

# MPLS Carrier Supporting Carrier

Help you help me

(C) Herbert Haas 2005/03/11

## Rationale



- **CsC allows one SP to allow another SP to use a segment of its backbone**
  - ♦ Basis technology: MPLS-VPNs
  - ♦ Customer SP might be an ISP or a BGP/MPLS VPN service provider
- **Useful...**
  - ♦ ...when PE routers cannot maintain all routes to offer the desired service
  - ♦ ...when SPs want to have good IP/MPLS connectivity between their sites while still managing external routes by themselves
  - ♦ ...because there is no need to store external routes of ISP customers on the edge of the central carrier—labels are sufficient

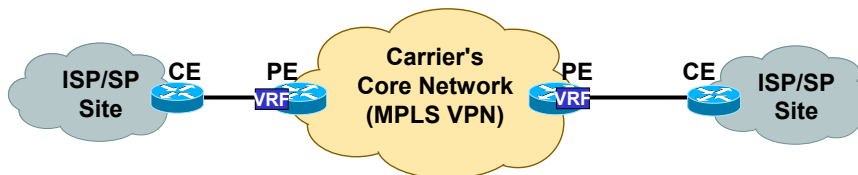
(C) Herbert Haas 2005/03/11

2

## Basic Principles (1)



- Carrier's core network runs MPLS-VPN
- ISP/SP sites are connected to carrier's PE routers via
  - ♦ Physical Interfaces
  - ♦ Logical Interfaces (ATM or FR PVCs)
  - ♦ Ethernet VLANs
- MPLS label exchange between PE and CE routers
  - ♦ Each site can use different PE-CE encapsulation
- Direct exchange of external routes between ISP/SPs sites



## Basic Principles (2)

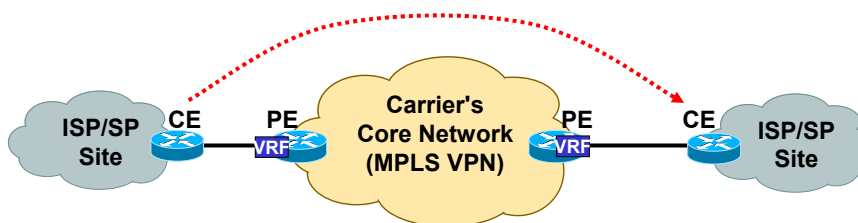


- **ISP/SP's routes + labels exchange between PE and CE**
  - ♦ OSPF | RIP | EIGRP | ISIS + LDP
  - ♦ Static routes + LDP
  - ♦ EBGP IPv4 + Label
- **Plus additional mechanism to only accept packets having a label which had been advertised by PE**

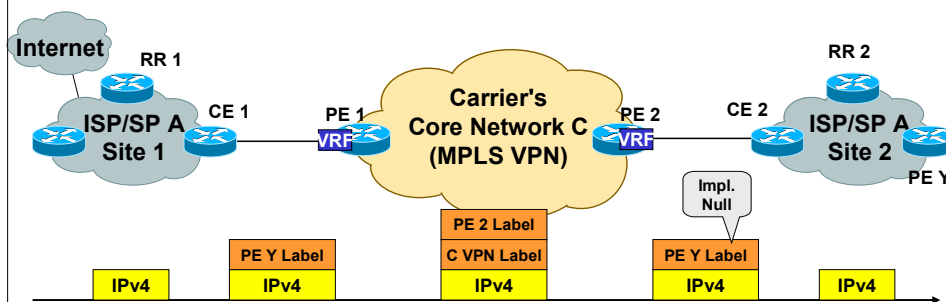
## Basic Principles (3)



- Direct exchange of external routes between ISP/SPs sites
  - ♦ Direct EBGP multihop IPv4/IPv6 or VPNv4
  - ♦ Direct IBGP IPv4/IPv6 or VPNv4
  - ♦ Indirect via route reflector IBGP sessions

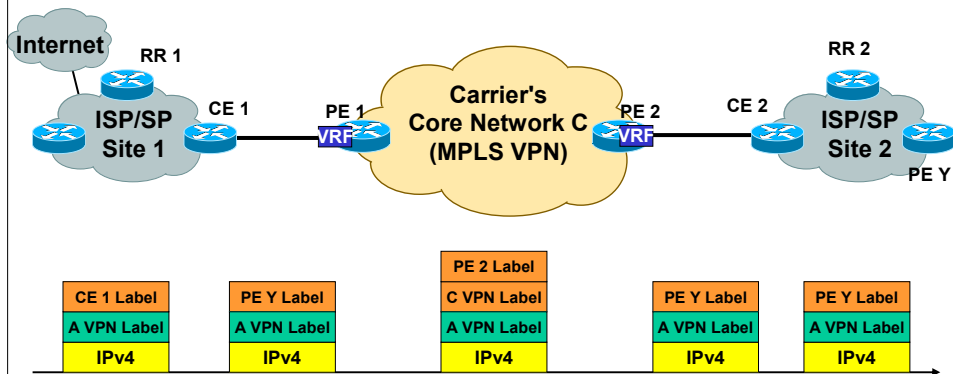


## Example 1



- Carrier providing connectivity for the ISPs/Enterprises

## Example 2



- Carrier providing connectivity for the SP's sites already offering mpls-vpn services (hierarchical VPNs)

## MPLS CSC Examples

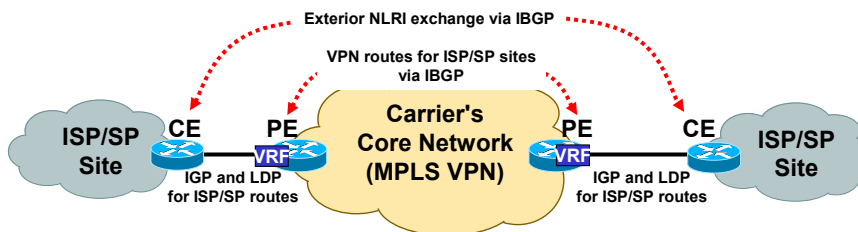


- Customer-ISP not running MPLS
- Customer-ISP running MPLS
- Customer-ISP running MPLS-VPN

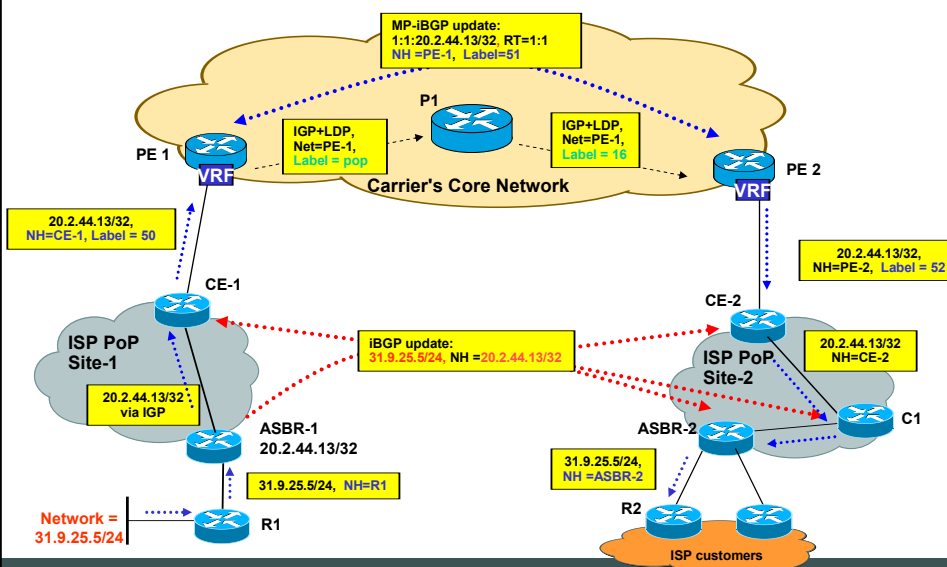
## Example 1: ISP/SP not running MPLS



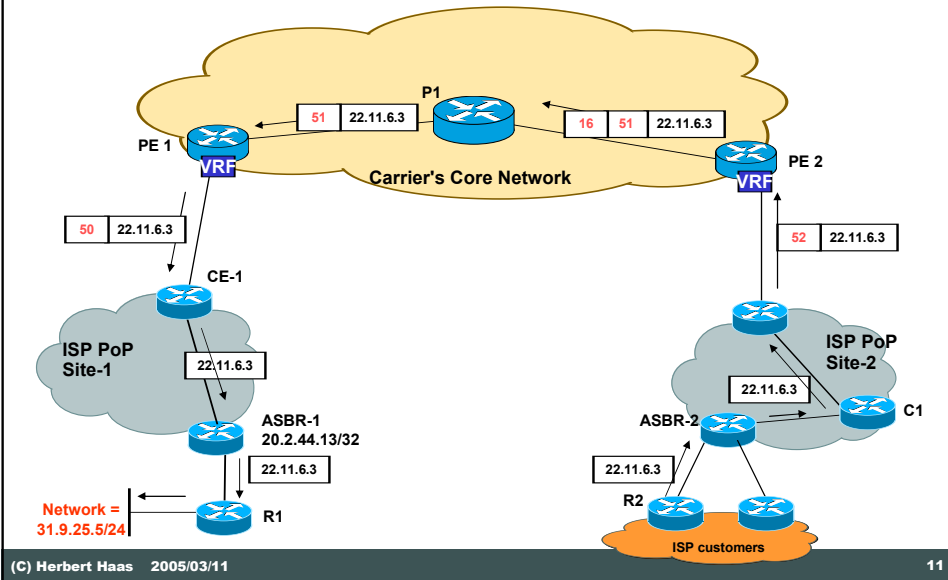
- Only the backbone carrier uses MPLS
  - ◆ Allows internal routes of ISP/SP to be exchanged between CE and PE
- ISP/SP runs IP only
  - ◆ But MPLS between CE and PE
  - ◆ iBGP full mesh to exchange external routes



## Example 1: ISP/SP not running MPLS



## Example 1: ISP/SP not running MPLS



## MPLS CSC Examples



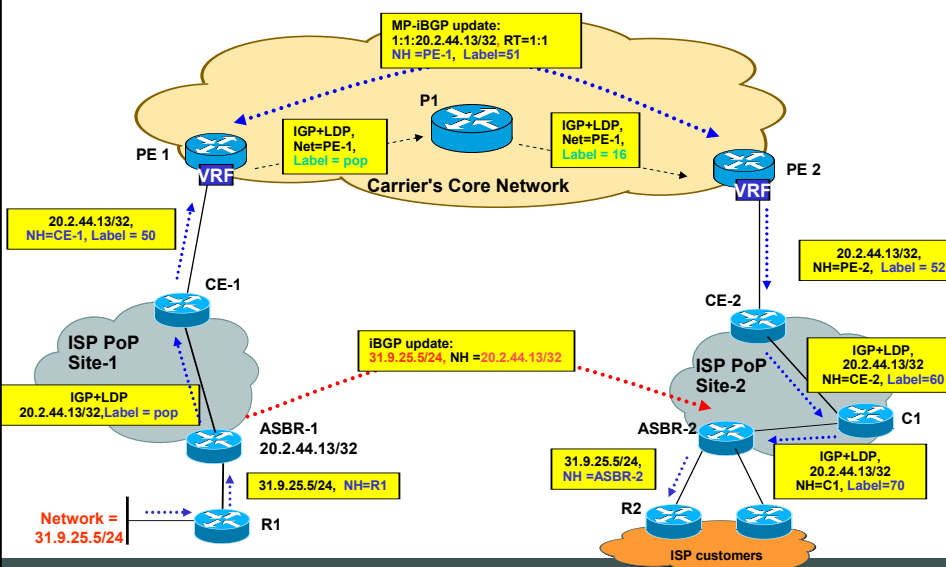
- Customer-ISP not running MPLS
- Customer-ISP running MPLS
- Customer-ISP running MPLS-VPN

## Example 2: Customer-ISP running MPLS

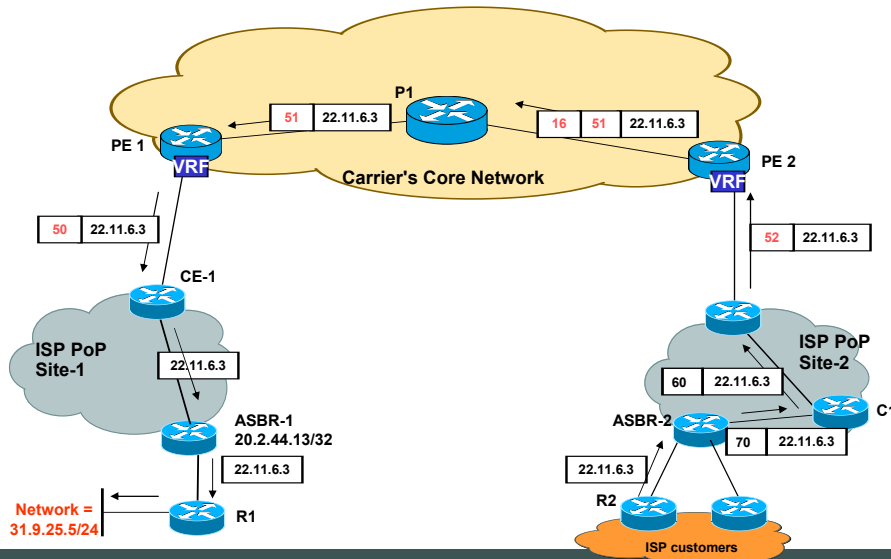


- No need of iBGP full-mesh
- iBGP sessions between ASBRs only
- On ASBR, next-hop addresses of external BGP routes is learned by IGP and LDP
  - ◆ BGP routes are external routes
  - ◆ Next-hop addresses are learned from internal routes

## Example 2: Customer-ISP running MPLS



## Example 2: Customer-ISP running MPLS



(C) Herbert Haas 2005/03/11

15

## MPLS CSC Examples



- Customer-ISP not running MPLS
- Customer-ISP running MPLS
- Customer-ISP running MPLS-VPN

(C) Herbert Haas 2005/03/11

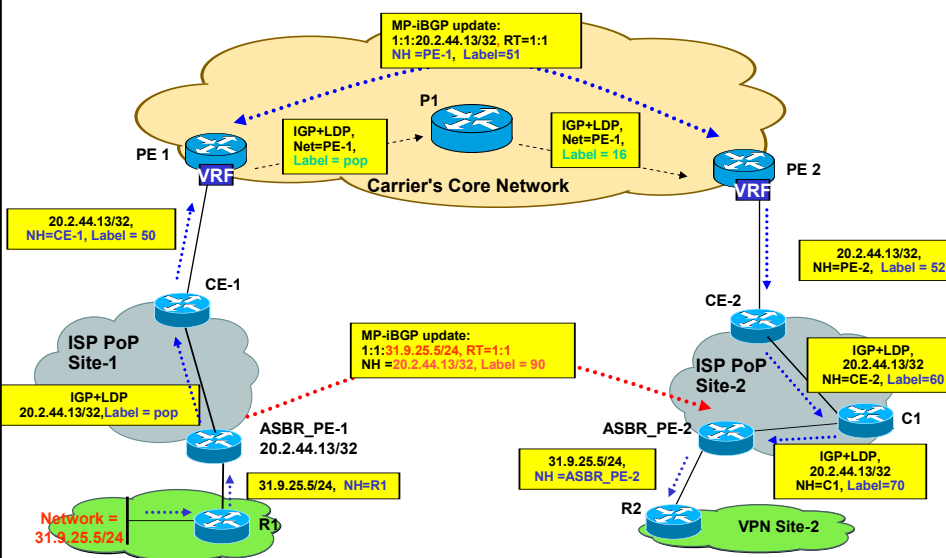
16

### Example 3: CSC for MPLS VPN

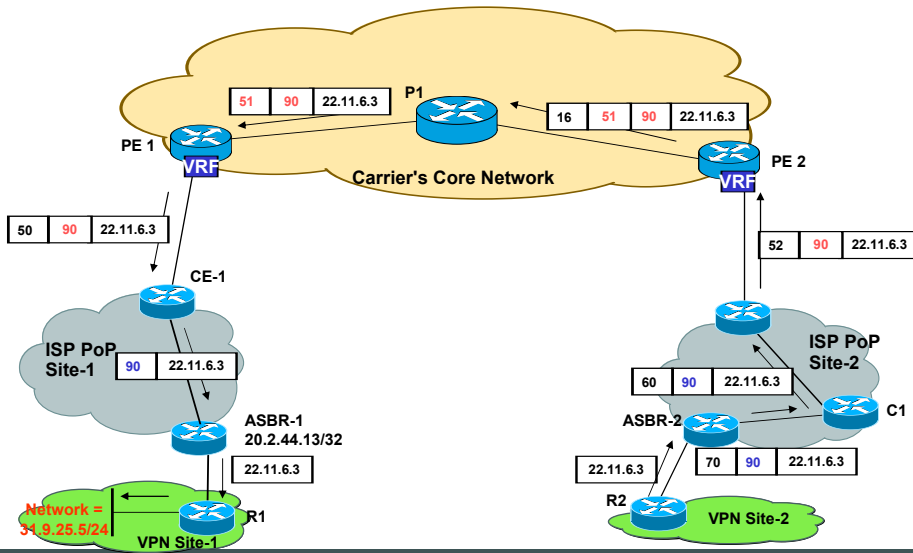


- A customer carrier may provide BGP/MPLS VPN services
  - ◆ Then external routes are VPN-IPv4 routes
  - ◆ Every site within the customer carrier must use MPLS
- Hierarchical VPNs
  - ◆ MPLS-VPN carrier has VPN customers that also provides MPLS-VPN services

### Example 3: CSC for MPLS VPN



### Example 3: CSC for MPLS VPN



## Configuration



## CsC IOS Commands (1)

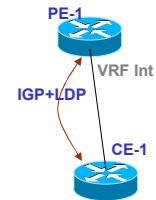


```
PE (conf)#int ser0/0
PE (conf-if)#ip vrf for rajiva
PE (conf-if)#mpls ip
PE (conf-if)#mpls ldp protocol ldp
PE (conf)#end
```

```
CE (conf)#int ser0/0
CE (conf-if)#mpls ip
CE (conf-if)#mpls ldp protocol ldp
CE (conf-if)#end
```

```
Sh mpls interface
Sh mpls ldp discovery
Sh mpls ldp bind
Sh mpls ldp neighbor
Sh mpls forward
```

```
Sh mpls interface [vrf <name>] all
Sh mpls ldp disc [vrf <name>] all
Sh mpls ldp bind vrf <name>
Sh mpls ip bind vrf <name>
Sh mpls ldp neighbor [vrf <name>] all
Sh mpls forward [vrf <name>]
```



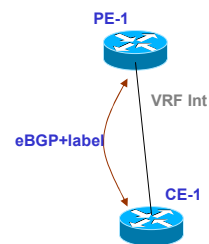
## CsC IOS Commands (2)



### Choice2: Enable eBGP+label on PE-CE:

```
PE (conf)#router bgp 1
PE (conf-router)#address-family ip vrf rajiva
PE (conf-router-af)#neighbor 200.1.61.6 remote-as 2
PE (conf-router-af)# neighbor 200.1.61.6 send-label
PE (conf-router-af)#end
```

```
CE (conf)#router bgp 2
CE (conf -router)#neighbor 200.1.61.5 remote-as 1
CE (conf-router)#neighbor 200.1.61.5 send-label
CE (conf-router)#end
```



1. No IGP needed on PE-CE
2. No LDP needed on PE-CE

## CsC IOS commands (3)



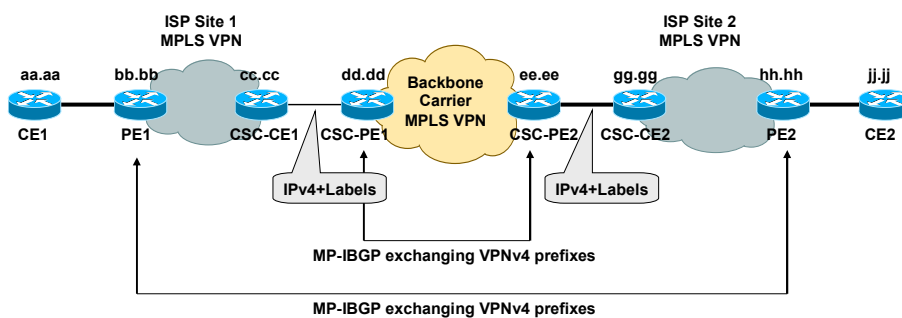
- On PE

```
Sh ip bgp vpn vrf <vrf> neighbor
Sh ip bgp vpn vrf <vrf> label
Sh mpls forward vrf <vrf>
```

- On CE

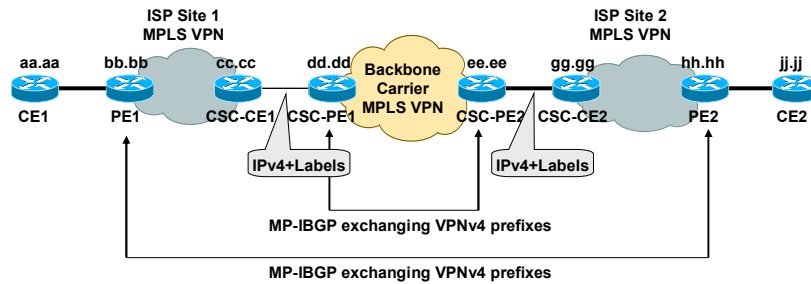
```
Sh ip bgp neighbor
Sh ip bgp labels
Sh mpls forward
```

## CsC using BGP configuration example



- Example CSC topology for exchanging IPv4 routes and MPLS labels

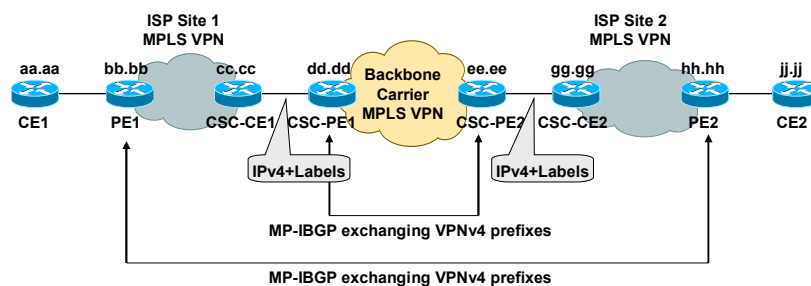
# CSC-PE1 Configuration



```
ip cef
!
ip vrf vpn1
rd 100:1
route-target export 100:1
route-target import 100:1
mpls label protocol ldp
!
interface Loopback0
ip address ee.ee.ee.ee 255.255.255.255
!
```

```
interface Ethernet5/0
ip vrf forwarding vpn1
ip address pp.0.0.2 255.0.0.0
!
router bgp 100
address-family ipv4 vrf vpn1
neighbor pp.0.0.1 remote-as 200
neighbor pp.0.0.1 activate
neighbor pp.0.0.1 as-override
neighbor pp.0.0.1 advertisement-interval 5
neighbor pp.0.0.1 send-label
```

# CSC-CE1 Configuration



```
ip cef
!
mpls label protocol ldp
!
interface Loopback0
ip address cc.cc.cc.cc 255.255.255.255
!
interface Ethernet3/0
ip address pp.0.0.1 255.0.0.0
```

```
router ospf 200
redistribute bgp 200 metric 3 subnets.
network cc.cc.cc.cc 0.0.0.0 area 200
network nn.0.0.0 0.255.255.255 area 200
passive-interface Ethernet3/0
!
router bgp 200
neighbor pp.0.0.2 remote-as 100
neighbor pp.0.0.2 send-label
redistribute connected
redistribute ospf 200 metric 4 match internal
```

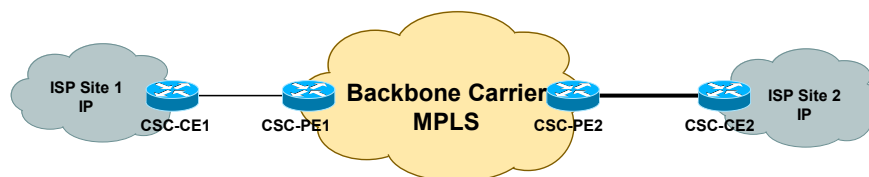
## PE1 Configuration



```
ip cef
!
ip vrf vpn2
rd 200:1
route-target export 200:1
route-target import 200:1
mpls label protocol ldp
!
interface Ethernet3/3
ip vrf forwarding vpn2
ip address mm.0.0.2 255.0.0.0
router ospf 200
redistribute connected subnets
network bb.bb.bb.bb 0.0.0.0 area 200
network nn.0.0.0 0.255.255.255 area 200
```

```
router bgp 200
no bgp default ipv4-unicast
neighbor hh.hh.hh.hh remote-as 200
neighbor hh.hh.hh.hh update-source Loopback0
!
address-family vpnv4
neighbor hh.hh.hh.hh activate
neighbor hh.hh.hh.hh send-community extended
bgp dampening 30
exit-address-family
!
address-family ipv4 vrf vpn2
neighbor mm.0.0.1 remote-as 300
neighbor mm.0.0.1 activate
neighbor mm.0.0.1 as-override
neighbor mm.0.0.1 advertisement-interval 5
```

## CsC using OSPF configuration example



- The following configuration examples show the configuration of each router in the CsC network
- OSPF is used to connect the customer carrier to the backbone carrier

## CSC-CE1 Configuration



```
interface Serial 0/0
ip address 38.0.0.2 255.0.0.0
mpls label protocol ldp
mpls ip
!
router ospf 200
log-adjacency-changes
redistribute connected subnets
network 14.14.14.14 0.0.0.0 area 200
network 38.0.0.0 0.255.255.255 area 200
network 46.0.0.0 0.255.255.255 area 200
```

- Only needs standard IGP + LDP to exchange routing updates and Labels with the Backbone ISP

## CSC-PE1 Configuration



```
router ospf 100
log-adjacency-changes
passive-interface ATM3/0/0.1
passive-interface Loopback100
network 11.11.11.11 0.0.0.0 area 100
network 33.0.0.0 0.255.255.255 area 100

ip vrf vpn1
rd 100:0
route-target export 100:0
route-target import 100:0
mpls label protocol ldp
!
router ospf 200 vrf vpn1
log-adjacency-changes
redistribute bgp 100 metric-type 1 subnets
network 19.19.19.19 0.0.0.0 area 200
network 46.0.0.0 0.255.255.255 area 200
!
```

```
router bgp 100
bgp log-neighbor-changes
timers bgp 10 30
neighbor 12.12.12.12 remote-as 100
neighbor 12.12.12.12 update-source Loopback0
!
address-family ipv4
neighbor 12.12.12.12 activate
neighbor 12.12.12.12 send-community extended
no synchronization
exit-address-family
!
address-family vpnv4
neighbor 12.12.12.12 activate
neighbor 12.12.12.12 send-community extended
exit-address-family
!
address-family ipv4 vrf vpn1
redistribute ospf 200 match internal external 1
external 2
no auto-summary
no synchronization
```