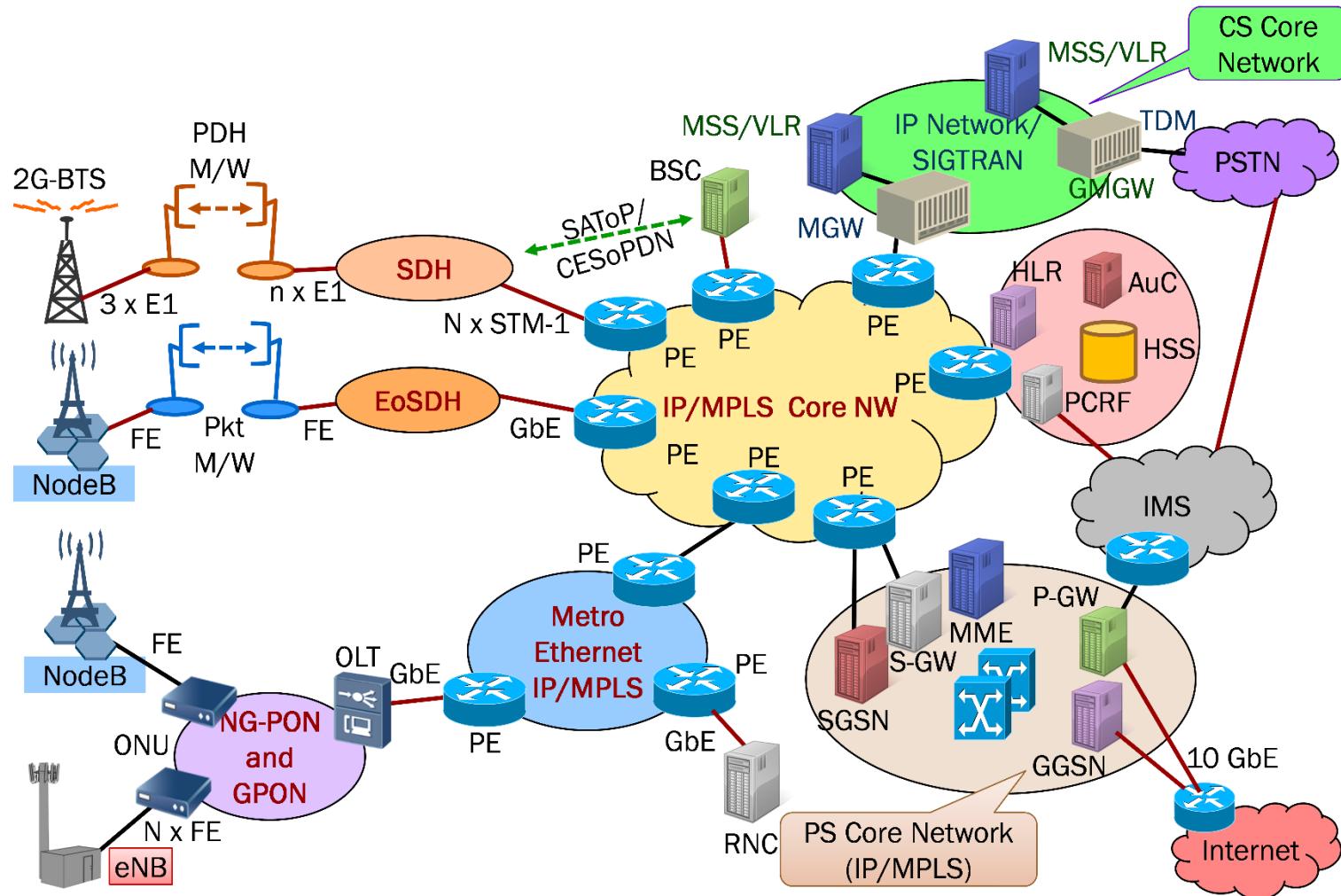
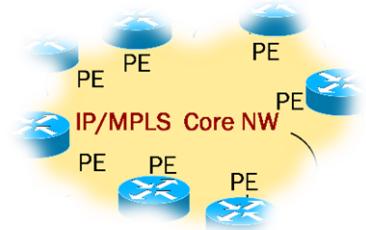


# Servicios Layer 2 VPLS / VPWS



# Servicios MPLS VPN



## Servicios MPLS VPN:

---

- Cuando se habla de VPNs, lo primero que se identifica es el uso de **encripción** para crear túneles IPs privados sobre una red, normalmente publica como Internet.
- Para esto, existen protocolos como ser IPSec o SSL, en función de los requerimientos de la arquitectura.
- En principio, podemos decir que no hay nada malo en esta aproximación, aunque también podemos considerar que la misma no es muy escalable.

## Servicios MPLS VPN:

---

- Esto ya que cada dispositivo final participando en un túnel VPN debe ser configurado con la información correcta de encripción y tunelización.
- Otra limitación está relacionado con que el control de la VPN está por completo en el Customer Edge device (CE).
- Estos problemas de servicios y escalabilidad resultaron en la necesidad de la introducción de las **VPN del tipo “Provider-based service”.**

## Servicios MPLS VPN:

---

- Para lograr esto, **el proveedor de servicios encapsula los datos** de los clientes para que puedan atravesar la red del service Provider.
- Dependiendo de la naturaleza del servicio VPN, **esta encapsulación puede ser en Capa 2 o en Capa 3**, incluyendo los headers correspondientes a cada uno.
- Los datos de los clientes deben ser transportados sin ningún tipo de cambios a través de la red del proveedor de servicios, desde un sitio hacia otro sitio del cliente.

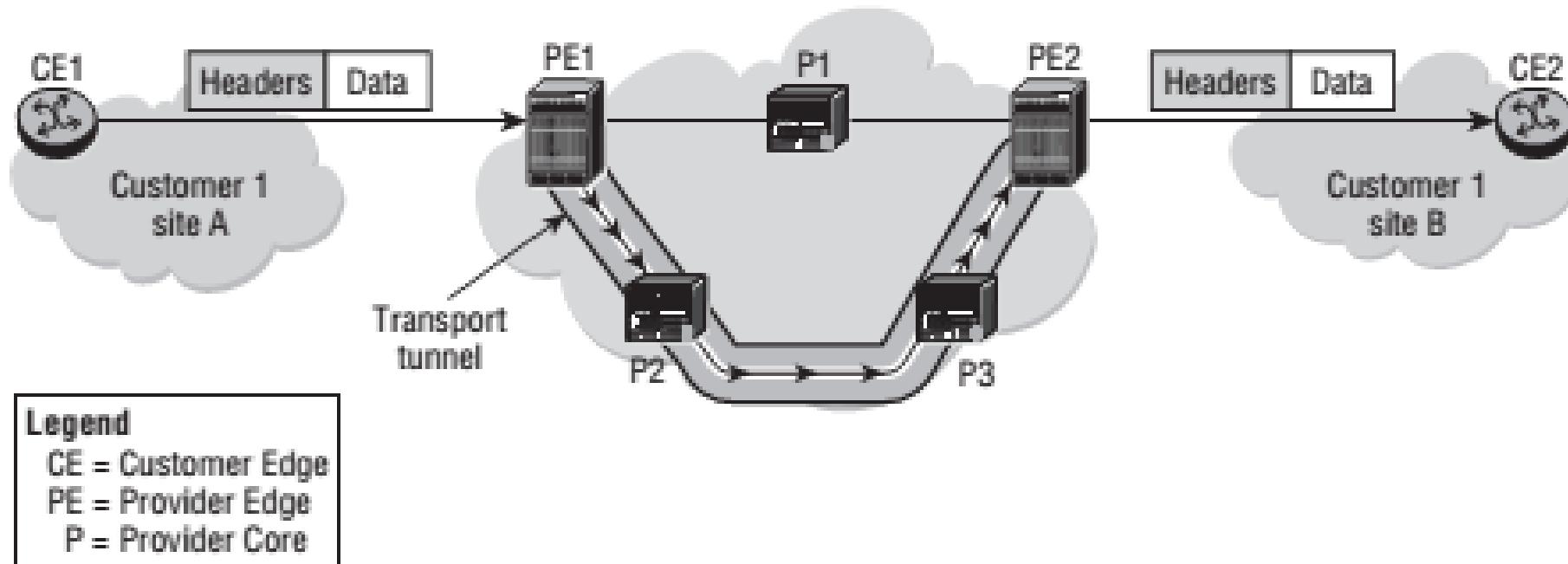
## Servicios MPLS VPN:

---

- De esta manera, el proveedor adjunta al paquete un header adicional para poder rutearlo.
- Luego quita esa información y forwardea el paquete utilizando información tanto de L2 como de L3 en el CE del destino.
- Luego, en lugar de rutear o conmutar el paquete en la red del proveedor de servicios utilizando información de los headers de L2 o L3 del cliente, se utiliza la información del header que es agregada en el borde de la red del service Provider.
- De esta forma, *los datos del cliente son efectivamente tunelizados* a través de la red del proveedor de servicios, *sin tener o recibir ningún tipo de cambios.*

## Servicios MPLS VPN:

- La siguiente figura muestra el arribo de un paquete desde el CE1 y como es conmutado a través de la red del proveedor de servicio mediante tuneles “Provider-based”



## Servicios MPLS VPN:

---

- Hay una serie de **puntos importantes** para marcar de la figura:
  - El primero es la existencia de un **túnel completamente transparente** al dispositivo CE.
  - El detalle de las características y la forma de la implementación del túnel **no son conocidas por el equipamiento instalado en el cliente.**
  - Toda la información del túnel en la red del proveedor es **configurada y controlada en los dispositivos PE.**

## Servicios MPLS VPN:

---

- Estos **PE** realizan:

- el set up del túnel
- marcan los paquetes entrantes desde los dispositivos CE
- luego, los forwardean.

- Los **routers P**:

- simplemente forwardean los paquetes basados en la información marcada en ellos por los dispositivos PE.

## Servicios MPLS VPN:

---

- El propósito de una red MPLS es:

*proveer un servicio de tunelización para forwardear paquetes de clientes a través de la red de un proveedor de servicios basado en otra información más que simplemente la dirección destino.*

## Servicios MPLS VPN:

---

- Para esto, la red MPLS **deberá marcar o etiquetar los paquetes** entrantes desde el CE con un header especial llamado label o etiqueta.
- Esta etiqueta es simplemente un **header adicional que es agregado a los paquetes** entrantes a la red.
- Luego, **se utilizaran estos headers para tomar decisiones de forwarding en la red**, en lugar de utilizar las direcciones IP.

## Servicios MPLS VPN:

---

- ***Finalmente, lo que obtendremos es que una red ruteada, de capa 3, se convierta en una red switcheada, de capa 2.***
- Esto provee una serie de ventajas sobre el routing tradicional.
- También es importante entender, que los routers de la red MPLS, deberán utilizar algún protocolo de señalización para distribuir los labels a través de la red.
- Para esto, podemos utilizar diferentes protocolos, como ser Label Distribution Protocol (LDP) o Resource Reservation Protocol (RSVP).

## Servicios MPLS VPN:

---

- Como decíamos anteriormente, las VPNs son una *forma de proveer a los clientes conexiones que parezcan ser privadas sobre la infraestructura de una red que es compartida por múltiples clientes.*
- Los proveedores de servicios entonces podrán ofrecer VPNs del tipo Layer 2 como así también del tipo Layer 3.
- De aquí, diferenciamos tres tipos de servicios que son soportados: VPWS, VPLS, and VPRN.

### Virtual Private Wire Service (VPWS)

- Es el servicio de Layer 2 más simple, **que emula una única conexión, cable o circuito entre dos ubicaciones.**
- Los clientes no tienen conocimiento de la red del proveedor, es un servicio que se conoce como LAN-to-LAN y actúa como una **simple conexión punto a punto entre dos sitios.**
- Se pueden emular conexiones Ethernet (epipe), Frame-Relay (fpipe), ATM (apipe) o TDM (cpipe).

## Servicios MPLS VPN:

---

### Virtual Private LAN Service (VPLS)

- Es un servicio de Layer 2 multipunto, que puede ser utilizado **para interconectar más de dos sitios de clientes.**
- Desde la perspectiva del cliente, una VPLS es **similar a una LAN layer 2** que interconecta diferentes puntos.
- La **única diferencia** entre los servicios VPWS y VPLS es la cantidad de puntos que permiten interconectar.

## Servicios MPLS VPN:

---

### Virtual Private Routed Network (VPRN)

- Es un servicio Layer 3 que **permite que la red del proveedor de servicios parezca un router IP que conecta dos o más ubicaciones.**
- Las VPNs del tipo VPRN permiten que **los dispositivos CE intercambien información de rutas con la VPRN**, como si esta fuera un router IP.

## Virtual Private Wire Service (VPWS):

---

- Las VPNs del tipo VPWS son el **servicio más sencillo que una red MPLS puede proveer**.
- Es un **servicio punto a punto**, emulando una conexión de Layer 2 entre dos sitios del cliente.
- El frame del cliente no es chequeado ni **tampoco se realiza aprendizaje de MAC address** por parte del VPWS.
- El Frame de Layer 2 del cliente es encapsulado con el correspondiente header MPLS y **switcheado a través de la red del proveedor de servicio en forma transparente**.

## Virtual Private Wire Service (VPWS):

---

- El PE de ingreso, recibe los datos del cliente en un **Service Access Point (SAP)** que es asociado con un servicio específico.
- **Este SAP puede ser** un puerto, un puerto con un VLAN tag específico o un circuit ID específico para los casos de ATM o Frame-Relay.
- Los datos del cliente son entonces **encapsulados con un service label en el PE de ingreso**.
- Debido a que se pueden configurar diferentes servicios en el PE, **el service label identifica a que servicio específico pertenecen los datos**.

## Virtual Private Wire Service (VPWS):

---

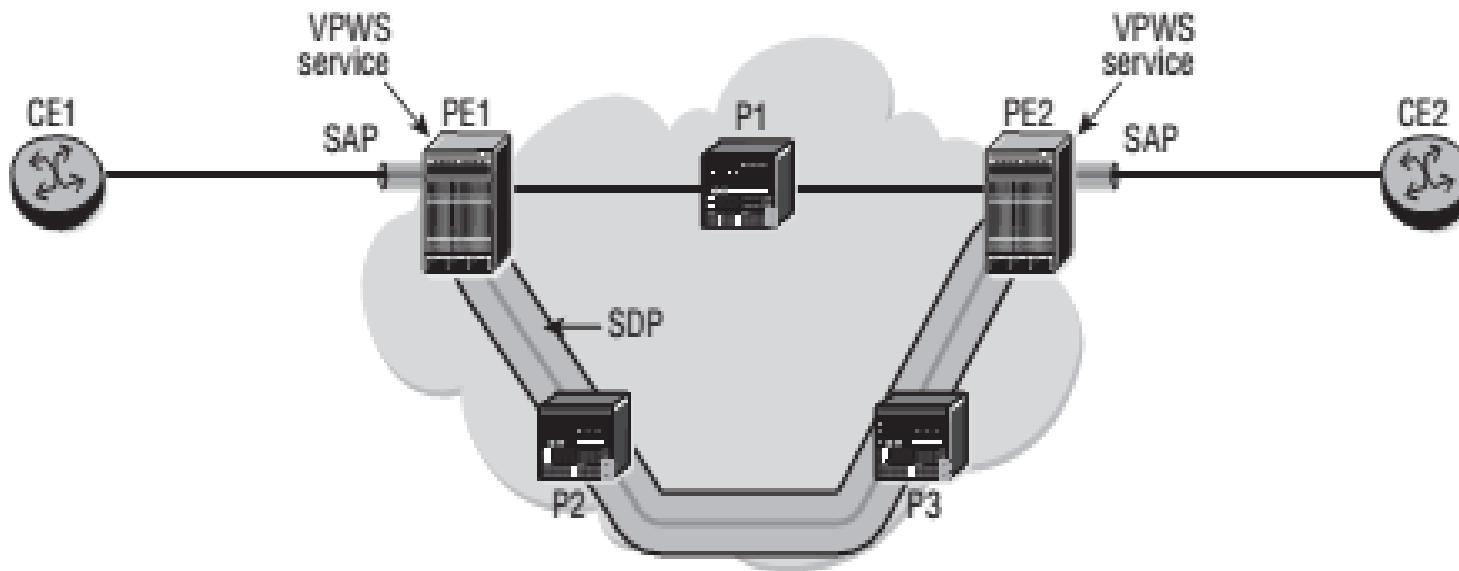
- Luego de que los datos son encapsulados con el service label correspondiente, deben ser forwardeados al correspondiente **Service Distribution Point (SDP)**, que es definido por el servicio.
- Para esto, un segundo label (outer) es agregado a los datos.
- Esta **label identifica al LSP que será utilizado para transportar los paquetes al extremo remoto del tunel**, es decir, el PE de egreso.

## Virtual Private Wire Service (VPWS):

---

- Para la **conmutación** de los datos a lo largo del LSP, **se utilizara el outer label**.
- Finalmente, el PE de egreso, **remueve el encapsulamiento MPLS del SDP**.
- El **label interior**, el service label, es utilizado para **identificar a que servicio pertenece** el dato, luego de que el label es removido, para que los paquetes puedan ser transmitidos en el SAP apropiado.

# Virtual Private Wire Service (VPWS):



**Legend**  
CE = Customer edge  
PE = Provider edge  
P = Provider core

Logical view is as if a single wire connects CE1 and CE2



## Virtual Private LAN Service (VPLS) :

---

- Son similares a las VPWS:
  - con **SAPs** para proveer acceso a los clientes
  - Con **SDPs** para proveer la conexión de transporte a través de la red al PE remoto del servicio.
- Sin embargo, *las VPLS son servicios multipunto que soportan múltiples puntos de acceso.*
- Por esto, los VPLS actúan lógicamente como un switch de capa 2 que interconecta todos los dispositivos CE que están conectados al servicio.

## Virtual Private LAN Service (VPLS) :

---

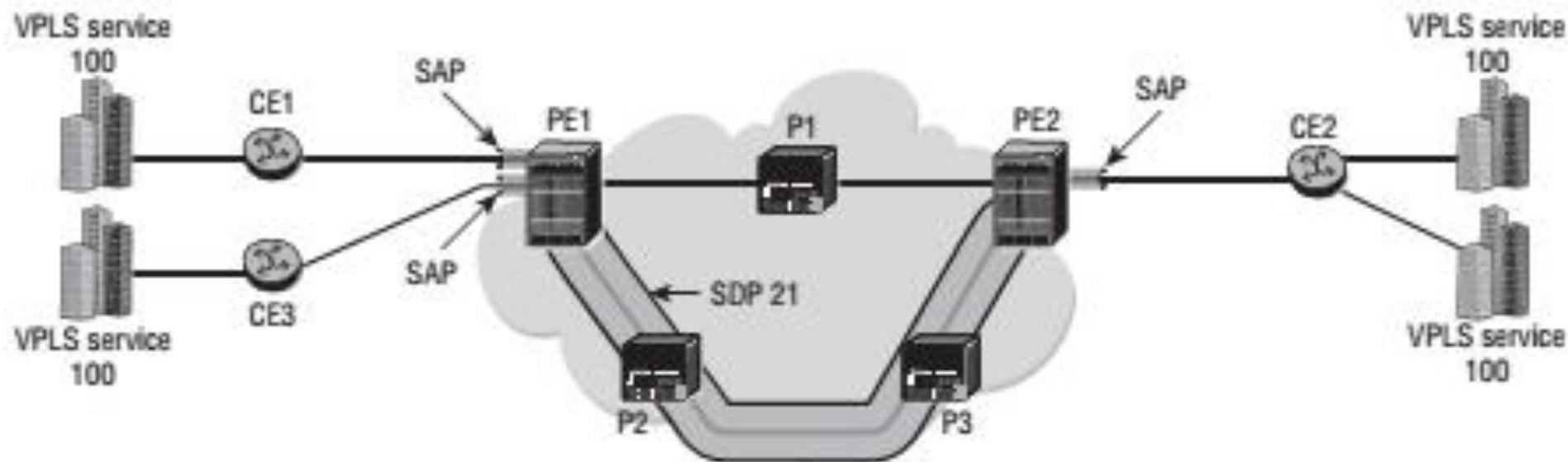
- Debido a que un VPLS emula el servicio de un switch Ethernet, una MAC address forwarding database (FDB) debe ser mantenida por cada VPLS.
- Cuando un frame unicast con una MAC address origen desconocida, llega a un SAP o a un SDP, la VPLS aprende la MAC address.
- Esto de la misma forma que un switch Ethernet lo realiza en cualquiera de sus puertos.
- Esto significa que el VPLS agrega y asocia la dirección MAC con un SAP y SDP en la FDB.

## Virtual Private LAN Service (VPLS) :

---

- Cuando un frame Ethernet arriba a un SAP o a un SDP, se realiza un lookup en la FDB para la dirección destino.
- Si se encuentra una entrada para esa dirección, el frame es forwardeado al SAP o SDP apropiado.
- Si no se encuentra ninguna entrada, el frame es “inundado” a todos los SAP y SDP, **similar al flooding de un switch.**

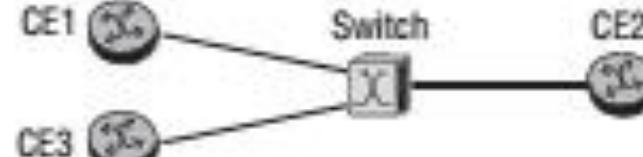
# Virtual Private LAN Service (VPLS) :



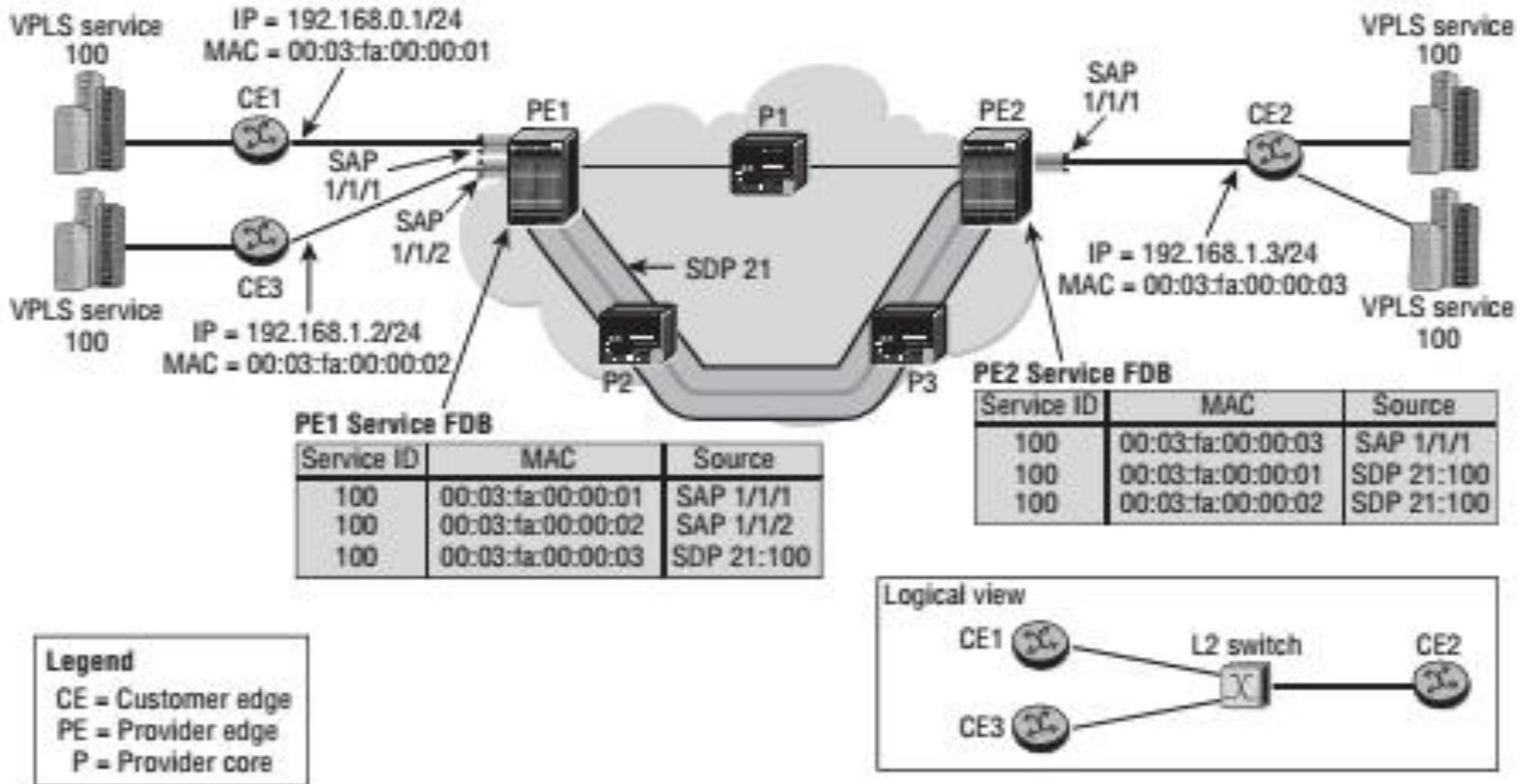
## Legend

- CE = Customer edge
- PE = Provider edge
- P = Provider core

## Logical view



# Virtual Private LAN Service (VPLS) :



## Legend

CE = Customer edge  
PE = Provider edge  
P = Provider core

## Virtual Private Routed Network (VPRN):

---

- Es una clase de VPNs que permite la **conexión de múltiples sitios en un dominio “ruteado”** sobre la red de un proveedor IP/MPLS, es decir es un servicio de Layer 3.
- Desde la perspectiva del cliente, **todos los sitios están conectados a una red privada completamente ruteada que es administrada por el proveedor de servicios.**
- Cada router PE que provee servicios VPRN mantiene por separado una tabla de forwarding IP para cada VPRN.

## Virtual Private Routed Network (VPRN):

---

- Esto, ya que cada cliente tiene su propio direccionamiento IP privado, evitando así solapamientos.
- Para lograr esto, utiliza VRF.
- Estos son tablas de routing virtuales, lógicas, privadas de cada cliente.
  - Permiten de manera segura aislar la información de routing:
    - de un cliente al otro,
    - de las rutas del propio Provider Core.

## Virtual Private Routed Network (VPRN):

---

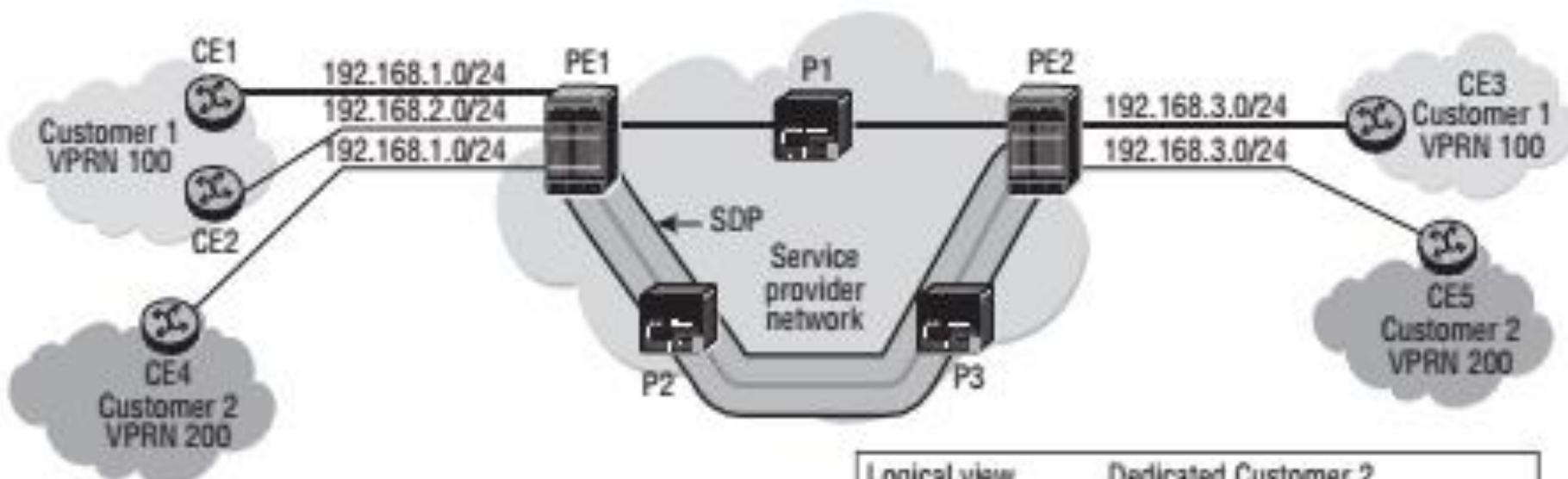
- Cada router CE se convierte en un routing peer de los routers PE del proveedor, al cual está directamente conectado.
- Entre estos dispositivos realizan el intercambio de rutas.
- Los dispositivos PE en la VPRN intercambian rutas entre ellos, de manera que todas las rutas puedan ser transmitidas a los CE remotos de cada cliente.

## Virtual Private Routed Network (VPRN):

---

- El transporte de datos de los clientes es similar al realizado en una VPWS o una VPLS.
- Con la principal diferencia que desde la perspectiva del cliente, **la red del proveedor aparece como una red ruteada.**
- Por lo que *el dispositivo del cliente debe realizar un peering con el equipamiento del proveedor utilizando algún protocolo de routing dinámico o en su defecto, rutas estáticas.*

# Virtual Private Routed Network (VPRN):

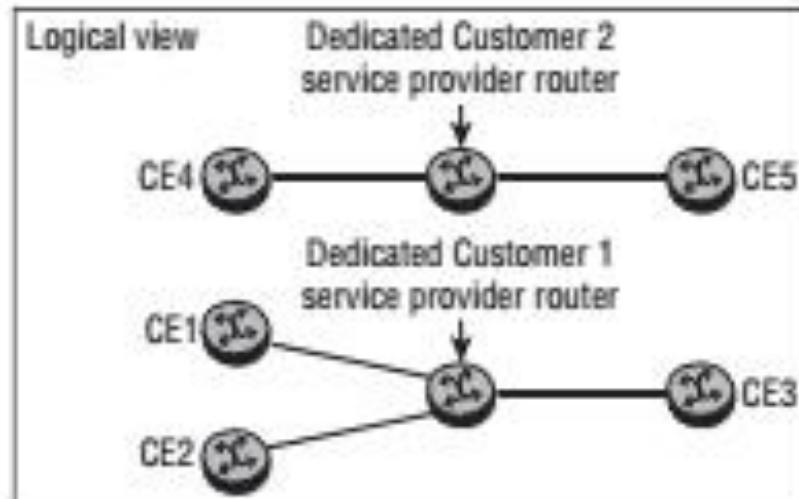


PE1 and PE2 has a VRF table for each VPRN

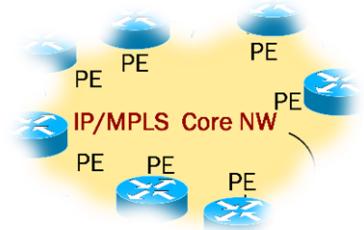
VRF 1	VRF 2
VPRN 100	VPRN 200
192.168.1.0/24	192.168.1.0/24
192.168.2.0/24	192.168.3.0/24
192.168.3.0/24	

Legend

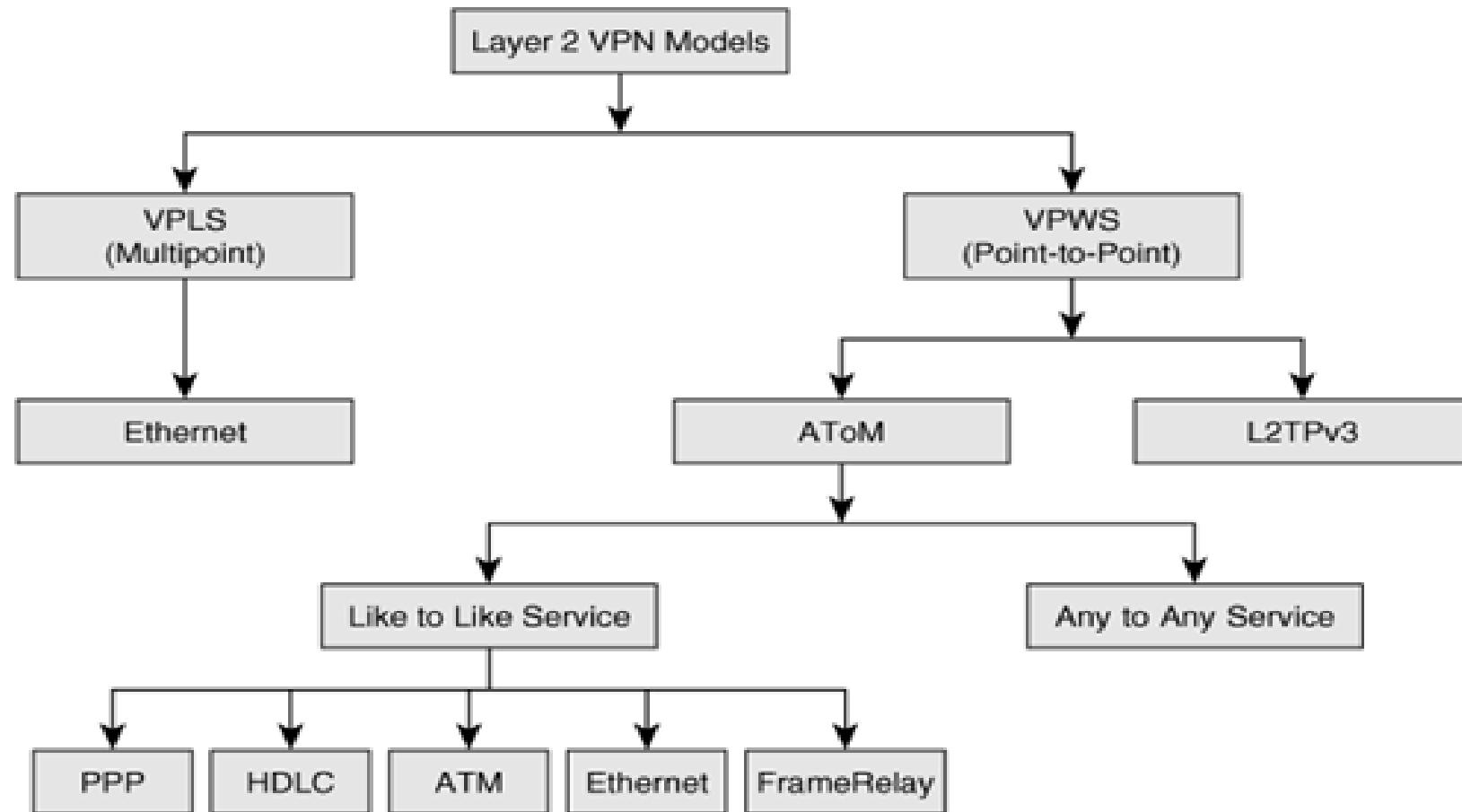
CE = Customer Edge  
PE = Provider Edge  
P = Provider Core



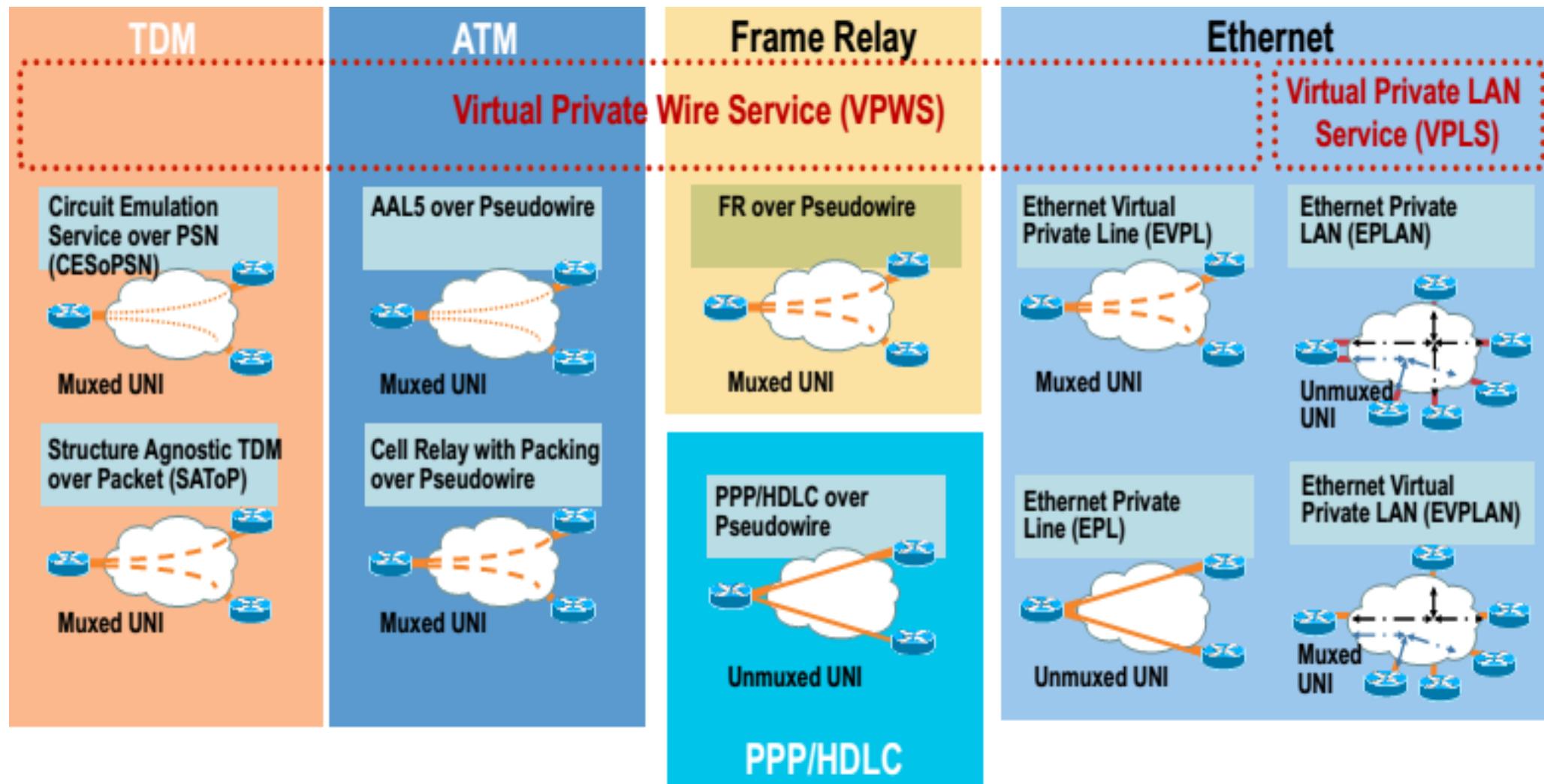
# Servicios Layer 2



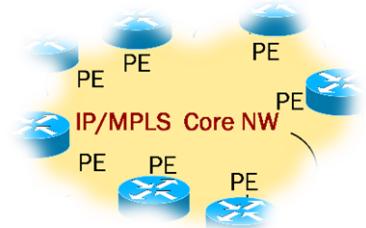
# Servicios Layer 2:



# Servicios Layer 2:



# Virtual Private Wire Service (VPWS):



## Virtual Private Wire Service (VPWS):

---

### Pseudowire Reference Model

- El modelo de referencia de los Pseudowire está basado en el grupo de trabajo del IETF PWE3 (Pseudo-wire Emulation Edge to Edge)
- Este provee el marco de referencia para la emulación de un vínculo edge to edge sobre una red basada en conmutación de paquetes.
- Un pseudowire es una conexión lógica entre dos routers provider edge (PE) que conectan dos pseudo-wire end-services (PWES) del mismo tipo.

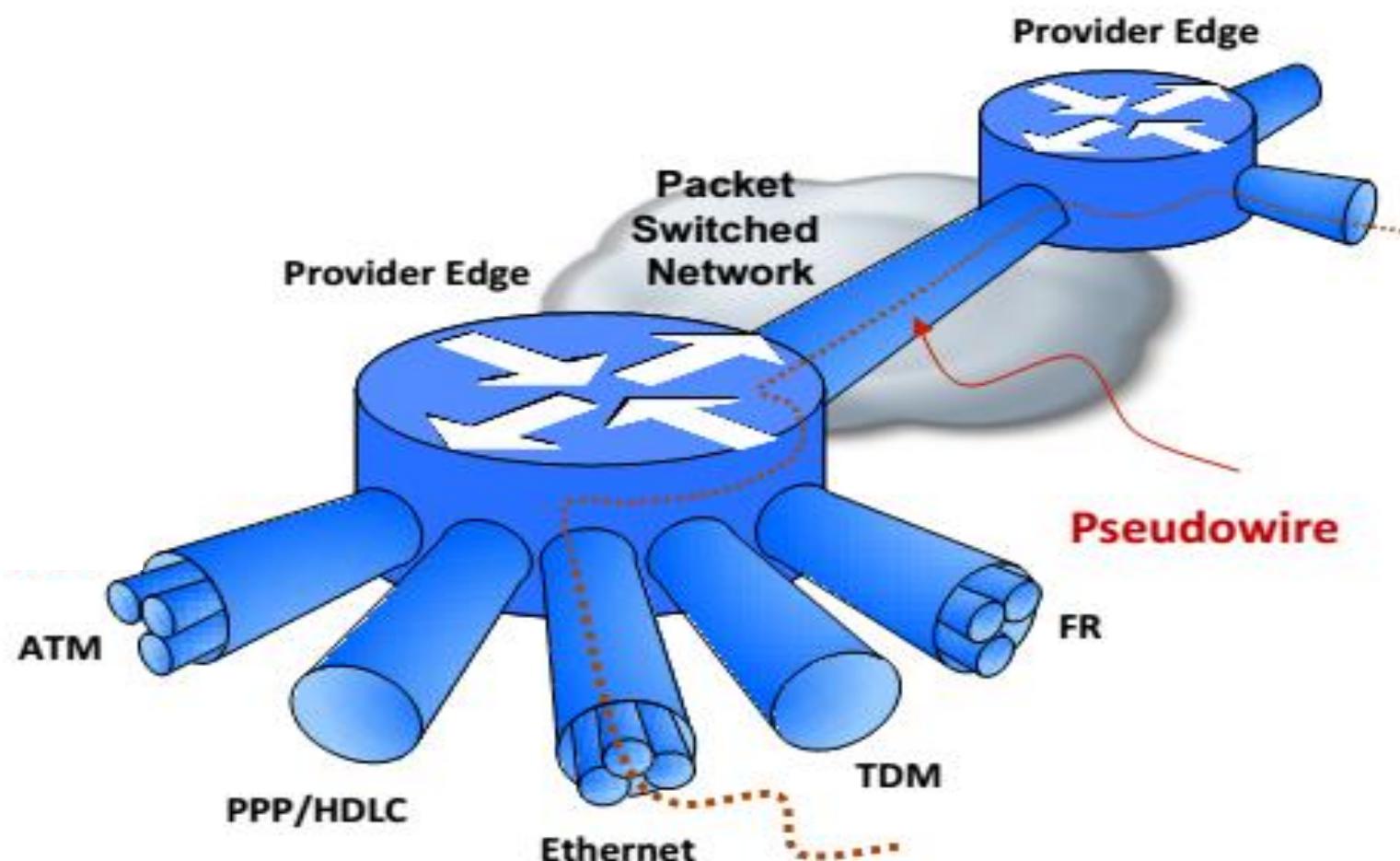
## Virtual Private Wire Service (VPWS):

---

•Estos **pseudo-wire end-services** (PWES) utilizados entre los dispositivos PE y CE pueden ser alguno de los siguientes:

- Ethernet, VLAN, or 802.1Q tunneling (QinQ)
- ATM VC or VP
- Frame Relay VC
- HDLC
- PPP

# Virtual Private Wire Service (VPWS):



## Virtual Private Wire Service (VPWS):

---

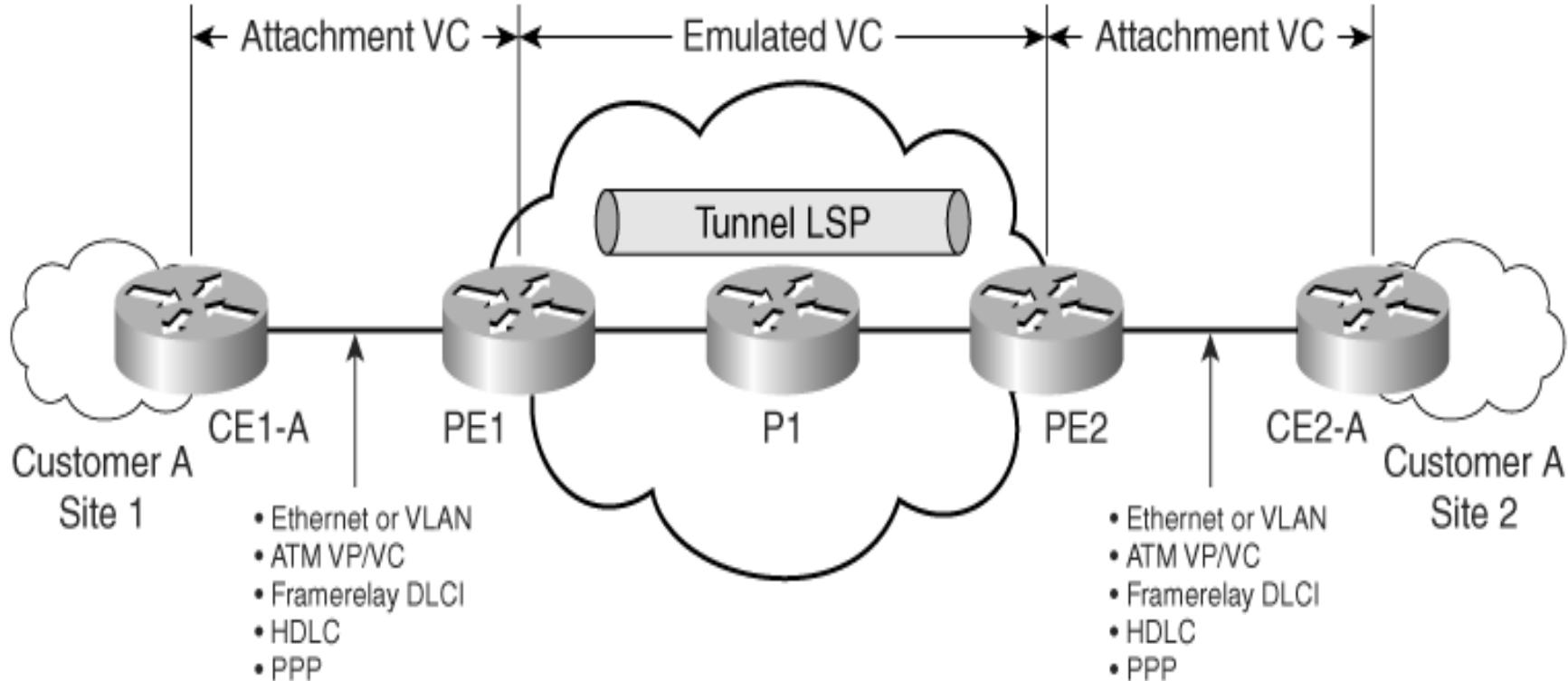
- Los routers PE son configurados como los **endpoints de una conexión pseudo-wire**.
- Luego de la creación del pseudowire, las PDU de Layer 2 son encapsuladas en el router de ingreso del PW.
- Esta PDU encapsulada es enviada sobre el PW hacia el router de egreso, donde los headers son reconstruidos y los frames son enviados en su formato original hacia el dispositivo CE.
- En estas, el ***Customer Edge (CE) considera al PW como un link o circuito “no compartido” sobre la red del proveedor.***

## Virtual Private Wire Service (VPWS):

---

- AToM o “Any Transport Over MPLS” es la implementación de Cisco para VPWS sobre redes IP/MPLS.
- La solución de VPWS o AToM según Cisco, provee la posibilidad que el backbone MPLS transporte el tráfico a nivel Layer 2 de los clientes, eliminando la necesidad de segmentar la red o separarla en dos partes una en cada extremo.
- De esta forma, el proveedor de servicio puede proveer el transporte a un servicio existente de L2 como ser (ATM, FR, HDLC o Ethernet) sobre el backbone MPLS.

# Virtual Private Wire Service (VPWS):



## Virtual Private Wire Service (VPWS):

---

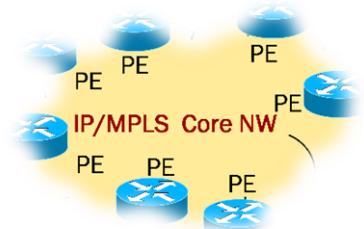
- En las redes VPWLS o AToM la inteligencia de la misma está limitada al router Provider Edge (PE).
- Por esto comúnmente se dice que AToM es una tecnología de borde que utiliza el backbone MPLS para crear servicios L2 point-to-point.
- Los proveedores de servicios, no necesitan realizar ningún cambio en los routers P en el core de la red MPLS, esto es ya que la inteligencia de la red está completamente en los routers PE del borde.

## Virtual Private Wire Service (VPWS):

---

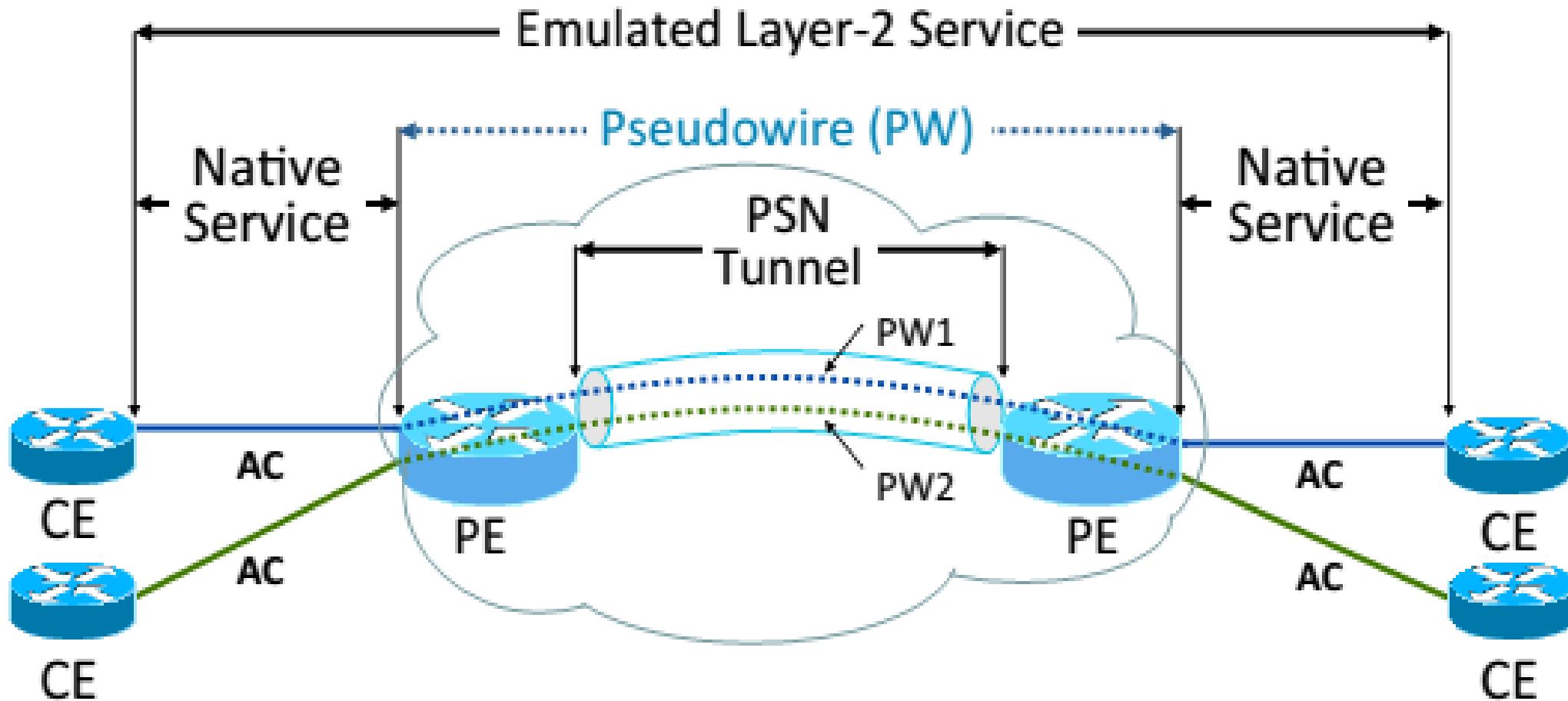
- Los clientes están dispuestos a migrar a la solución MPLS VPN por tres razones:
  - La primera razón es que quieren mantener un control completo sobre su red y la forma en que está construida.
  - La segunda razón es que tienen los equipos heredados (por ejemplo, IBM FEP) que ejecutan protocolos que no se pueden transportar a través de redes IP puras.
  - La tercer razón es que no necesitan realizar ningún cambio en la red, ni de topología, ni de protocolos ni de equipamientos.

# VPWS: Arquitectura / Terminología



## VPWS Arquitectura / Terminología:

- En la siguiente figura podemos observar un túnel PSN que conecta dos routers PE del proveedor del servicio.



## VPWS Arquitectura / Terminología:

---

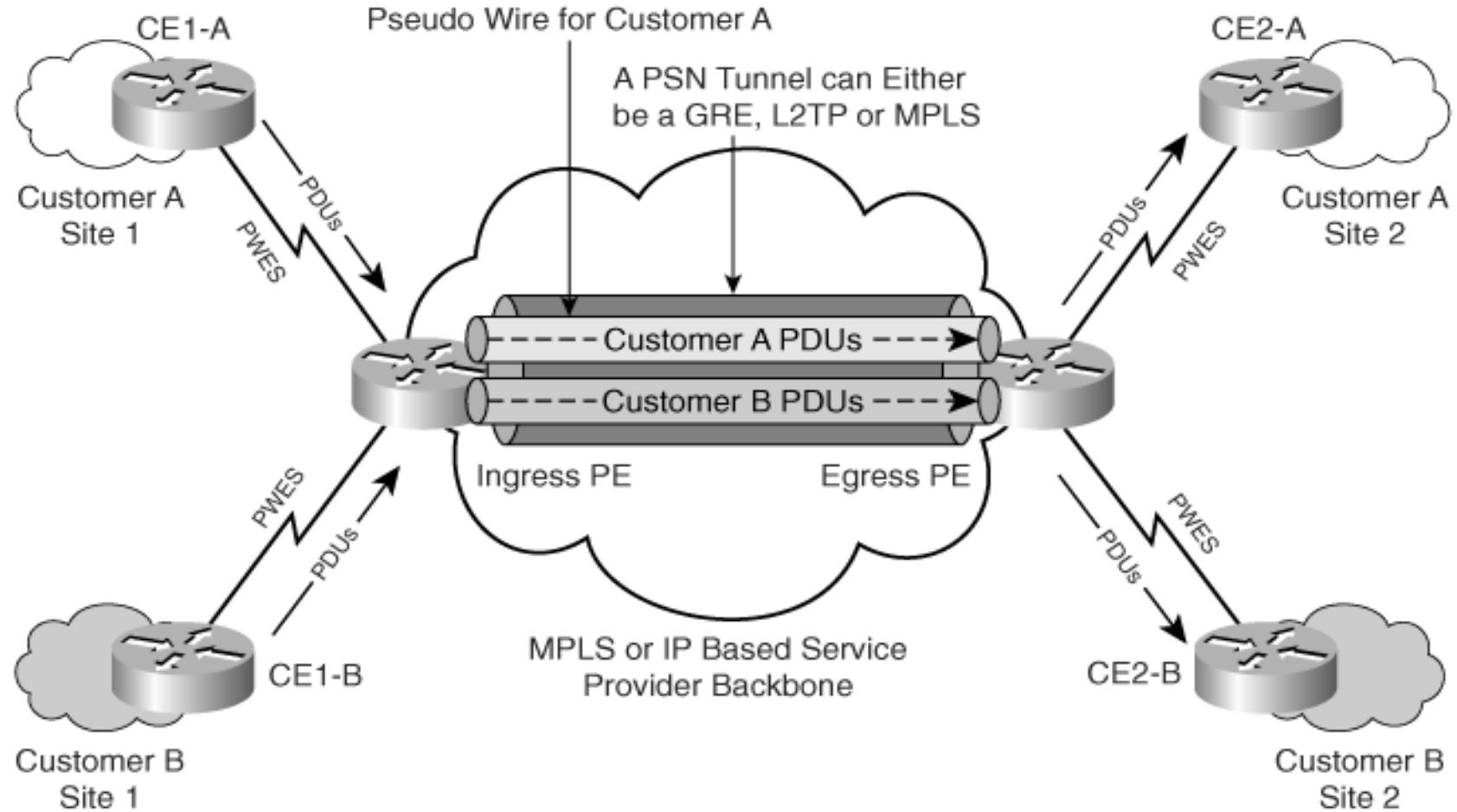
- Dentro de estos túneles PSN pueden existir uno o más pseudowires
- Estos conectan con los “**attachment circuits**” (AC) en los routers PE.
- Un Attachment circuit (AC) es un *circuito físico o virtual (VC) que conecta a un CE con un PE.*
- Un AC puede ser entre otras cosas Frame Relay, ATM, Ethernet port, Ethernet Vlan, HDLC o PPP.

## VPWS Arquitectura / Terminología:

---

- La Packet Switched Network (PSN) utiliza IP o MPLS como el mecanismo para el forwarding de paquetes.
- Los endpoints de un pseudowire son dos routers PE conectados a ACs del mismo tipo.
- Las tramas que el router PE recibe en el AC son encapsuladas y enviadas a través del pseudowire al router PE remoto.
- El router PE de egreso recibe los paquetes desde el pseudowire y le quita el encapsulamiento.
- Este PE de salida extrae y forwardea el frame al AC

# VPWS Arquitectura / Terminología:



## VPWS Arquitectura / Terminología:

---

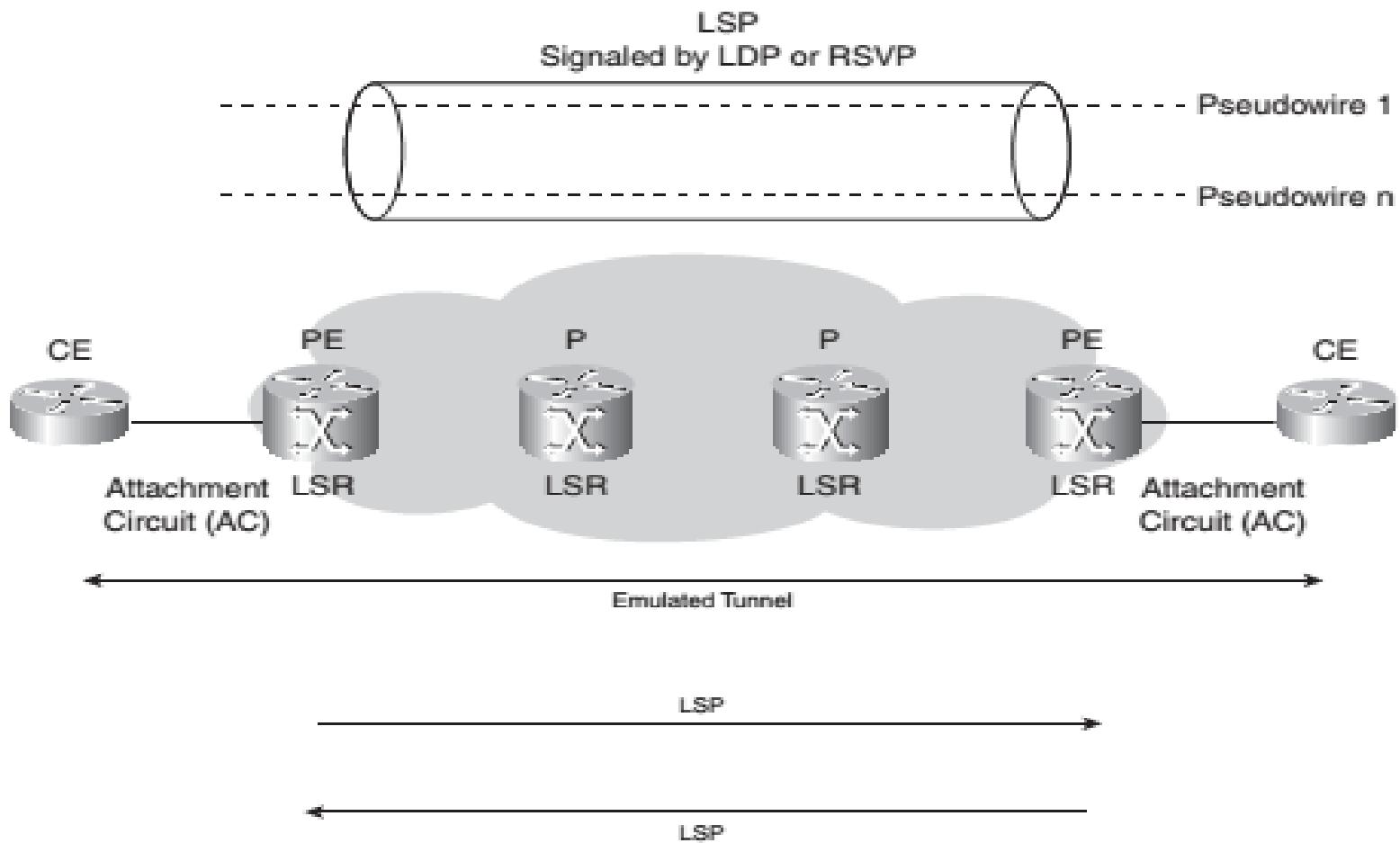
- Este túnel PSN no es ni más ni menos que un LSP entre los dos routers PE.
- Como tal, tiene una etiqueta que está asociada con este LSP denominada “tunnel label”
- Con este tunnel label, podremos identificar a que PSN tunnel pertenecen los frames transportados del cliente.
- Este tunnel label recibe las tramas desde el router PE local o de ingreso hacia el PE remoto a través del backbone MPLS.

## VPWS Arquitectura / Terminología:

---

- Para **multiplexar varios pseudowires** dentro de un PSN tunnel, el router PE utiliza **otra etiqueta** para identificar el pseudowire.
- Esta etiqueta o label es denominada **VC o PW label**, ya que se utiliza para **identificar el virtual circuit o el pseudowire en la que el frame está multiplexado.**
- *Un LSP es unidireccional.*
- *Por esto, dos LSP deben existir entre un par de routers PE, uno por cada dirección.*

# VPWS Arquitectura / Terminología:

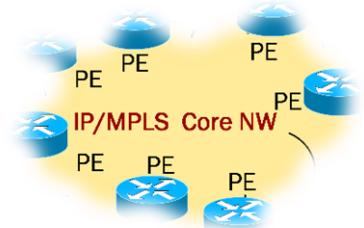


## VPWS Arquitectura / Terminología:

---

- Antes que el pseudowire esté listo para commutar paquetes, los routers PE deben realizar el establecimiento del mismo.
- En esta etapa de establecimiento, los routers PE intercambian la información necesaria para el acuerdo del tipo de servicio a realizar.
- Por ejemplo necesitan intercambiar información del método de encapsulamiento a utilizar.

# VPWS: Plano de Datos



## VPWS Plano de Datos:

---

- A medida que el **router PE de ingreso** recibe los frames desde el CE, este los forwardea a través del backbone hacia el **router de egreso utilizando un label stack con dos etiquetas**:
  - la tunnel label
  - la VC label.

## VPWS Plano de Datos:

---

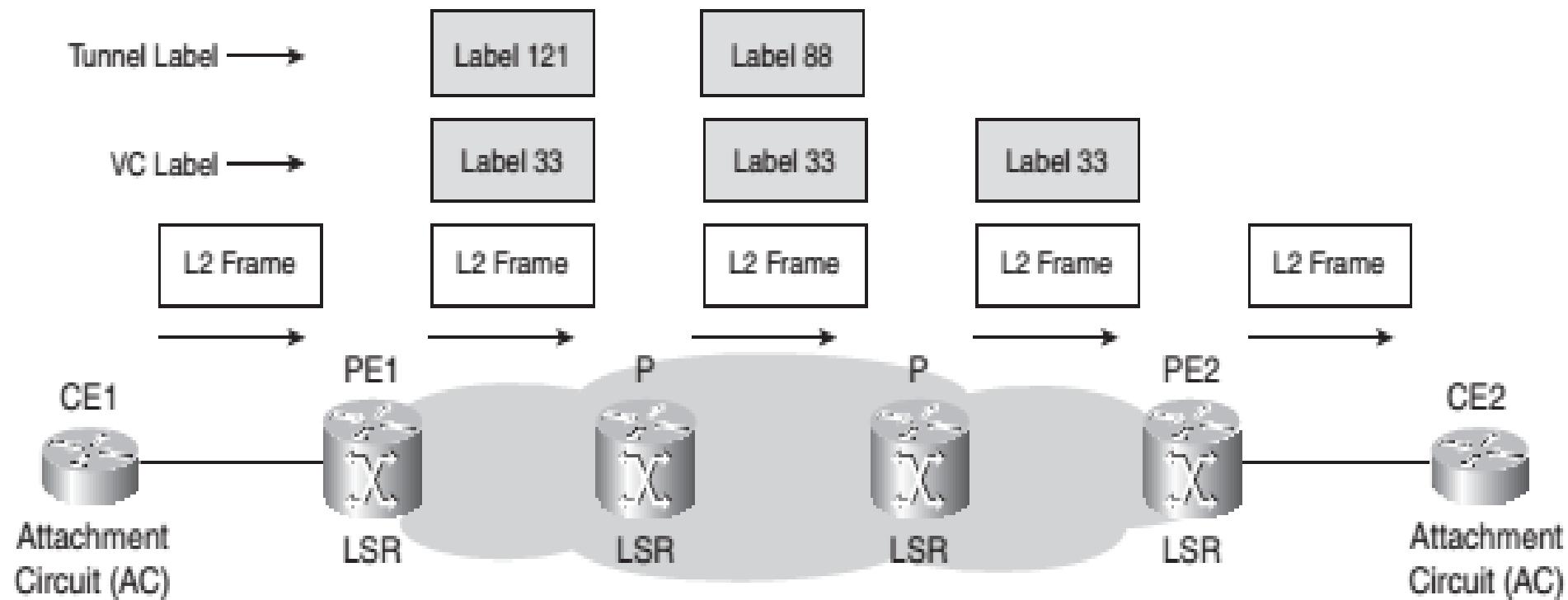
- **La etiqueta del túnel:**
- Es la etiqueta superior de la pila de etiquetas.
- Le dice a todos los LSRs intermedios a cual LSR de salida debe ser forwardeado el frame.
- La etiqueta exterior propaga el paquete desde el router PE de ingreso al router de egreso PE correspondiente.
- Esta etiqueta exterior *es utilizada entonces para el PE-to-PE LSP*.

## VPWS Plano de Datos:

---

- **La etiqueta de VC:**
- Es siempre la etiqueta de la parte inferior de la pila de etiquetas.
- Identifica el AC de salida en el PE de salida
- Es *utilizada por el router PE de egreso para forwardear el paquete hacia la interfaz de salida apropiada.*

# VPWS Plano de Datos:



## VPWS Plano de Datos:

---

- El router PE1 de ingreso realiza:

- Primero un push del VC label (label 33).
- Luego, realiza el push del tunnel label.

- Luego de esto:

- el paquete es forwardreado de acuerdo a este tunnel label
- salto a salto
- hasta que el paquete alcance el router PE de egreso, el PE2.

## VPWS Plano de Datos:

---

- Es importante notar que:
  - Cuando el paquete alcanza el router PE de egreso, el tunnel label ya fue removido.
  - Esto es debido a que se utiliza PHP entre el último router P y el router de egreso PE.
  - Esto permite que el router de egreso PE solamente realice un look up del VC label en la LFIB y realice el forwarding del frame en el AC correcto.
  - También es importante notar que los routers P de la red, no necesitan inspeccionar el VC label, estando completamente aislados de la solución o del servicio L2.

## VPWS Plano de Datos:

---

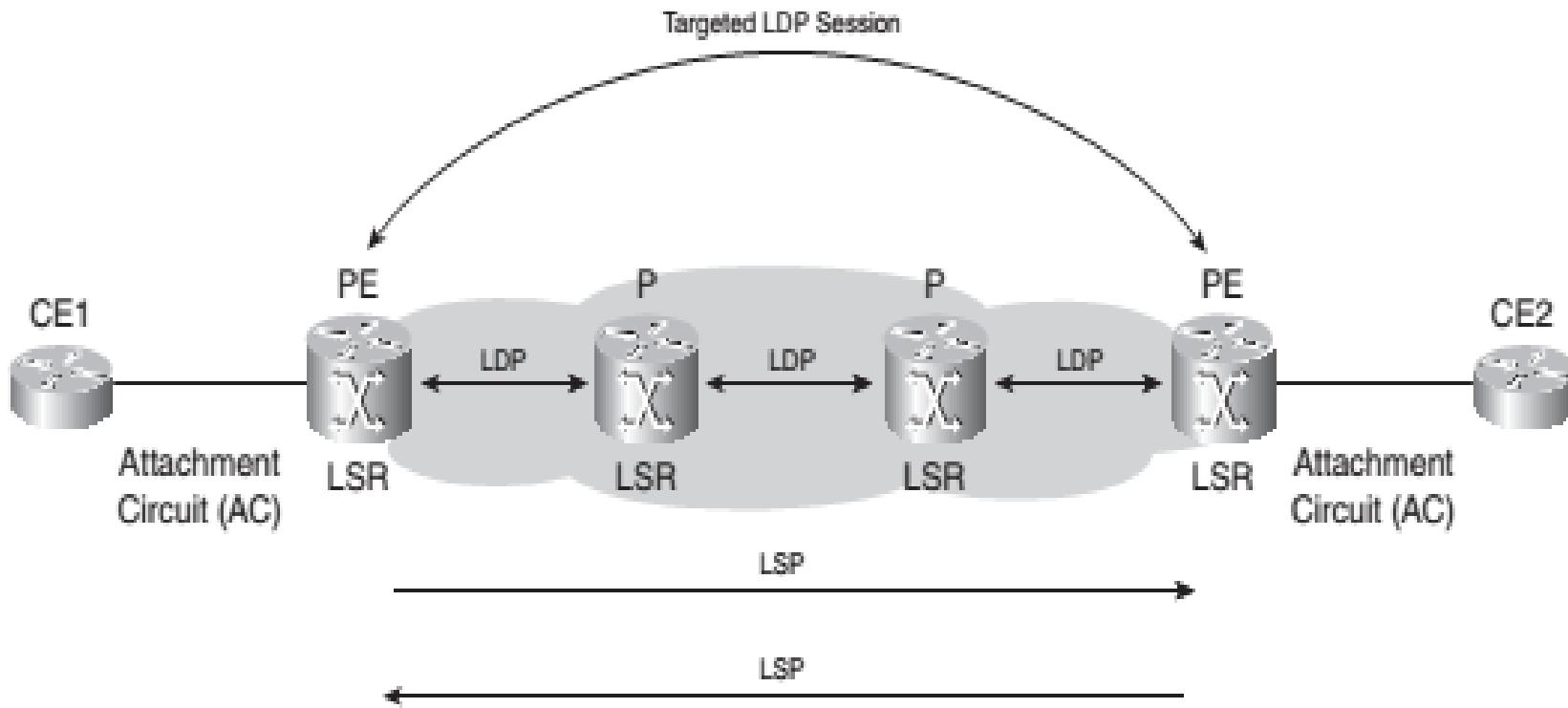
- Para la distribución de las *etiquetas utilizadas para el tunnel label, no es necesario ningún protocolo adicional*, siendo posible utilizar tanto LDP como RSVP.
- Sin embargo, *para los VC labels*, es necesario que sean asociadas con un determinado AC y publicados al PE remoto.
- Para esto, es necesario realizar una sesión targeted LDP.

## VPWS Plano de Datos:

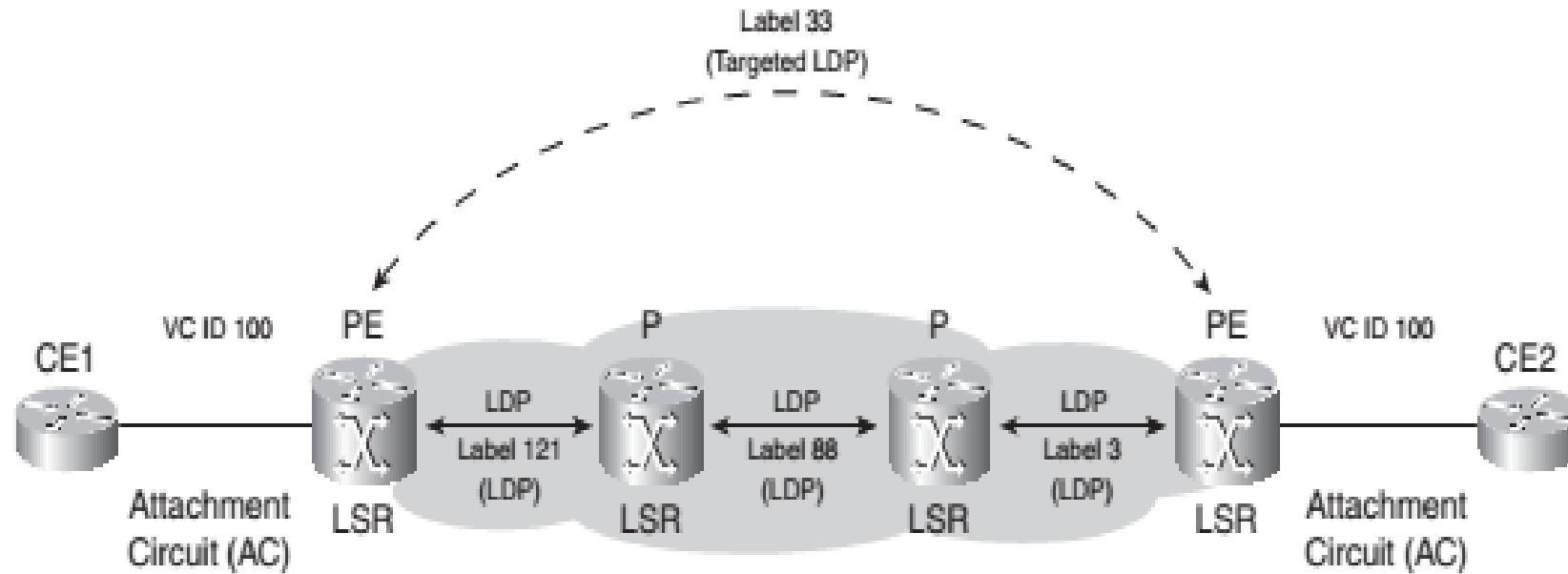
---

- *Una sesión LDP targeted* debe ser establecida entre los routers PE para señalizar el pseudowire.
- El propósito principal de esta de esta sesión LDP es la de **publicar el VC label** que está asociado con el pseudowire.
- Esta etiqueta es publicada en un **Label Mapping message** utilizando el downstream unsolicited label advertisement mode.
- Para hacer esta publicación, **LDP utiliza un tipo especial de TLV**.

# VPWS Plano de Datos:



# VPWS Plano de Datos:

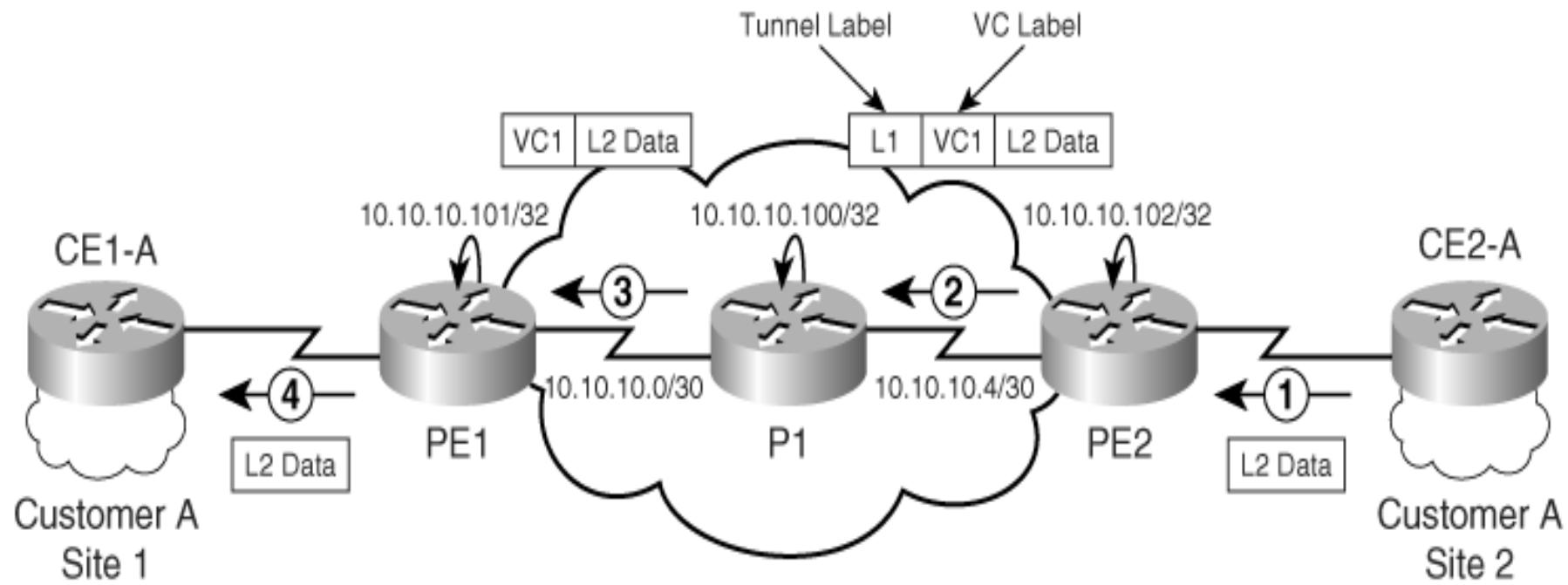


- ***El router PE de egreso:***

- Envía un mensaje LDP Label Mapping Message que indica el valor de label a utilizar para el VC Forwarding Equivalence Class (VC-FEC).
- Este VC FEC es un nuevo elemento de LDP definido específicamente para este propósito.
- La información de VC es intercambiada utilizando downstream unsolicited label distribution procedures.

*Este valor de label es luego utilizado por el router de ingreso PE como el segundo o label interno del stack impuestos a los frames le correspondiente VC-FEC.*

# VPWS Plano de Datos:



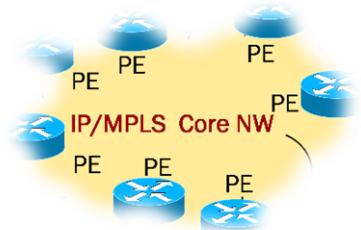
## VPWS Plano de Datos:

---

### ➤ PASOS:

- ✓ Los datos de layer 2 desde el CE2-A son recibidos en el PE2.
- ✓ Este PE2 realiza una label imposition en el paquete con el tunnel label L1 y el VC label VC1.
- ✓ PE2 utiliza el tunnel label L1 para forwardear los paquetes de datos hacia PE1.
- ✓ En el P1, el tunnel label L1 es quitado y el paquete resultante es forwardeado hacia el PE1.
- ✓ Este PE1 utiliza el inner label o VC label VC1 para forwardear el paquete hacia la interfaz de salida correspondiente en PE1.

# VPWS: Plano de Control



## VPWS Plano de Control:

---

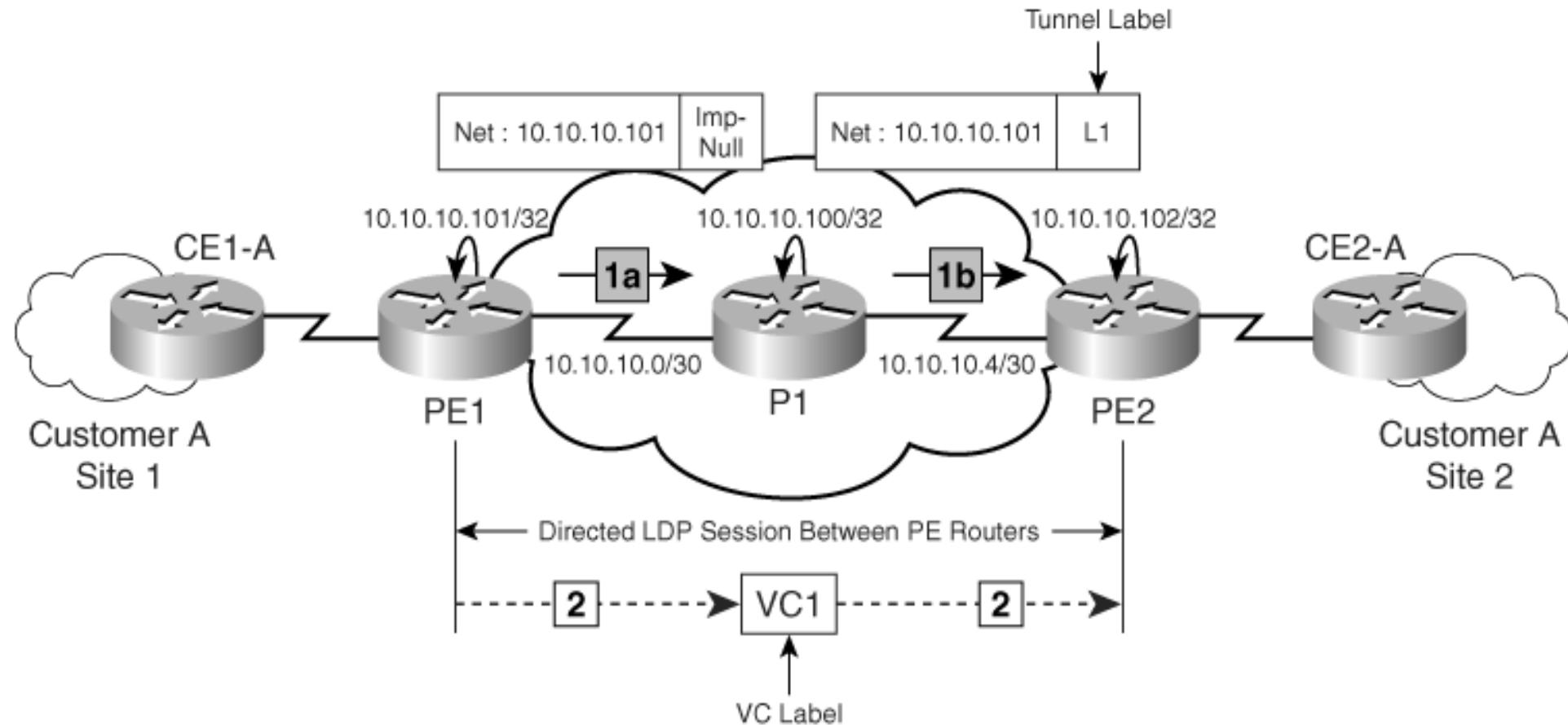
- Para establecer el LSP entre los routers PE:
  - El proceso de propagación de la etiqueta comienza cuando PE1 publica el pop label (implicit-null) para su propia dirección de loopback (en este caso 10.10.10.101)
- El router de backbone P1 publica un valor de etiqueta para L1 hacia el router PE2.
- Una **sesión LDP dirigida** debe ser también creada entre el PE1 y el PE2 para intercambiar el VC label.

## VPWS Plano de Control:

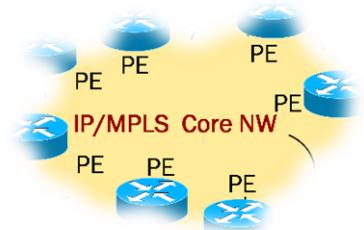
---

- Para proveer servicios AToM, cualquier par de PE requiere un sesión dirigida LDP similar.
- El router PE1 asigna una etiqueta local de VC1 para ser la etiqueta de VC para el circuito específico.
- La etiqueta VC, VC1, se anuncia al router PE2 utilizando la sesión LDP dirigido entre PE1 y PE2.
- El router PE2 ahora forma una pila de etiquetas.
  - La etiqueta de más arriba, la etiqueta del túnel, tiene el valor L1 y reenvía los paquetes a la PE1.

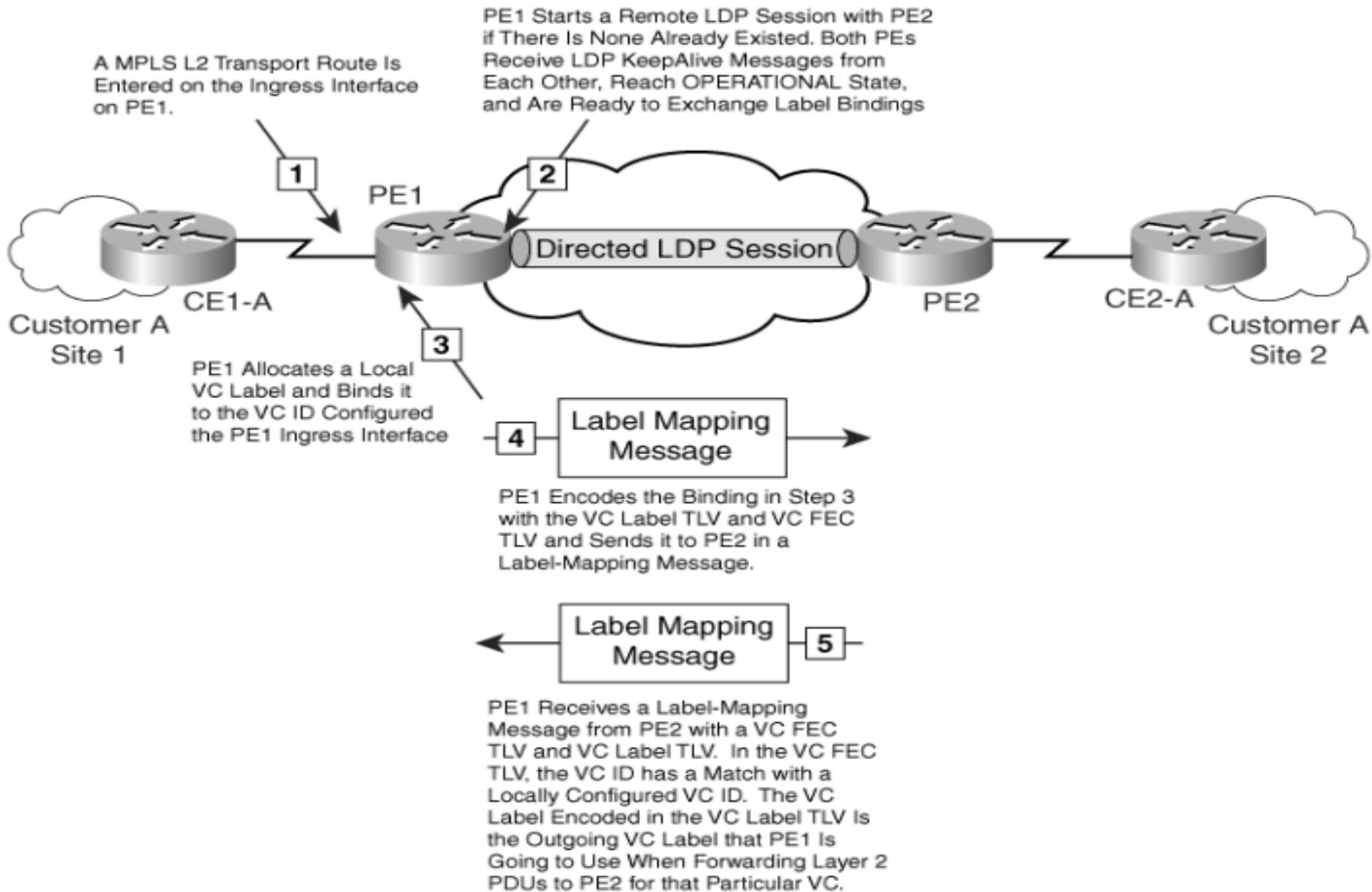
# VPWS Plano de Control:



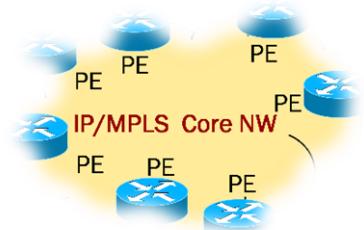
# VPWS: LDP Label Mapping Procedure



# VPWS LDP Label Mapping Procedure:



# VPWS: LDP Label Mapping Message



## VPWS LDP Label Mapping Message:

---

- El **mensaje Label Mapping** que es publicado en la sesión targeted LDP contiene los siguientes TLVs:

- Pseudowire identifier (PW ID) FEC TLV

- ✓ Identifica el pseudowire a la que la etiqueta está asociada.

- Label TLV

- ✓ Es el TLV que LDP utiliza para publicar el label MPLS.

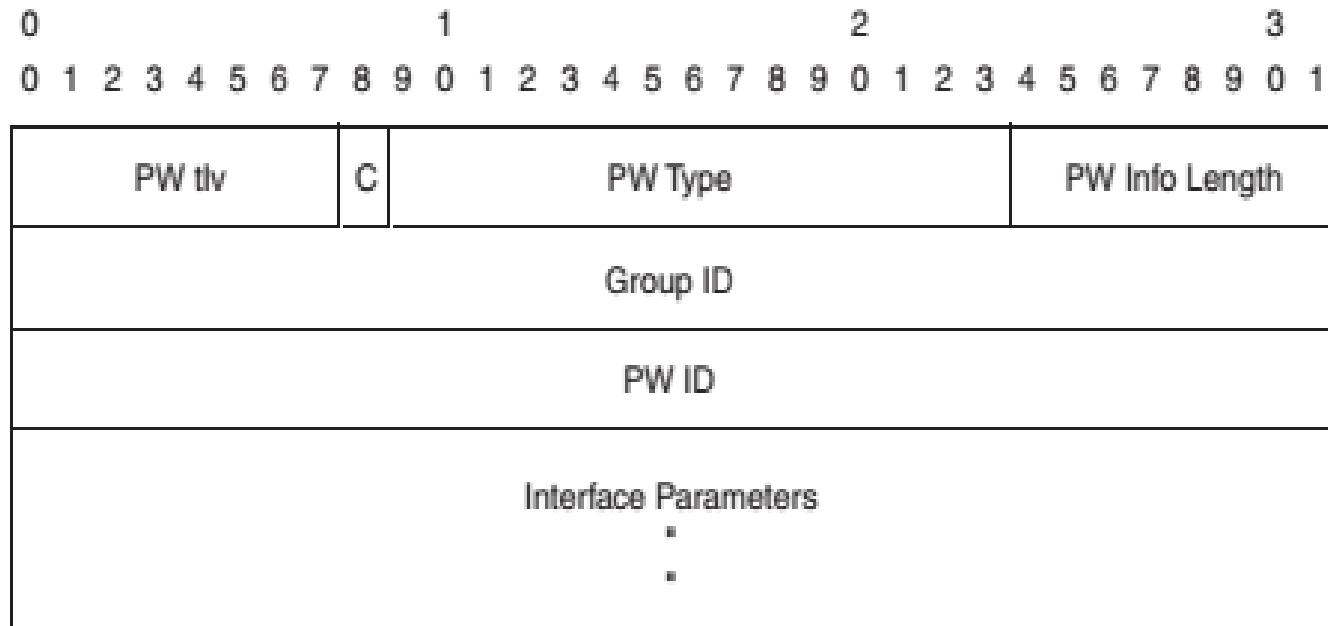
## VPWS LDP Label Mapping Message:

---

- El PW ID FEC TLV contiene los siguientes elementos:

- C-bit
- PW Type
- Group ID
- PW ID
- Interface Parameters

# VPWS LDP Label Mapping Message:



The PW TLV value is set to 128 to indicate that this is the PW ID FEC TLV.

## VPWS LDP Label Mapping Message:

---

### C-Bit

- Este campo esta en 1 indicando que una control word esta presente, y en 0 en caso contrario

### PW Type

- Este es un campo de 15 bits que representa el tipo de pseudowire. Los valores posibles son los siguientes:

# VPWS LDP Label Mapping Message:

## *PW Type Assigned by IANA*

<b>PW Type</b>	<b>Description</b>
0x0001	Frame Relay DLCI <sup>1</sup>
0x0002	ATM AAL5 SDU <sup>2</sup> VCC <sup>3</sup> transport
0x0003	ATM transparent cell transport
0x0004	Ethernet Tagged Mode
0x0005	Ethernet
0x0006	HDLC
0x0007	PPP
0x0008	SONET/SDH CEM <sup>4</sup>
0x0009	ATM n-to-one VCC cell transport
0x000A	ATM n-to-one VPC cell transport

## *PW Type Assigned by IANA (Continued)*

<b>PW Type</b>	<b>Description</b>
0x000B	IP Layer 2 Transport
0x000C	ATM one-to-one VCC cell mode
0x000D	ATM one-to-one VPC cell mode
0x000E	ATM AAL5 PDU <sup>5</sup> VCC transport
0x000F	Frame-Relay Port mode
0x0010	SONET/SDH CEP <sup>6</sup>
0x0011	Structure-Agnostic E1 over Packet (SAToP)
0x0012	Structure-Agnostic T1 (DS1) over Packet (SAToP)
0x0013	Structure-Agnostic E3 over Packet (SAToP)
0x0014	Structure-Agnostic T3 (DS3) over Packet (SAToP)
0x0015	CESoPSN <sup>7</sup> basic mode
0x0016	TDMoIP <sup>8</sup> basic mode
0x0017	CESoPSN TDM <sup>9</sup> with CAS <sup>10</sup>
0x0018	TDMoIP TDM with CAS
0x0019	Frame Relay DLCI

## VPWS LDP Label Mapping Message:

---

### Group ID

- Este campo identifica un grupo de pseudowires.
- Cisco IOS asigna el mismo Group ID a todos los ACs en la misma interface.

## VPWS LDP Label Mapping Message:

---

### PW ID

- Este es un campo de 32 bits, que se utiliza como identificador de conexión que en conjunto con el PW type identifica en forma completa el pseudowire.
- En CISCO IOS se especifica el PW ID en ambos PE routers con el comando:

```
#xconnect peer-router-id vcid
```

- El PW ID es el VC ID que se observa en las salidas de los comandos CISCO

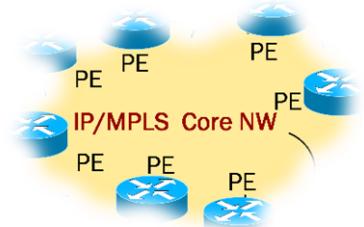
# VPWS LDP Label Mapping Message:

---

## Interface Parameters

- Este campo describe algunos parámetros específicos de las interfaces como ser MTU de la interfaz hacia el router CE, una descripción, la requested VLAN ID.
- Es importante notar que si el parámetro de MTU no coincide en ambos extremos, el pseudowire no es señalizado.
- Debido a que los LSP son unidireccionales, un pseudowire puede ser formado solo si otro LSP existe en la dirección opuesta entre el mismo par de routers PE.
- El PW ID FEC TLV se utiliza para identificar y machear dos LSP opuestos entre un par de routers PE

# VPWS: Implementación Ethernet over MPLS (AToM)



## Implementación Ethernet over MPLS (AToM):

---

- La solución AToM para el transporte de Ethernet sobre MPLS es estrictamente punto a punto.
- Es decir, los frames Ethernet son transportados desde un PE de ingreso a un PE de egreso.
- Esto es el equivalente a un servicio LAN-to-LAN bridgeado sobre links WAN punto a punto.
- Un servicio de conexión multipunto, LAN-like sobre MPLS se denomina VPLS
- El AC puede ser un Puerto Ethernet o una vlan ID 802.1q.

## Implementación Ethernet over MPLS (AToM):

---

- Para cada uno de los dos tipos de ACs:
  - LDP señaliza un tipo diferente de VC o PW type en el PW ID FEC TLV mediante la sesión targeted LDP entre ambos routers PE.
    - VC Type 5 es utilizado para Ethernet port mode
    - VC Type 4 se utiliza para Ethernet vlan mode.

## Implementación Ethernet over MPLS (AToM):

---

- En el Ethernet VLAN mode:

- un VLAN header que tiene significado para los routers PE está presente, en otras palabras, los routers PE revisan este vlan header.

- En el Ethernet port mode:

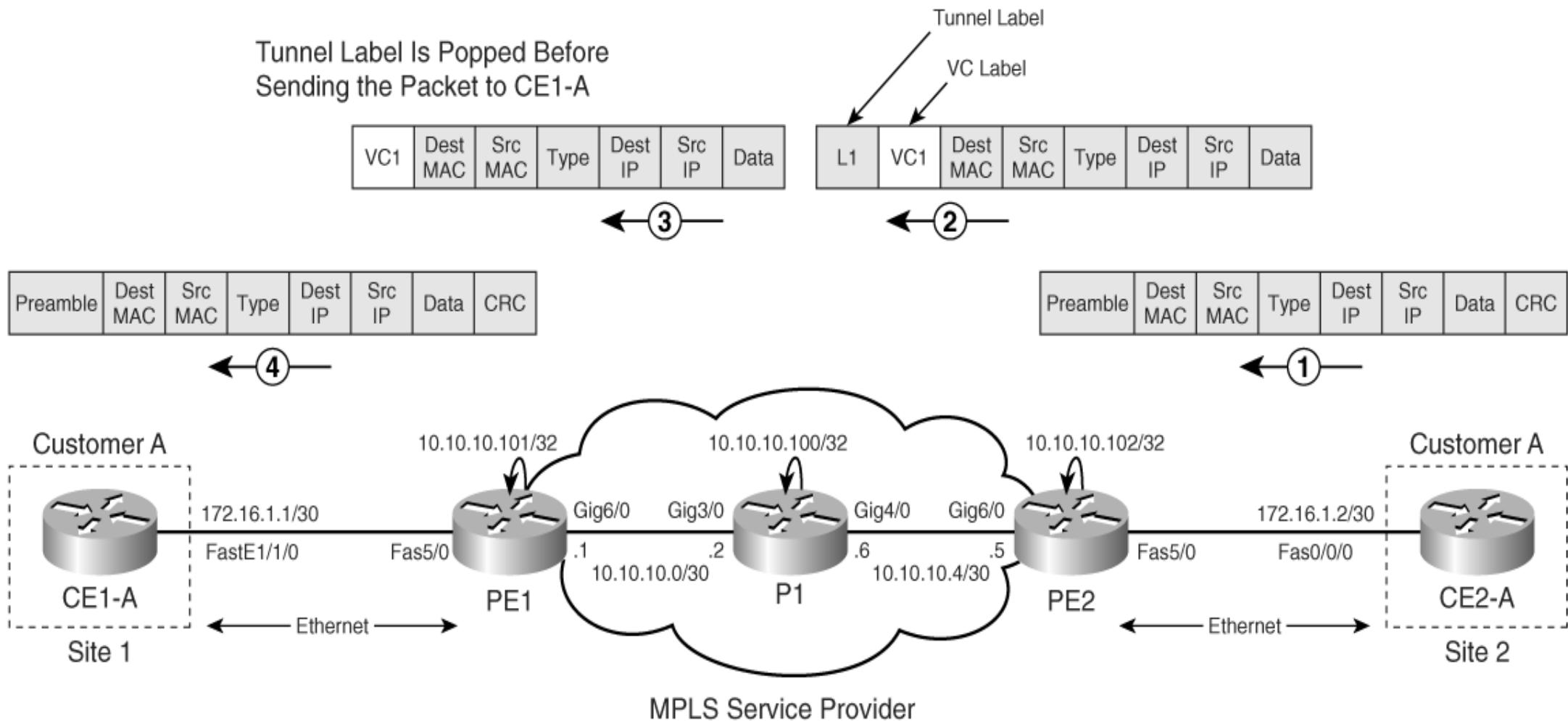
- el VLAN header puede o no estar presente en el frame. En este caso, si el VLAN header existe, el router PE no lo inspecciona, simplemente lo transporta en forma transparente.

## Router-Based Ethernet over **MPLS—Port Mode**:

---

- En este modo, toda la trama Ethernet, sin el preámbulo o FCS es transportada como un único paquete AToM.
- La utilización de control Word es opcional.
- En este modo, las interfaces utilizan VC type 5 o 0x0005

# Router-Based Ethernet over MPLS—Port Mode:

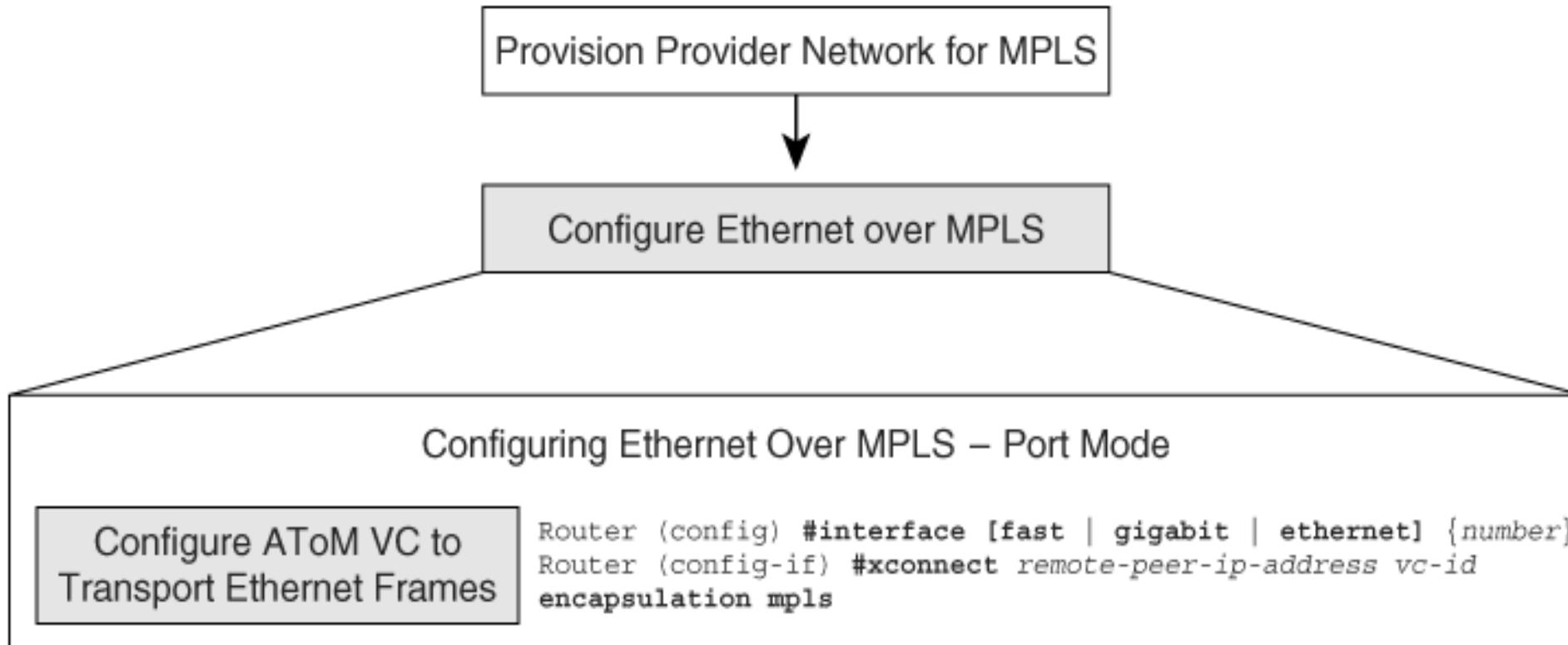


## Router-Based Ethernet over MPLS—Port Mode:

---

- Como muestra la figura anterior:
- el CE1-A y el CE2-A intercambian frames Ethernet sobre la red de backbone del proveedor de servicio.
- El router PE2 recibe un frame Ethernet desde el CE2-A y lo encapsula en como un frame MPLS.
- Este frame es luego forwardeado a través de la red de backbone del proveedor hasta llegar a PE1.
- Este router PE1 remueve el header MPLS y reproduce el frame Ethernet original para enviarlo hasta CE1-A

# Router-Based Ethernet over MPLS—Port Mode:



## Router-Based Ethernet over MPLS—Port Mode:

---

- Los pasos para la configuración del port mode Ethernet over MPLS se fundamentan en la definición del xconnect bajo la internet Ethernet correcta, para habilitar el transporte de frames Ethernet desde los dispositivos CE finales sobre la red MPLS.
- Ejemplo: Enable Transport of Ethernet over MPLS on PE Router

- PE1(config)#interface FastEthernet5/0
- PE1(config-if)#xconnect 10.10.10.102 100 encapsulation mpls
- PE2(config)#interface FastEthernet5/0
- PE2(config-if)#xconnect 10.10.10.101 100 encapsulation mpls

# Router-Based Ethernet over MPLS—Port Mode:

```
hostname PE1
```

```
!
```

```
interface FastEthernet5/0
```

```
no ip address
```

```
xconnect 10.10.10.102 100 encapsulation mpls
```

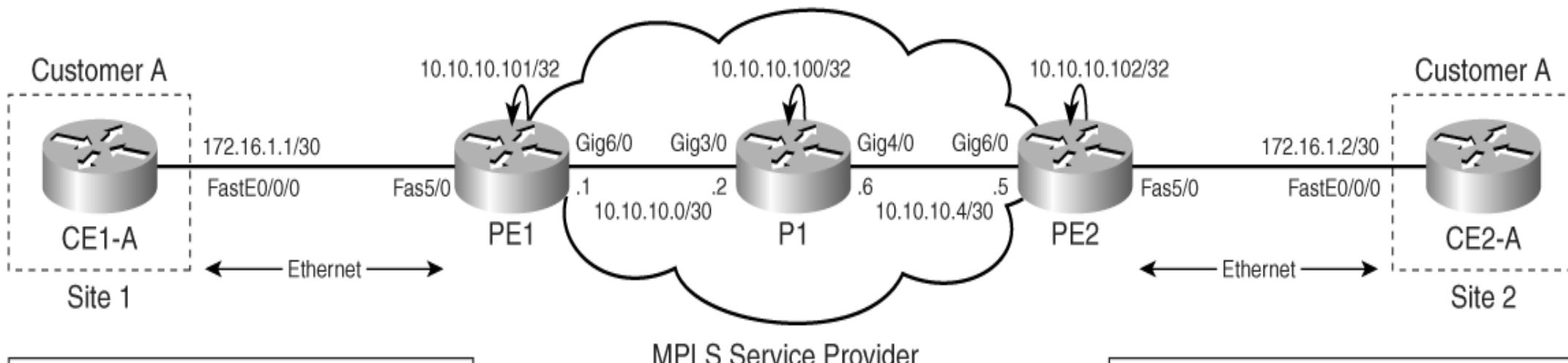
```
hostname PE2
```

```
!
```

```
interface FastEthernet5/0
```

```
no ip address
```

```
xconnect 10.10.10.101 100 encapsulation mpls
```



```
hostname CE1-A
```

```
!
```

```
interface FastEthernet1/1/0
```

```
ip address 172.16.1.1 255.255.255.252
```

```
hostname CE2-A
```

```
!
```

```
interface FastEthernet0/0/0
```

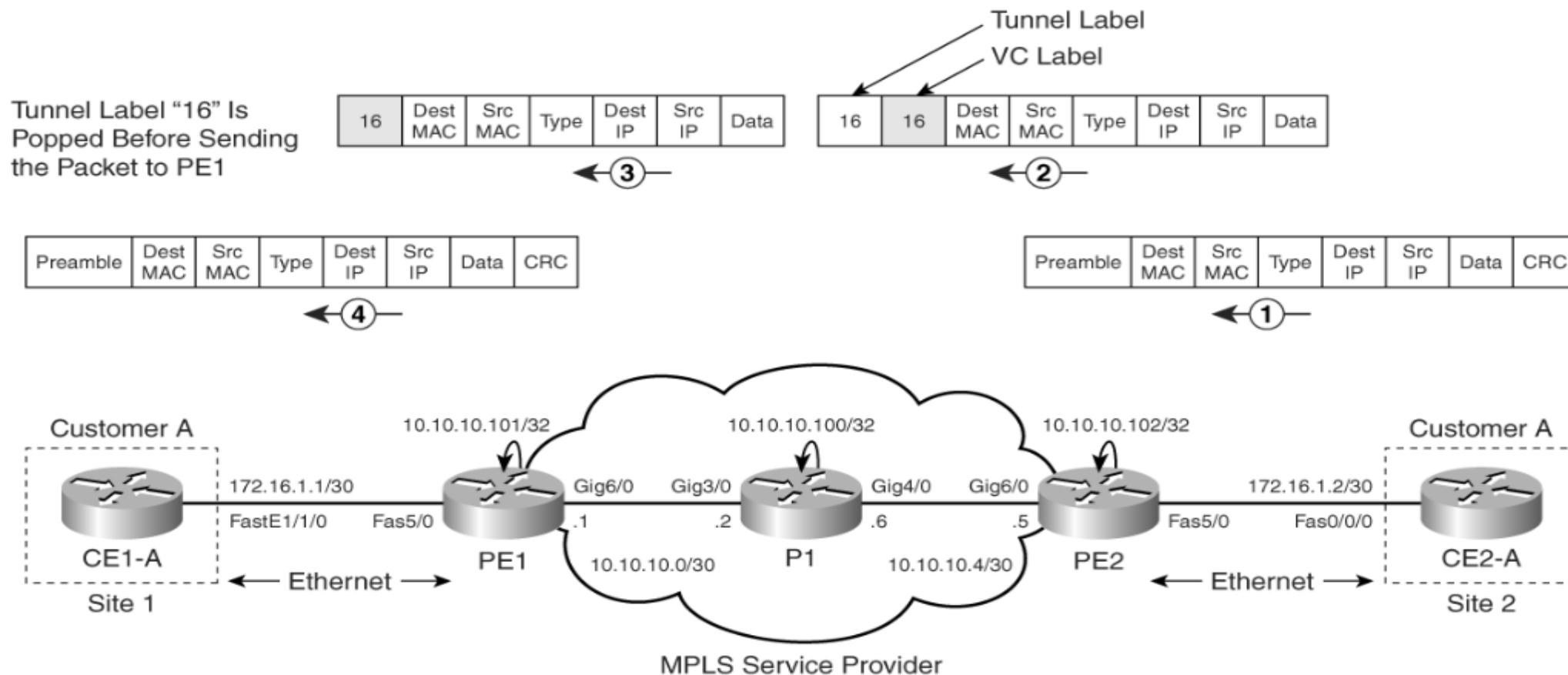
```
ip address 172.16.1.2 255.255.255.252
```

## Router-Based Ethernet over MPLS—Port Mode:

---

- Como muestra la siguiente figura:
- Durante las operaciones del plano de control, PE1 distribuye el VC label 16 para el AC conectado contra el CE1-A.
- Este VC label es propagado a través de la red MPLS hacia el PE2, que utiliza este VC label 16 en el plano de forwarding para los paquetes que se originan desde el CE2-A destinados al CE1-A.
- El tunnel label 16 o IGP label 16, que es asignado por P1 para la interfaz de loopback de PE1 10.10.10.101 guía el paquete desde PE2 hacia PE1 para los datos originándose desde el CE2-A hacia el CE1-A.

# Router-Based Ethernet over MPLS—Port Mode:



```
PE1#show mpls forwarding-table
```

Local Label	Outgoing Label or VC	Prefix	Bytes	Label or Tunnel Id	Switched	Outgoing Interface	Next Hop
16	No Label	12ckt (100)	14198	none		point2point	
17	Pop Label	10.10.10.4/30	0			Gi6/0	10.10.10.2
18	Pop Label	10.10.10.100/32	0			Gi6/0	10.10.10.2
19	17	10.10.10.102/32	0			Gi6/0	10.10.10.2

```
P1#show mpls forwarding-table
```

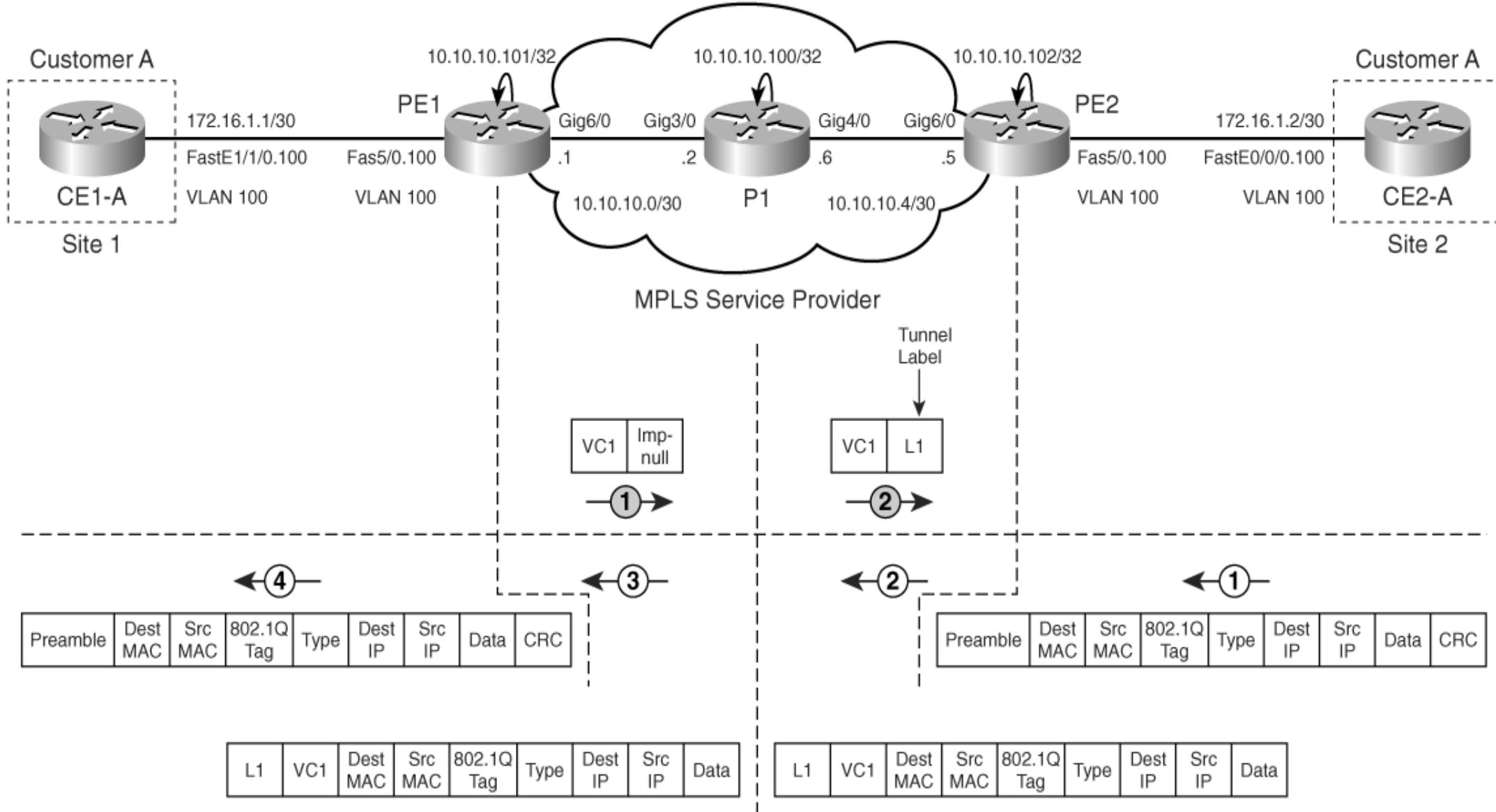
Local Label	Outgoing Label or VC	Prefix	Bytes	Label or Tunnel Id	Switched	Outgoing Interface	Next Hop
16	Pop Label	10.10.10.101/32	265060			Gi3/0	10.10.10.1
17	Pop Label	10.10.10.102/32	271497			Gi4/0	10.10.10.5

## Router-Based Ethernet over **MPLS—VLAN Mode**:

---

- En el siguiente ejemplo, vemos que el encapsulamiento de la interfaz entre el CE y el router PE soporta VLANs.
- Diferentes subinterfaces en el router PE conectan diferentes VLANs.
- La interfaz del PE1 conectada al CE1-A es configurada como una **VLAN subinterfaces**.
- Esta VLAN subinterfaces es utilizada para el forwarding AToM.

# Router-Based Ethernet over MPLS—VLAN Mode:



## Router-Based Ethernet over MPLS—VLAN Mode:

---

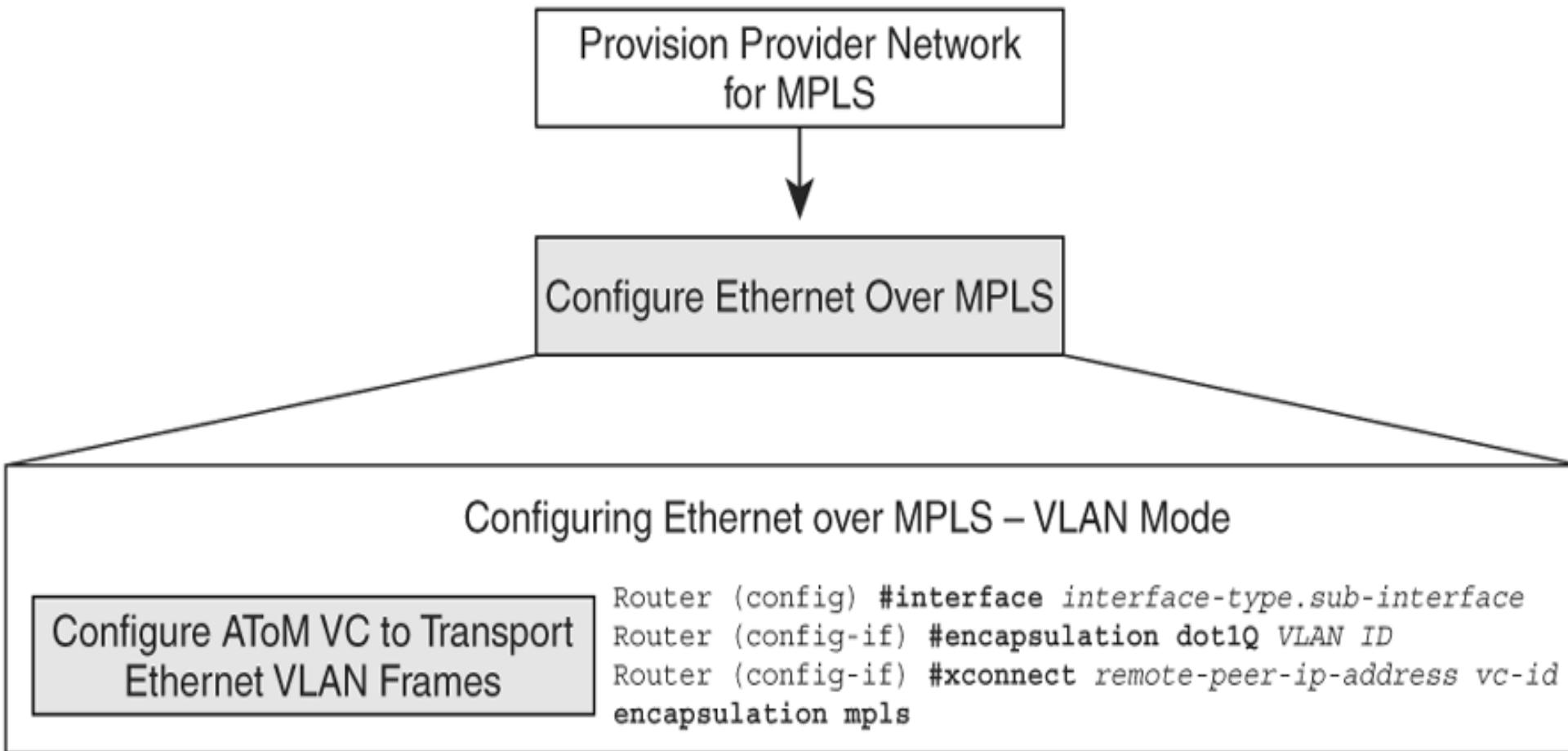
- En la figura anterior, se observa que PE1 y PE2 son configurados con EoMPLS para propagar la VLAN 100 a través del backbone.
- PE1 y PE2 tiene cada uno una interfaz de loopback con dirección IP 10.10.10.101 y 10.10.10.102 respectivamente.
- Esta dos direcciones son utilizadas como LDP peer ID.
- La subinterfaz fas5/0.100 esta configurada con la VLAN 100 en ambos routers PE.

## Router-Based Ethernet over MPLS—VLAN Mode:

---

- Esta subinterfaz está configurada para EoMPLS.
- Los frames Ethernet son encapsulados con MPLS y forwardeados hacia el router PE del otro extremo.
- El VC identifier (100) asocia la conexión con el otro extremo.
- Es requerido que en ambos extremos se utilice el mismo valor de identificador de VC.
- No es requerimiento que el VLAN identifier sea el mismo en ambos extremos.

# Router-Based Ethernet over MPLS—VLAN Mode:



## Router-Based Ethernet over MPLS—VLAN Mode:

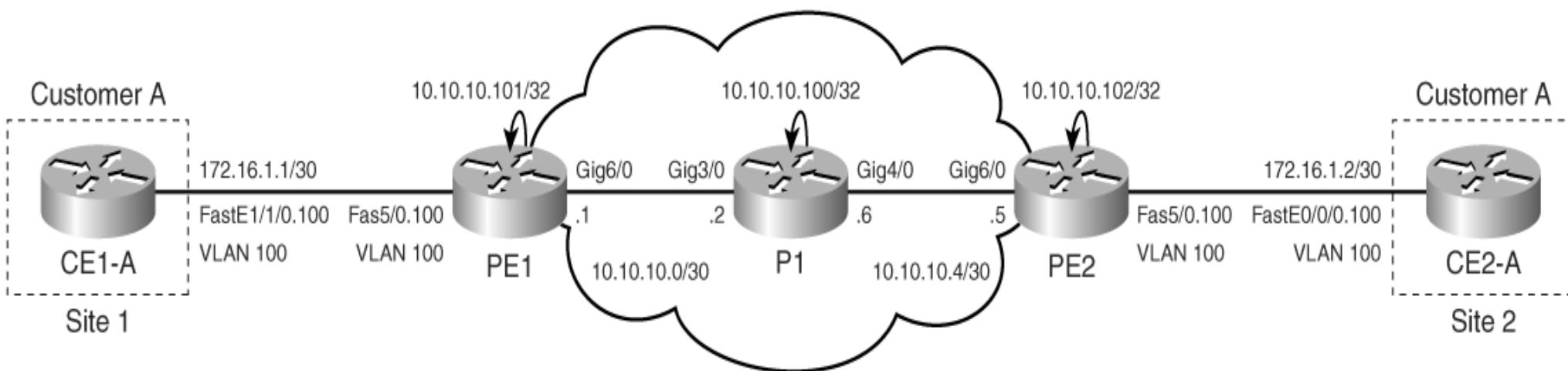
---

- PE1(config)#interface FastEthernet5/0.100
  - PE1(config-subif)# encapsulation dot1Q 100
  - PE1(config-subif)# no cdp enable
  - PE1(config-subif)# xconnect 10.10.10.102 100 encapsulation mpls
- 
- PE2(config)#interface FastEthernet5/0.100
  - PE2(config-subif)# encapsulation dot1Q 100
  - PE2(config-subif)# no cdp enable
  - PE2(config-subif)# xconnect 10.10.10.101 100 encapsulation mpls

# Router-Based Ethernet over MPLS—VLAN Mode:

```
hostname PE1
!
interface FastEthernet5/0.100
encapsulation dot1Q 100
no cdp enable
xconnect 10.10.10.102 100 encapsulation mpls
```

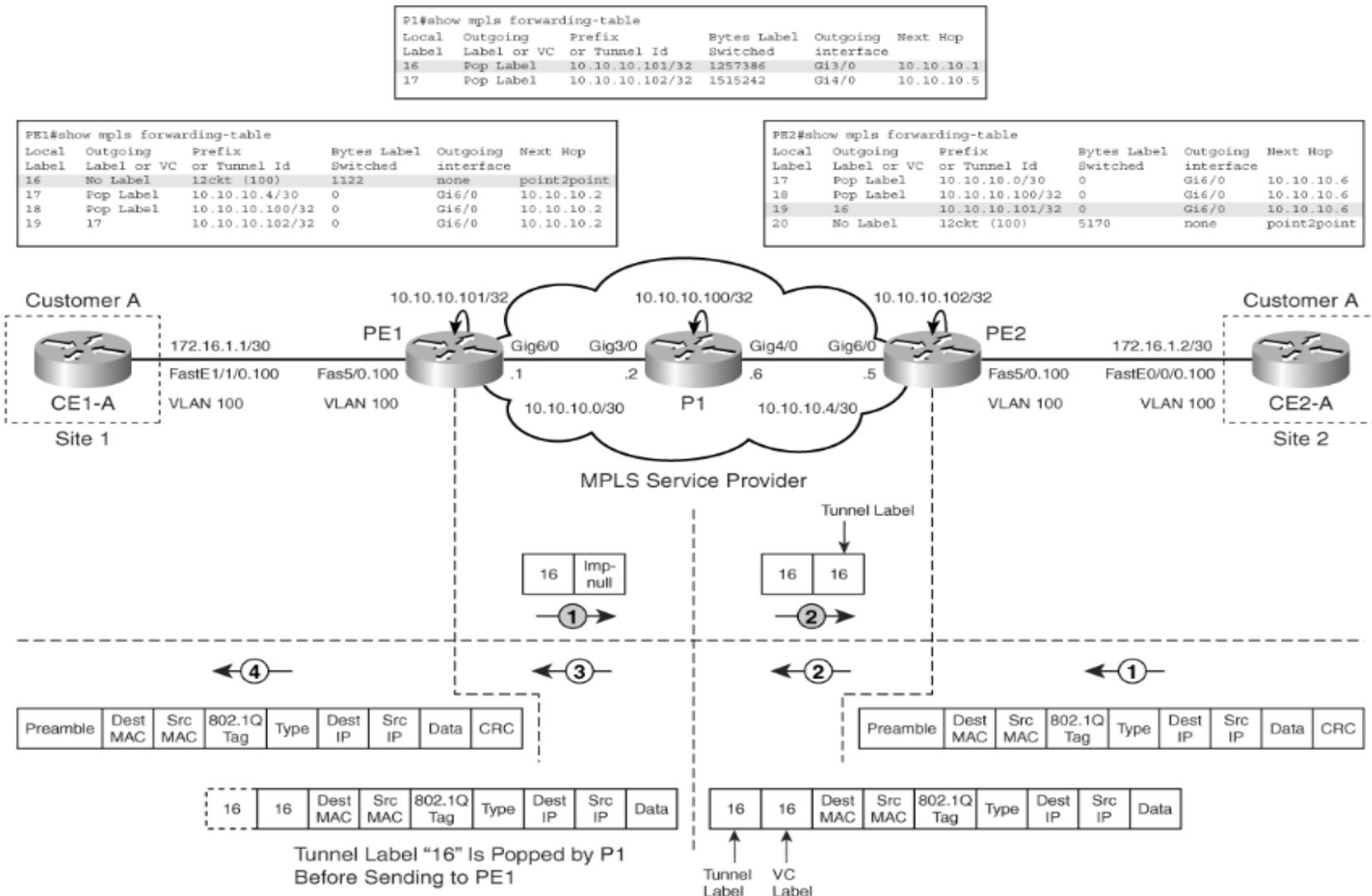
```
hostname PE2
!
interface FastEthernet5/0.100
encapsulation dot1Q 100
no cdp enable
xconnect 10.10.10.101 100 encapsulation mpls
```



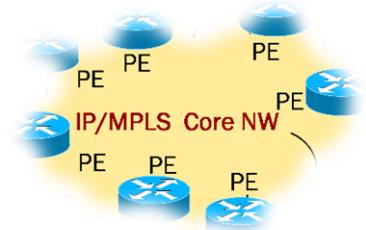
```
hostname CE1-A
!
interface FastEthernet1/1/0.100
encapsulation dot1Q 100
ip address 172.16.1.1 255.255.255.252
```

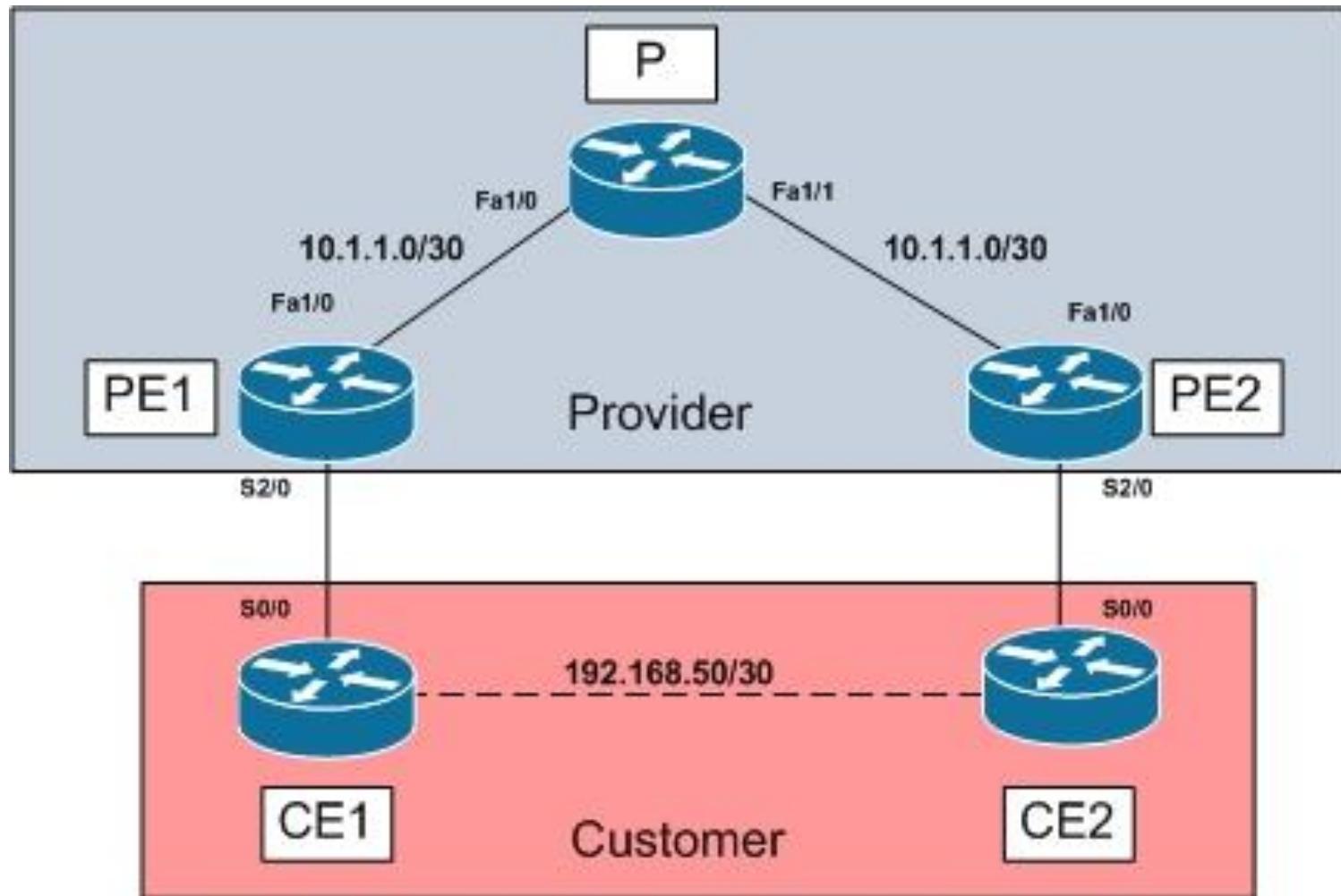
```
hostname CE2-A
!
interface FastEthernet0/0/0.100
encapsulation dot1Q 100
ip address 172.16.1.2 255.255.255.252
```

# Router-Based Ethernet over MPLS—VLAN Mode:



# VPWS: LAB





## P Router:

```
hostname P
!
mpls label protocol ldp
mpls ldp router-id lo0 force
!
interface Loopback0
 ip address 1.1.1.1 255.255.255.255
!
interface FastEthernet1/0
 ip address 10.1.1.1 255.255.255.252
 duplex auto
 speed auto
 mpls ip
!
interface FastEthernet1/1
 ip address 10.1.2.1 255.255.255.252
 duplex auto
 speed auto
 mpls ip
!
router ospf 100
 router-id 1.1.1.1
 log-adjacency-changes
 network 1.1.1.1 0.0.0.0 area 0
 network 10.1.1.0 0.0.0.3 area 0
 network 10.1.2.0 0.0.0.3 area 0
```

## PE1 Router:

```
hostname PE1
!
mpls label protocol ldp
mpls ldp router-id lo0 force
!
pseudowire-class one
  encapsulation mpls
!
interface Loopback0
  ip address 2.2.2.2 255.255.255.255
!
interface FastEthernet1/0
  ip address 10.1.1.2 255.255.255.252
  duplex auto
  speed auto
  mpls ip
!
interface Serial2/0
  no ip address
  xconnect 3.3.3.3 100 pw-class one
!
router ospf 100
  router-id 2.2.2.2
  log-adjacency-changes
  network 2.2.2.2 0.0.0.0 area 0
  network 10.1.1.0 0.0.0.3 area 0
```

## PE2 Router:

```
hostname PE2
!
mpls label protocol ldp
mpls ldp router-id lo0 force
!
pseudowire-class one
  encapsulation mpls
!
interface Loopback0
  ip address 3.3.3.3 255.255.255.255
!
interface FastEthernet1/0
  ip address 10.1.2.2 255.255.255.252
  duplex auto
  speed auto
  mpls ip
!
interface Serial2/0
  no ip address
  xconnect 2.2.2.2 100 pw-class one
!
router ospf 100
  router-id 3.3.3.3
  log-adjacency-changes
  network 2.2.2.2 0.0.0.0 area 0
  network 10.1.2.0 0.0.0.3 area 0
```

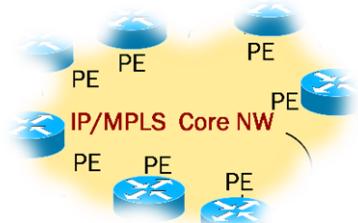
## CE1 Router:

```
hostname CE1
!
interface Serial0/0
 ip address 192.168.50.1 255.255.255.252
```

## CE2 Router:

```
hostname CE2
!
interface Serial0/0
 ip address 192.168.50.2 255.255.255.252
```

# Virtual Private LAN Service (VPLS):

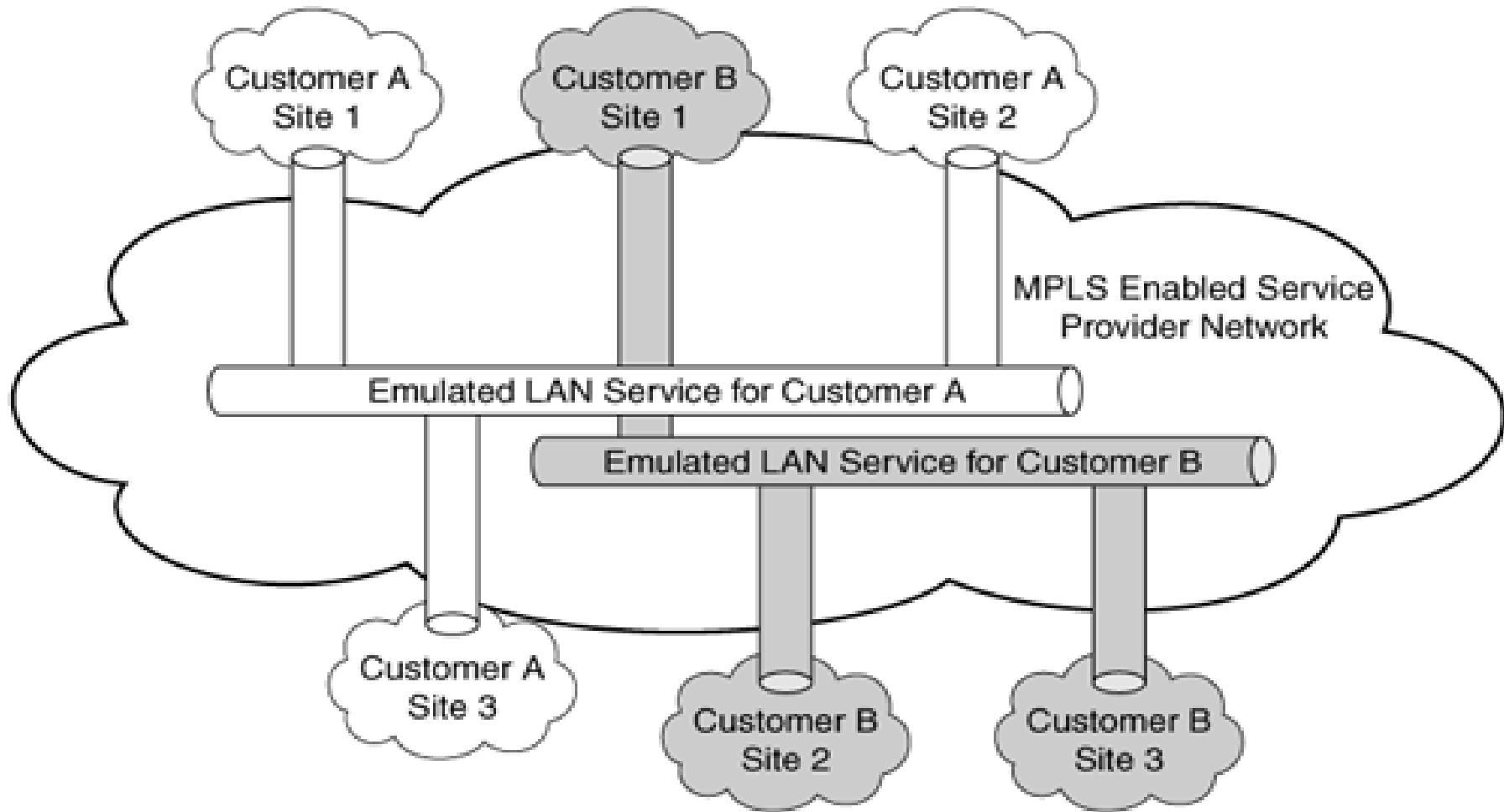


## Virtual Private LAN Service (VPLS):

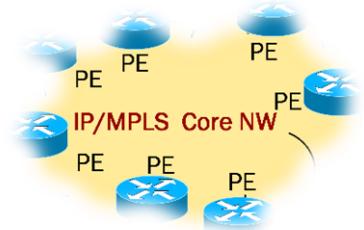
---

- Permite que **múltiples LANs Ethernet** de sitios de diferentes clientes sean conectadas en conjunto a través de la red del proveedor de servicio.
- Emulando estar *dentro de un único segmento LAN* para los clientes.
- De esta forma, el SP ofrece un servicio en el cual *cada cliente se puede comunicar con sitios remotos como si estuviera conectado en un segmento LAN Ethernet privado*

# Virtual Private LAN Service (VPLS):



# VPLS: Componentes



# Virtual Private LAN Service (VPLS) – Componentes:

---

Tiene seis componentes:

- **Attachment circuit (AC):**
- Es un circuito de capa 2 punto a punto entre el router CE en el sitio del cliente y el router provider edge PE en la red del operador.
- En VPLS, los attachment circuits soportados actualmente son Ethernet en:
  - port mode, donde la interfaz solo envía y acepta frames Ethernet sin tagear.
  - 802.1Q vlan o trunk mode, donde la interfaz es configurada como un trunk 802.1Q y solo envía y recibe frames Ethernet tagueados y en vlan nativa

## Virtual Private LAN Service (VPLS) – Componentes:

---

- Dot1q túnel mode, En este modo, un túnel 802.1Q es configurado y un vlan tag de acceso es agregado al paquete en la interfaz de ingreso del túnel y removido en la interfaz de egreso del túnel.

Los paquetes independientemente de estar con o sin tag 802.1q son forwardeados a través del túnel 802.1Q

## Virtual Private LAN Service (VPLS) – Componentes:

---

- **Packet Switched Network (PSN) tunnels.**
  - Estos PSN tunnels son construidos entre dos dispositivos PE en una PSN para proveer servicios de transporte para un o mas pseudowire (emulated VC).
  - En una arquitectura VPLS, el transporte es previsto sobre los LSP MPLS

## Virtual Private LAN Service (VPLS) – Componentes:

---

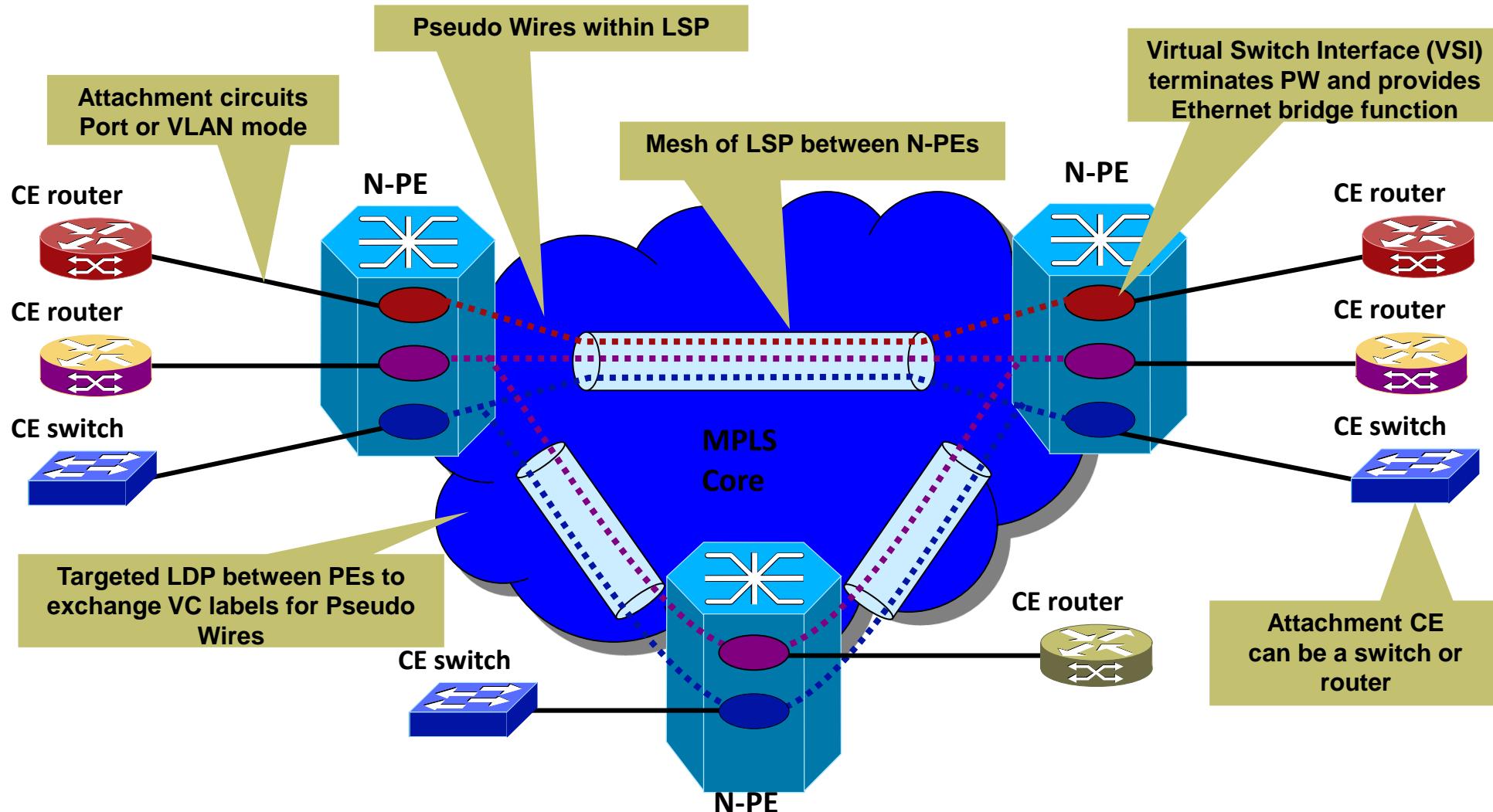
- **Pseudowires:**

- Un pseudo wire es un circuito virtual emulado que conecta dos attachment circuits (AC) en dos routers PE diferentes a través de la red MPLS.

- **Auto-Discovery.**

- Es un mecanismo que habilita múltiples routers PE que participan en un dominio VPLS para encontrarse entre ellos.
- Esto automatiza la creación del mesh LSP.
- En el caso de ausencia de auto-discovery el SP debe explícitamente identificar los PE que son parte de la instancia VPLS.

# Virtual Private LAN Service (VPLS) – Componentes:



# Virtual Private LAN Service (VPLS) – Componentes:

---

- **Auto configuration:**

- Es un mecanismo que automáticamente establece pseudo wires (emulated VC) para CE recientemente descubiertos.

- **Virtual Switching Instance (VSI) o Virtual Forwarding Instance (VFI).**

- Estos son entidades virtuales de forwarding de capa 2 que definen la membresía al dominio VPLS.
- Asemeja switches virtuales en los routers PE

# Virtual Private LAN Service (VPLS) – Componentes:

---

## Características VSI/VFI:

### Flooding / Forwarding

MAC table instances per customer (port/vlan) for each PE

VFI will participate in learning and forwarding process

Associate ports to MAC, flood unknowns to all other ports

### Address Learning / Aging

LDP enhanced with additional MAC List TLV (label withdrawal)

MAC timers refreshed with incoming frames

### Loop Prevention

Create full-mesh of Pseudo Wire VCs (EoMPLS)

Unidirectional LSP carries VCs between pair of N-PE Per

A VPLS use “split horizon” concepts to prevent loops

## Virtual Private LAN Service (VPLS) – Componentes:

---

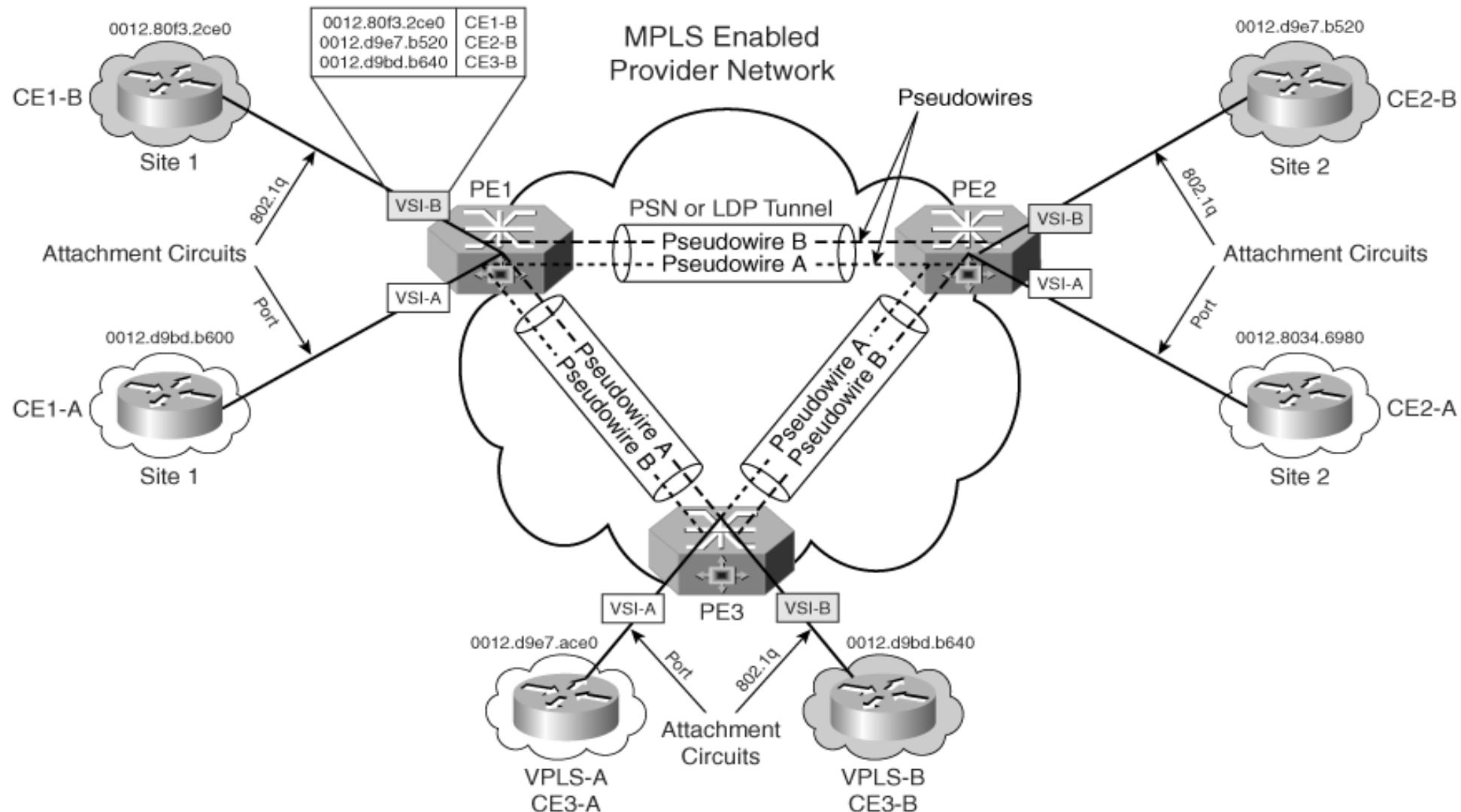
- Un **dominio VPLS** consiste en:
  - *interfaces Ethernet o VLNs que pertenecen a la misma (virtual) LAN pero que están conectadas a múltiples dispositivos PE.*
- **Por ejemplo**, el dominio VPLS para el Customer As consiste en interfaces Ethernet conectadas a routers del Customer As en diferentes sitios.
- **Los VSI aprenden direcciones MAC remotas** y son responsables del correcto forwarding del tráfico del cliente a los nodos destinos apropiados.

## Virtual Private LAN Service (VPLS) – Componentes:

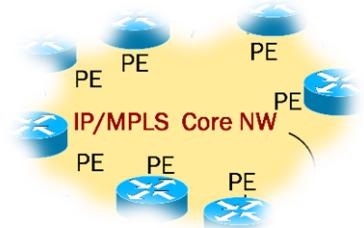
---

- **El VSI es responsable de** garantizar que cada dominio VPLS este libre de loops.
- **Adicionalmente es responsable de** diferentes funciones como:
  - MAC address management,
  - dynamic MAC address learning en puertos físicos y VC,
  - aging de MAC address,
  - MAC address withdrawal,
  - Flooding
  - forwarding de datos.

# Virtual Private LAN Service (VPLS) – Componentes:



# VPLS: Señalización



## Virtual Private LAN Service (VPLS) – Señalización:

---

- VPLS requiere:

*“Una malla completa de pseudowire entre los routers PE para cada instancia VPLS”*

- Cuando se configura la instancia VPLS en el router PE, **se debe especificar los vecinos de este router PE VPLS.**
- Esto significa que debe especificar ***todos los routers PE remotos para este router PE para esta instancia VPLS.***

## Virtual Private LAN Service (VPLS) – Señalización:

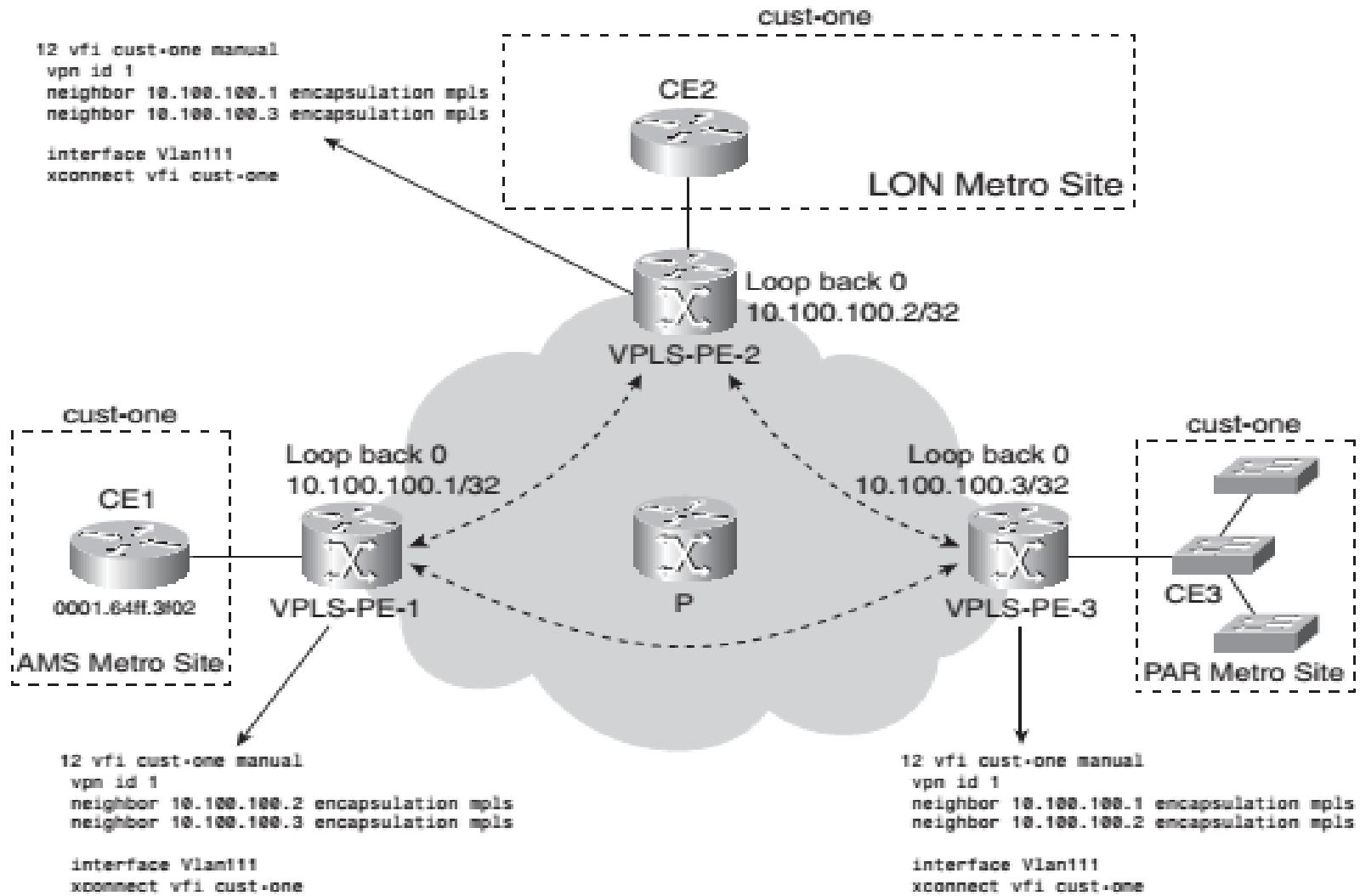
---

- Los routers PE forman entonces una **sesión targeted LDP entre ellos** en una malla completa.

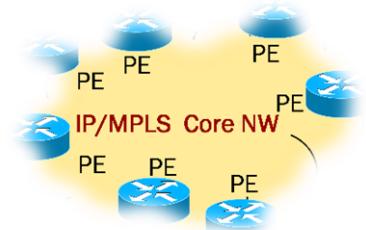
*Las targeted LDP señalan cada sesión VC o pseudowire entre un par de routers PE y anuncia las etiquetas VC.*

- **El VC ID es el identificador de VPN** (VPN ID) que se debe asignar a una instancia VPLS por medio de la configuración.
- Cada pseudowire entre un par de routers PE para esa instancia **VPLS tiene que VC ID**.
- **Sin embargo**, la etiqueta VC local que el router asigna para esa instancia VPLS es diferente para cada pseudowire.

# Virtual Private LAN Service (VPLS) – Señalización:



# VPLS: MAC address learning



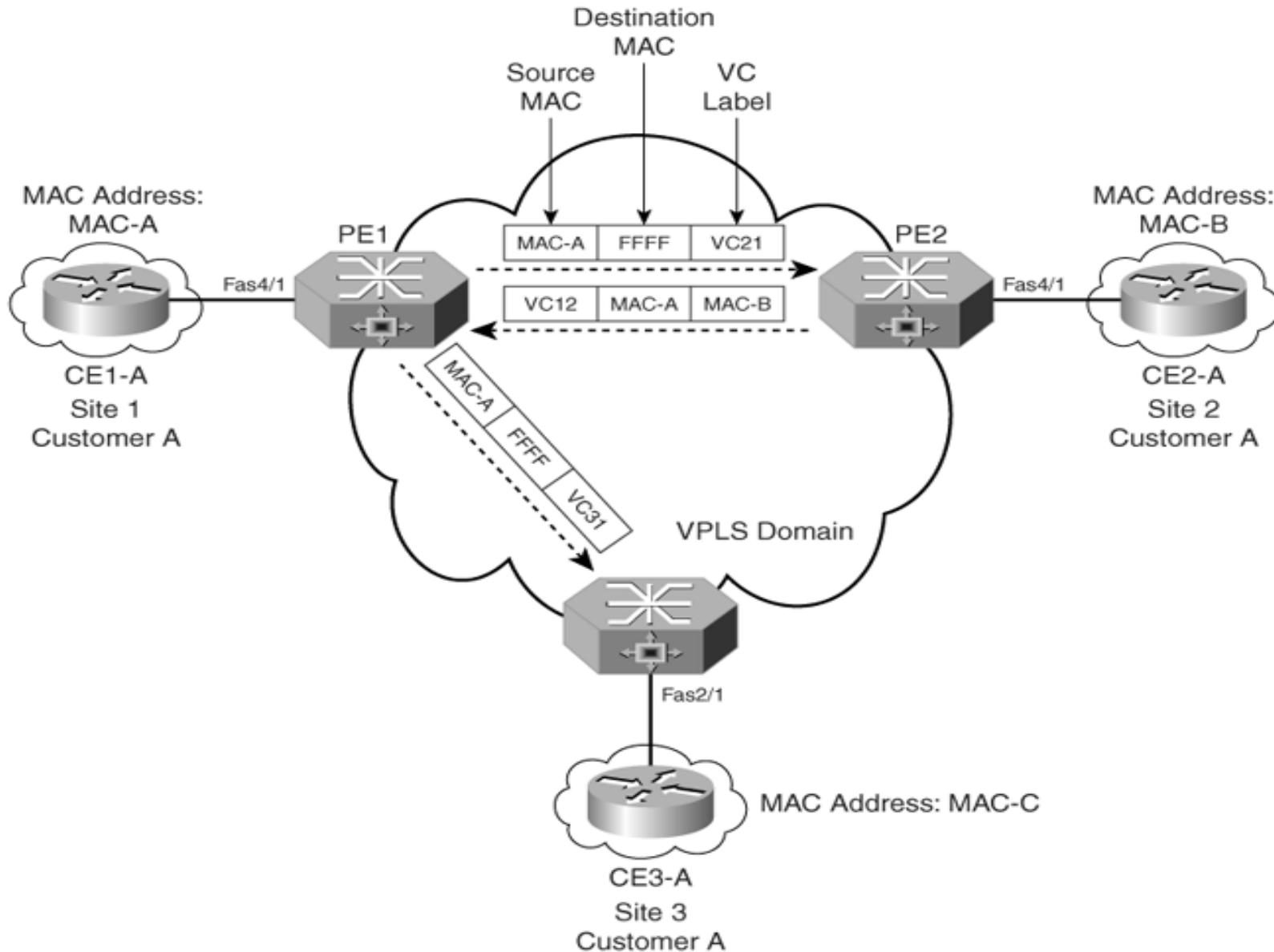
## Virtual Private LAN Service (VPLS) – Operación:

---

MAC Address learning.

- Utilizando sesiones LDP dirigidas, cada PE publica un VC label mapping.
- Este es utilizada como parte del label stack impuesto a los frames Ethernet por el router PE de ingreso durante el forwardeo de paquetes.
- Las implementaciones VPLS de Cisco aprenden MAC address utilizando el mecanismo del estándar 802.1d, para aprender, olvidar y filtrar MAC address.

# Virtual Private LAN Service (VPLS) – Operación:



## Virtual Private LAN Service (VPLS) – Operación:

---

- En el ejemplo anterior:
  - La red VPLS para el Customer A es una full mesh de pseudo wireless Ethernet.
  - A la instancia VPLS de cada cliente se le asigna un Virtual Circuit Identifier (VCI) único.
  - El VC emulado formado entre los routers PE consiste en LSP bidireccionales.
  - Las MAC address son aprendidas vía directed LDP label mappings entre los routers PE.

## Virtual Private LAN Service (VPLS) – Operación:

---

1. PE1 y PE2 tiene conectividad IGP y se pueden comunicar vía un túnel LSP.

PE1 distribuye un local label VC12 para su attached circuit y este label es propagado hacia PE2.

PE2 distribuye un local label VC21 y envía este label VC hacia PE1.

2. Un paquete desde CE1-A destinado a CE2-A requiere conocimiento de la mac address de CE2-A, MAC-B.

## Virtual Private LAN Service (VPLS) – Operación:

---

PE1 y PE2 no tienen la información en la ubicación de MAC-B (CE2-A) y MAC-A (CE1-A).

Por lo tanto:

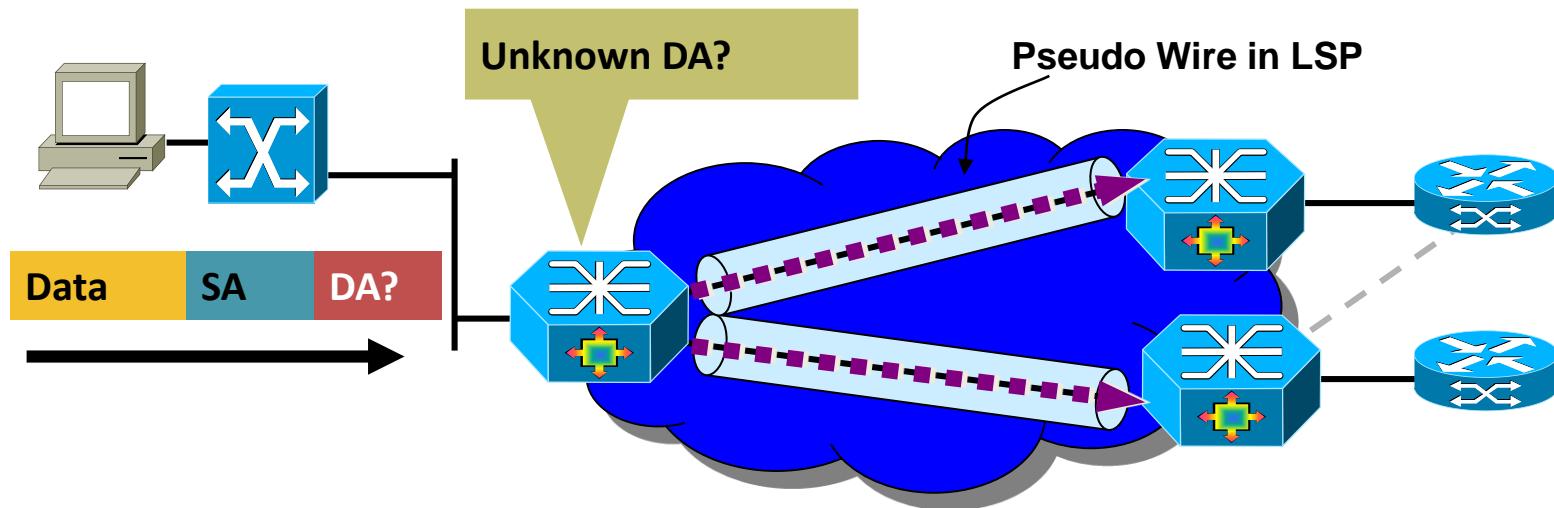
- ✓ Cuando los paquetes salen de CE1-A, la source MAC address es MAC-A.
- ✓ Debido a que CE1-A no tiene conocimiento de la CE2-A (MAC-B) un broadcast es enviado y renviado por PE1 hacia PE2 y PE3.
- ✓ PE1 envía un paquete de broadcast con source MAC address de CE1-A (MAC-A) a los otros peers PE2 y PE3 en el dominio VPLS.

## Virtual Private LAN Service (VPLS) – Operación:

---

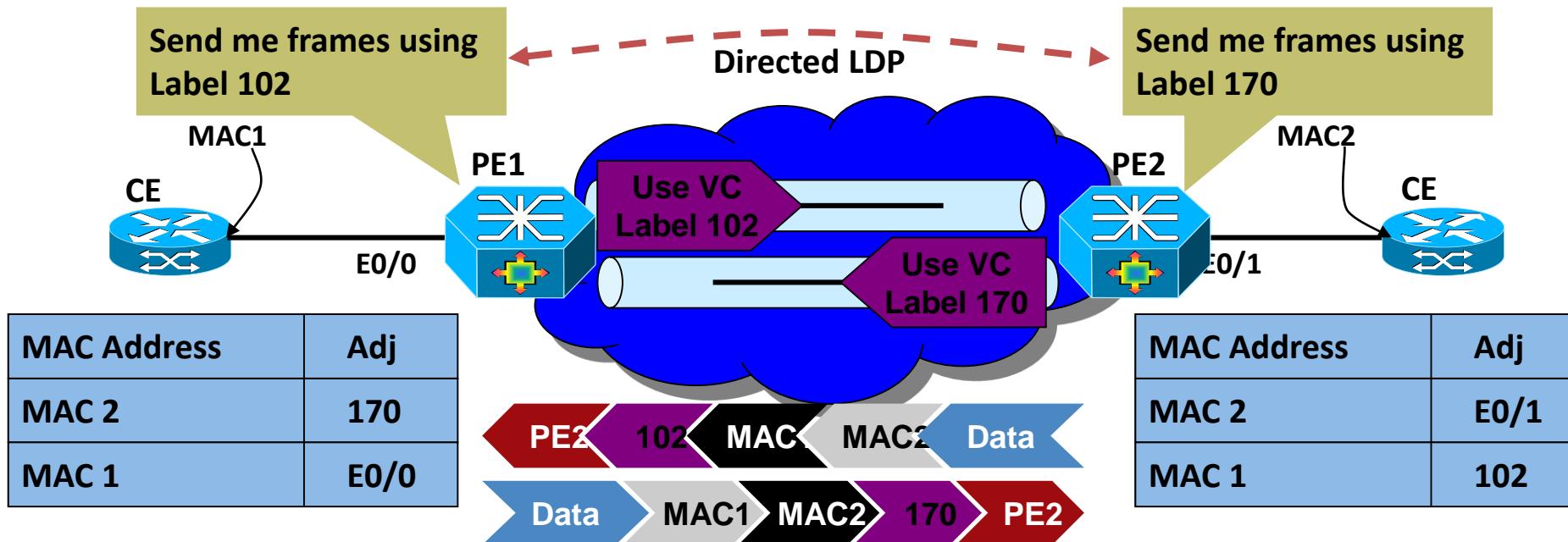
- ✓ Este paquete de broadcast es enviado con VC label VC21 a PE2 que fue aprendido desde PE2 durante la formación de las sesiones LDP dirigidas entre PE1 y PE2.
  - ✓ De la misma forma, este paquete de broadcast es también enviado con VC label VC31 a PE3.
3. PE2 recibe el paquete desde PE1 y asocia la source MAC address MAC-A con el inner label (VC label) VC21 y por lo tanto, concluye que la source mac address MAC-A esta detrás de la red PE1.
  4. Debido a que VC21 fue inicialmente asignada y propagada por PE2 a PE1 durante el establecimiento de la sesión LDP dirigida, PE2 ahora puede asociar la MAC-A con VC21

# VPLS Flooding & Forwarding



- Flooding (Broadcast, Multicast, Unknown Unicast)
- Dynamic learning of MAC addresses on PHY and VCs
- Forwarding
  - Physical Port
  - Virtual Circuit

## MAC Address Learning and Forwarding



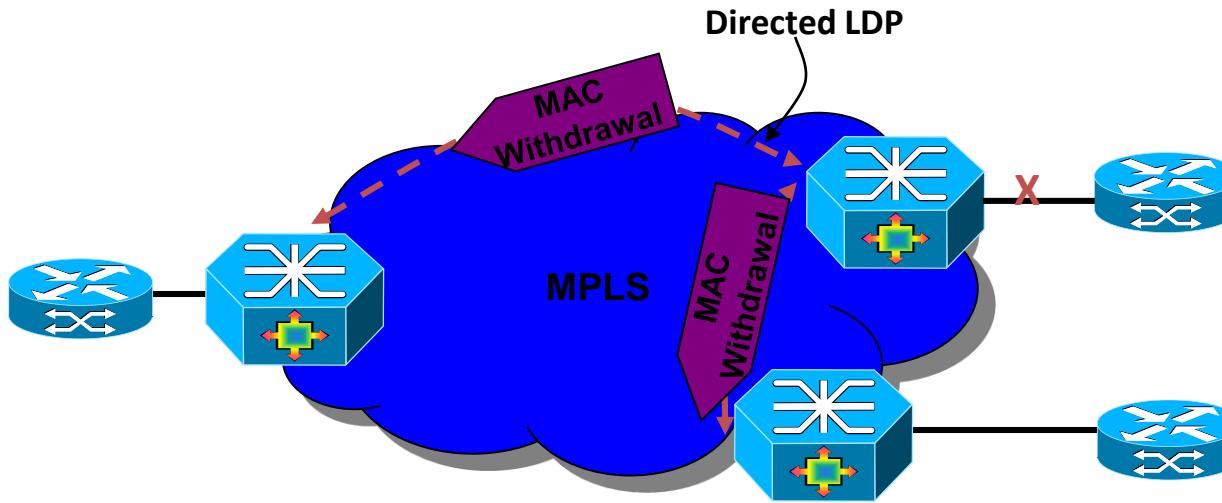
Broadcast, Multicast, and Unknown Unicast are learned via the received label associations

Two LSPs associated with a VC (Tx & Rx)

If inbound or outbound LSP is down

Then the entire Pseudo Wire is considered down

# MAC Address Withdrawal Message



Message speeds up convergence process

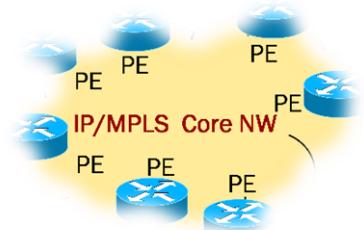
Otherwise PE relies on MAC Address Aging Timer

Upon failure PE removes locally learned MAC addresses

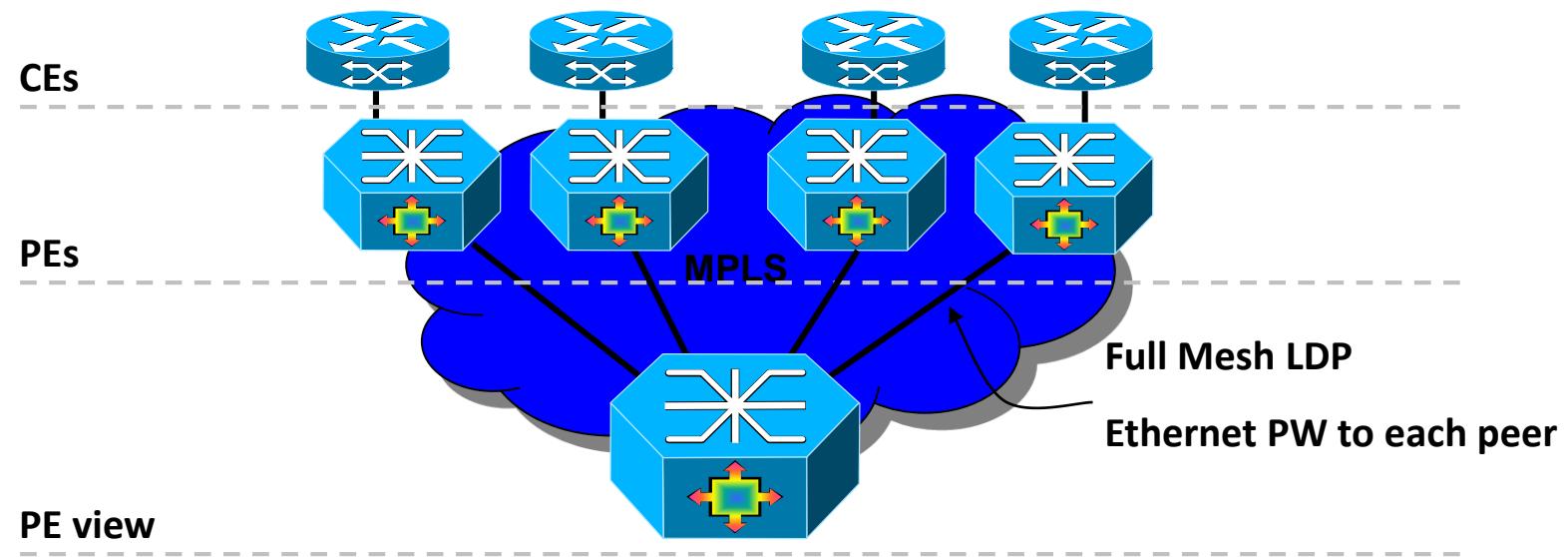
Send LDP Address Withdraw (RFC3036) to remote PEs in VPLS (using the Directed LDP session)

New MAC List TLV is used to withdraw addresses

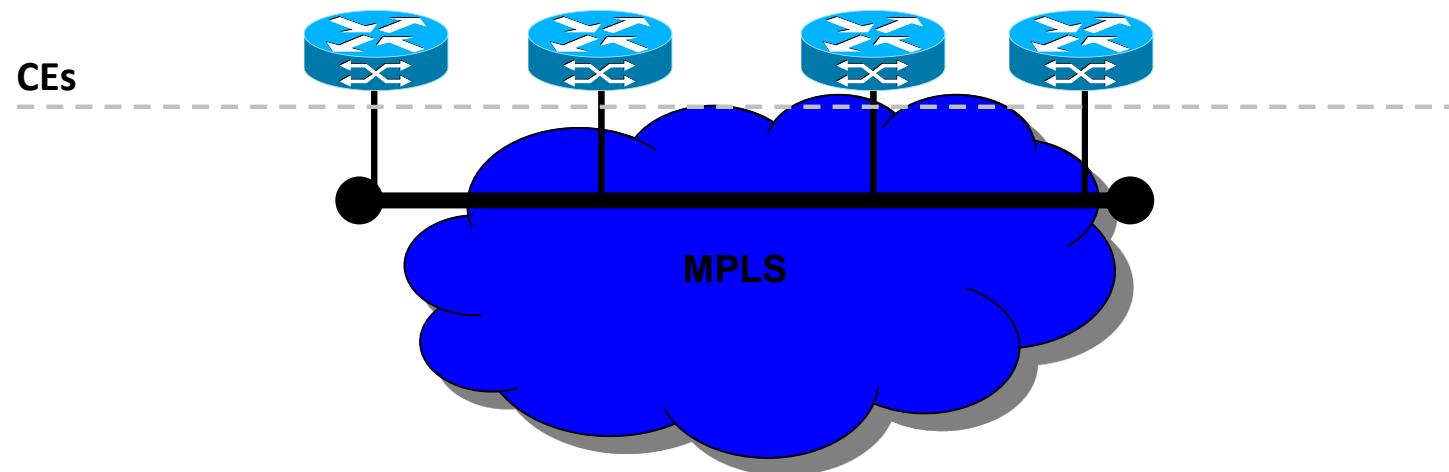
# VPLS: Topologías



# VPLS Topology – PE View



# VPLS Topology – CE View



VPLS define dos tipos de arquitectura:

## Direct Attachment (Flat)

Section 04 of Draft-ietf-l2vpn-vpls-ldp

## Hierarchical or H-VPLS

Section 10 of Draft-ietf-l2vpn-vpls-ldp

Comprising of two access methods

Ethernet Edge (EE-H-VPLS) – QinQ tunnels

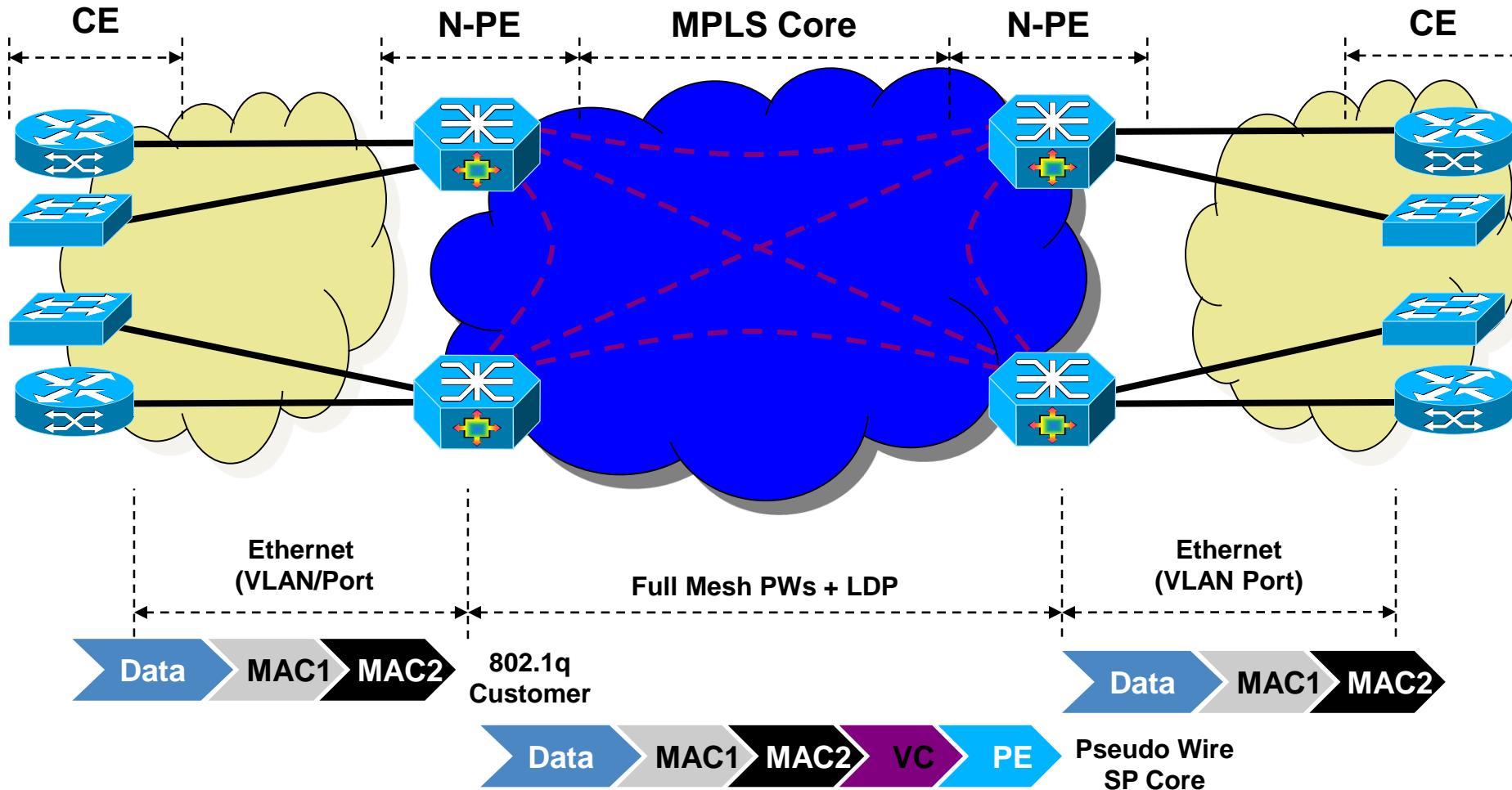
MPLS Edge (ME-H-VPLS) - PWE3 Pseudo Wires (EoMPLS)

# Directed attachment (Flat) Characteristics

- Suitable for simple/small implementations
- 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 (no QinQ)
- Potential signaling and packet replication overhead
  - ✓ Large amount of multicast replication over same physical
  - ✓ CPU overhead for replication

## VPLS Topology:

# Direct Attachment VPLS (Flat Architecture)



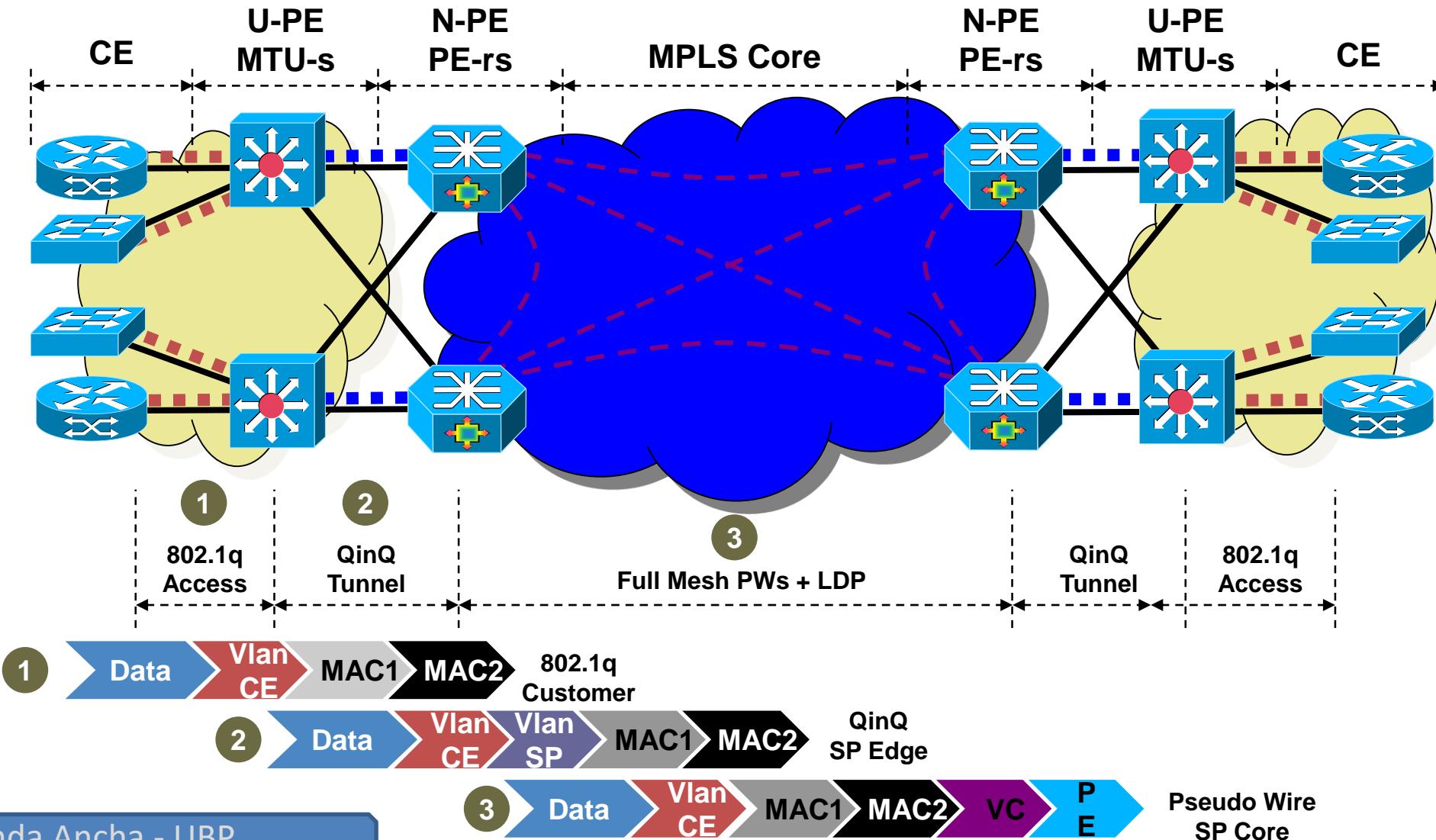
# Hierarchical VPLS (H-VPLS)

- 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)

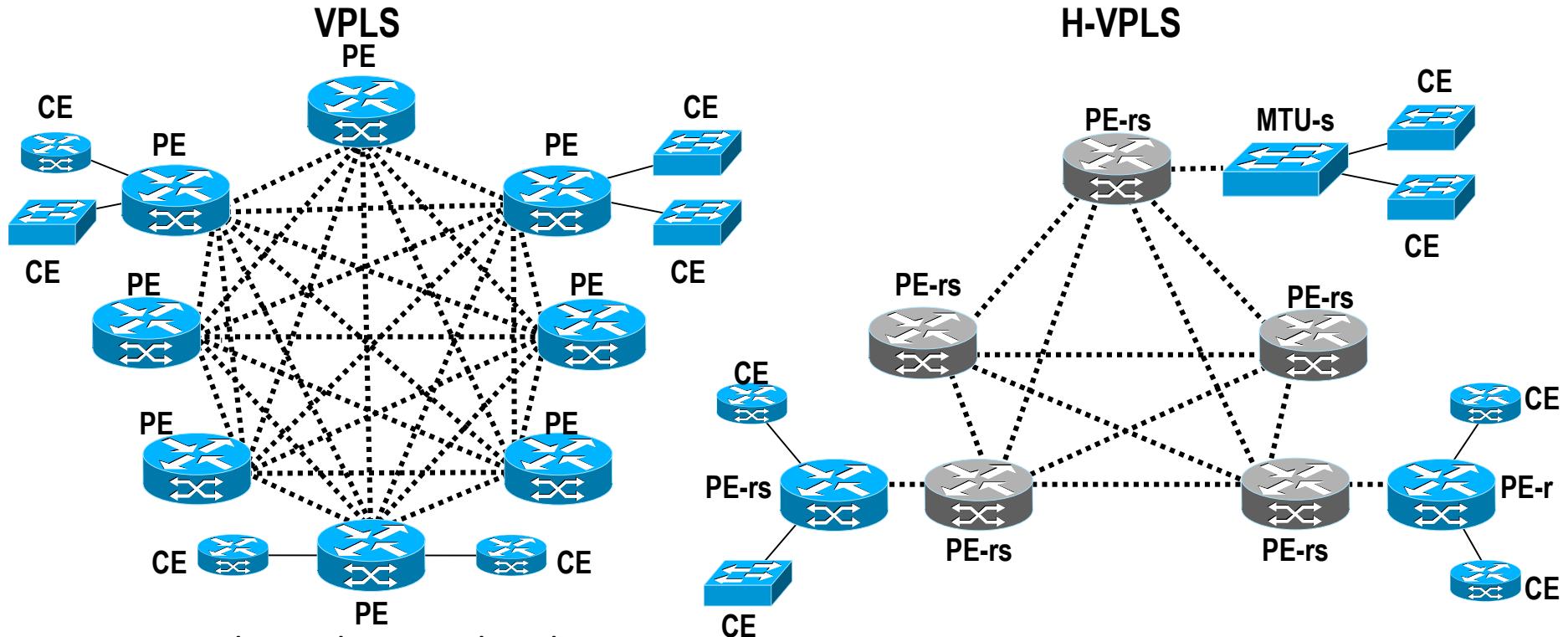
Some additional H-VPLS terms

<b>MTU-s</b>	Multi-Tenant Unit Switch capable of bridging (U-PE)
<b>PE-r</b>	Non bridging PE router
<b>PE-rs</b>	Bridging and Routing capable PE

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



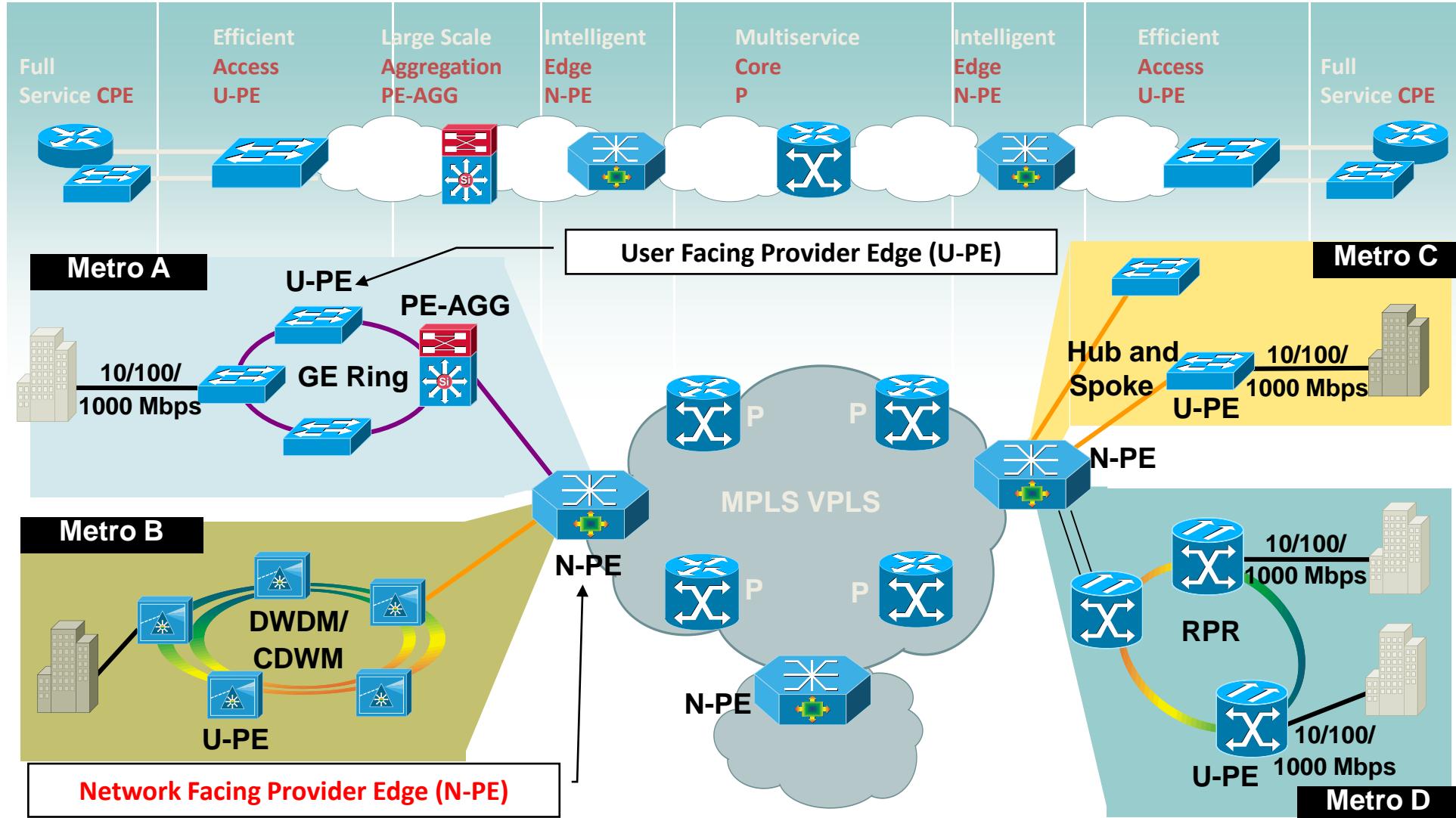
## Why H-VPLS?



- 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

# VPLS Topology:

## Ethernet Edge Topologies



## VPLS Topology—Single PE or Direct Attachment:

---

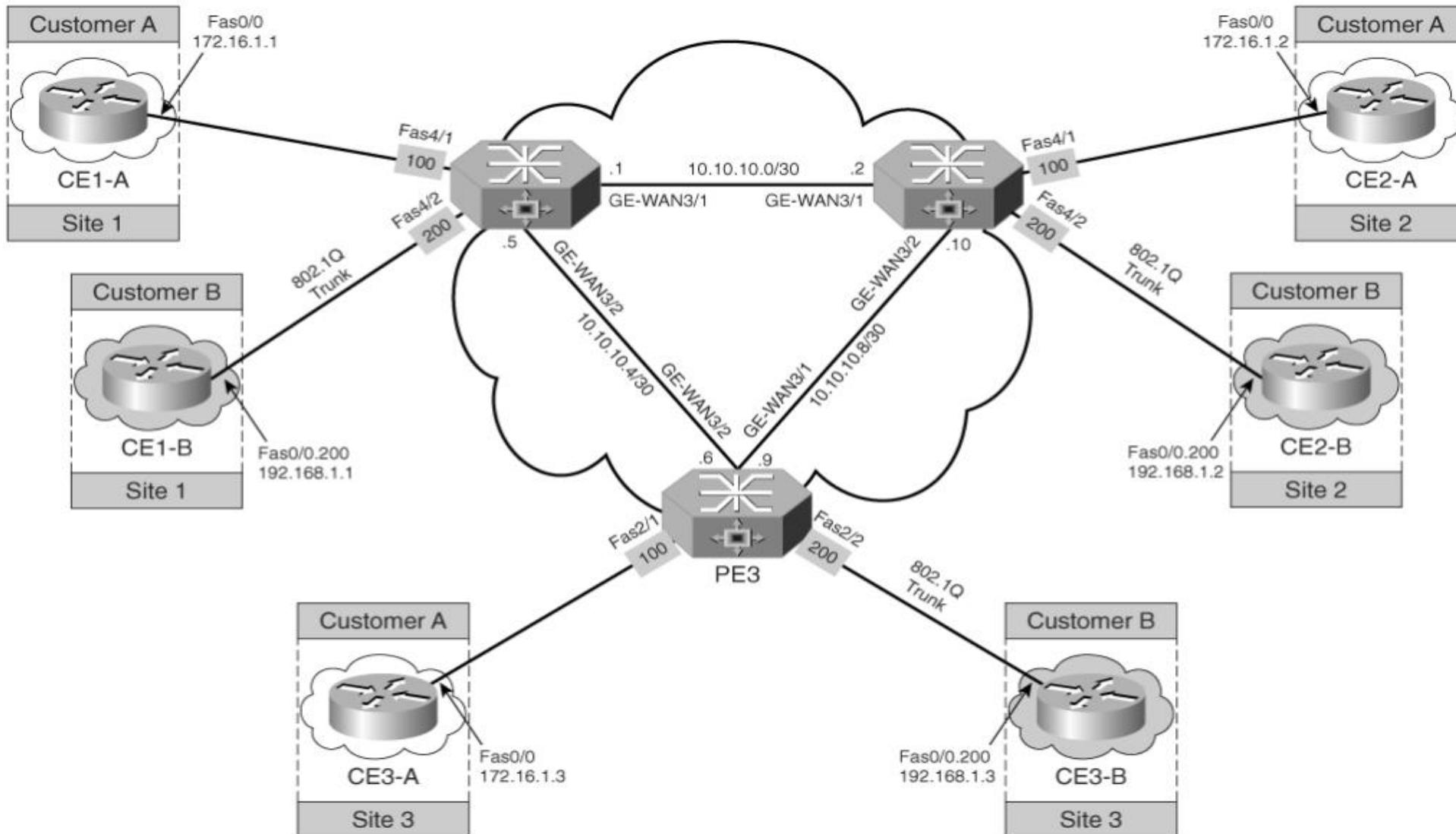
- ✓ Esta arquitectura utiliza un **esquema plano**
- ✓ Soporta los modos del tipo puertos **Ethernet, 802.1Q y dot1q.**
- ✓ Los CEs están directamente conectados a los routers PE.
- ✓ Implica la **creación de un VSI separado por cada cliente.**
- ✓ El tráfico originado por los clientes desde un CE en un frame Ethernet nativo o en un frame con VLAN tagged es **encapsulado con MPLS con el stack AToM.**

## VPLS Topology—Single PE or Direct Attachment:

---

- ✓ Las VPLS Direct Attachment además utilizan una **full mesh** de LDP dirigidos y túneles LSP entre todos los routers PE.
- ✓ Aunque esto crea **sobrecarga de señalización**, el verdadero perjuicio para el despliegue a gran escala es los requisitos de **replicación de paquetes** para cada VC aprovisionado en un router PE.
- ✓ Debido a estos problemas de escala, **esta solución es únicamente recomendable para implementaciones simples.**

# VPLS Topology—Single PE or Direct Attachment:

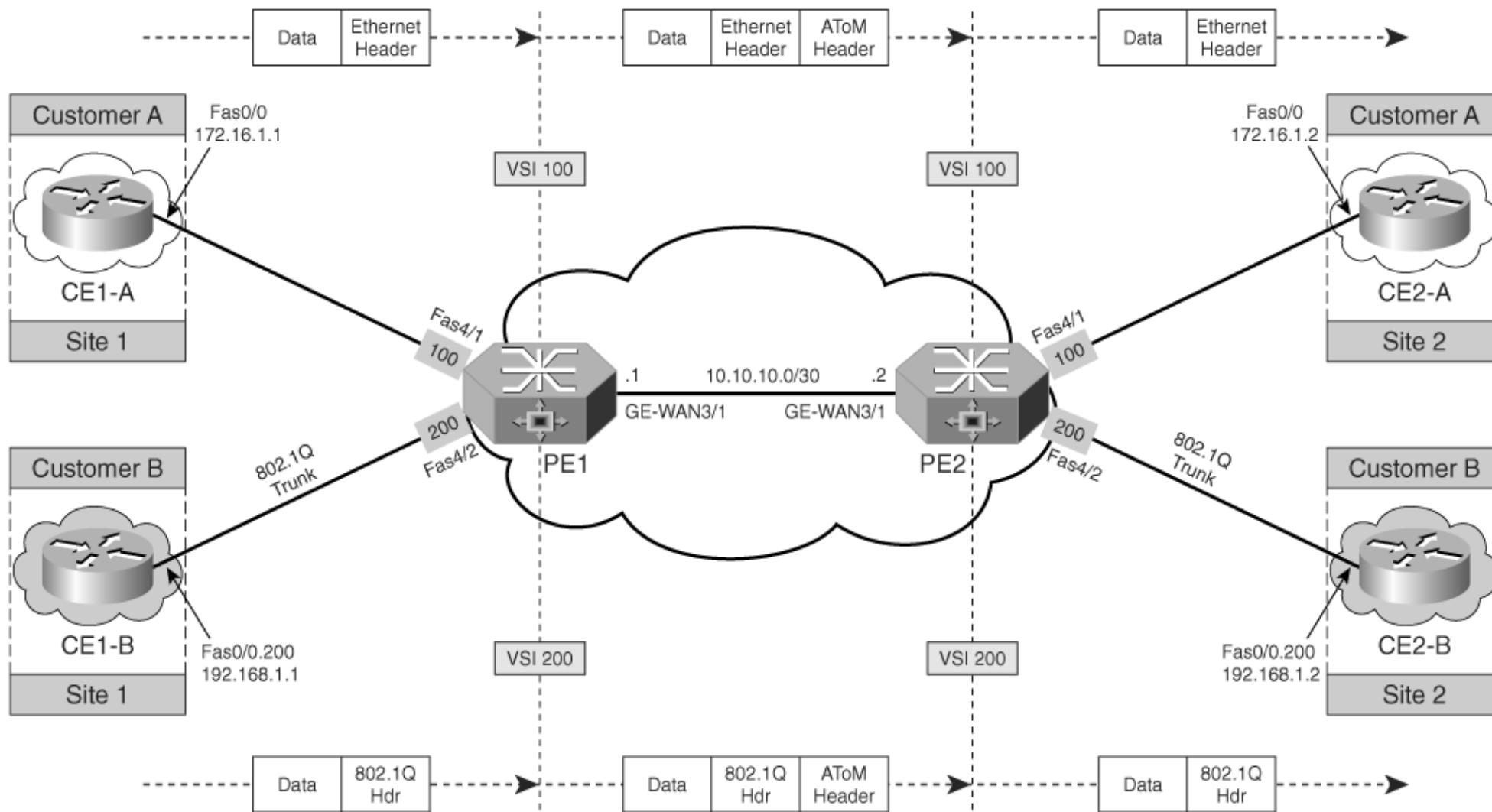


## VPLS Topology—Single PE or Direct Attachment:

---

- ✓ La red del cliente A tiene los routers CE, CE1-A, CE2-A y CE3-A conectados a los routers PE, PE1, PE2 y PE3 respectivamente.
- ✓ Los links a CE1-A, CE2-A y CE3-A están configurados como switch Access port son PE1, PE2 y PE3.
- ✓ La red del cliente B tiene los routers CE, CE1-B, CE2-B y CE3-B que están configurados como puertos 802.1Q trunks en los routers PE1, PE2 y PE3 respectivamente.

# VPLS Topology—Single PE or Direct Attachment:



## VPLS Topology—Single PE or Direct Attachment:

---

- ✓ La red del cliente A tiene los routers CE, CE1-A, CE2-A y CE3-A conectados a los routers PE, PE1, PE2 y PE3 respectivamente.
- ✓ Los links a CE1-A, CE2-A y CE3-A están configurados como switch Access port son PE1, PE2 y PE3.
- ✓ La red del cliente B tiene los routers CE, CE1-B, CE2-B y CE3-B que están configurados como puertos 802.1Q trunks en los routers PE1, PE2 y PE3 respectivamente.

## VPLS Service Configuration Flowchart on PE Router:

---

- ✓ En este ejemplo la red VPLS del Cliente A utiliza el port mode y la red VPLS del Cliente B utiliza el 802.1Q vlan mode.
- ✓ Step 1:
  - ✓ Configuramos la interfaz del router PE conectada al dispositivo CE del Cliente A como Access port (untagged) y la interfaz del router PE conectada al dispositivo CE del Cliente B como 802.1Q trunk port (tagged)

# VPLS Service Configuration Flowchart on PE Router:

---

```
PE1(config)#vlan 100
PE1(config-vlan)#state active
PE1(config-vlan)#vlan 200
PE1(config-vlan)#state active
PE1(config)#interface fastEthernet 4/1
PE1(config-if)#description VPLS Customer A (CE1-A)
PE1(config-if)#switchport
PE1(config-if)#switchport access vlan 100
PE1(config-if)#switchport mode access
PE1(config-if)#interface FastEthernet4/2
PE1(config-if)#description VPLS Customer B (CE1-B)
PE1(config-if)#switchport
PE1(config-if)#switchport trunk encapsulation dot1q
PE1(config-if)#switchport trunk allowed vlan 200
PE1(config-if)#switchport mode trunk
```

## VPLS Service Configuration Flowchart on PE Router:

---

- ✓ Step 2.
  - ✓ Definir el VFI y asociarlo a la interfaz conectado con el CE.
  - ✓ En este paso, el VFI es configurado y luego de definirlo, este es asociado a uno o mas attachment circuits (interfaces, subinterfaces or virtual circuits).
- ✓ El VFI especifica el VPN ID de un dominio VPLS, la dirección de otros routers PE en este dominio y el tipo de señalización y encapsulamiento para el túnel para cada peer.

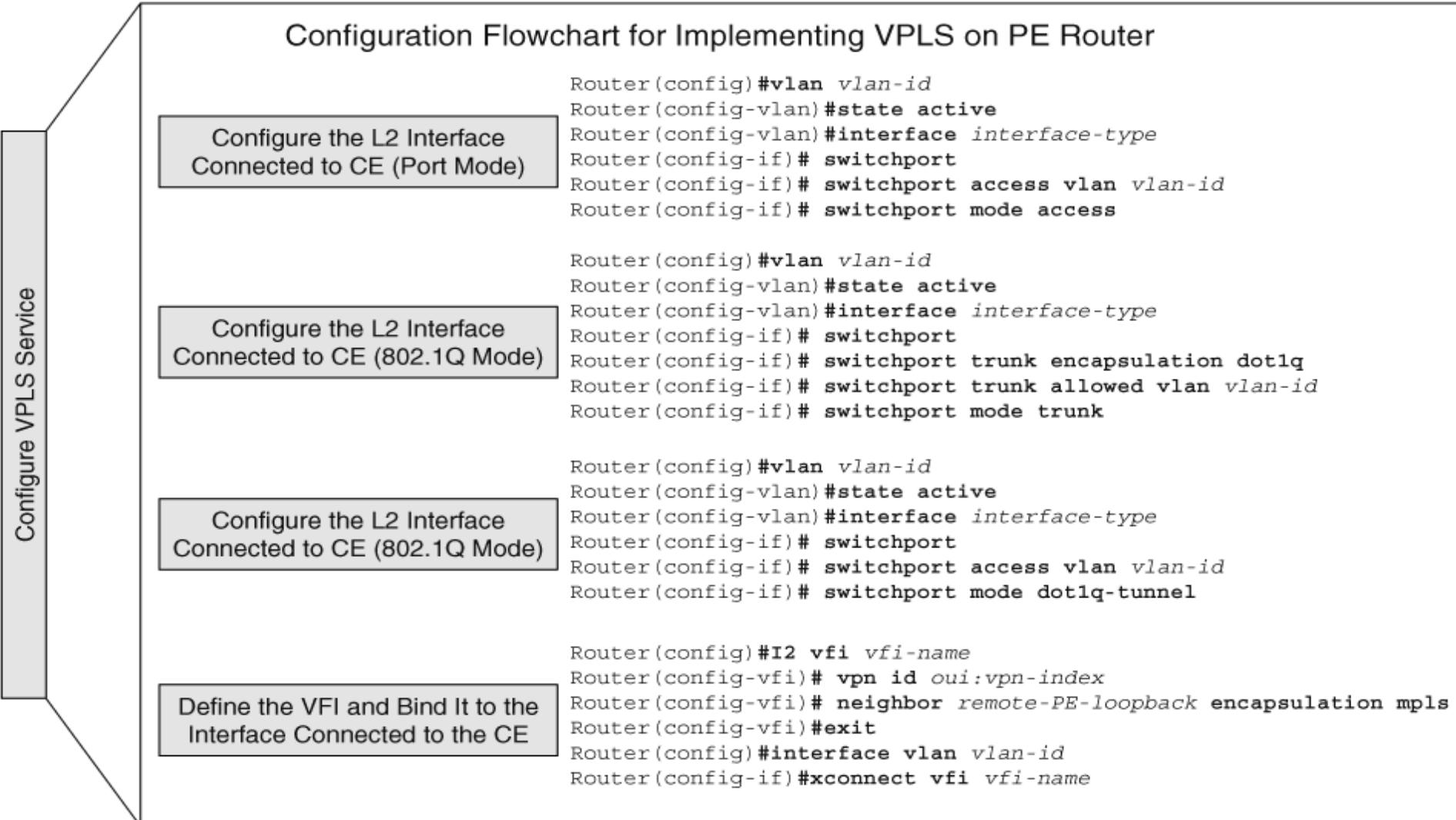
## VPLS Service Configuration Flowchart on PE Router:

---

- ✓ El MPLS VPN ID es utilizado para identificar las VPNs con un VPN identification number, como describe la RFC 2685.
- ✓ Múltiples VPNS se pueden configurar en un routers y podemos utilizar el VPN ID para identificar una en particular.

# VPLS Service Configuration Flowchart on PE Router:

Provision the Provider Network for MPLS Forwarding



# VPLS Service Configuration Flowchart on PE Router:

---

```
PE1(config)#12 vfi Cust_A manual  
PE1(config-vfi)# vpn id 100  
PE1(config-vfi)# neighbor 10.10.10.102 encapsulation mpls  
PE1(config-vfi)# neighbor 10.10.10.103 encapsulation mpls  
  
PE1(config)#12 vfi Cust_B manual  
PE1(config-vfi)#vpn id 200  
PE1(config-vfi)#neighbor 10.10.10.102 encapsulation mpls  
PE1(config-vfi)#neighbor 10.10.10.103 encapsulation mpls  
  
PE1(config)#interface vlan 100  
PE1(config-if)#xconnect vfi Cust_A  
PE1(config-if)#interface vlan 200  
PE1(config-if)#xconnect vfi Cust_B
```

## VPLS Topology—Hierarchical VPLS—Distributed PE Architecture:

---

- ✓ Con el servicio de VPLS Jerárquico (H-VPLS), los routers PE no se encuentran directamente contactos al equipamiento del cliente.
- ✓ Este modelo jerárquico *introduce una nueva capa en la capa de acceso* hacia el equipamiento de cliente.
- ✓ Esto se denomina PE distribuidos.
- ✓ Es por esto, que en esta arquitectura, tenemos dos o mas tipos de dispositivos PE.

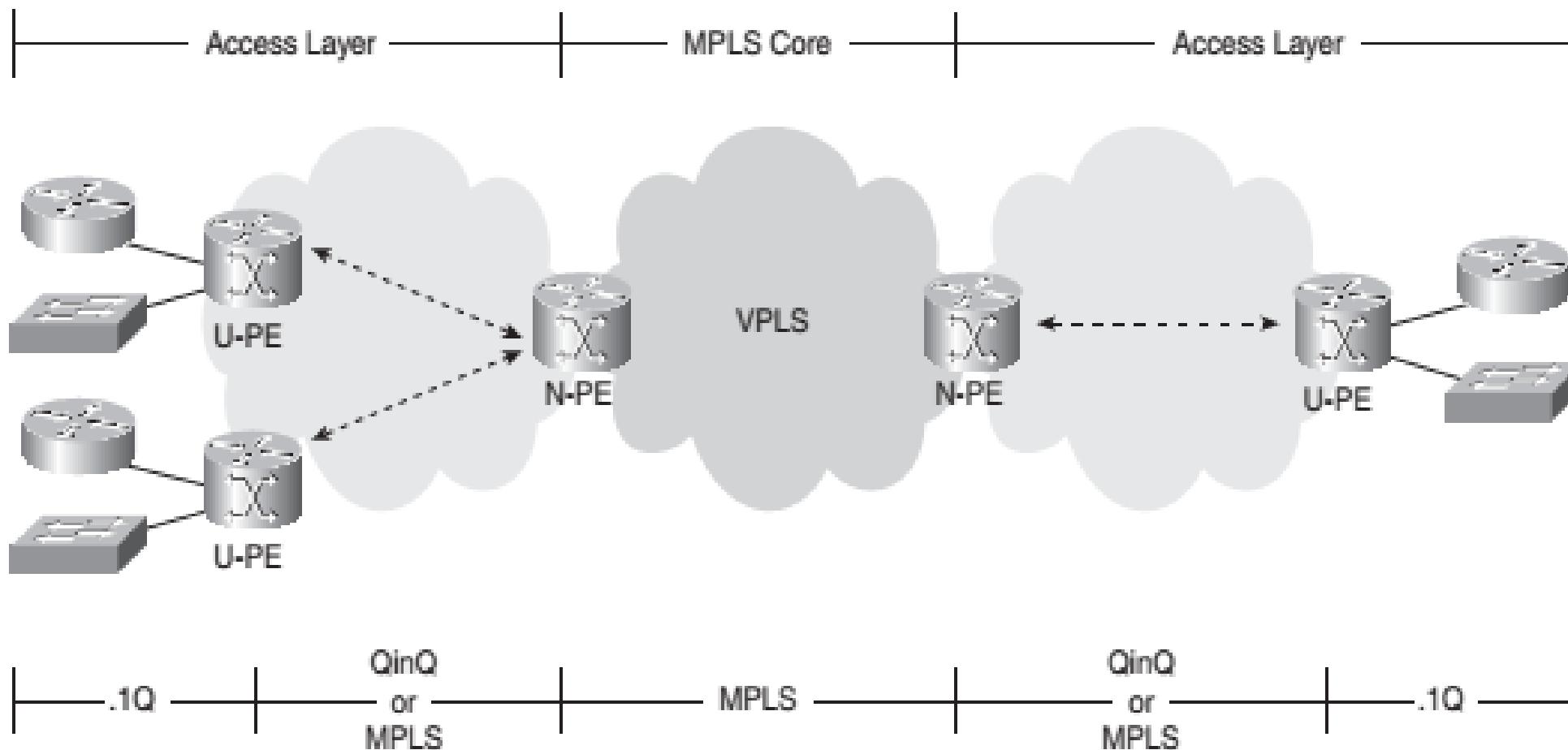
## VPLS Topology—Hierarchical VPLS—Distributed PE Architecture:

---

- ✓ Estos se denominan **routers N-PE y U-PE**.
- ✓ Los N-PE son routers PE que “apuntan hacia la red”
- ✓ Los U-PE son routers que “apuntan” hacia el usuario.

Este modelo jerárquico provee como beneficio *menor señalización en la red del core* y *menor replicación de paquetes en los routers N-PE*,

# VPLS Topology—Hierarchical VPLS—Distributed PE Architecture:



## VPLS Topology—Hierarchical VPLS—Distributed PE Architecture:

---

### ✓ User facing PE (u-PE):

- ✓ Los dispositivos CE se conectan directamente con el u-PE.
- ✓ Un u-PE típicamente tiene una única conexión al dispositivo de red PE (n-PE) ubicado en el backbone MPLS.
- ✓ El forwarding VPLS es realizado basado en los VSI ( MAC address learning y switching)

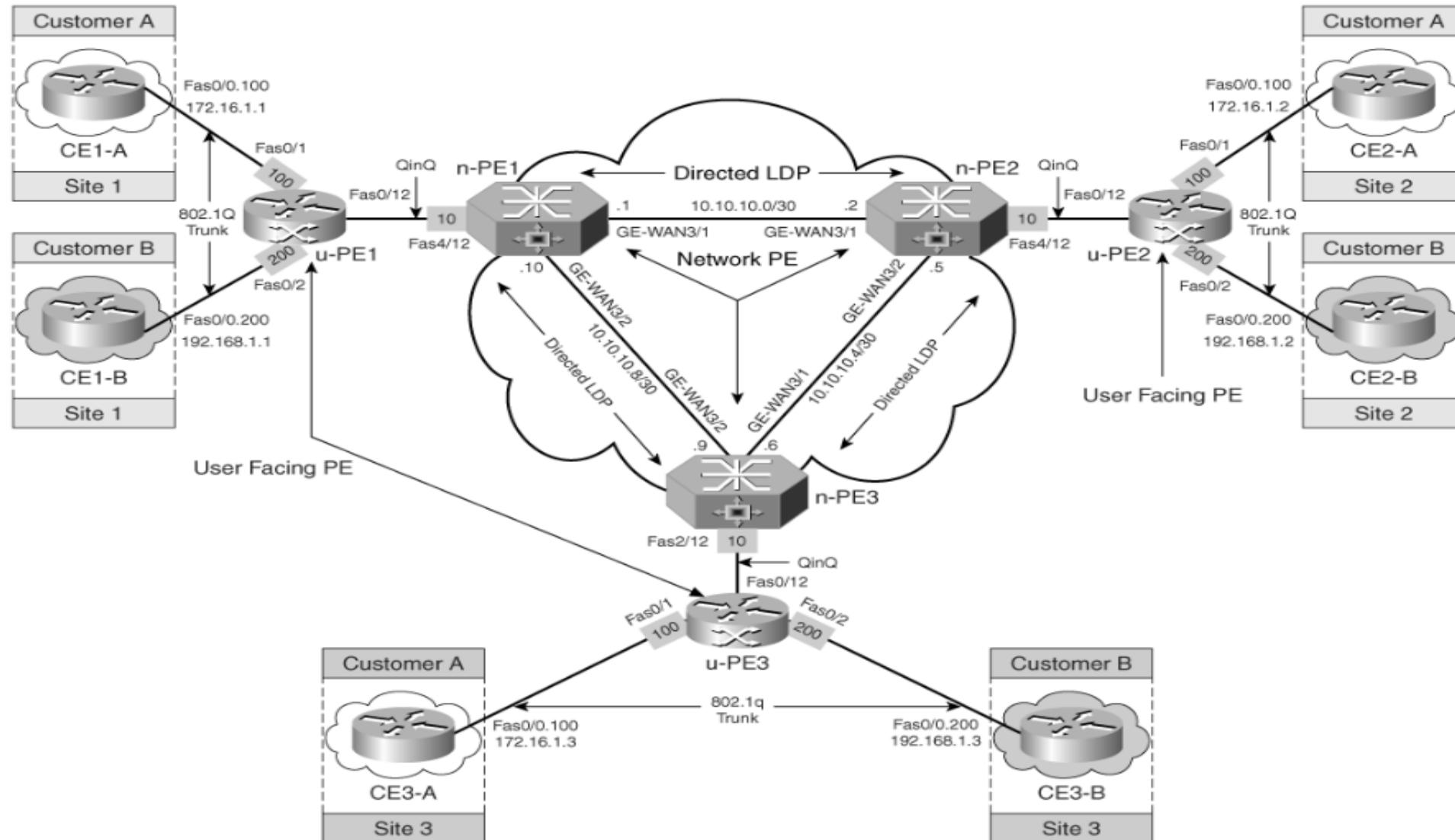
## VPLS Topology—Hierarchical VPLS—Distributed PE Architecture:

---

### ✓ Network PE (n-PE):

- ✓ Los u-PE se conectan contra los n-PE donde la información VPLS es forwardeado basada en los VSI.
- ✓ La forma de implementación mas común utiliza encapsulamiento y tunneling 802.1Q. (Q-in-Q)
- ✓ Para cada servicio VPLS, un single-spoke pseudo wire es establecido entre el uPE y nPE.
- ✓ Este pseudo wire es terminado en una instancia de virtual bridge en el u-PE

# VPLS Topology—Hierarchical VPLS—Distributed PE Architecture:

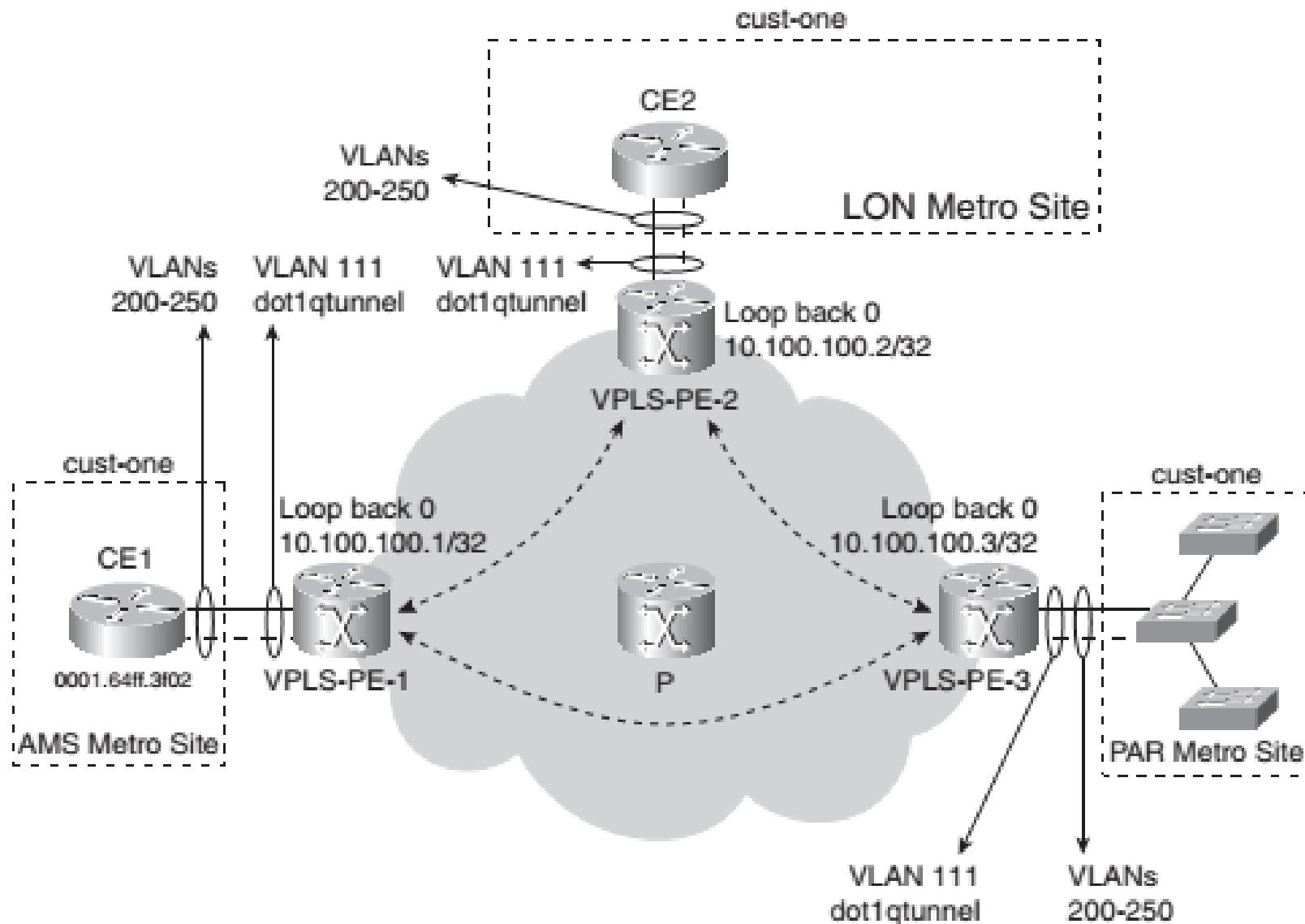


## VPLS Topology—Hierarchical VPLS—Distributed PE Architecture:

---

- ✓ Q-in-Q es posible en VPLS.
- ✓ Esto implica que:
  - ✓ La VLAN del cliente es encapsulada dentro de otra VLAN (P-VLAN o provider vlan)
  - ✓ Permitiendo una conmutación multi-vlan de clientes transportado en forma transparente entre múltiples sitios interconectados en la red MPLS.
- ✓ Esta P-VLAN es mapeada en un VFI en el router N-PE.

# VPLS Topology—Hierarchical VPLS—Distributed PE Architecture:



## VPLS Topology—Hierarchical VPLS—Distributed PE Architecture:

---

- ✓ Los **spoke pseudo wires** son encapsulados con Q-in-Q
- ✓ Esto *permite separación entre clientes* manteniendo en forma intacta la información de VLAN específica de los clientes.
- ✓ Los Clientes A y Clientes B tiene dispositivos CE ubicados en sitios diferentes que son conectados a la VPLS utilizando túneles de accesos Q-in-Q.
- ✓ Cada cliente tiene su propia separación interna de VLANs. Los Clientes A pertenecen a la vlan 100 y los Clientes B a la 200.

## VPLS Topology—Hierarchical VPLS—Distributed PE Architecture:

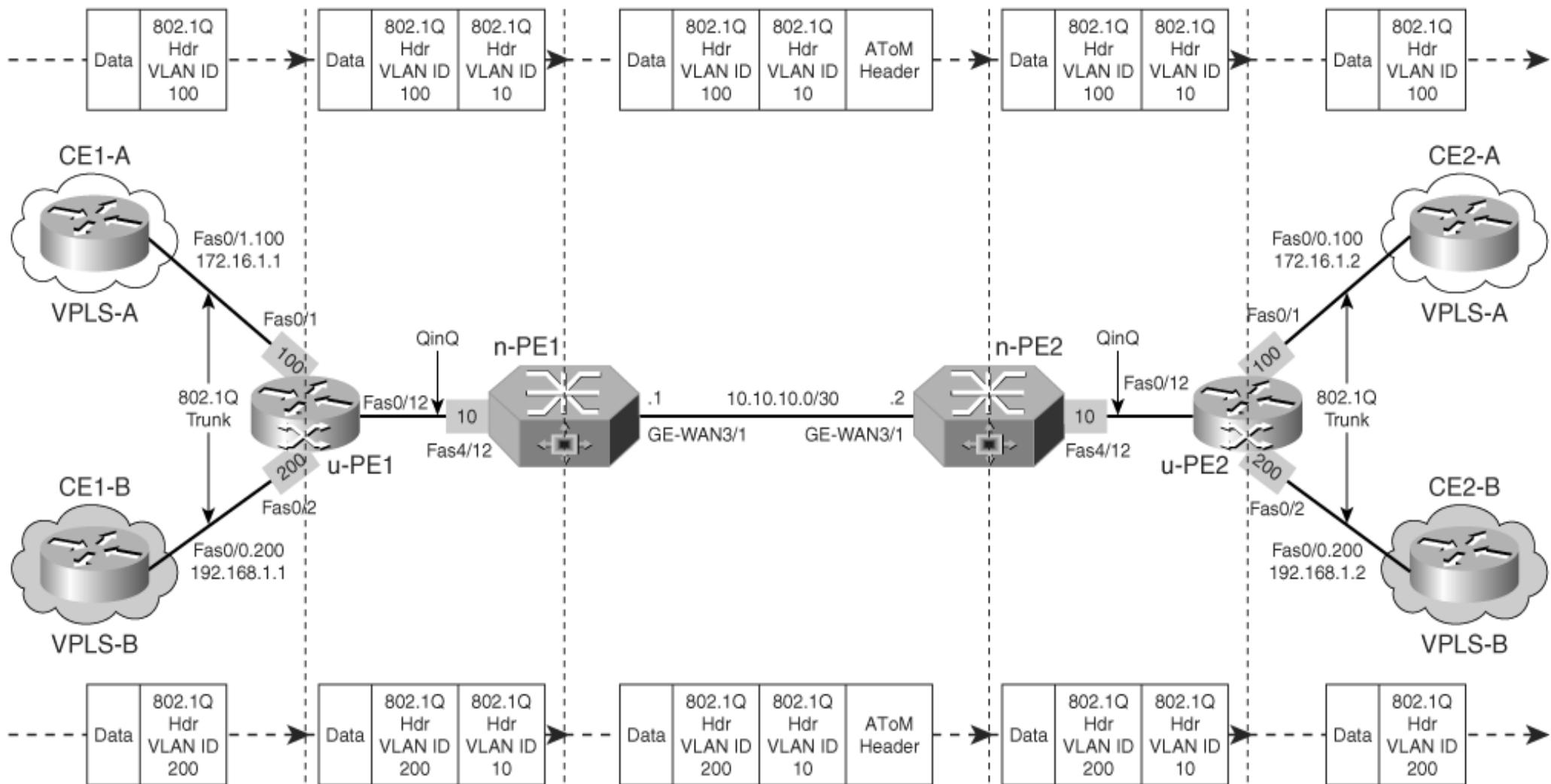
---

- ✓ El objetivo es **asegurar conectividad vlan-to-vlan entre diferentes sitios pertenecientes a los clientes A y a los clientes B**
- ✓ El tráfico es tageado por los switches de los clientes, con el vlan tag apropiado y enviado hacia el SP (vlan tag 100 para el Cliente A y vlan tag 200 para el Cliente B).
- ✓ Los u-PE agregan un vlan tag adicional (Vlan tag 10) para el tráfico originado por cada dispositivo CE.
- ✓ Este tráfico es entonces enviado por el u-PE hacia el network PE, donde este es procesado de acuerdo al VSI de cada cliente.

## VPLS Topology—Hierarchical VPLS—Distributed PE Architecture:

- ✓ Outer Vlan tags 100 y 200 son reemplazados por el AToM label stack (LSP label, VC label) y es enviado a través de la red de backbone MPLS.
- ✓ Las implementaciones H-vPLS entonces:
  - ✓ Eliminan la necesidad de una red full mesh de túneles
  - ✓ Eliminan el full mesh de pseudo wires por servicios entre todos los dispositivos participantes en la implementación VPLS.
  - ✓ Esta minimiza la duplicación de paquetes y el overhead de señalización ya que una menor cantidad de pseudo wires son requeridos para brindar el servicio.

# VPLS Topology—Hierarchical VPLS—Distributed PE Architecture:



# Configuration Flowchart for H-VPLS Using Q-in-Q Mode:

---

## ✓ Step 1:

### ✓ Configurar la interface conectada al u-PE para 802.1Q

```
PE1(config)#vlan 10  
PE1(config-vlan)#state active  
PE1(config-vlan)#interface FastEthernet4/12  
PE1(config-if)# description link to u-PE1  
PE1(config-if)# switchport  
PE1(config-if)# switchport access vlan 10  
PE1(config-if)# switchport mode dot1q-tunnel
```

# Configuration Flowchart for H-VPLS Using Q-in-Q Mode:

---

- ✓ PE2(config)#vlan 10
- ✓ PE2(config-vlan)#state active
- ✓ PE2(config-if)#interface FastEthernet4/12
- ✓ PE2(config-if)# description link to u-PE2
- ✓ PE2(config-if)# switchport
- ✓ PE2(config-if)# switchport access vlan 10
- ✓ PE2(config-if)# switchport mode dot1q-tunnel
- ✓ PE3(config)#vlan 100
- ✓ PE3(config-vlan)#state active
- ✓ PE3(config-if)#interface FastEthernet4/12
- ✓ PE3(config-if)# description link to u-PE3
- ✓ PE3(config-if)# switchport
- ✓ PE3(config-if)# switchport access vlan 10
- ✓ PE3(config-if)# switchport mode dot1q-tunnel

## Configuration Flowchart for H-VPLS Using Q-in-Q Mode:

---

### ✓ Step 2:

- ✓ Definir el VFI y asociarlo con la interfaz conectada al CE.
- ✓ El VFI en el n-PE especifica el VPN ID de un dominio VPLS, la dirección de cada router PE en este dominio y el tipo de señalización y encapsulamiento para cada peer.

# Configuration Flowchart for H-VPLS Using Q-in-Q Mode:

```
n-PE1(config-vfi)#12 vfi QinQ manual  
n-PE1(config-vfi)# vpn id 10  
n-PE1(config-vfi)# neighbor 10.10.10.101 encapsulation mpls  
n-PE1(config-vfi)# neighbor 10.10.10.103 encapsulation mpls  
n-PE1(config-vlan)#interface vlan 10  
n-PE1(config-if)#xconnect vfi QinQ
```

---

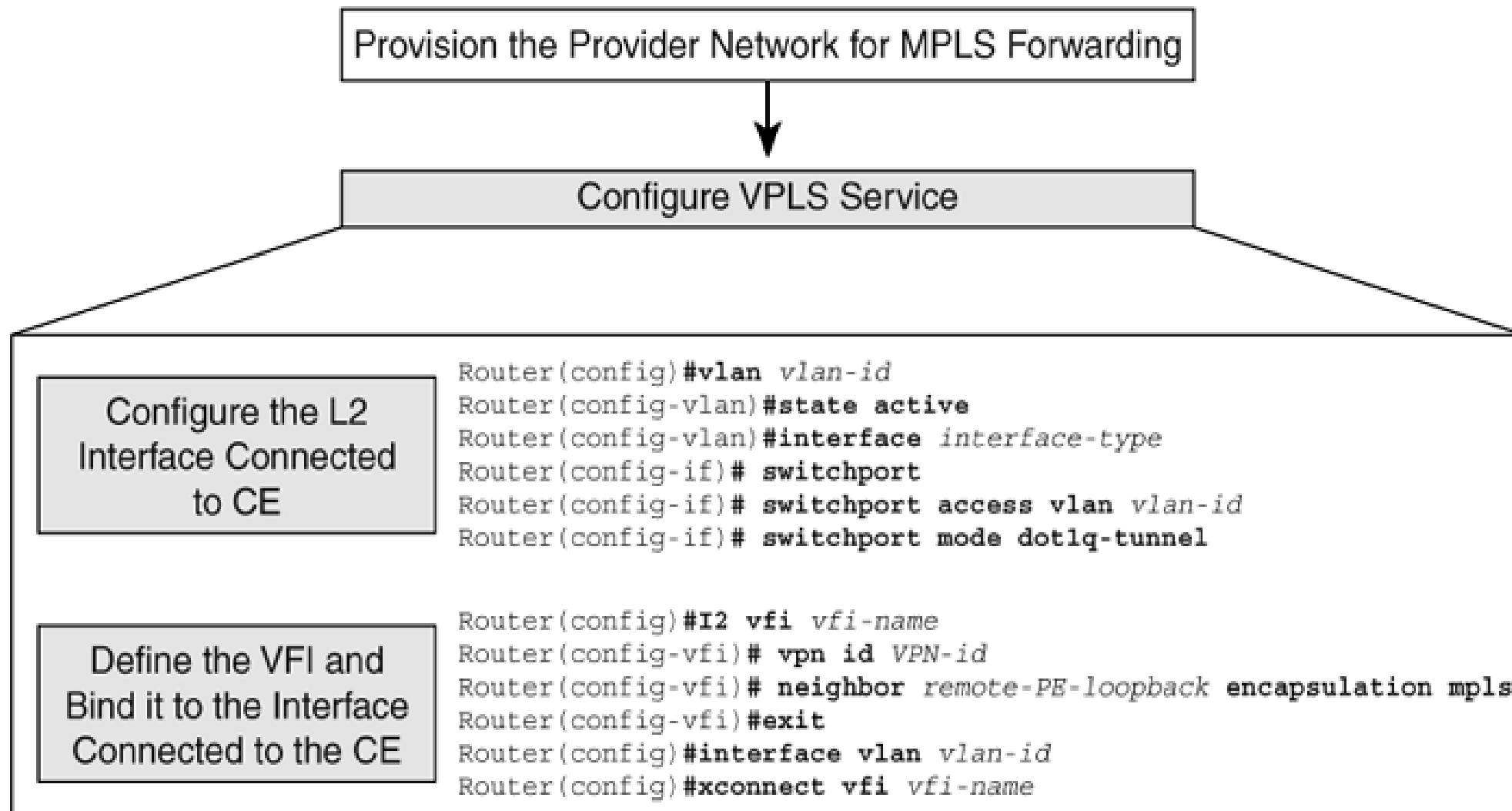
```
n-PE2(config-vfi)#12 vfi QinQ manual  
n-PE2(config-vfi)# vpn id 10  
n-PE2(config-vfi)# neighbor 10.10.10.101 encapsulation mpls  
n-PE2(config-vfi)# neighbor 10.10.10.103 encapsulation mpls  
n-PE2(config-vlan)#interface vlan 10  
n-PE2(config-if)#xconnect vfi QinQ
```

# Configuration Flowchart for H-VPLS Using Q-in-Q Mode:

---

```
n-PE3(config)#12 vfi QinQ manual  
n-PE3(config-vfi)# vpn id 10  
n-PE3(config-vfi)# neighbor 10.10.10.101 encapsulation mpls  
n-PE3(config-vfi)# neighbor 10.10.10.102 encapsulation mpls  
n-PE3(config-vlan)#interface vlan 10  
n-PE3(config-if)#xconnect vfi QinQ
```

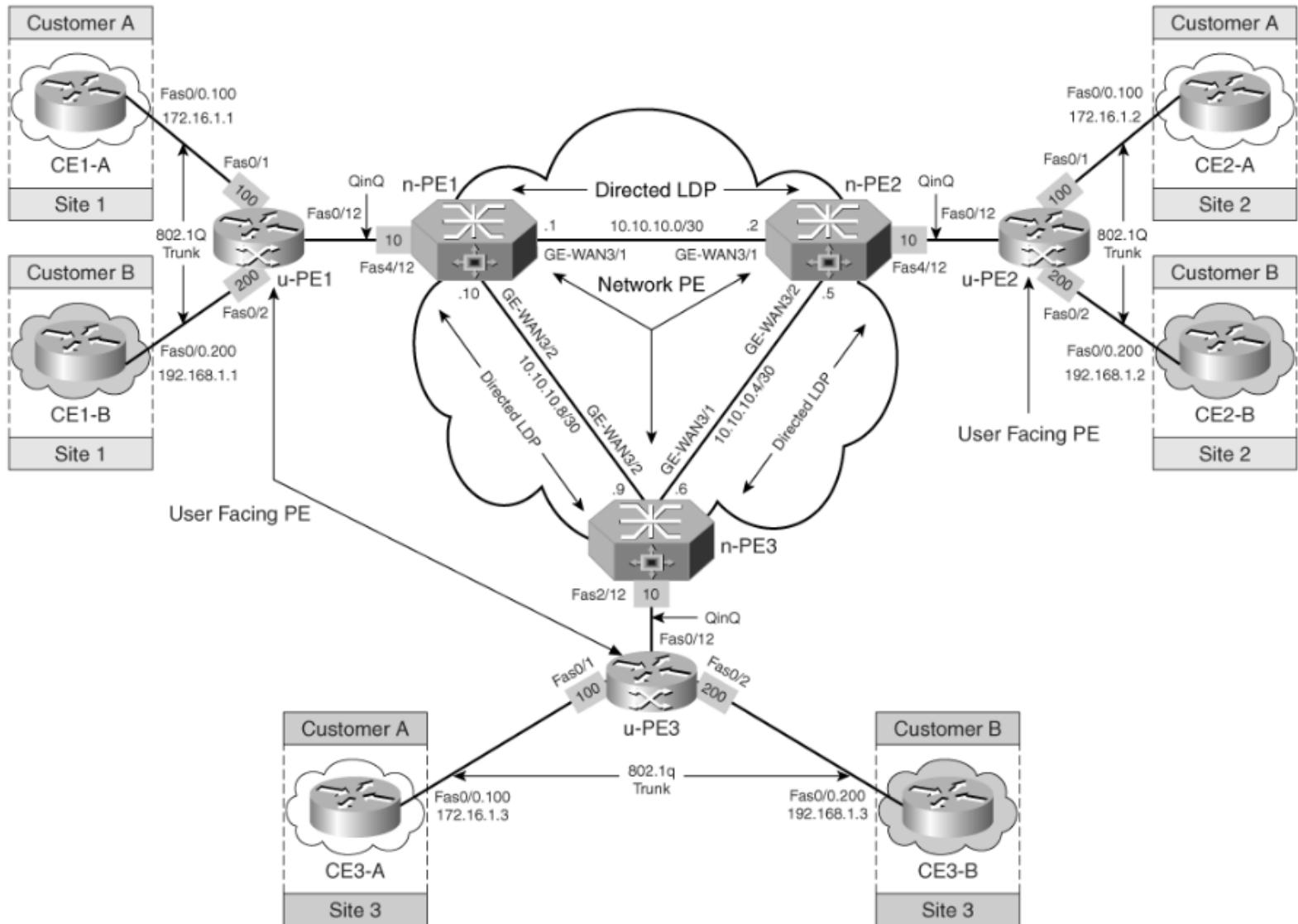
# Configuration Flowchart for H-VPLS Using Q-in-Q Mode:



# Configuration Flowchart for H-VPLS Using Q-in-Q Mode:

Command	Description
Router(config)#l2 vfi vfi-name manual	Enables the L2 VFI manual configuration mode.
Router(config-vfi)#vpn id oui:vpn-index	Configures a VPN ID for a VPLS domain. The emulated VCs bound to this L2 VRF use this VPN ID for signaling.
Router(config-vfi)#neighbor remote-router-id encapsulation mpls	Specifies the remote peering router ID and the tunnel encapsulation type or the pseudo-wire property to be used to set up the emulated VC.
Router(config-vfi)#xconnect vfi vfi name	Specifies the L2 VFI that you are binding.
Router(config-vfi)#shutdown	Disconnects all emulated VCs previously established under the L2VFI and prevents the establishment of new attachment circuits.
Router(config)#mpls label protocol {ldp   tdp}	Specifies the default label distribution protocol for a platform.
Router(config)#mpls ldp logging neighbor-changes (optional)	Determines logging neighbor changes.
Router(config)#mpls ldp discovery {hello   targeted-hello} {holdtime   interval} seconds	Configures the interval between transmission of LDP (TDP) discovery hello messages or the hold time for a LDP transport connection.
Router(config)#mpls ldp router-id loopback0 force	Assigns the loopback as the source of TDP/LDP messages.
Router(config)#mpls ip	Enables label switching of IPv4 packets on an interface.
Router(config-if)#mls qos trust [cos   dscp   ip-precedence]	Sets the trusted state of an interface to specify that the ToS bits in the incoming packets contain a DSCP value.
Router(config-if)#switchport	Modifies the switching characteristics of the Layer 2-switched interface.
Router(config-if)#switchport trunk encapsulation dot1q	Sets the switch port encapsulation format to 802.1Q. Modifies the switching characteristics of the Layer 2-switched interface.
Router(config-if)#switchport mode trunk switchport trunk allowed vlan	Sets the interface to a trunking VLAN Layer 2 interface. Sets the list of allowed VLANs.
Router(config-if)#switchport access vlan vlan-id	Sets the VLAN when the interface is in access mode.
Router(config-if)#switchport mode [access  trunk  dot1q-tunnel]	Sets the interface as an 802.1Q tunnel port.
Router(config-if)#l2protocol-tunnel [cdp   stp   vtp]	Enables protocol tunneling on an interface.

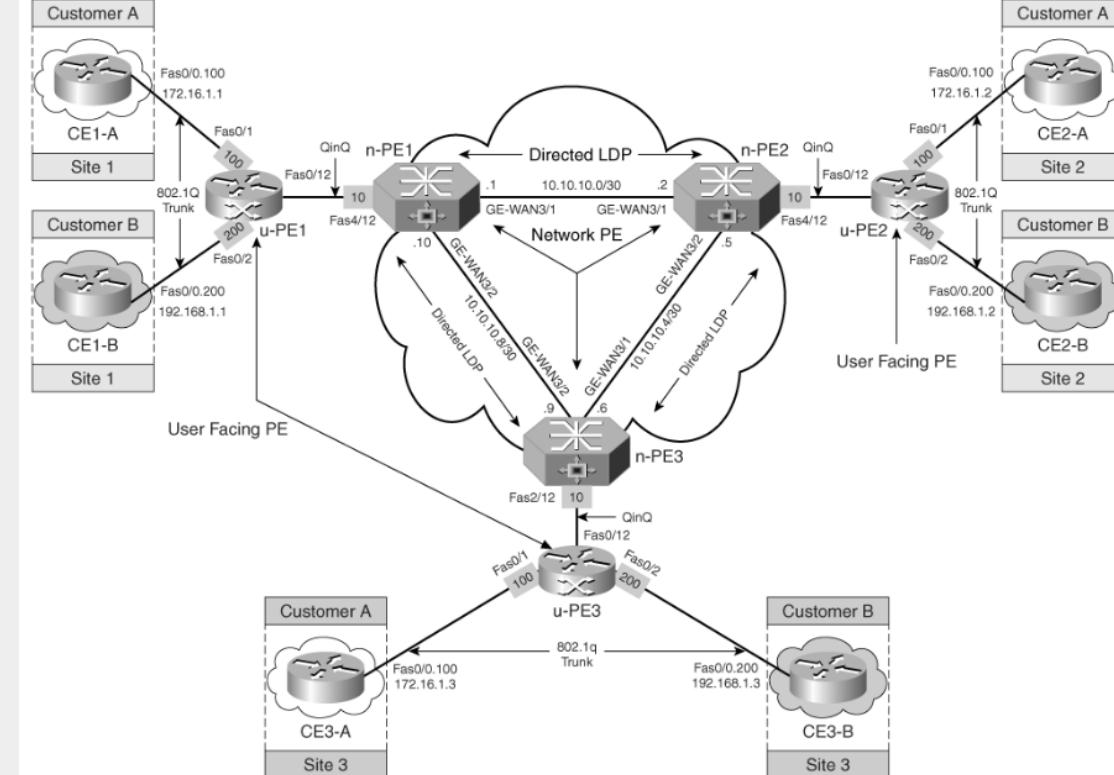
# VPLS LAB:



# VPLS LAB:

## n-PE1

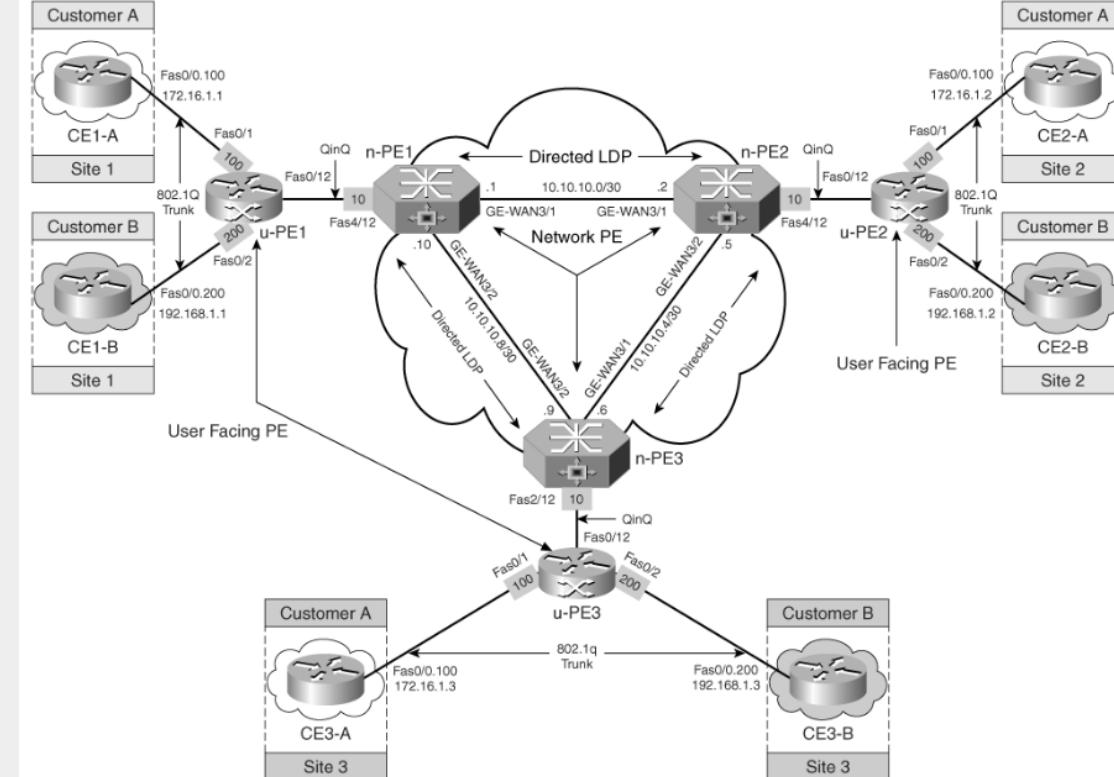
```
!n-PE1
hostname n-PE1
!
mpls label protocol ldp
mpls ldp discovery targeted-hello accept
mpls ldp router-id Loopback0
!
l2 vfi QinQ
  vpn id 10
    neighbor 10.10.10.102 encapsulation mpls
    neighbor 10.10.10.103 encapsulation mpls
!
vlan internal allocation policy ascending
vlan dot1q tag native
!
interface Loopback0
  ip address 10.10.10.101 255.255.255.255
!
interface FastEthernet4/12
  no ip address
  switchport
  switchport access vlan 10
  switchport mode dot1q-tunnel
!
interface Vlan10
  no ip address
  xconnect vfi QinQ
```



# VPLS LAB:

## n-PE2

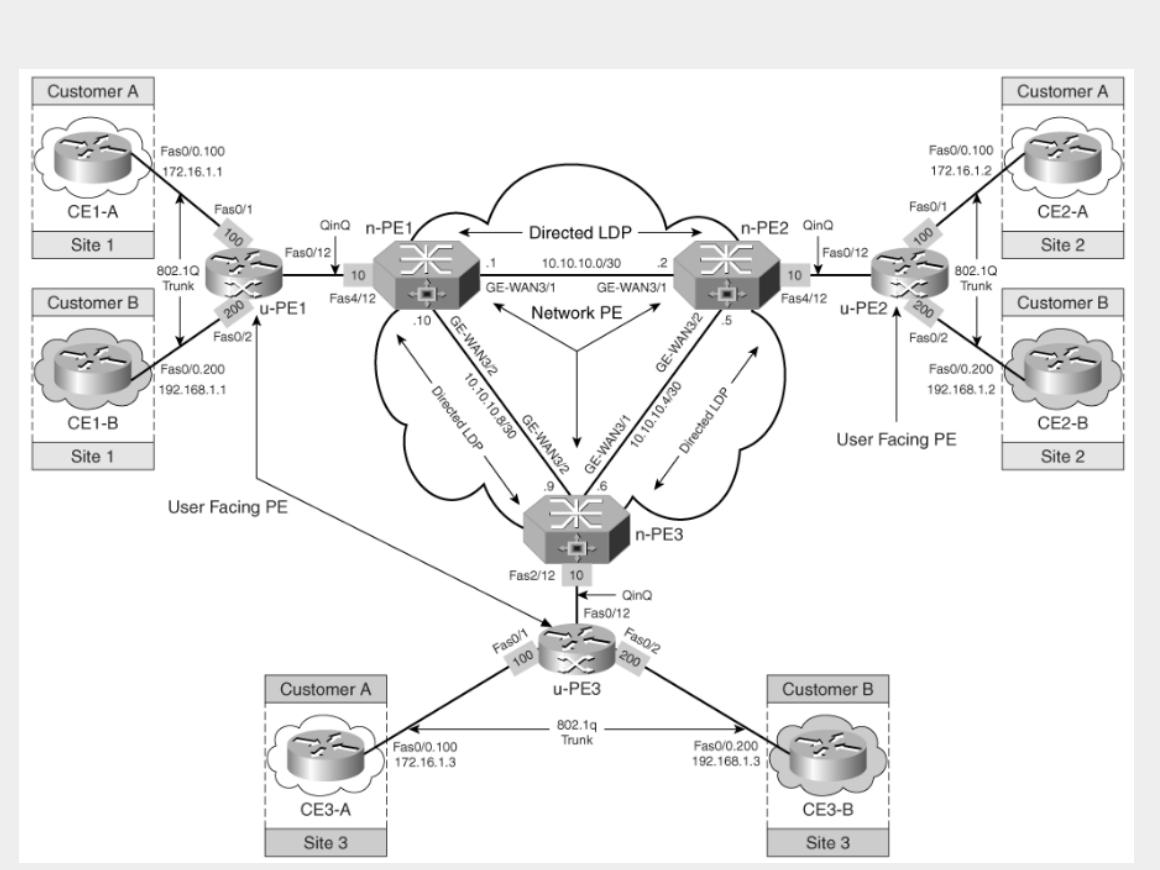
```
!n-PE2
hostname n-PE2
!
mpls label protocol ldp
mpls ldp discovery targeted-hello accept
mpls ldp router-id Loopback0
!
l2 vfi QinQ
  vpn id 10
    neighbor 10.10.10.101 encapsulation mpls
    neighbor 10.10.10.103 encapsulation mpls
!
vlan internal allocation policy ascending
vlan dot1q tag native
!
interface Loopback0
  ip address 10.10.10.102 255.255.255.255
!
interface FastEthernet4/12
  no ip address
  switchport
  switchport access vlan 10
  switchport mode dot1q-tunnel
!
interface Vlan10
  no ip address
  xconnect vfi QinQ
```



# VPLS LAB:

## n-PE3

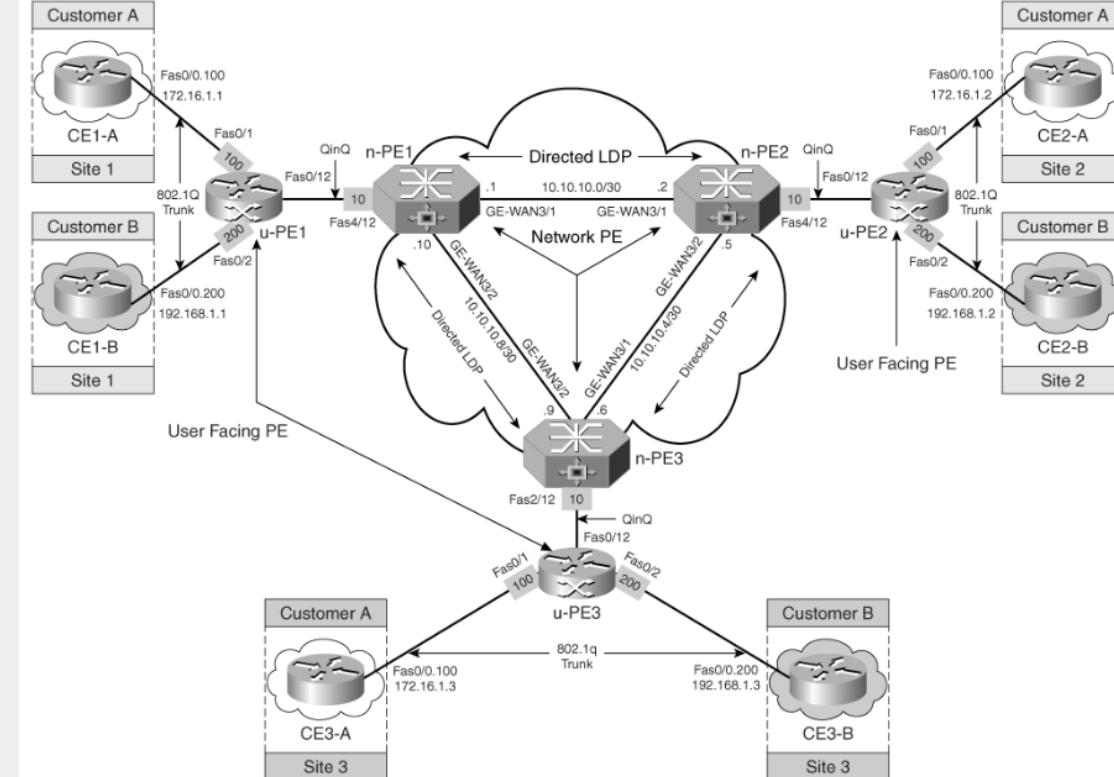
```
!n-PE3
hostname n-PE3
!
mpls label protocol ldp
mpls ldp discovery targeted-hello accept
mpls ldp router-id Loopback0
!
l2 vfi QinQ
  vpn id 10
    neighbor 10.10.10.101 encapsulation mpls
    neighbor 10.10.10.102 encapsulation mpls
!
vlan internal allocation policy ascending
vlan dot1q tag native
!
interface Loopback0
  ip address 10.10.10.103 255.255.255.255
!
interface FastEthernet2/12
  no ip address
  switchport
  switchport access vlan 10
  switchport mode dot1q-tunnel
!
interface Vlan10
  no ip address
  xconnect vfi QinQ
```



# VPLS LAB:

## u-PE1

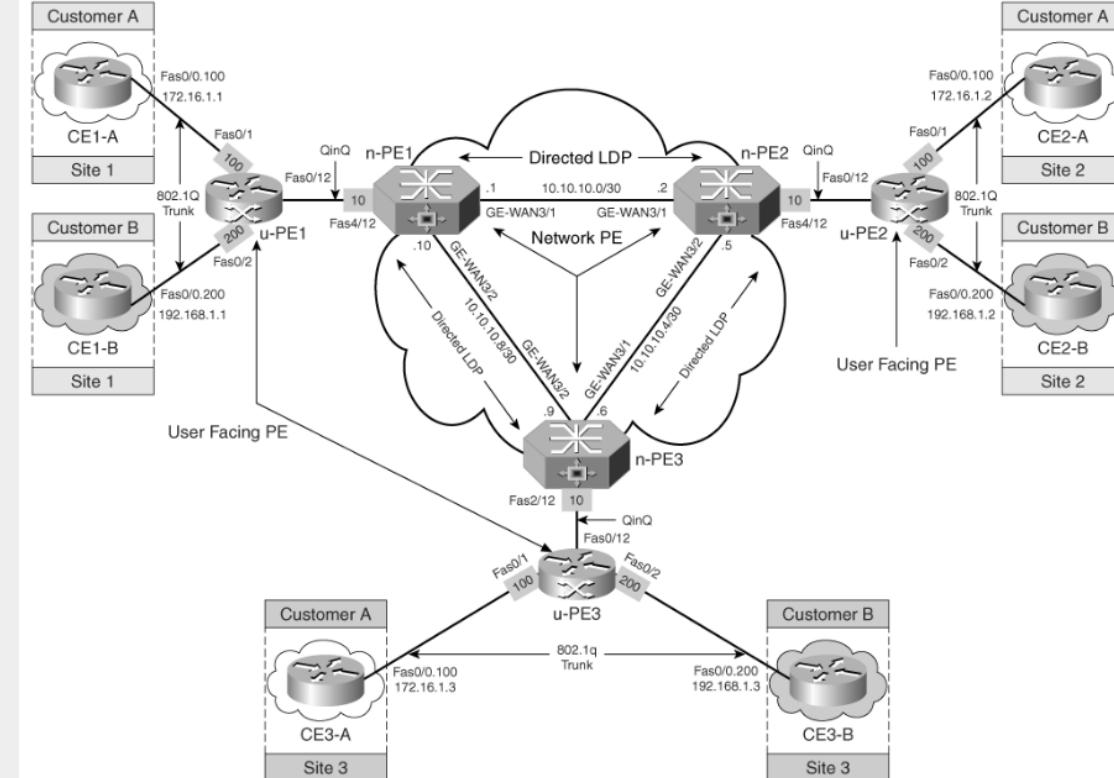
```
!u-PE1
hostname u-PE1
!
vlan 100,200
!
interface FastEthernet0/1
description connected to CE1-A
switchport access vlan 100
switchport trunk encapsulation dot1q
switchport mode dot1q-tunnel
no cdp enable
spanning-tree bpduguard enable
!
interface FastEthernet0/2
description connected to CE1-B
switchport access vlan 200
no switchport trunk encapsulation dot1q
switchport mode dot1q-tunnel
no cdp enable
spanning-tree bpduguard enable
!
interface FastEthernet0/12
description connected to n-PE1
no switchport trunk encapsulation dot1q
switchport trunk allowed vlan 100,200
switchport mode trunk
```



# VPLS LAB:

## u-PE2

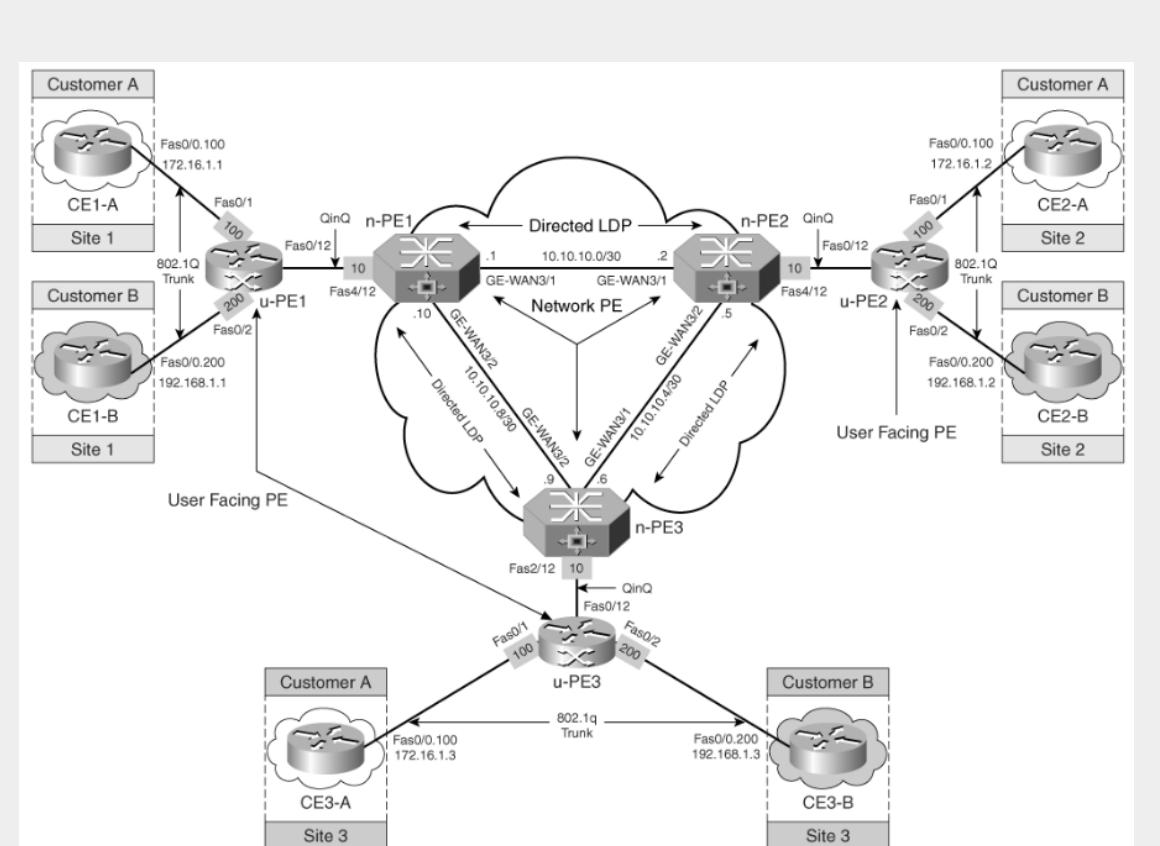
```
!u-PE2
hostname u-PE2
!
vlan 100,200
!
interface FastEthernet0/1
description connected to CE2-A
switchport access vlan 100
no switchport trunk encapsulation dot1q
switchport mode dot1q-tunnel
no cdp enable
spanning-tree bpdulfiler enable
!
interface FastEthernet0/2
description connected to CE2-B
switchport access vlan 200
no switchport trunk encapsulation dot1q
switchport mode dot1q-tunnel
no cdp enable
spanning-tree bpdulfiler enable
!
interface FastEthernet0/12
description connected to n-PE2
switchport trunk encapsulation dot1q
switchport trunk allowed vlan 100,200
switchport mode trunk
```



# VPLS LAB:

## u-PE3

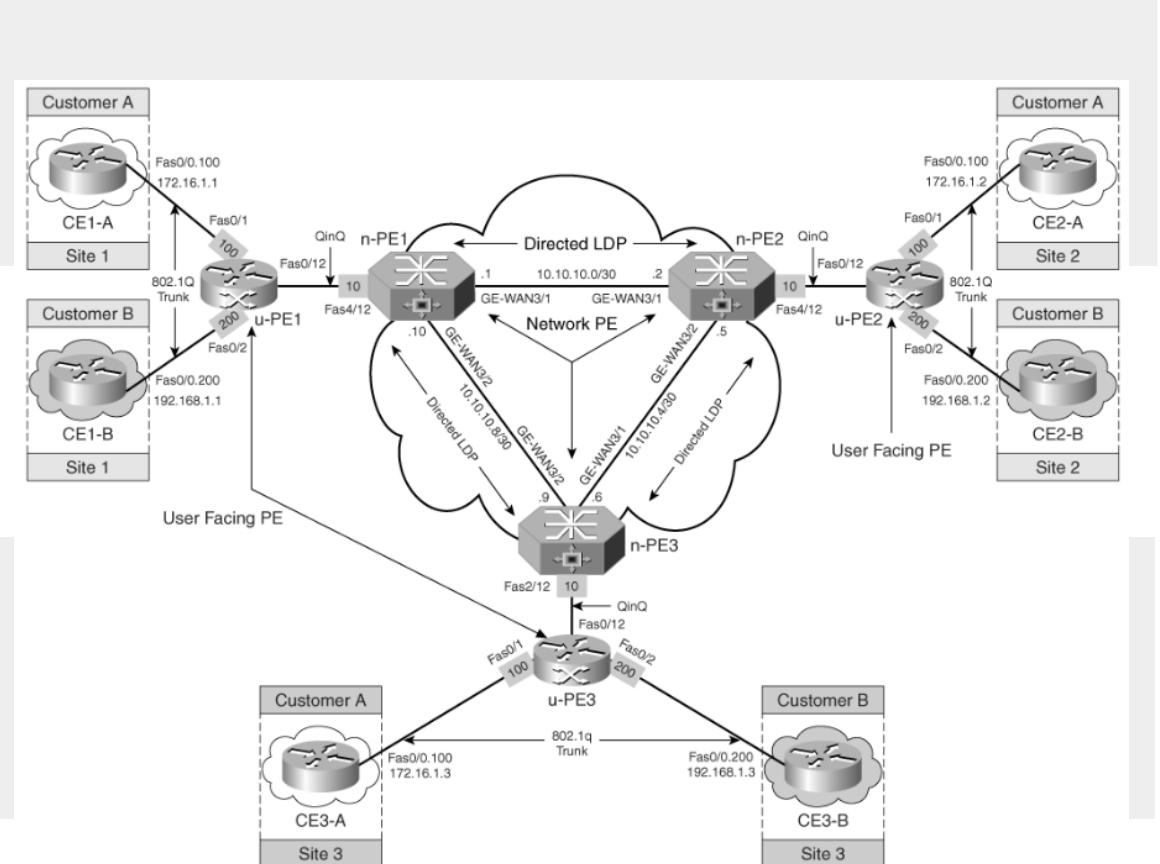
```
!u-PE3
hostname u-PE3
!
vlan 100,200
!
interface FastEthernet0/1
description connected to CE3-A
switchport access vlan 100
no switchport trunk encapsulation dot1q
switchport mode dot1q-tunnel
no cdp enable
spanning-tree bpdulfiler enable
!
interface FastEthernet0/2
description connected to CE3-B
switchport access vlan 200
no switchport trunk encapsulation dot1q
switchport mode dot1q-tunnel
no cdp enable
spanning-tree bpdulfiler enable
!
interface FastEthernet0/12
description connected to n-PE3
switchport trunk encapsulation dot1q
switchport trunk allowed vlan 100,200
switchport mode trunk
```



# VPLS LAB:

## CE1-A

```
!CE1-A
hostname CE1-A
!
interface FastEthernet0/0.100
  encapsulation dot1Q 100
  ip address 172.16.1.1 255.255.255.0
```



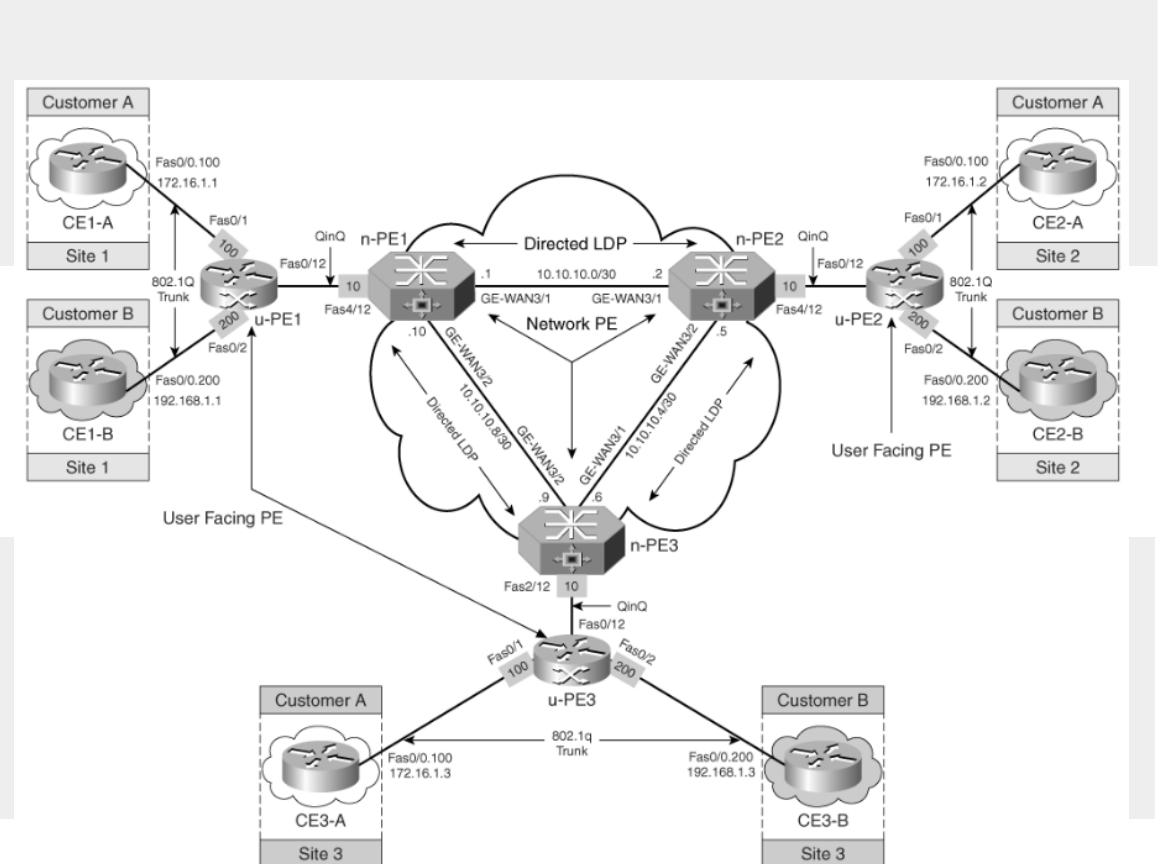
## CE1-B

```
!CE1-B
hostname CE1-B
!
interface FastEthernet0/0.200
  encapsulation dot1Q 200
  ip address 192.168.1.1 255.255.255.0
```

# VPLS LAB:

## CE2-A

```
!CE2-A
hostname CE2-A
!
interface FastEthernet0/0.100
  encapsulation dot1Q 100
  ip address 172.16.1.1 255.255.255.0
```



## CE2-B

```
!CE2-B
hostname CE2-B
!
interface FastEthernet0/0.200
  encapsulation dot1Q 200
  ip address 192.168.1.2 255.255.255.0
```

# VPLS LAB:

---

## CE3-A

```
!CE3-A
hostname CE3-A
!
interface FastEthernet0/0.100
encapsulation dot1Q 100
ip address 172.16.1.1 255.255.255.0
```

## CE3-B

```
!CE3-B
hostname CE3-B
!
interface FastEthernet0/0.200
encapsulation dot1Q 200
ip address 192.168.1.3 255.255.255.0
```

¿dudas?