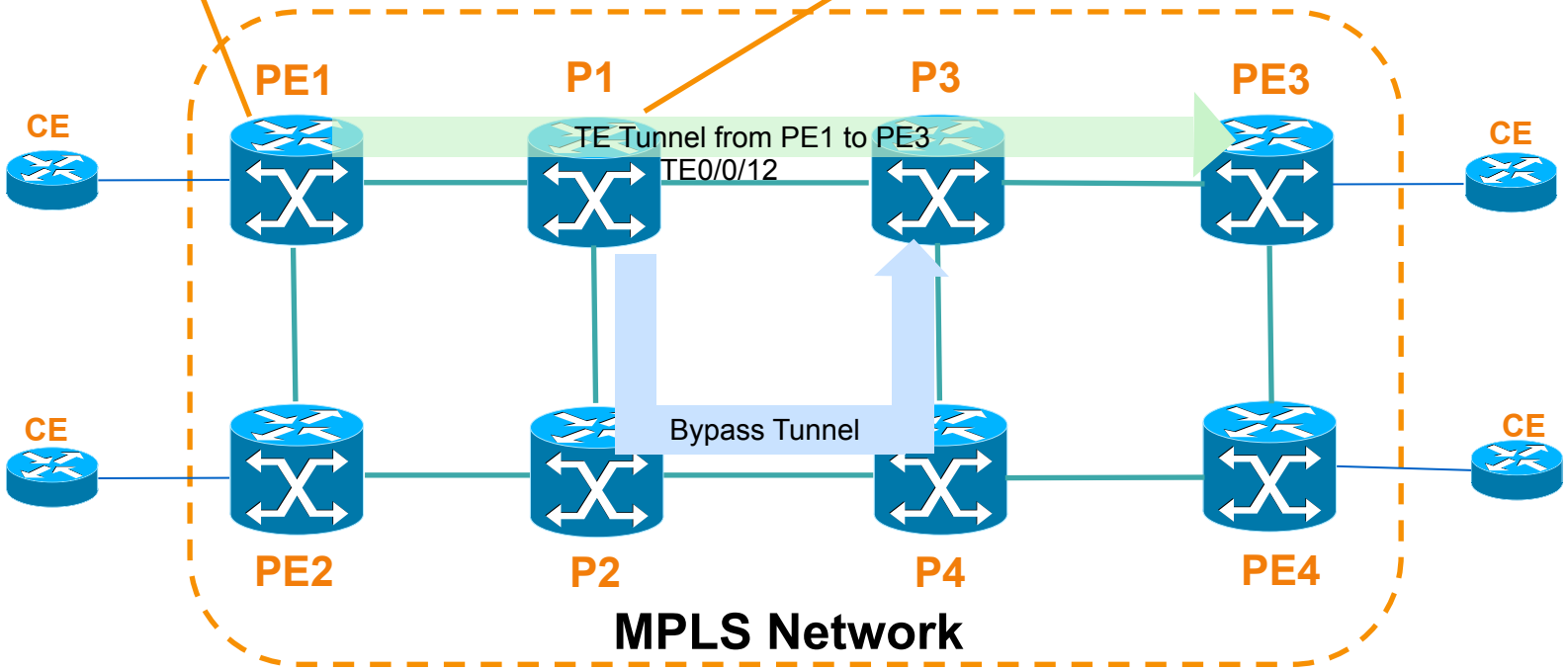
```
interface tunnel 1000
ip unnumbered loopback0
tunnel mode mpls traffic-eng
tunnel destination 3.3.3.3
tunnel mpls traffic-eng bandwidth 256
tunnel mpls traffic-eng path-option 1 explicit name LINK_PRO_P1_P3
tunnel mpls traffic-eng autoroute announce
```



Configuration of Link Protection(3) ---- Fast Re-Route

```
PE1(config)# interface tunnel 100
PE1(config-if)# tunnel mpls traffic-eng fast-reroute
```

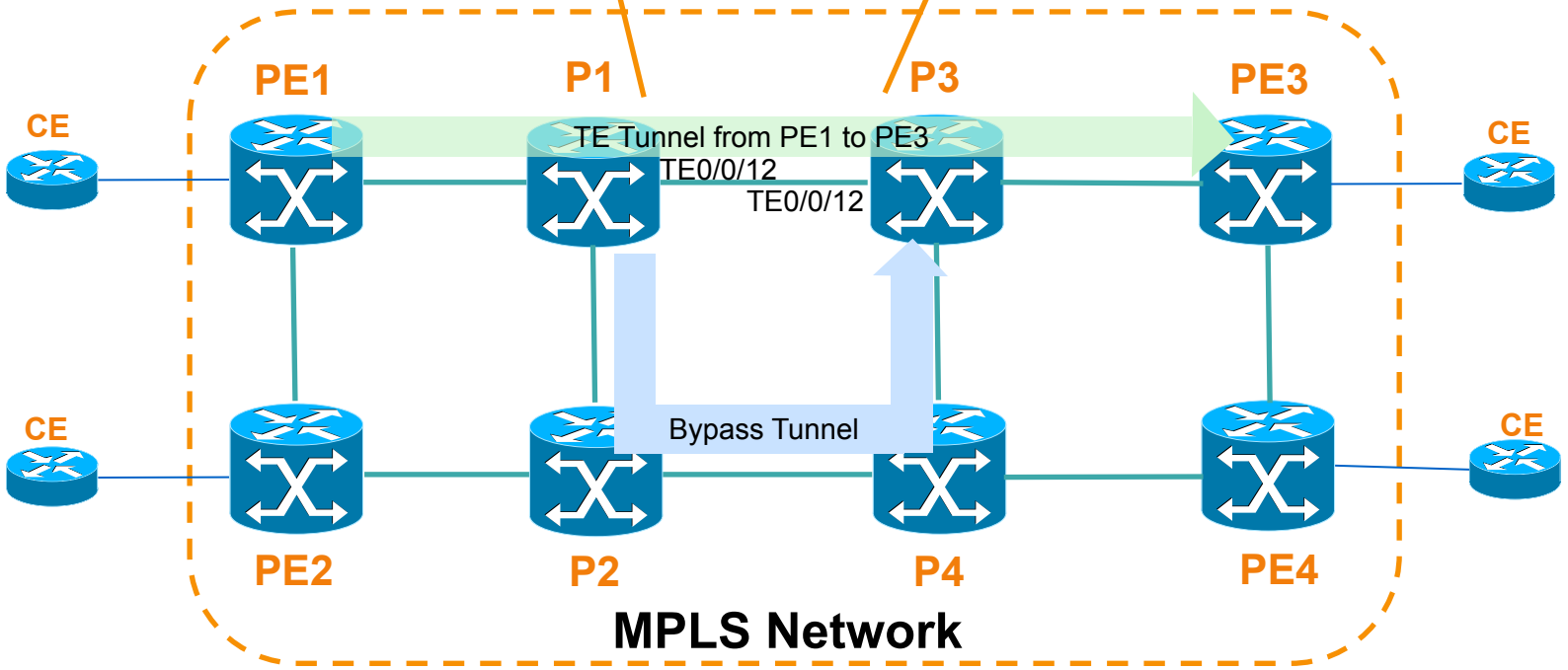
```
P1(config)# interface te 0/0/12
P1(config-if)# mpls traffic-eng backup-path tunnel 1000
```



Configuration of Link Protection(4) ---- RSVP Signalling Hello

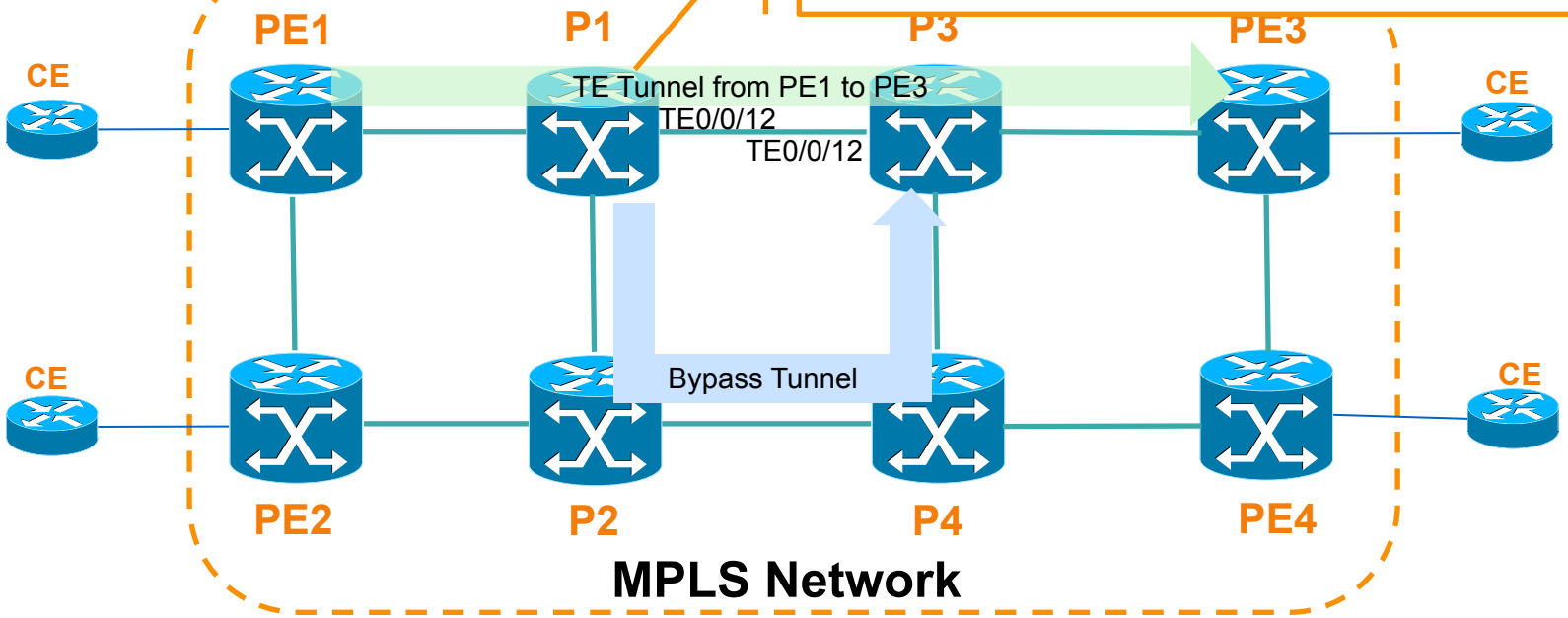
```
P1(config)# ip rsvp signalling hello
P1(config)# interface te 0/0/12
P1(config-if)# ip rsvp signalling hello
```

```
P3(config)# ip rsvp signalling hello
P3(config)# interface te 0/0/12
P3(config-if)# ip rsvp signalling hello
```



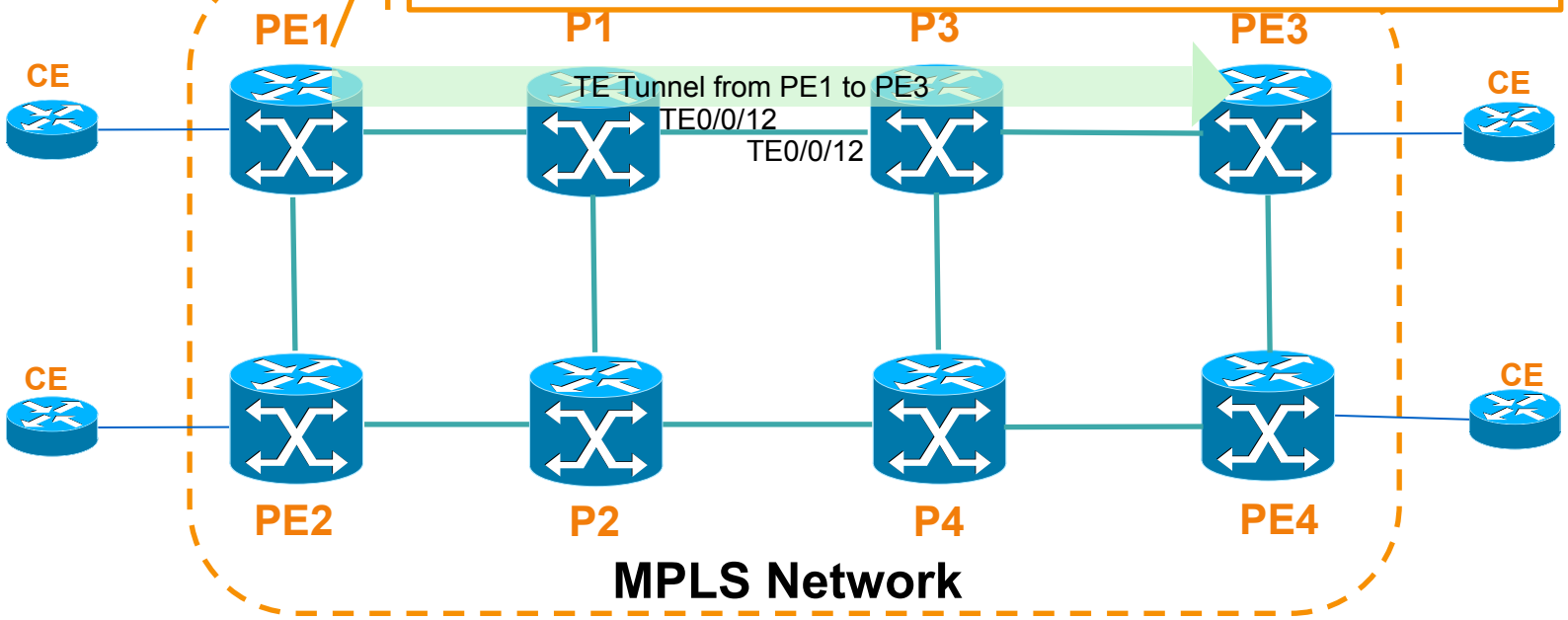
Check Fast Reroute

```
P1#show mpls traffic-eng tunnels backup
Router2_t1000
  LSP Head, Admin: up, Oper: up
  Tun ID: 1000, LSP ID: 134, Source: 172.16.15.2
  Destination: 172.16.15.5
  Fast Reroute Backup Provided:
    Protected i/fs: Te0/0/12
    Protected LSPs/Sub-LSPs: 1, Active: 0
    Backup BW: any pool unlimited; inuse: 256 kbps
    Backup flags: 0x0
```



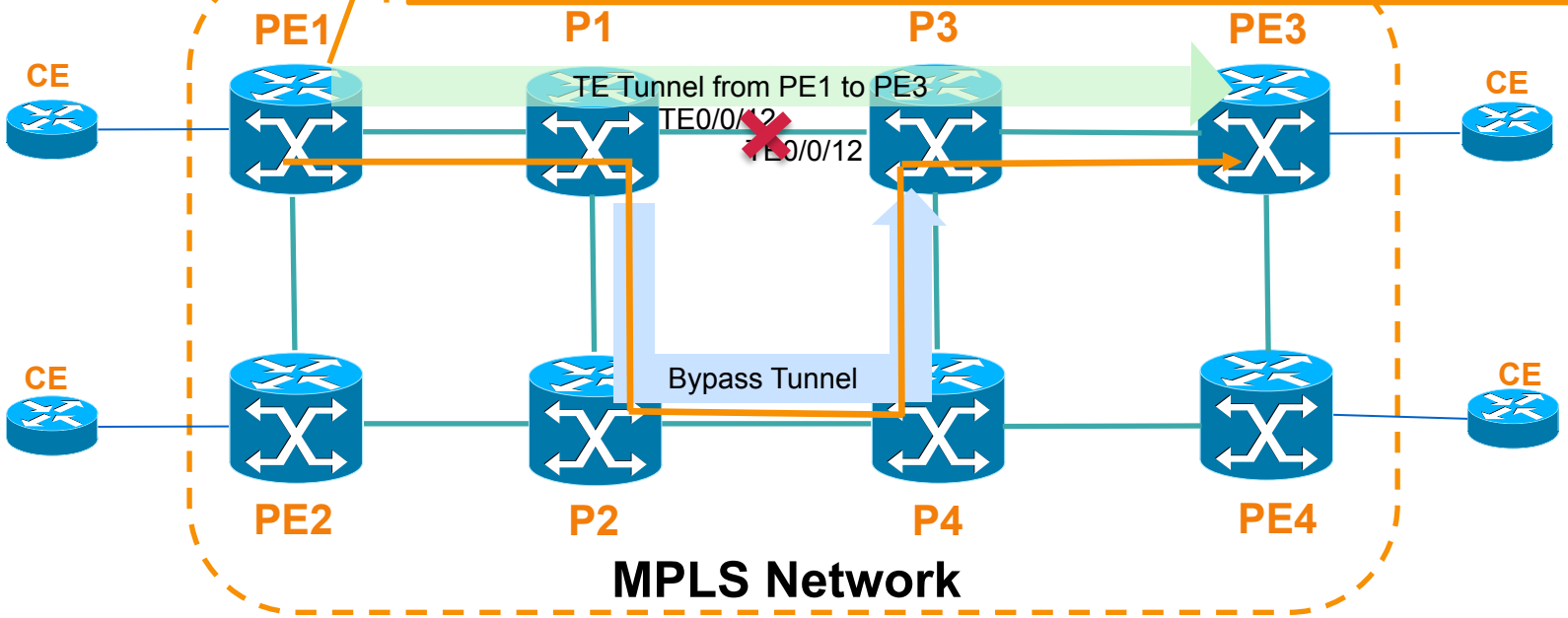
When the Protected Link is UP

```
Router1#traceroute mpls traffic-eng tunnel 100
Tracing MPLS TE Label Switched Path on Tunnel100, timeout is 2
seconds

Type escape sequence to abort.
  0 172.16.10.2 MRU 1512 [Labels: 201 Exp: 0]
L 1 172.16.10.1 MRU 1512 [Labels: 521 Exp: 0] 3 ms
L 2 172.16.12.2 MRU 1500 [Labels: implicit-null Exp: 0] 3 ms
! 3 172.16.10.26 3 ms
```



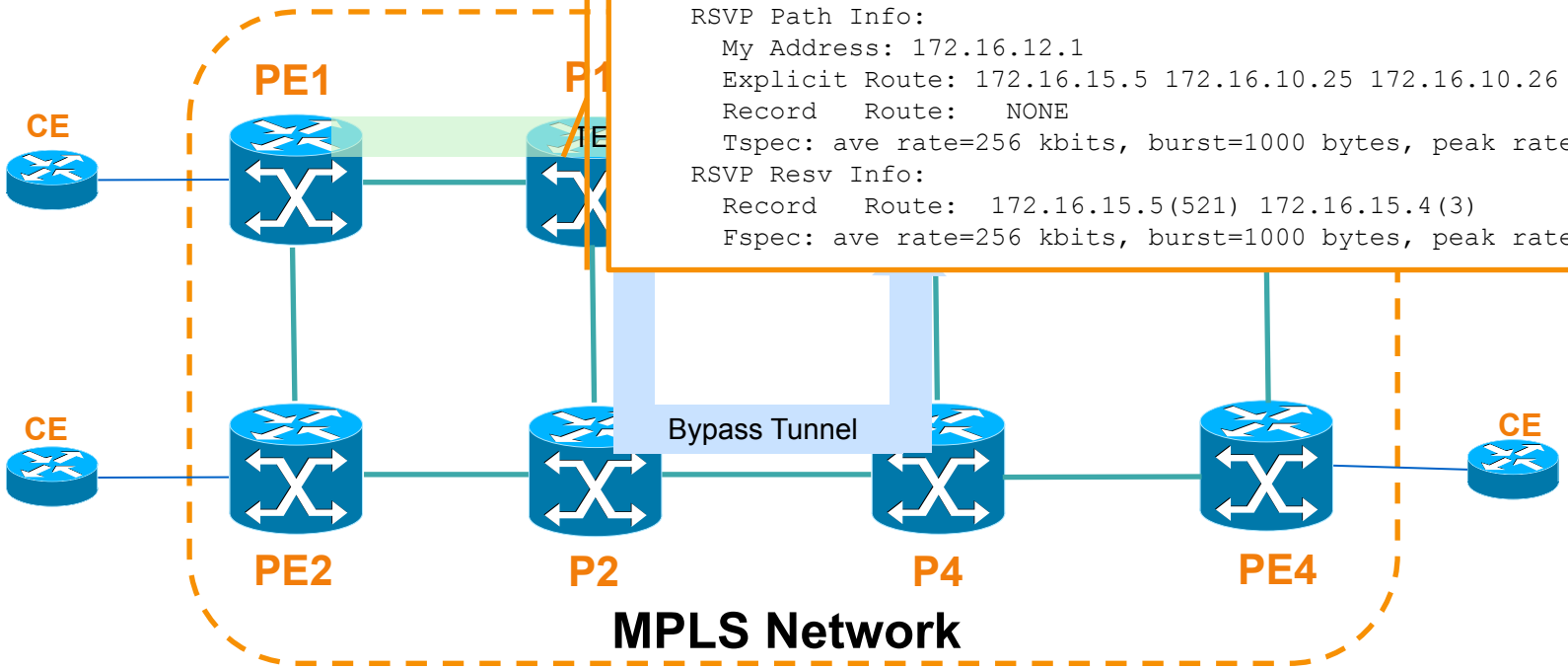
When the Protected Link is Down

```
Router1#traceroute mpls traffic-eng tunnel 100
Tracing MPLS TE Label Switched Path on Tunnel100, timeout is 2
seconds

Type escape sequence to abort.
  0 172.16.10.2 MRU 1512 [Labels: 201 Exp: 0]
L 1 172.16.10.1 MRU 1512 [Labels: 1108/521 Exp: 0/0] 4 ms
L 2 172.16.12.13 MRU 1512 [Labels: 816/521 Exp: 0/0] 3 ms
L 3 172.16.12.9 MRU 1512 [Labels: 521 Exp: 0] 3 ms
D 4 172.16.10.26 3 ms
! 5 172.16.10.26 3 ms
```



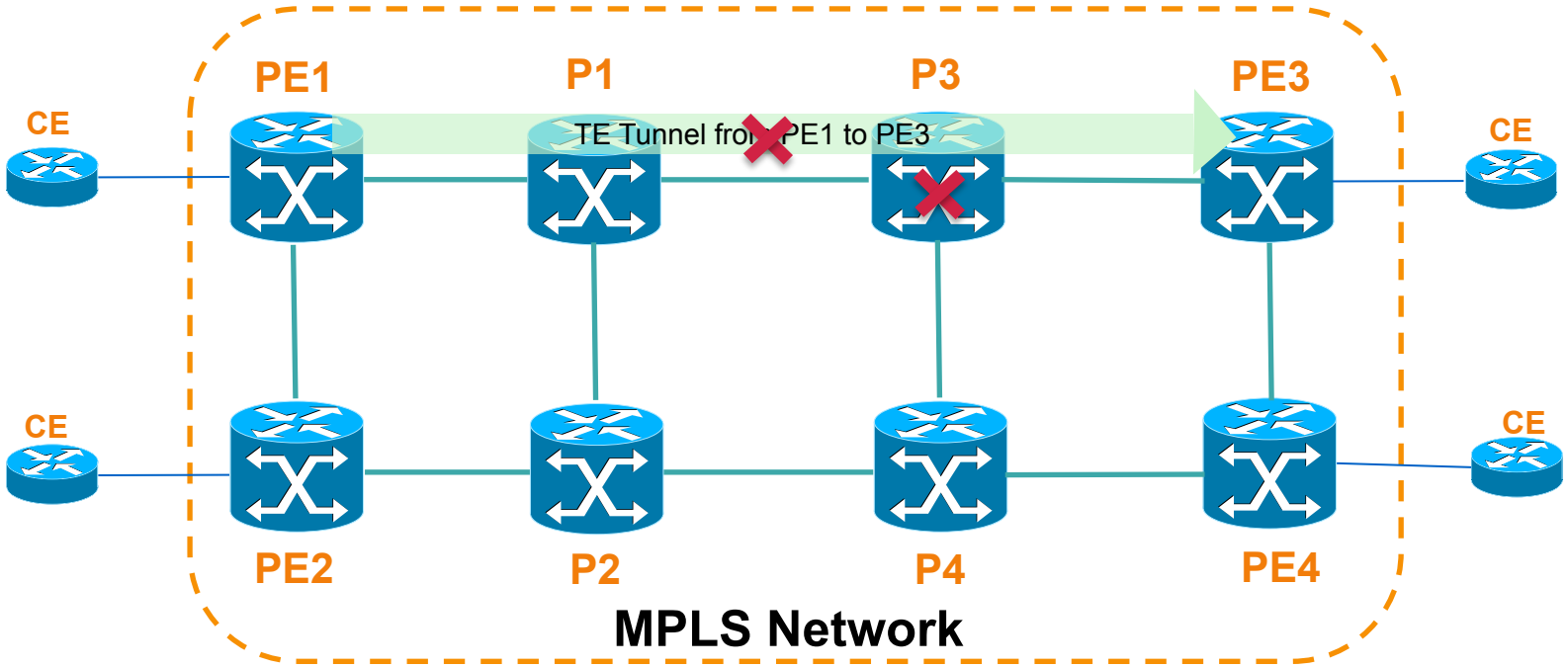
When the Protected Link is Down

```
P1#show mpls traffic-eng tunnels property fast-reroute

P2P TUNNELS/LSPs:

LSP Tunnel PE1_t100 is signalled, connection is up
  InLabel  : GigabitEthernet0/0/0, 201
  Prev Hop : 172.16.10.2
  OutLabel : TenGigabitEthernet0/0/12, 521
  Next Hop : 172.16.12.2
  FRR OutLabel : Tunnel1000, 521 (in use)
  RSVP Signalling Info:
       Src 172.16.15.1, Dst 172.16.15.4, Tun_Id 100, Tun_Instance 3649
    RSVP Path Info:
      My Address: 172.16.12.1
      Explicit Route: 172.16.15.5 172.16.10.25 172.16.10.26 172.16.15.4
      Record   Route:   NONE
      Tspec: ave rate=256 kbits, burst=1000 bytes, peak rate=256 kbits
    RSVP Resv Info:
      Record   Route:  172.16.15.5(521) 172.16.15.4(3)
      Fspec: ave rate=256 kbits, burst=1000 bytes, peak rate=256 kbits
```



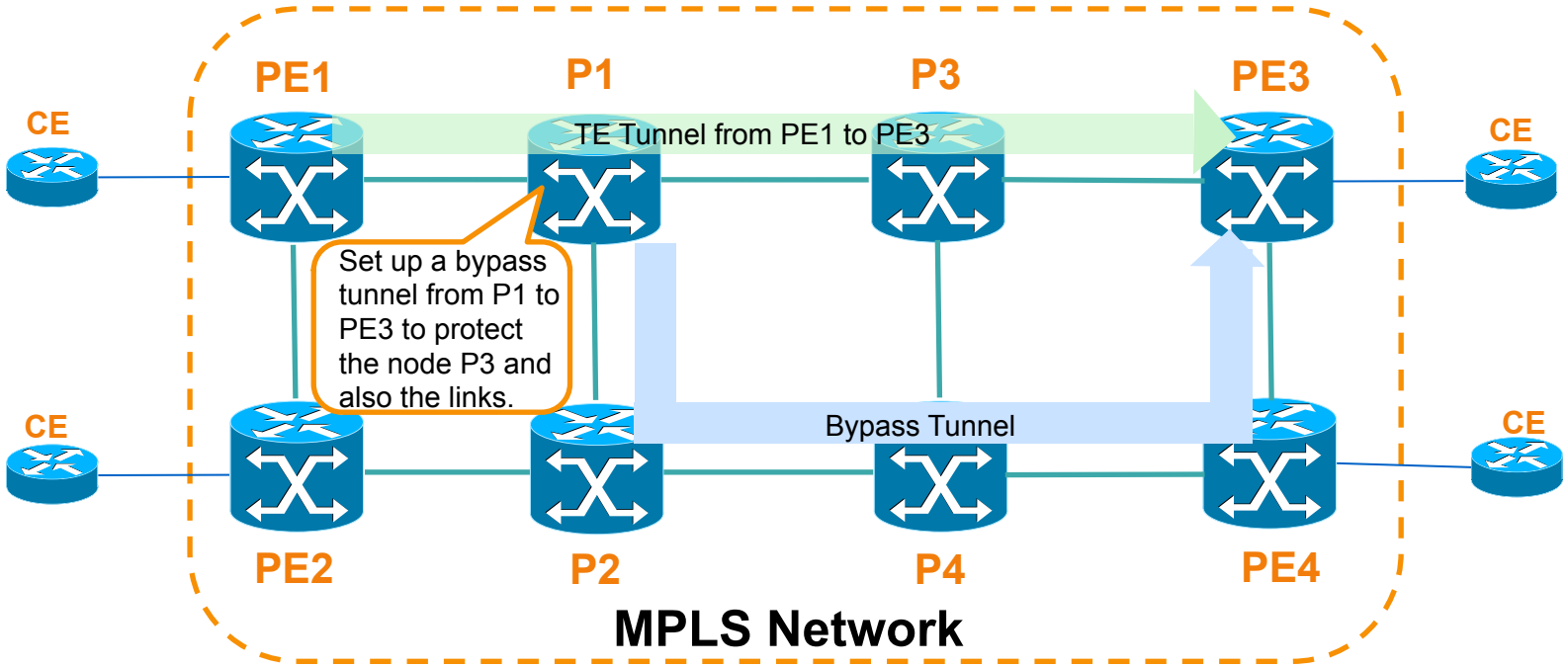
Node Failure

- If one node on the LSP fails, TE tunnel also will be torn down.



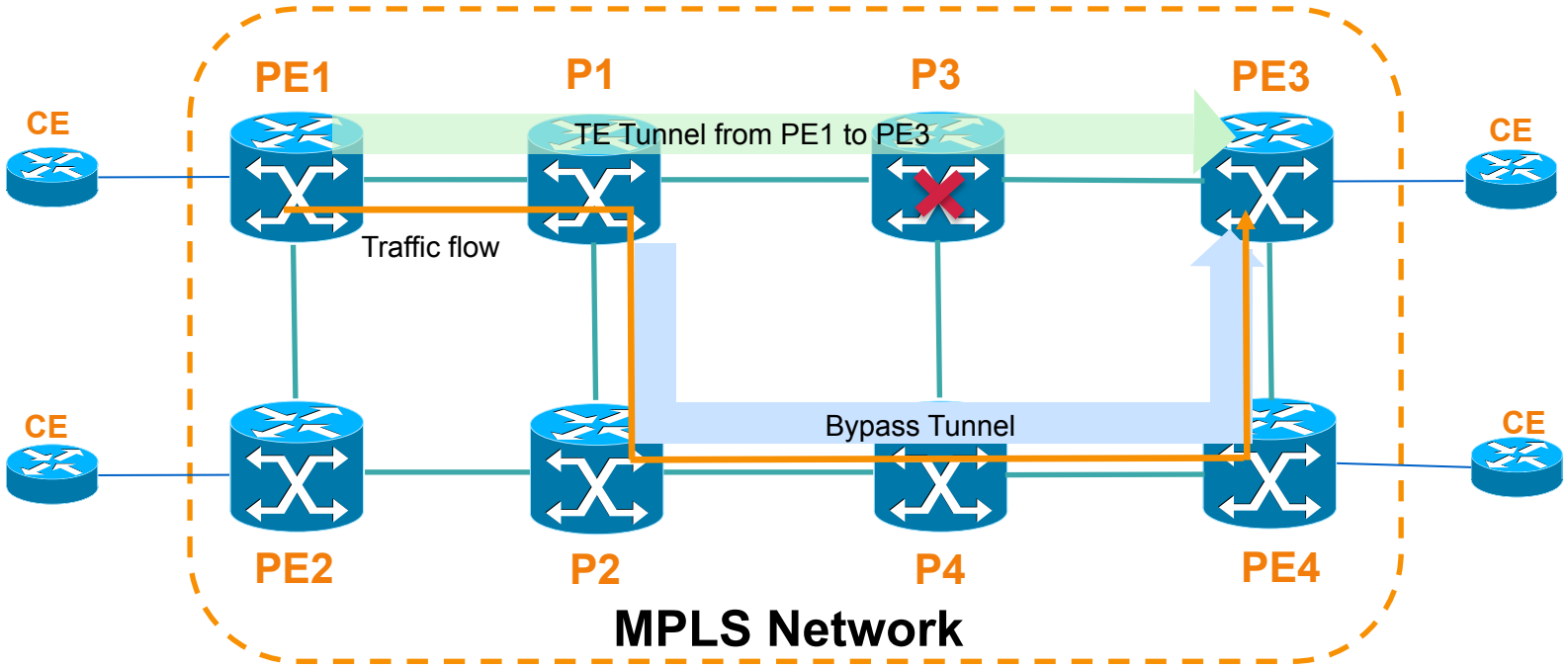
Fast Re-Route ---- Node Protection

- Node protection is similar to link protection in most ways, the backup tunnel is set up to bypass the protected node.



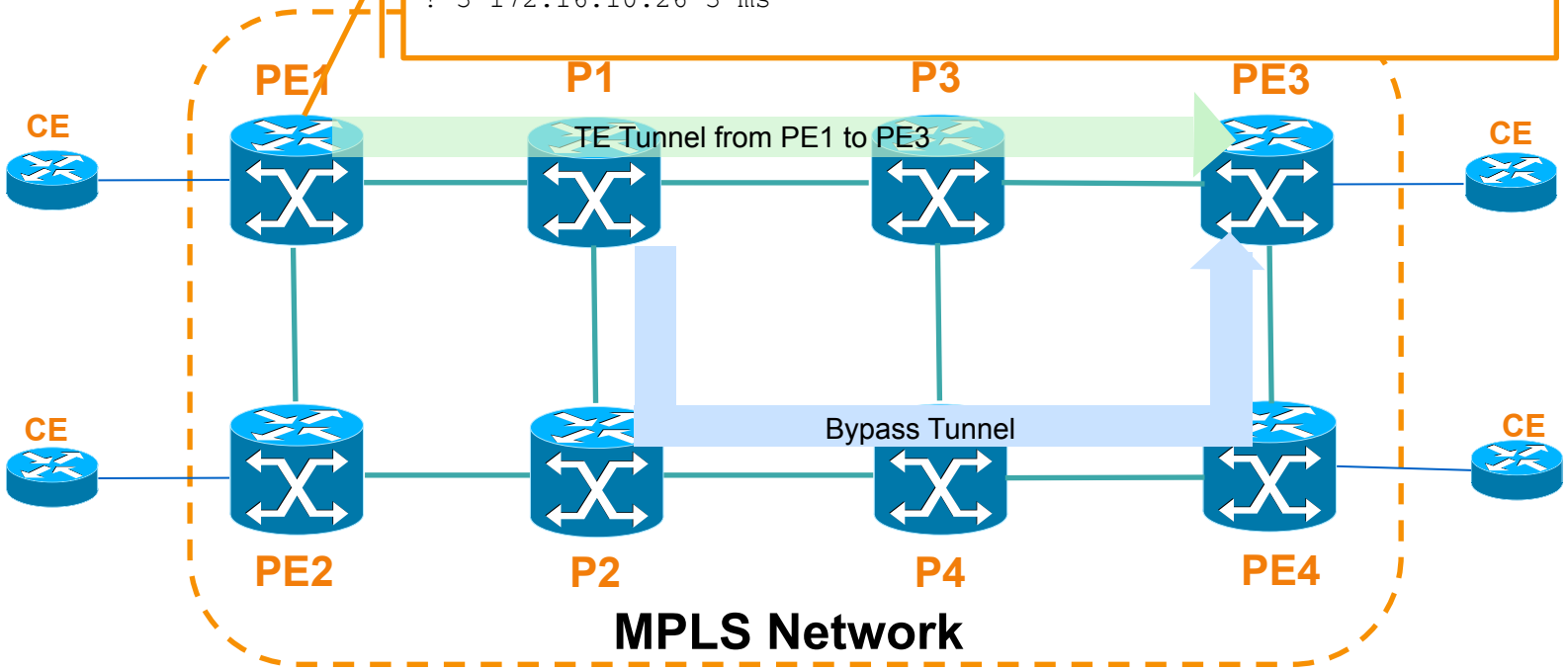
Fast Re-Route ---- Node Protection

- The backup tunnel can reroute the traffic to bypass the failure node.
- Backup tunnels also provide protection from link failures, because they bypass the failed link and the node.



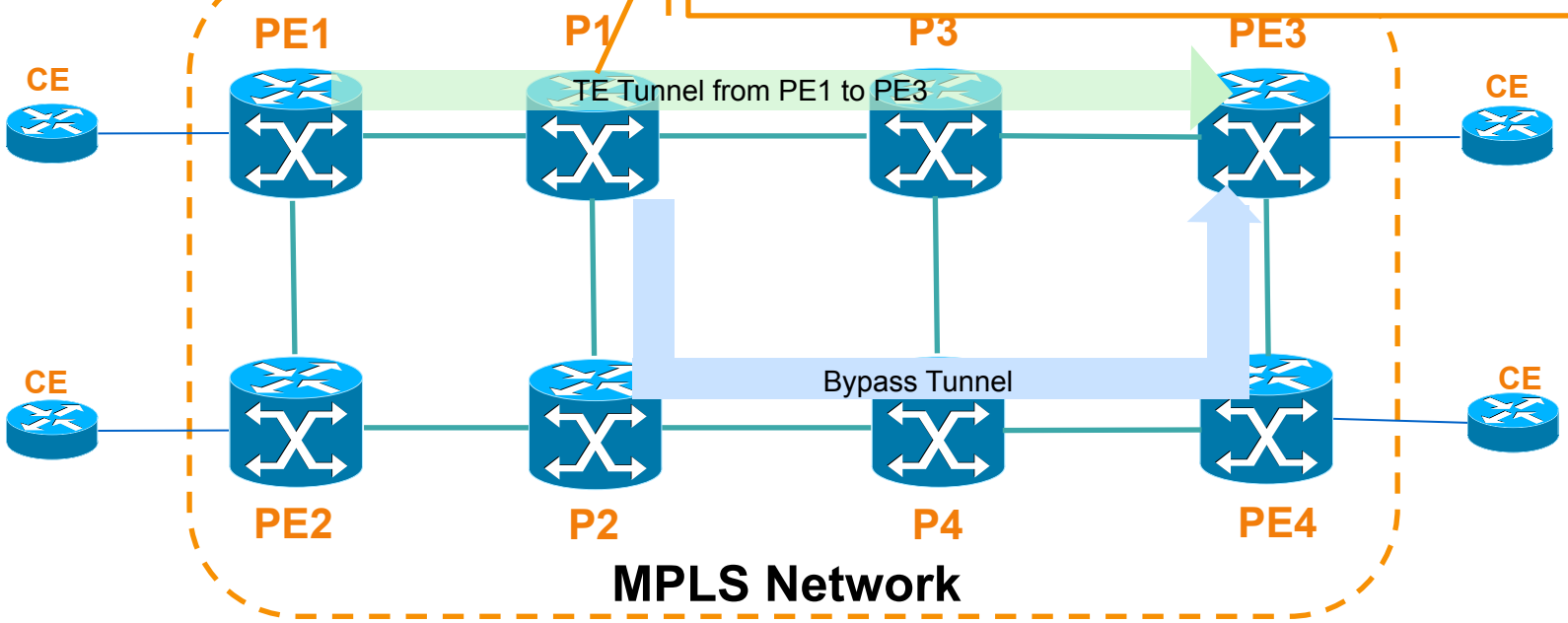
Primary LSP Before Any Failure

```
PE1#traceroute mpls traffic-eng tunnel 100
Tracing MPLS TE Label Switched Path on Tunnel100, timeout is 2
seconds

Type escape sequence to abort.
  0 172.16.10.2 MRU 1512 [Labels: 203 Exp: 0]
L 1 172.16.10.1 MRU 1512 [Labels: 522 Exp: 0] 3 ms
L 2 172.16.12.2 MRU 1500 [Labels: implicit-null Exp: 0] 3 ms
! 3 172.16.10.26 3 ms
```



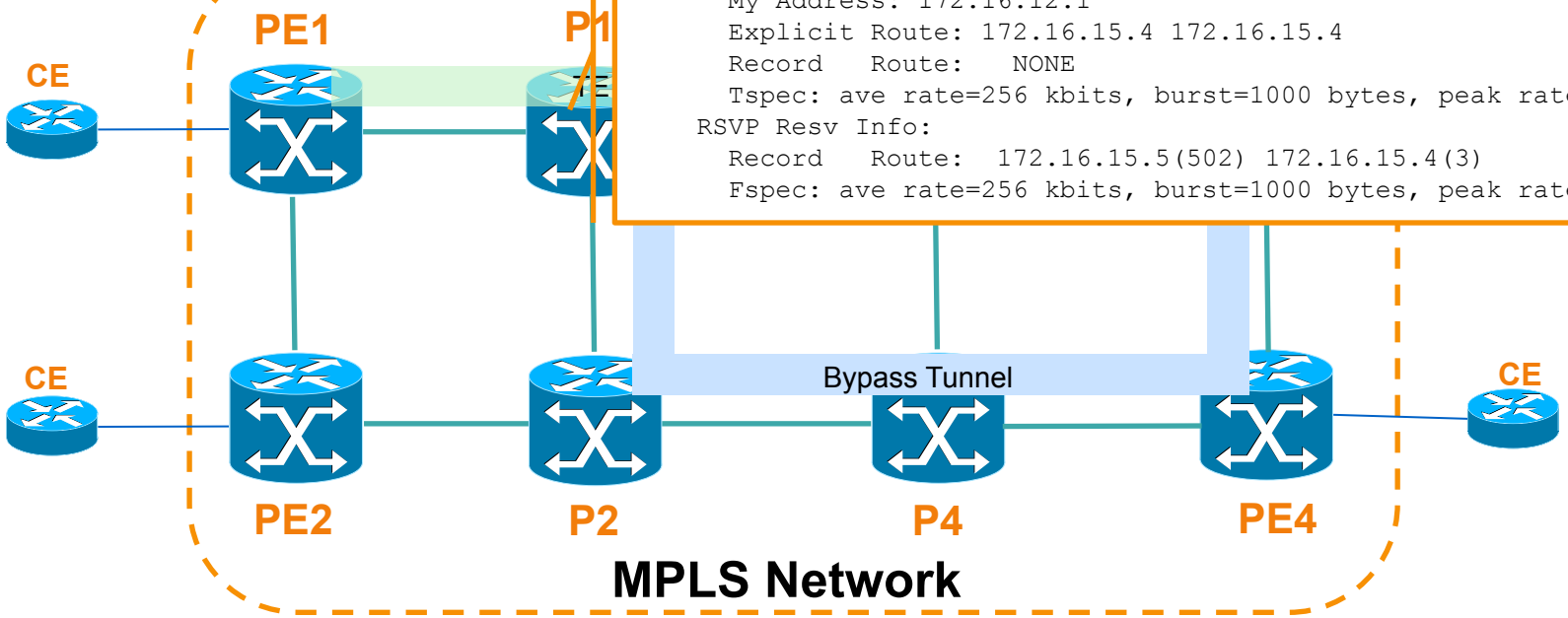
Bypass LSP Before Any Failure

```
P1#traceroute mpls traffic-eng tunnel 1500
Tracing MPLS TE Label Switched Path on Tunnel100, timeout is
2 seconds

Type escape sequence to abort.
  0 172.16.12.14 MRU 1512 [Labels: 1113 Exp: 0]
L 1 172.16.12.13 MRU 1512 [Labels: 804 Exp: 0] 4 ms
L 2 172.16.12.9 MRU 1512 [Labels: 616 Exp: 0] 3 ms
L 3 172.16.12.18 MRU 1500 [Labels: implicit-null Exp: 0] 4 ms
! 4 172.16.10.33 4 ms
```



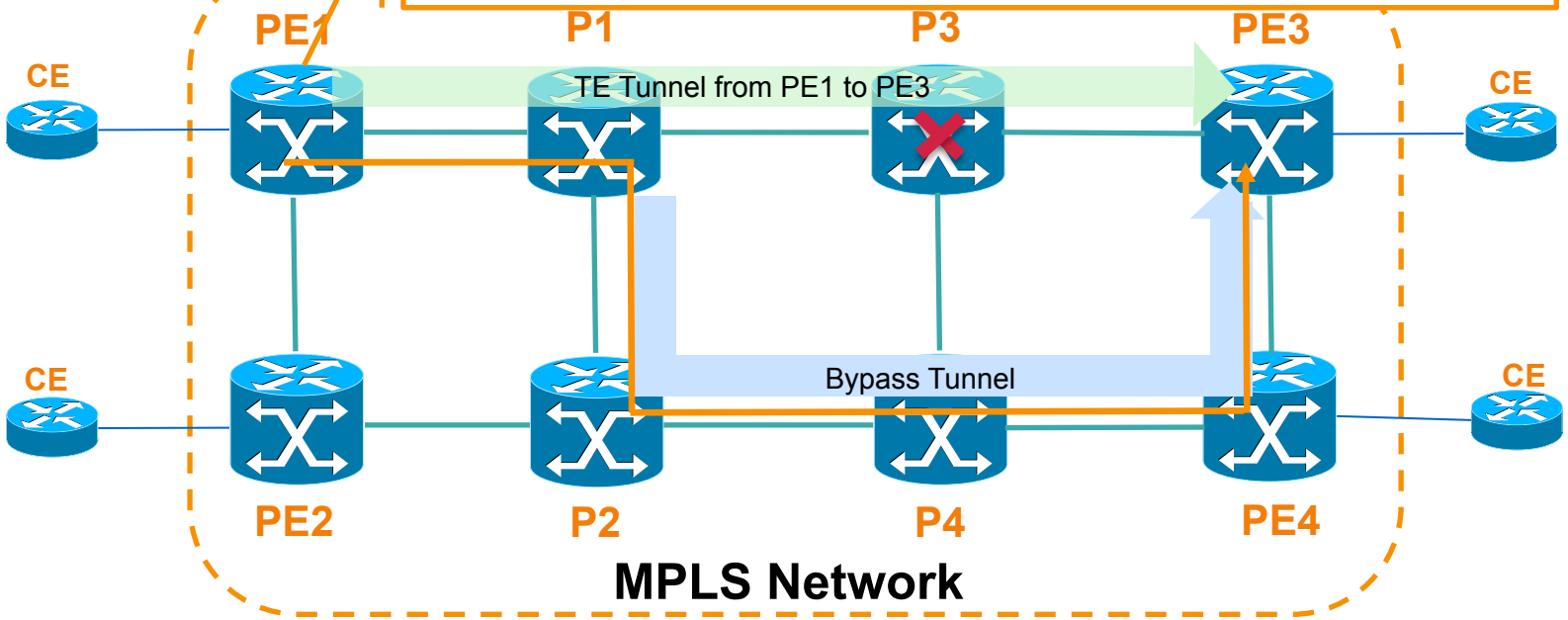
Check Fast Reroute after the Node is Down

```
P1#show mpls traffic-eng tunnels property fast-reroute

P2P TUNNELS/LSPs:

LSP Tunnel PE1_t100 is signalled, connection is up
  InLabel  : GigabitEthernet0/0/0, 202
  Prev Hop : 172.16.10.2
  OutLabel : TenGigabitEthernet0/0/12, 502
  Next Hop : 172.16.12.2
  FRR OutLabel : Tunnel1500, implicit-null (in use)
  RSVP Signalling Info:
       Src 172.16.15.1, Dst 172.16.15.4, Tun_Id 100, Tun_Instance 3684
    RSVP Path Info:
      My Address: 172.16.12.1
      Explicit Route: 172.16.15.4 172.16.15.4
      Record   Route:   NONE
      Tspec: ave rate=256 kbits, burst=1000 bytes, peak rate=256 kbits
    RSVP Resv Info:
      Record   Route:  172.16.15.5(502) 172.16.15.4(3)
      Fspec: ave rate=256 kbits, burst=1000 bytes, peak rate=256 kbits
```

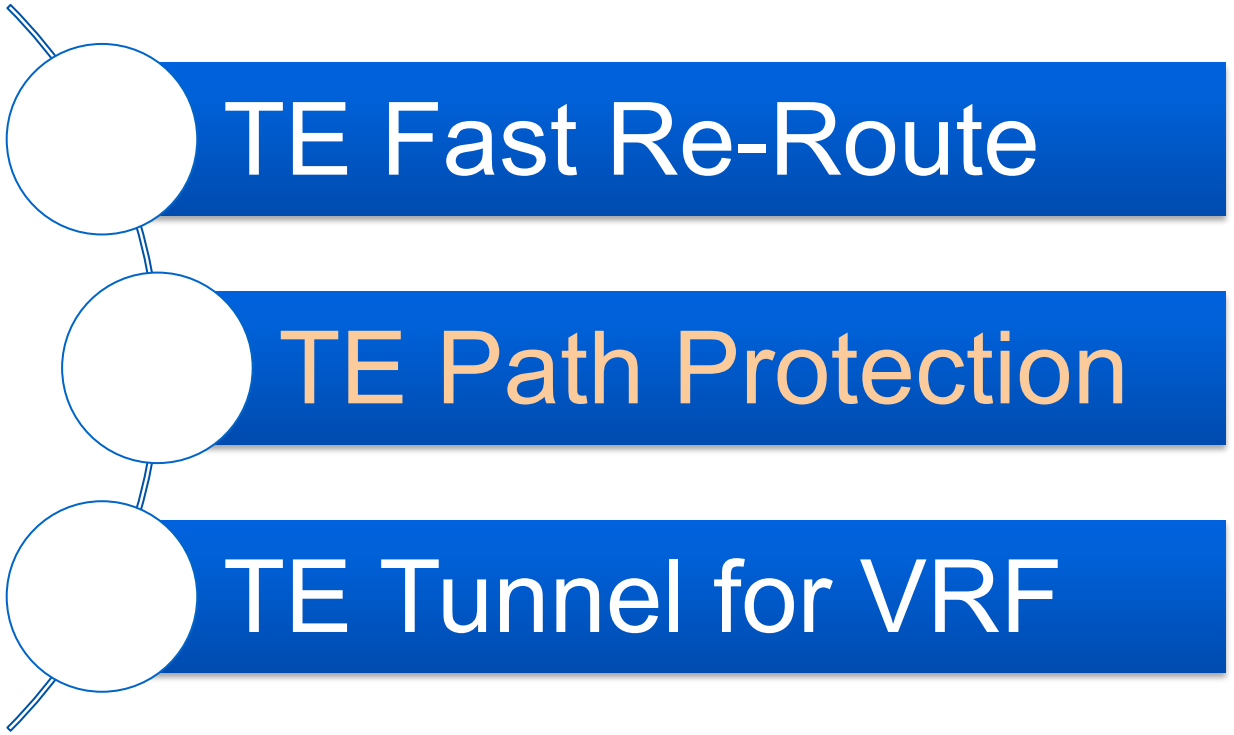


Check LSP after the Node is Down

```
PE1#traceroute mpls traffic-eng tunnel 100
Tracing MPLS TE Label Switched Path on Tunnel100, timeout is 2
seconds

Type escape sequence to abort.
  0 172.16.10.2 MRU 1512 [Labels: 202 Exp: 0]
L 1 172.16.10.1 MRU 1512 [Labels: implicit-null Exp: 0] 4 ms
L 2 172.16.12.13 MRU 1512 [Labels: 804 Exp: 0] 3 ms
L 3 172.16.12.9 MRU 1512 [Labels: 616 Exp: 0] 4 ms
L 4 172.16.12.18 MRU 1500 [Labels: implicit-null Exp: 0] 4 ms
! 5 172.16.10.33 4 ms
```



MPLS TE Services

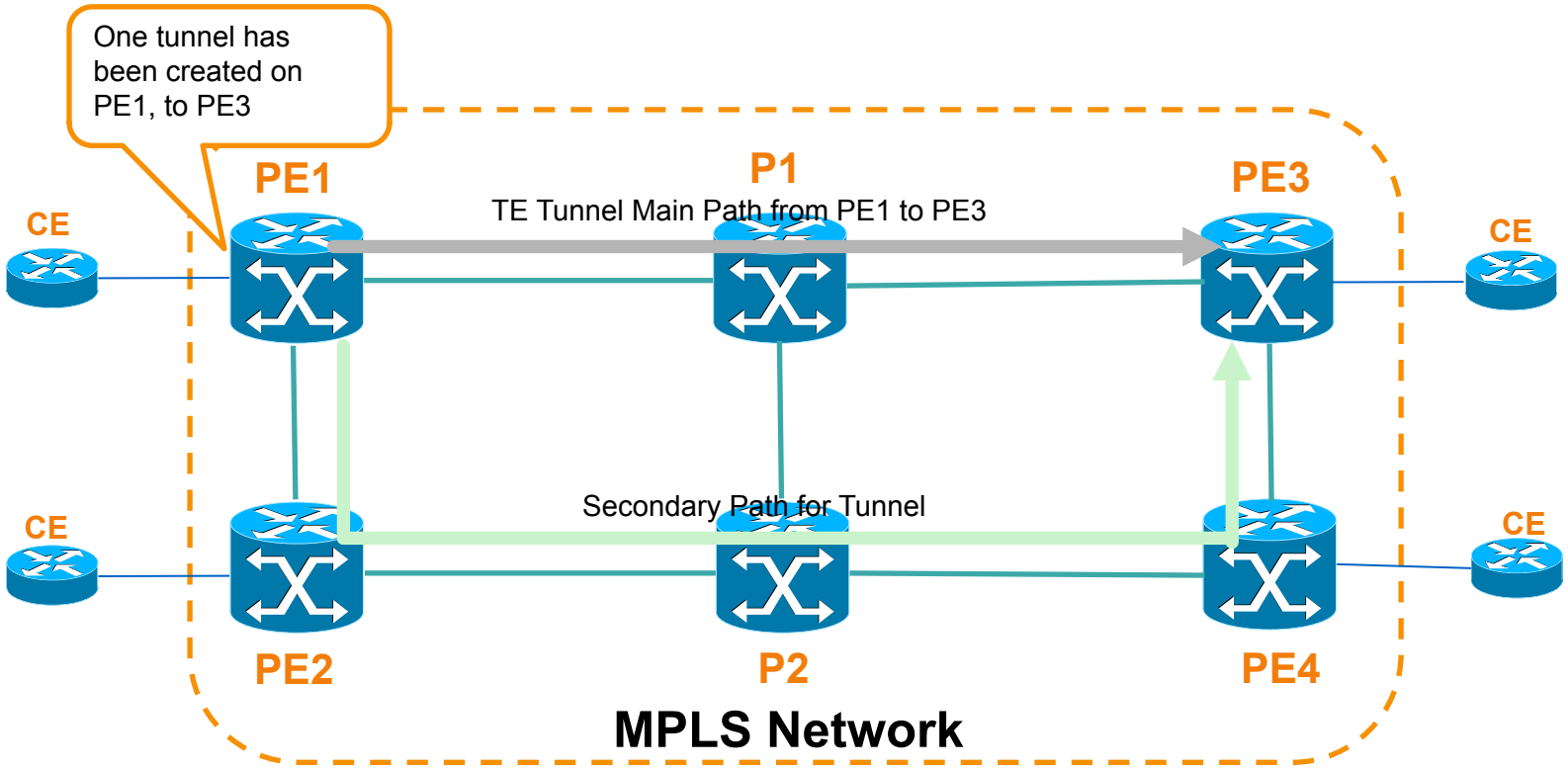


MPLS TE Path Protection

- Path protection provides an end-to-end failure recovery mechanism for MPLS TE tunnels.
- If the ingress node detects a failure of the primary LSP, it switches traffic to a backup LSP. After the primary LSP recovers, traffic switches back to the primary LSP.
- Path protection reduces the time required to recalculate a route in case of a failure within the MPLS tunnel.

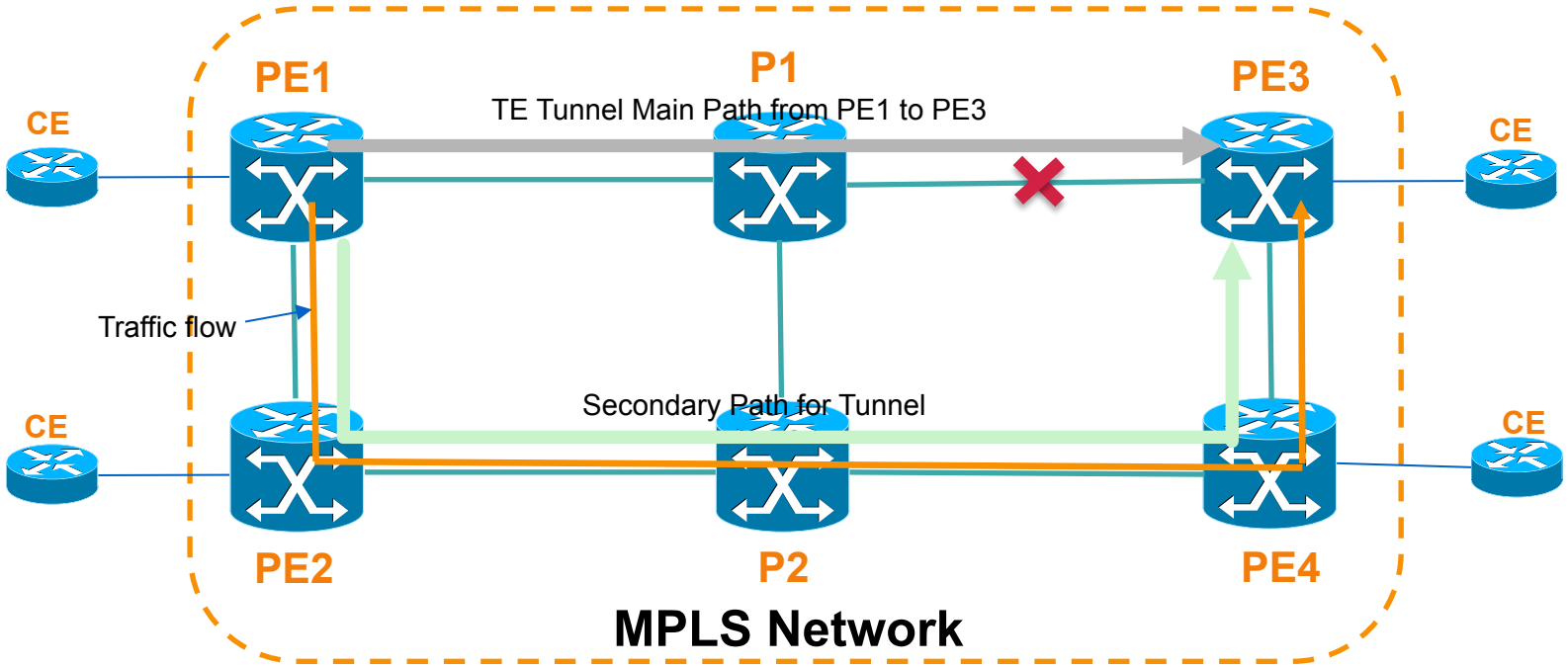
TE Path Protection Example

- Under on tunnel, both main path and secondary path are created.



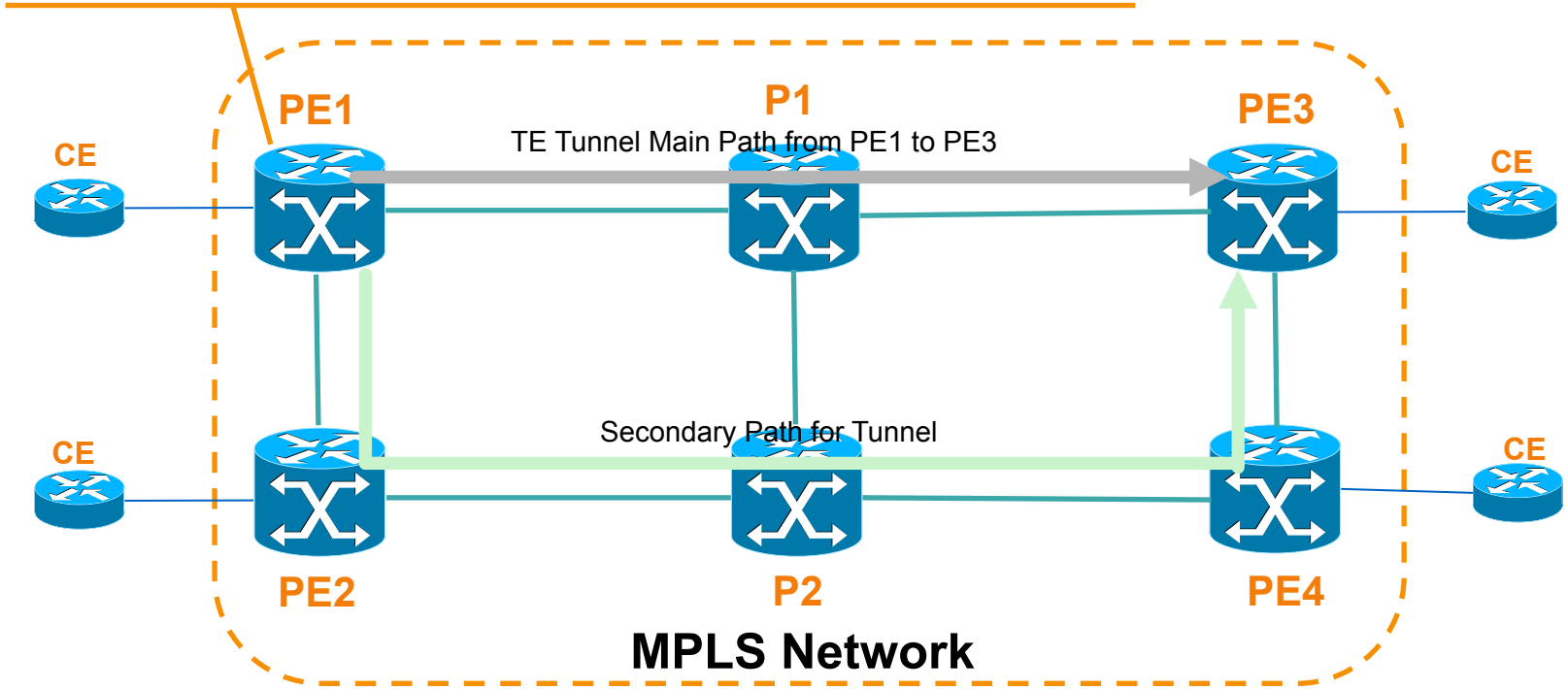
TE Path Protection Example

- When main path is not available, secondary path will be active. Traffic will go via secondary path.



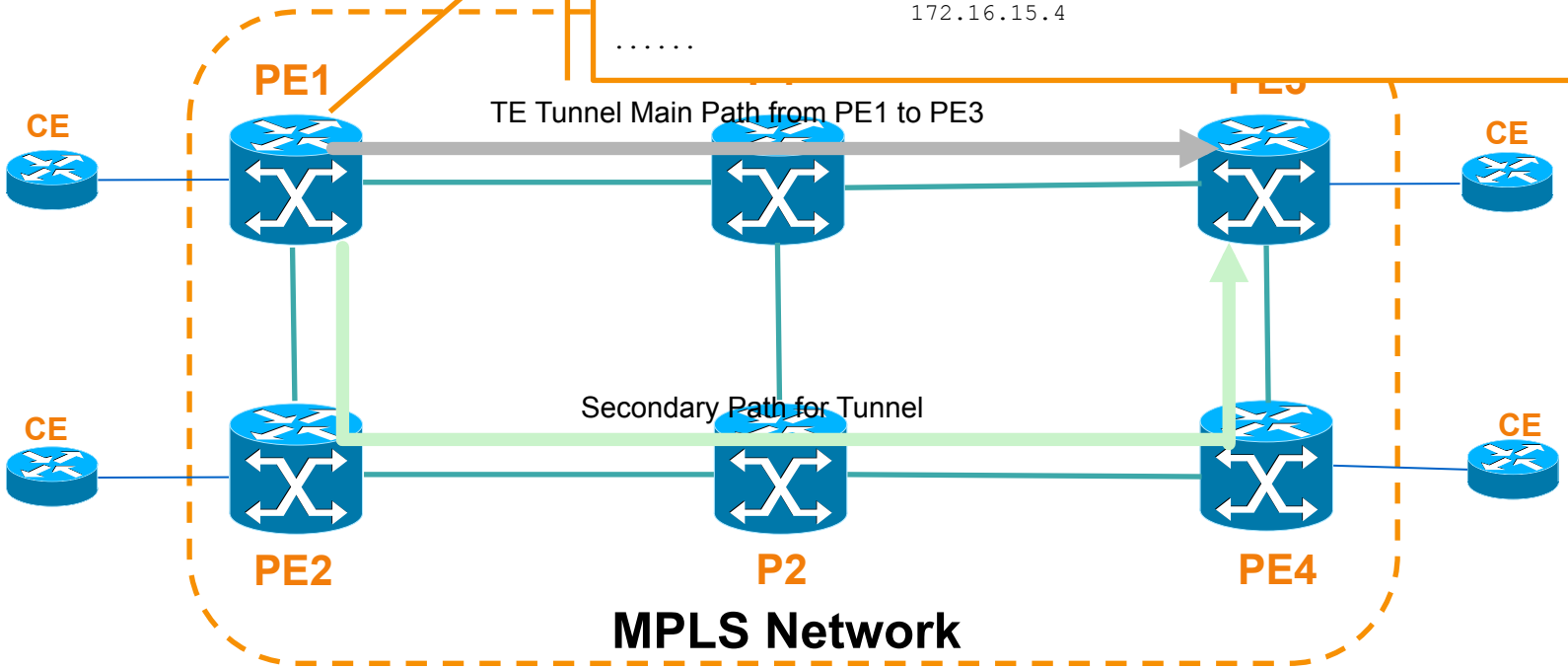
TE Path Protection Configuration

```
PE1(config)# interface tunnel 2000
PE1(config-if)# tunnel mpls traffic-eng path-option 10 explicit
name MAIN_PE1_PE3
PE1(config-if)# tunnel mpls traffic-eng path-option protect 10
explicit name PRO_PE1_PE3
```



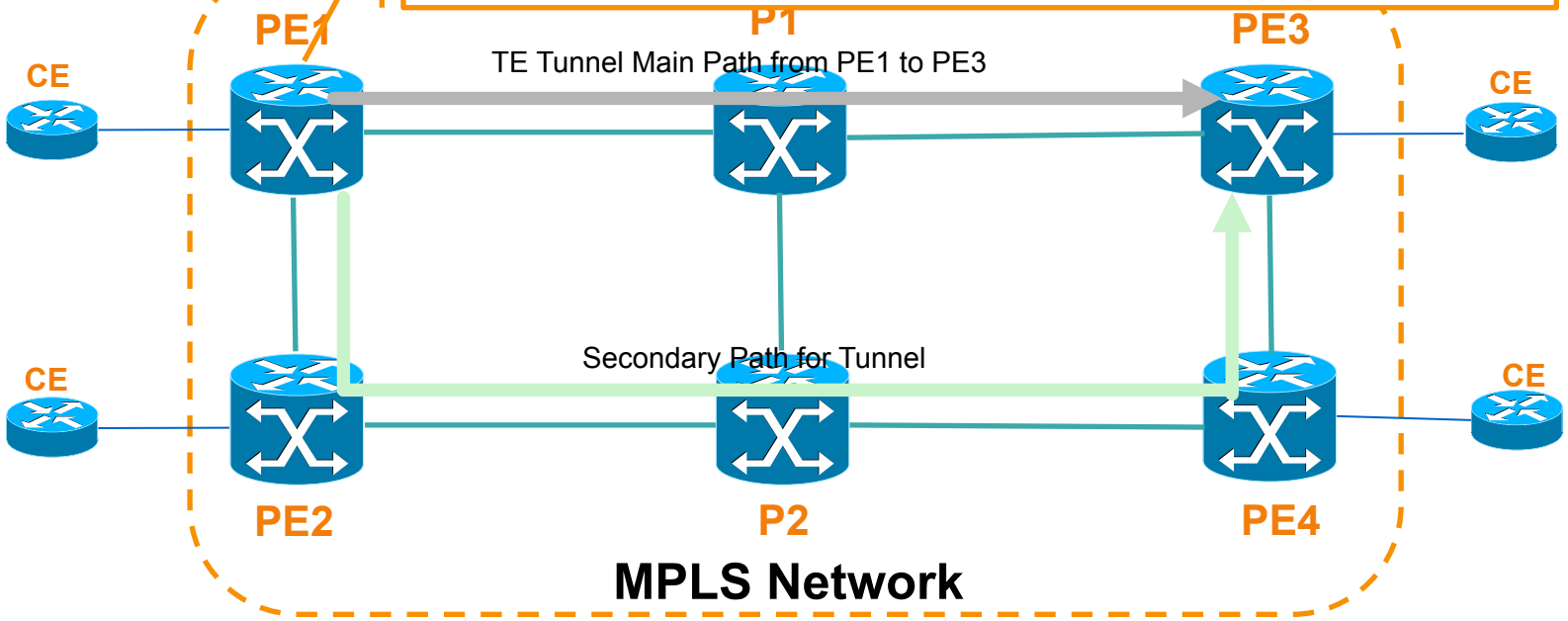
Check the Path Protection

```
PE1#show mpls traffic-eng tunnels protection | begin t2000
PE1_t2000  LSP Head, Tunnel2000, Admin: up, Oper: up
  Src 172.16.15.2, Dest 172.16.15.4, Instance 10
  Fast Reroute Protection: None
  Path Protection: 0 Common Link(s), 0 Common Node(s)
    Primary lsp path:172.16.12.1 172.16.12.2
                     172.16.10.25 172.16.10.26
                     172.16.15.4
    Protect lsp path:172.16.12.14 172.16.12.13
                     172.16.12.10 172.16.12.9
                     172.16.12.17 172.16.12.18
                     172.16.10.34 172.16.10.33
                     172.16.15.4
......
```



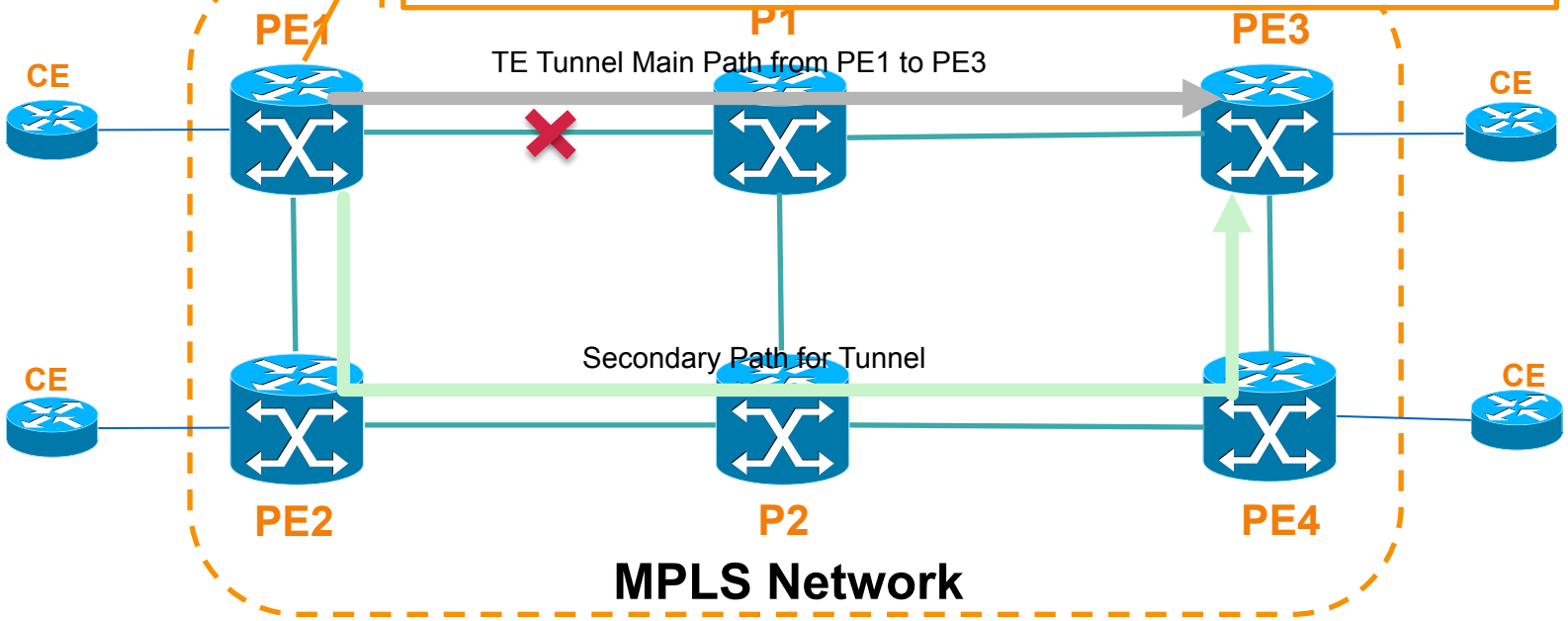
Check LSP before Any Failure

```
PE1#traceroute mpls traffic-eng tunnel 100
Tracing MPLS TE Label Switched Path on Tunnel100, timeout is 2
seconds
......
Type escape sequence to abort.
  0 172.16.12.1 MRU 1512 [Labels: 501 Exp: 0]
L 1 172.16.12.2 MRU 1500 [Labels: implicit-null Exp: 0] 5 ms
! 2 172.16.10.26 4 ms
```



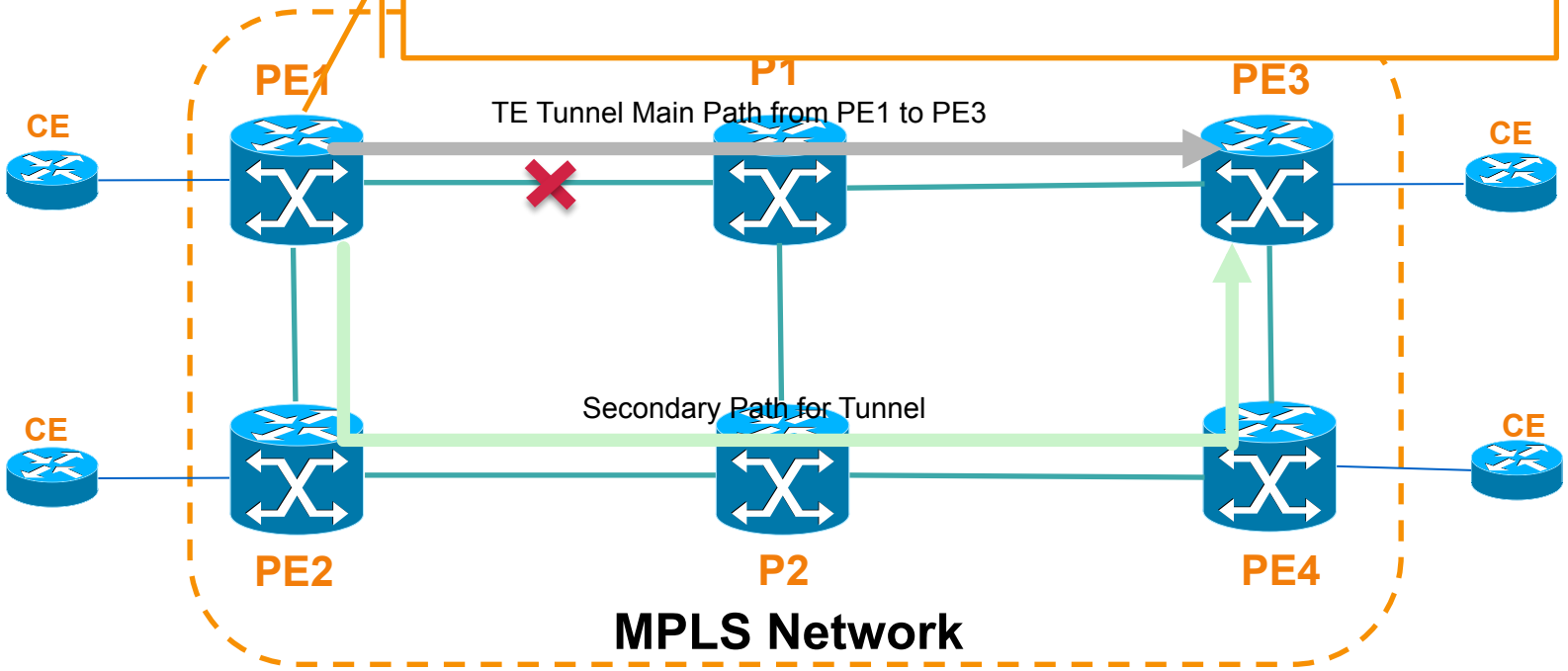
Check LSP when One Link is Down

```
PE1#traceroute mpls traffic-eng tunnel 100
Tracing MPLS TE Label Switched Path on Tunnel100, timeout is 2
seconds
......
Type escape sequence to abort.
  0 172.16.12.14 MRU 1512 [Labels: 1113 Exp: 0]
L 1 172.16.12.13 MRU 1512 [Labels: 811 Exp: 0] 4 ms
L 2 172.16.12.9 MRU 1512 [Labels: 619 Exp: 0] 4 ms
L 3 172.16.12.18 MRU 1500 [Labels: implicit-null Exp: 0] 3 ms
! 4 172.16.10.33 3 ms
```

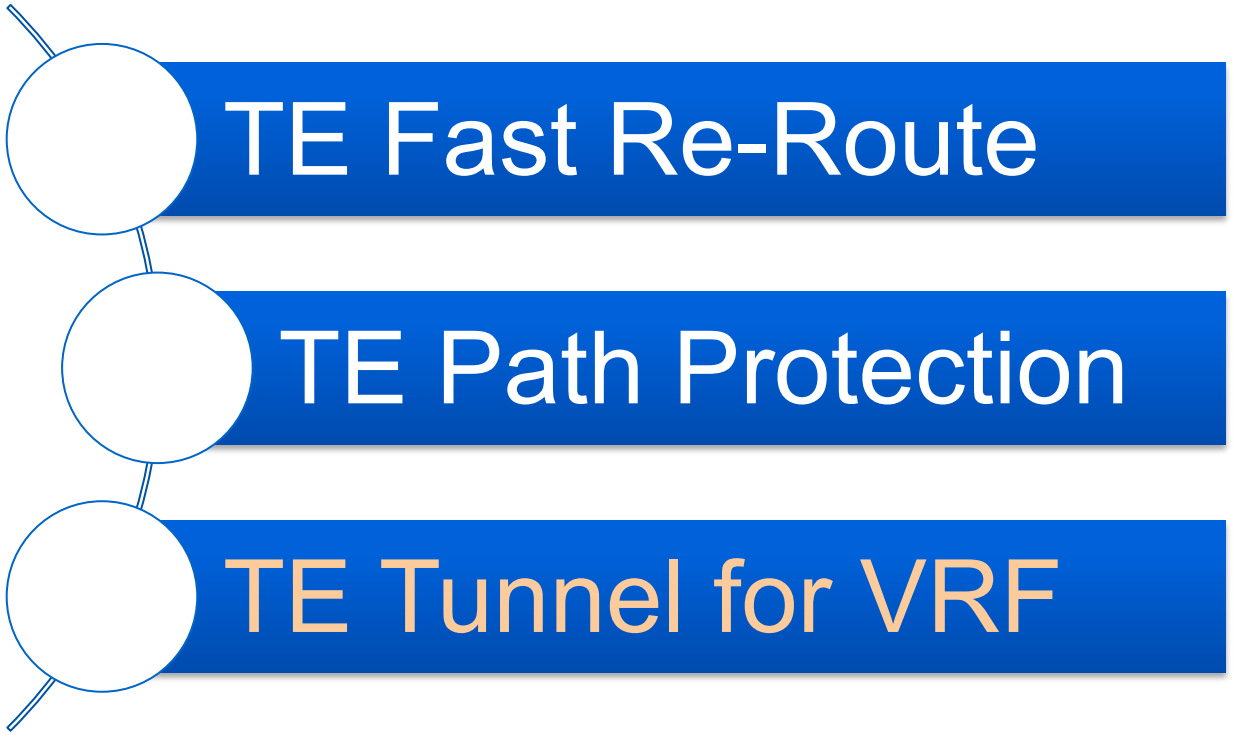


Check Path Protection when One Link is Down

```
Router2#show mpls traffic-eng tunnels protection | begin t2000

Router2_t2000
  LSP Head, Tunnel2000, Admin: up, Oper: up
  Src 172.16.15.2, Dest 172.16.15.4, Instance 11
  Fast Reroute Protection: None
  Path Protection: Backup lsp in use.
```

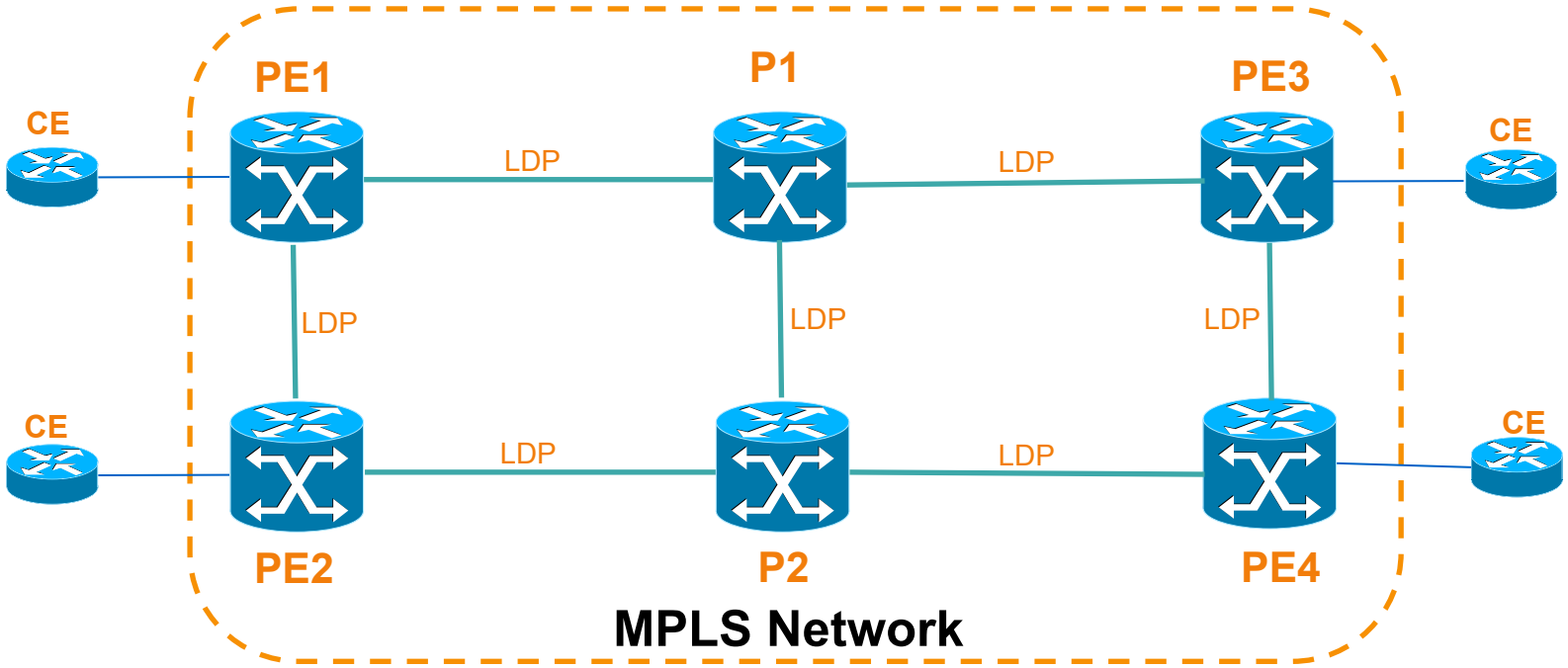


MPLS TE Services



LDP for VPN Tunnel LSP

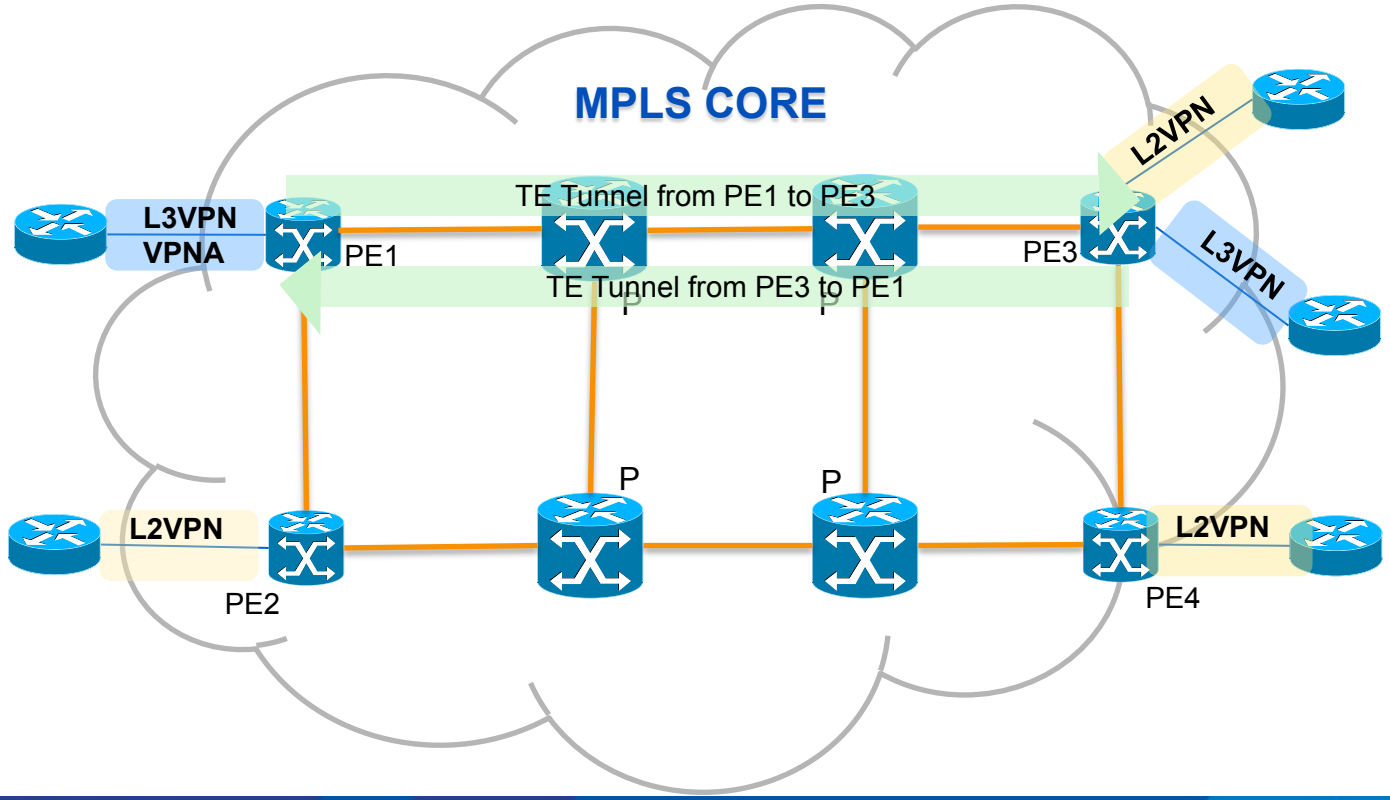
- In many ISPs, LDP LSP is the tunnel LSP for MPLS L3VPN service. LDP labels are the outer label for traffic forwarding.



TE Tunnel for VPN Tunnel LSP

- MPLS TE tunnels also can be used for VPN tunnel LSP.

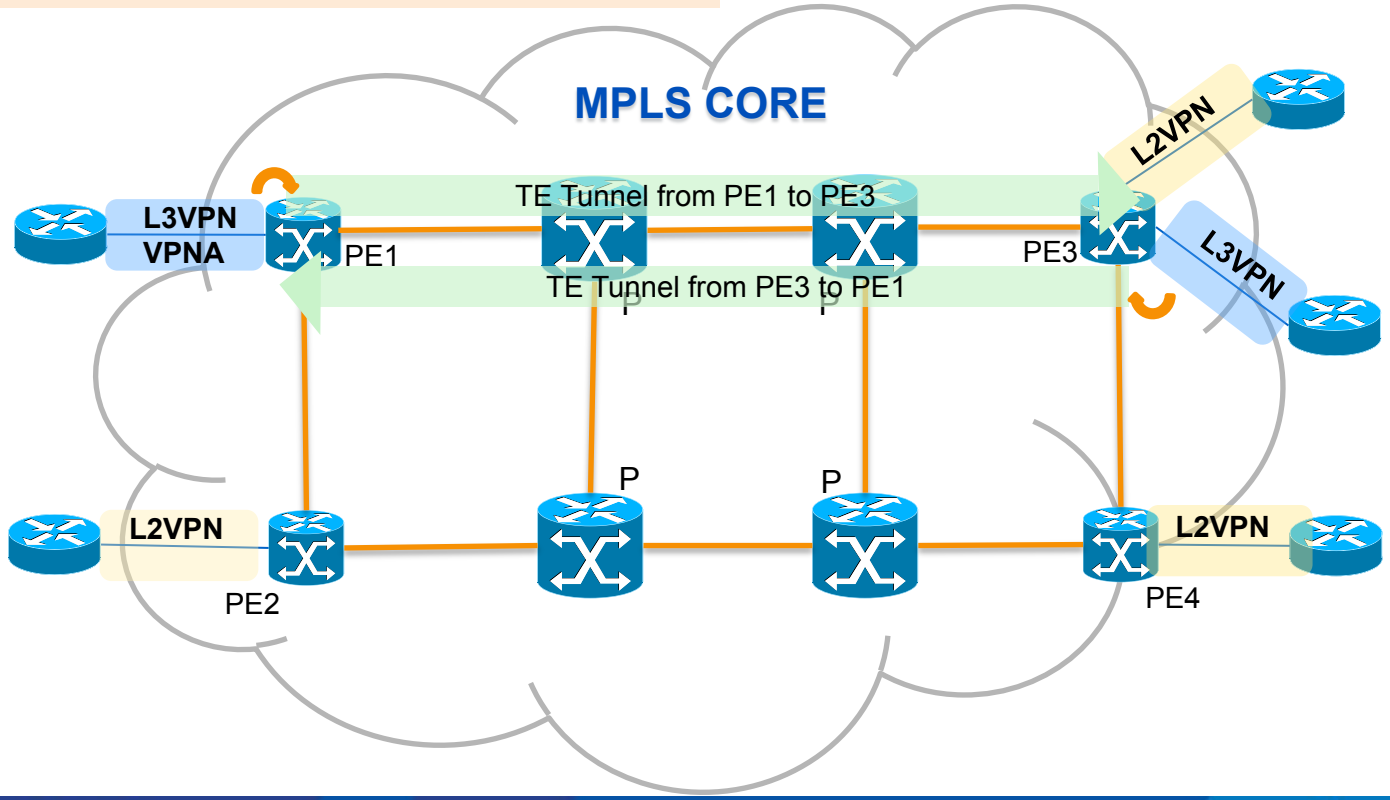
1. Set up TE tunnels between PEs



TE Tunnel for VPN Tunnel LSP

- MPLS TE tunnels also can be used for VPN tunnel LSP.

1. Set up TE tunnels between PEs
2. Guide the VPN traffic into the TE tunnel



Configuration of TE Tunnel for VPN Tunnel LSP

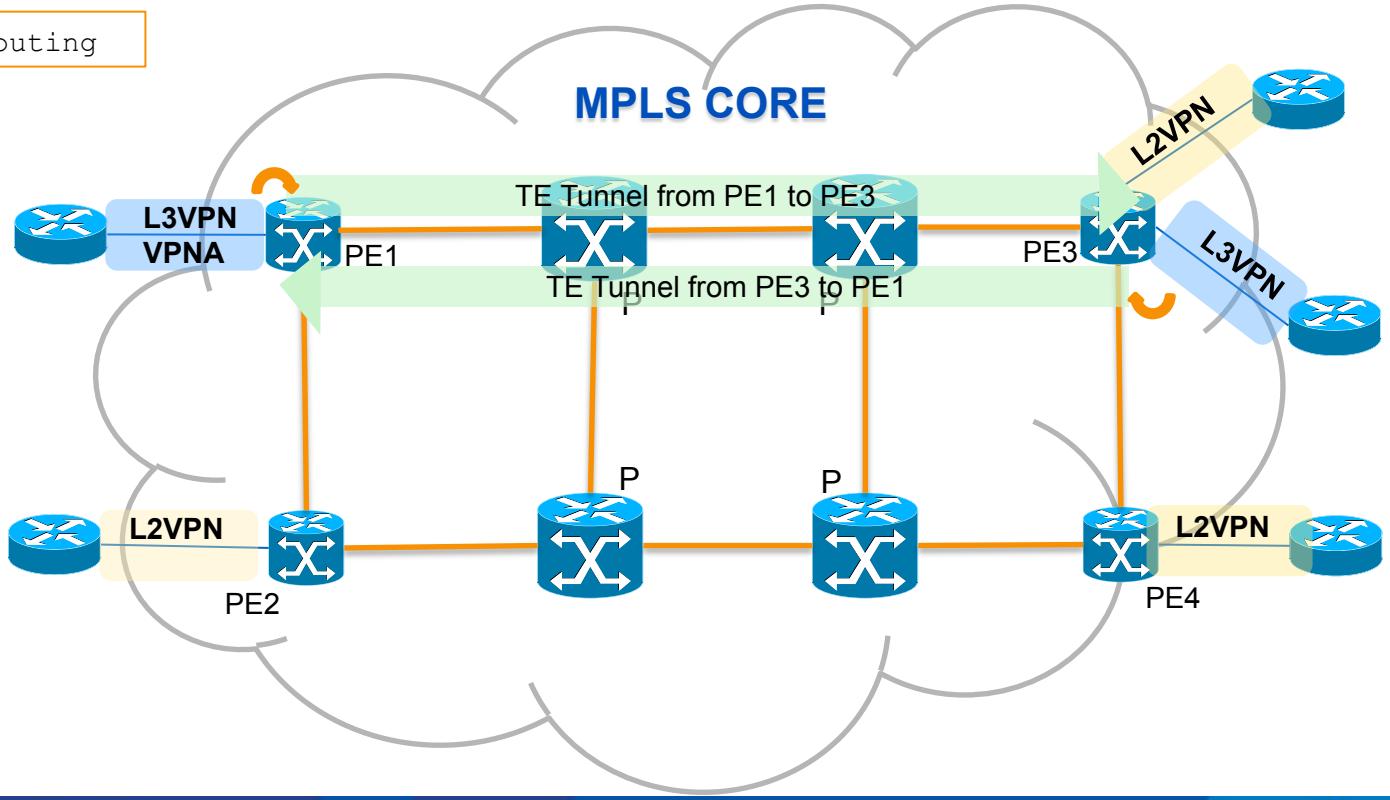
```
PE1(config)# interface tunnel 1000
PE1(config-if)# tunnel mpls traffic-eng autoroute announce
```

OR

```
PE1(config)# ip route 172.16.15.4 255.255.255.255 Tunnel1000
```

OR

```
Policy Routing
```



Questions?

APNIC