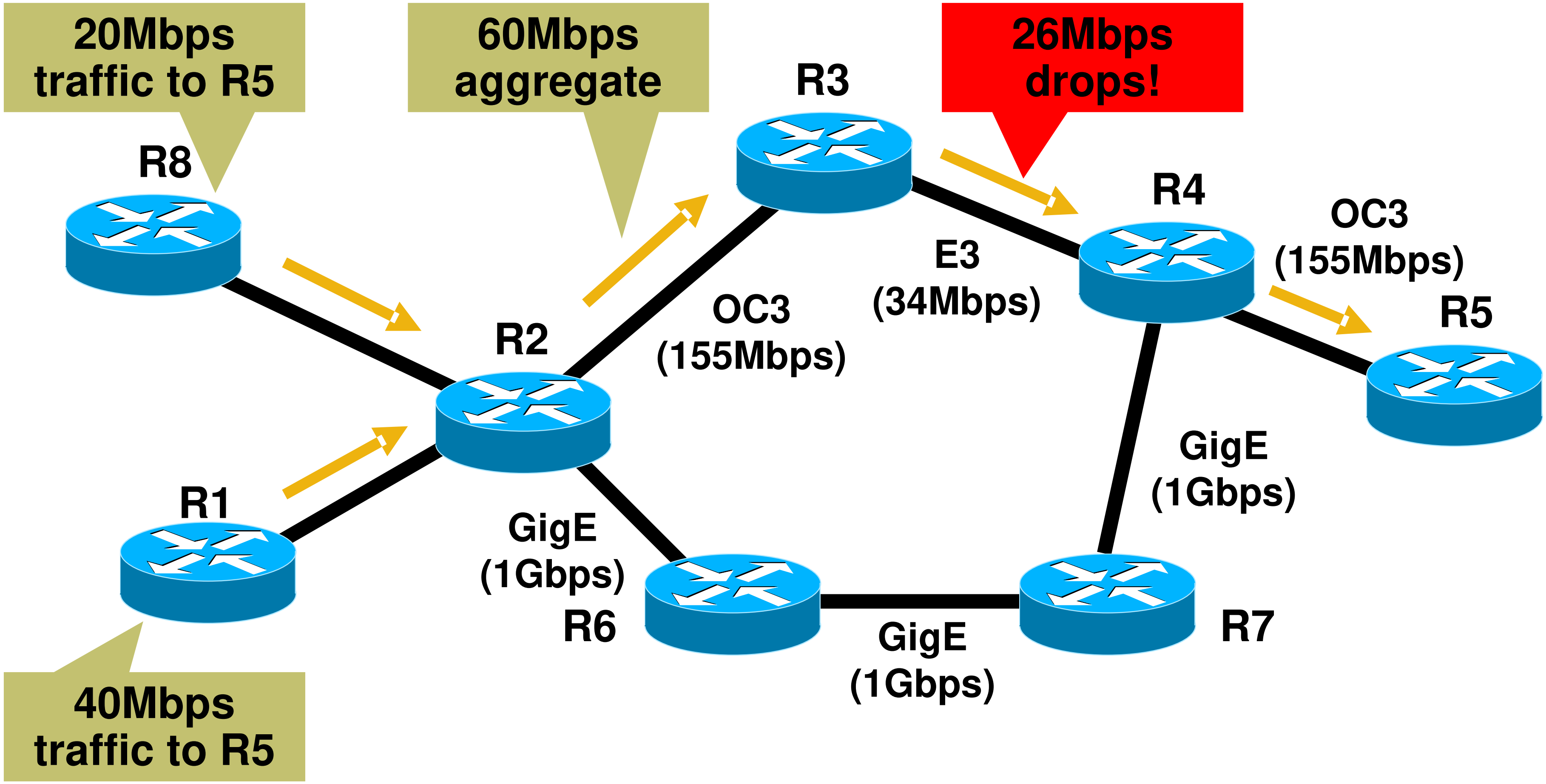
Traffic Engineering (TE) & Multiprotocol Label Switching (MPLS)

universidade de aveiro
deti.ua.pt

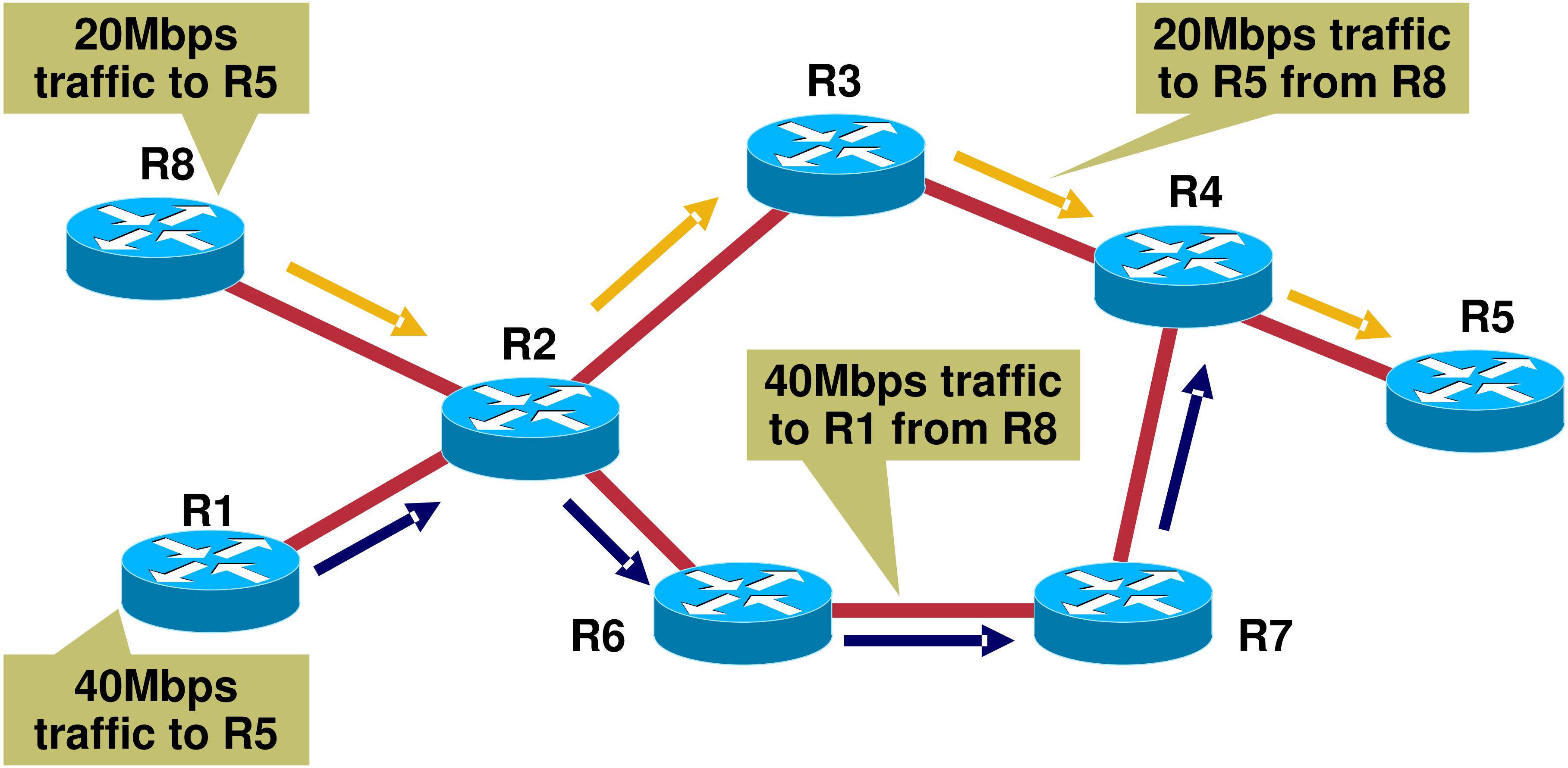
Traffic Engineering (TE)

- Network Engineering
  - Build your network to carry your predicted traffic!
  - Traffic patterns are impossible to predict!
  - Routing is based on the destination and does not allow to take the maximum possible advantage of the network resources.
  - IP source routing (using options field of IP header) is not usable in practice due to security reasons.
- Traffic Engineering
  - Manipulate your traffic path to fit your network!
    - Can be done with routing protocol costs (difficult deployment), or MPLS.
    - With RIP or OSPF or ANY OTHER IGP it is not possible to condition multiple traffic flows.
  - Increase efficiency of bandwidth resources.
    - Prevent over-utilized (congested) links whilst other links are under-utilized.
  - Ensure the most desirable/appropriate path for some/all traffic.
    - Override the shortest path selected by the routing protocols.

Shortest Path and Congestion



A TE Solution



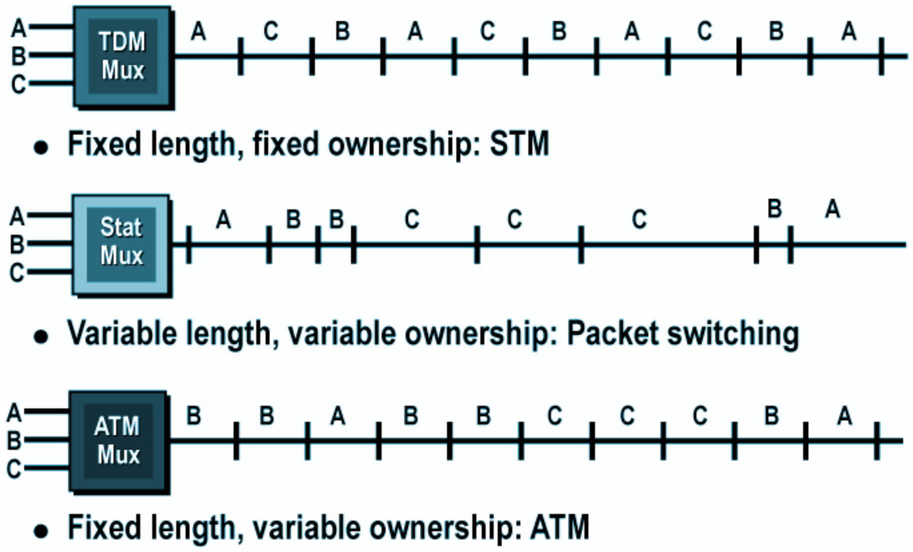
Tunnels are UNI-DIRECTIONAL

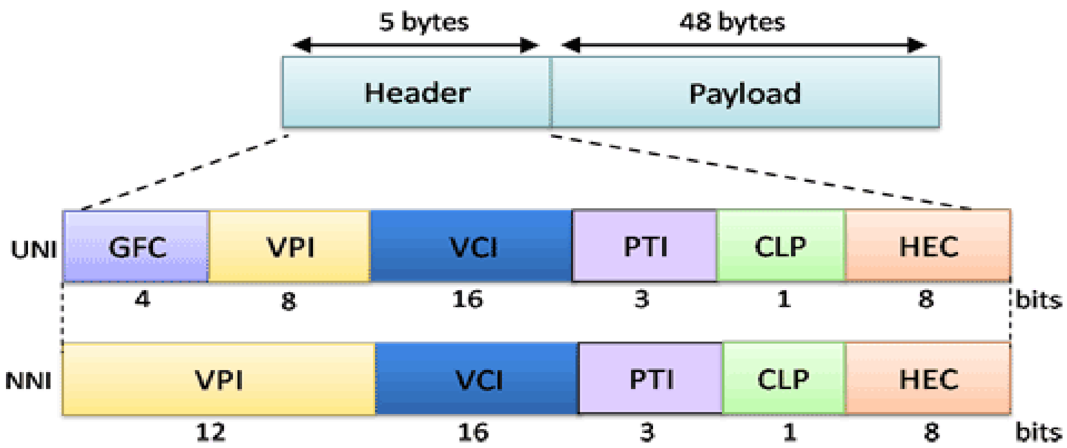
Normal path: R8 > R2 > R3 > R4 > R5

Tunnel path: R1 > R2 > R6 > R7 > R4

Asynchronous Transfer Mode (ATM)

- ATM is a blend of Synchronous Transfer Mode (STM) and packet switching.
  - It has variable assignment, based on the arrival rate and delay sensitivity of the traffic.
  - However, after the assignment occurs, uses fixed-length time slots called cells.
    - Delay-sensitive traffic has immediate assignment
    - Data traffic can be temporarily buffered before being transmitted.
- Is a form of cell switching using small fixed-sized data units called cells.
  - 53 bytes: 5 header and 48 data.

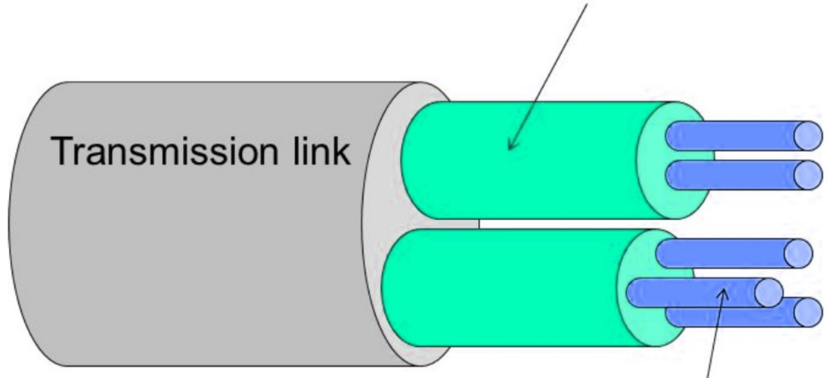


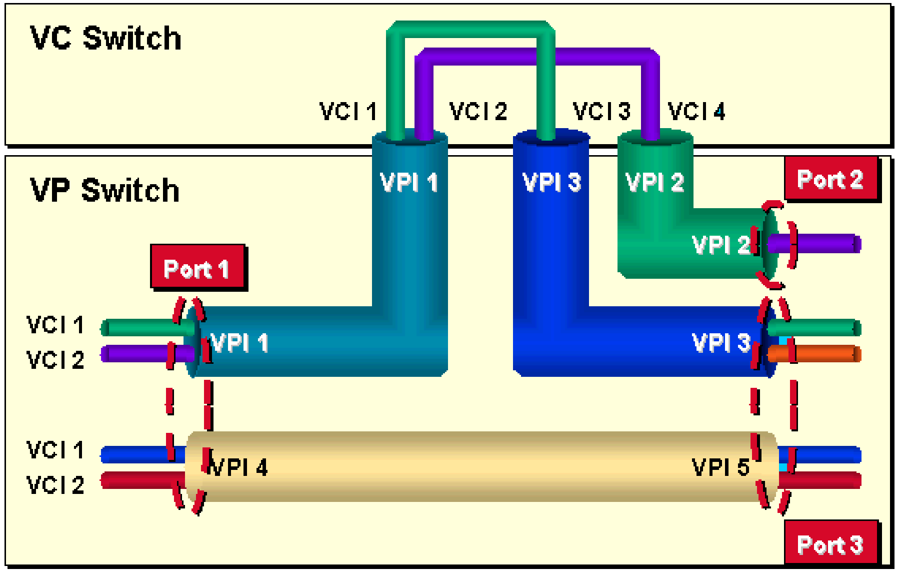


UNI (User-Network Interface).
NNI (Network-Network Interface).

ATM Connections and Switching

- ATM is connection-oriented.
  - A connection (an ATM channel) must be established before any cells are sent.
  - Two levels of ATM connections:
    - Virtual path connections.
    - Virtual channel connections.
    - Indicated by two fields in the cell header:
    - Virtual Path Identifier: VPI.
    - Virtual Channel Identifier: VCI.
- Switching based on VPI/VCI.

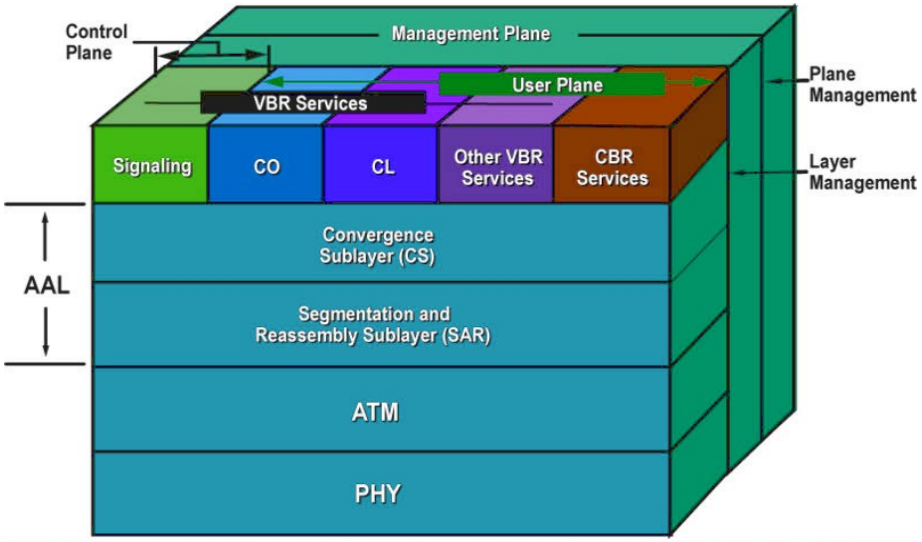




Port in	VPI/VCI	Port out	VPI/VCI
1	1/1	2	2/4
1	1/2	2	3/3
1	4/1	3	5/1
1	4/2	3	5/2

ATM Adaptation Layer (AAL)

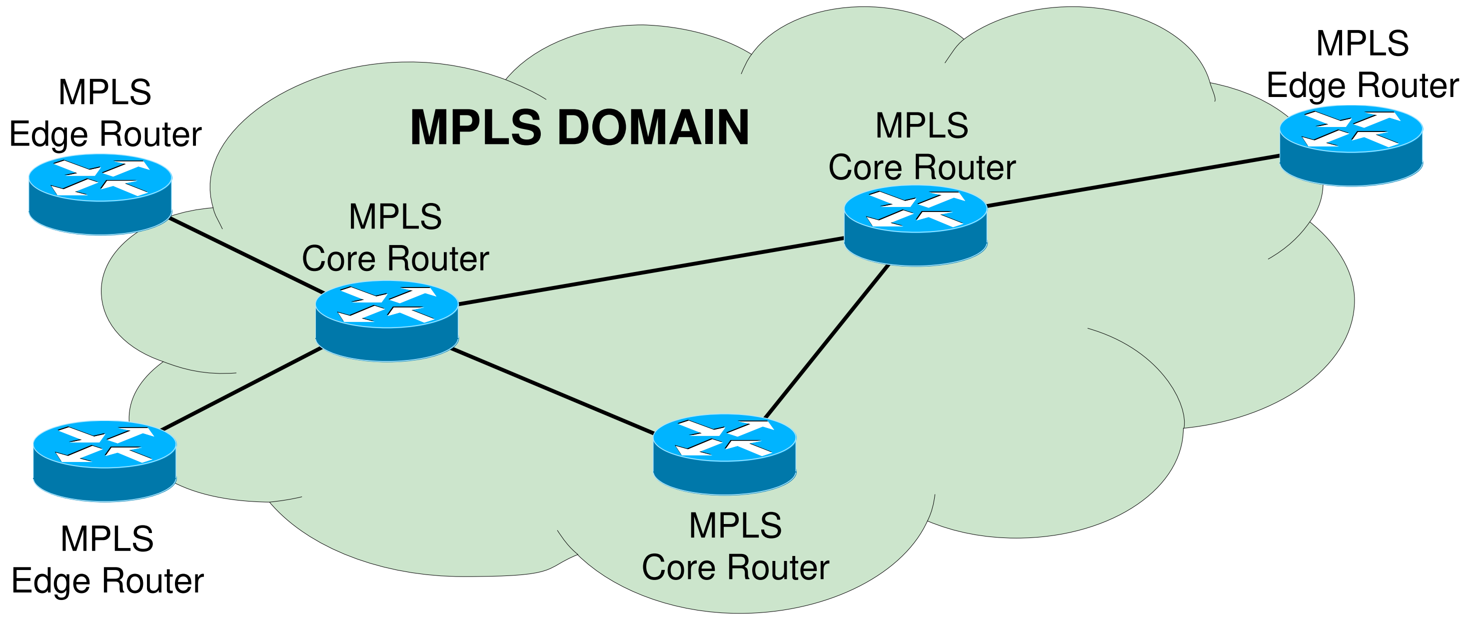
- AAL is responsible for providing specific transport services to the higher layer protocols.
- The AAL is divided into:
  - Convergence Sublayer (CS) - manages the flow of data to and from SAR sublayer.
  - Segmentation and Reassembly Sublayer (SAR) - breaks data into cells at the sender and reassembles cells into larger data units at the receiver.
- ITU-T has defined four AAL service classes based on combinations of these three characteristics
  - Class A is a constant bit rate (CBR), delay-sensitive, connection-oriented service or a circuit emulation service.
  - Class B is a variable bit rate (VBR) service requiring time synchronization between sender and receiver (e.g., real-time compressed audio and video).
  - Classes C and D are delay-insensitive VBR services.
- Four AAL protocol types were defined to support the four service classes.
  - AAL 1 and AAL 5; And not in use anymore: AAL 2 and AAL 3/4.
  - Each type describes the format of the SAR-PDU (or the cell Payload field) and related operational procedures.



Service Class	A	B	C	D
Connection Mode	Connection-Oriented			Connectionless
Bit Rate	Constant	Variable		
End-to-End Timing Relationship	Required		Not Required	
Users	Circuit Emulation (e.g., Voice)	Packet Video and Compressed Voice	Connection-Oriented Data (e.g., Frame Relay)	Connectionless Data (e.g., SMDS, IP)
Sugested AAL Type	1	2	3/4, 5	3/4, 5

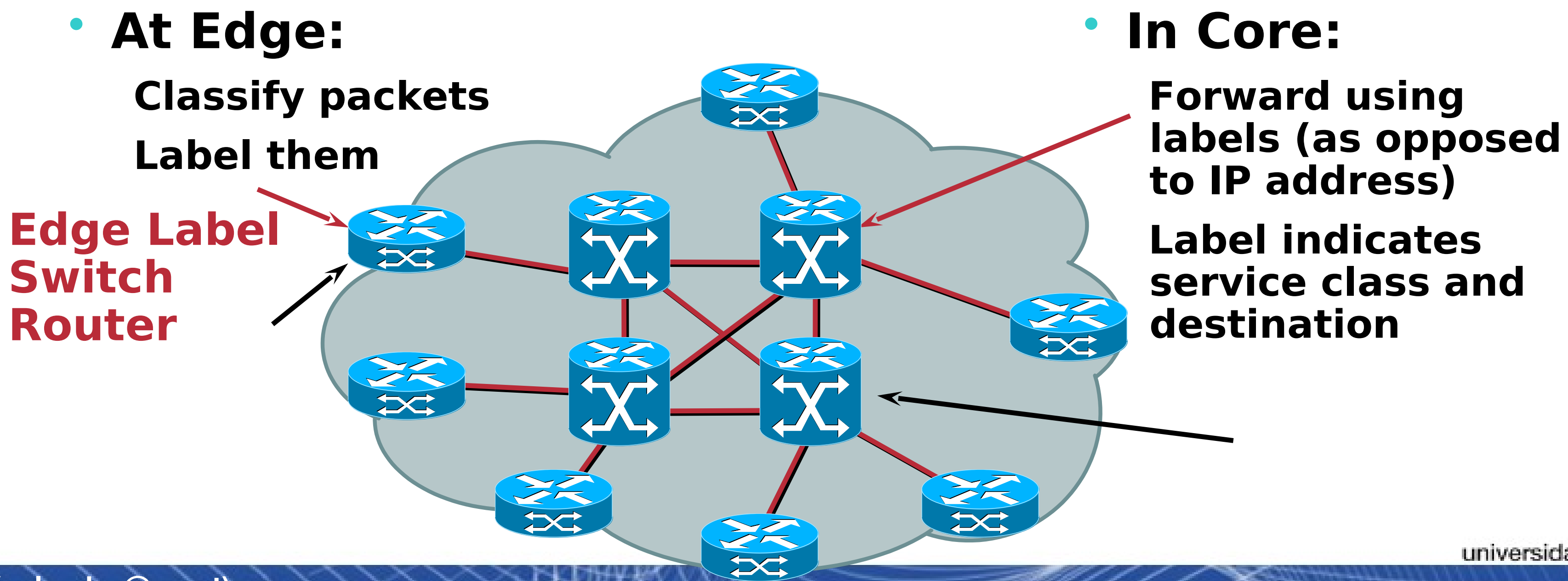
Multiprotocol Label Switching (MPLS)

- Packets are labeled at the source with the label of the first hop.
- As a packet travels from one router to the next, each router makes an independent forwarding decision for that packet based on a label.
- Advantages
  - Simplification of the packet routing process on routers.
  - Traffic engineering capability.
  - Simplification of the network management (a single protocol layer).



MPLS Fundamentals

- Based on the label-swapping and forwarding paradigm.
- As a packet enters an MPLS network, it is assigned a label based on its **Forwarding Equivalence Class (FEC)** as determined at the edge of the MPLS network.
- FECs are groups of packets forwarded over the same **Label Switched Path (LSP)** by **Label Switching Routers (LSR).**
- Need a mechanism that will create and distribute labels to establish LSP paths.
- Separated into two planes:
  - Control Plane - Responsible for maintaining correct label tables among Label Switching Routers.
  - Forwarding Plane - Uses label carried by packet and label table maintained by LSR to forward the packet.

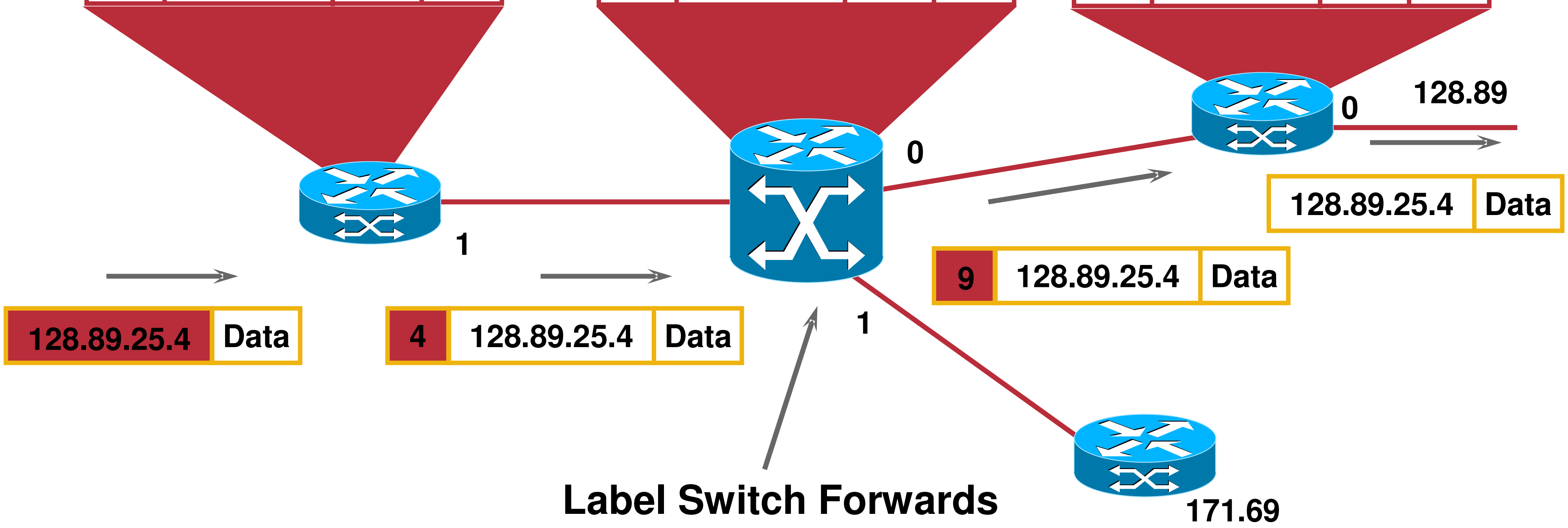


MPLS Switching

In Label	Address Prefix	Out I'face	Out Label
	128.89	1	4
	171.69	1	5
...	...	...	...

In Label	Address Prefix	Out I'face	Out Label
4	128.89	0	9
5	171.69	1	7
...	...	...	...

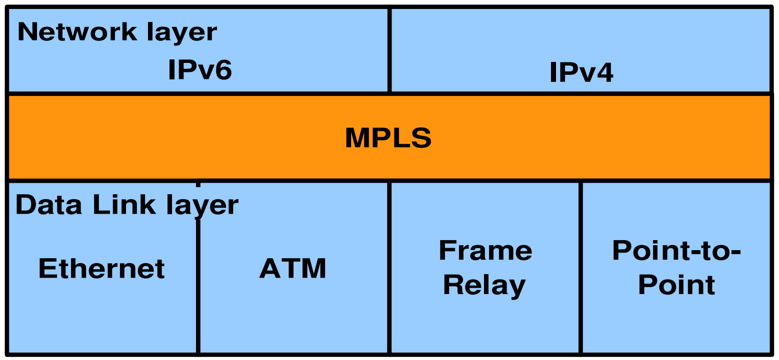
In Label	Address Prefix	Out I'face	Out Label
9	128.89	0	
...	...	...	...



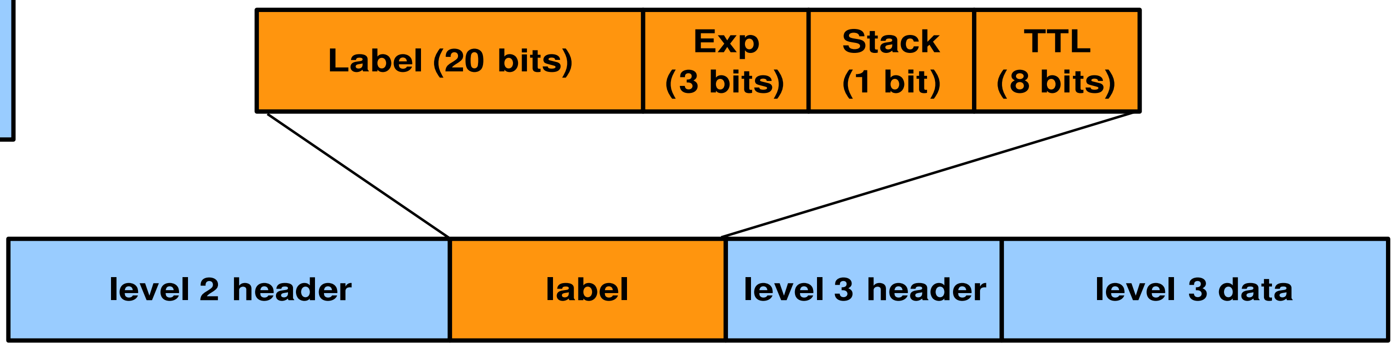
Label Switch Forwards Based on Label

MPLS Labels

- On some Data Link (level 2) technologies, label is given by the appropriate fields of their header.
  - ATM technology : VPI (Virtual Path ID) and VCI (Virtual Channel ID) fields.
  - Frame Relay technology: DLCI (Data Link Connection Identifier) field.
- On other Data Link technologies (Point-to-Point, Ethernet), the label is inserted between layer 2 and layer 3 headers.
- Label is a 20-bit field that carries the actual value of the Label.
- TTL field is IP independent – Similar purpose.

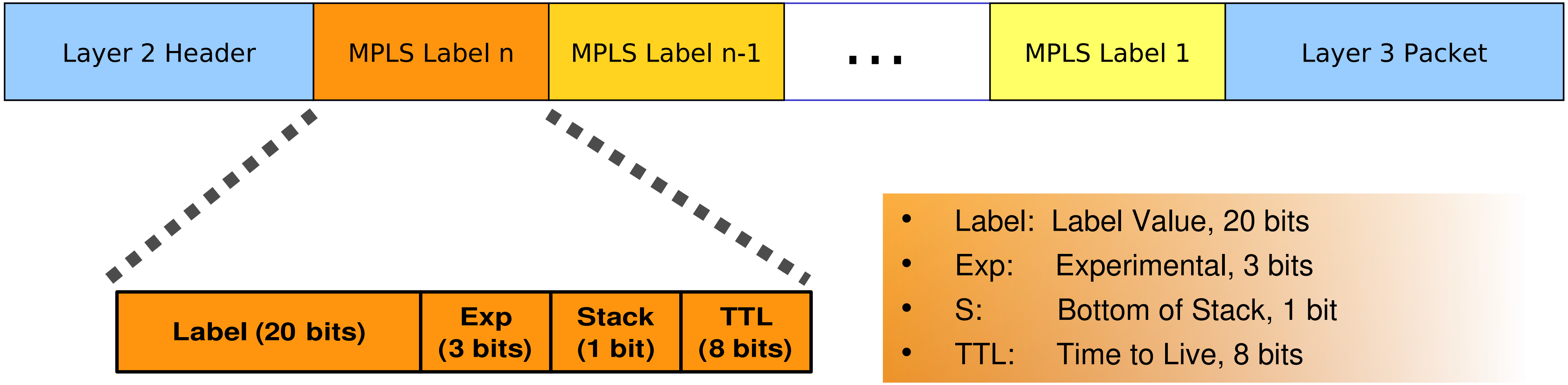


- Label: Label Value, 20 bits
- Exp: Experimental, 3 bits
- S: Bottom of Stack, 1 bit
- TTL: Time to Live, 8 bits



MPLS Label Stacking

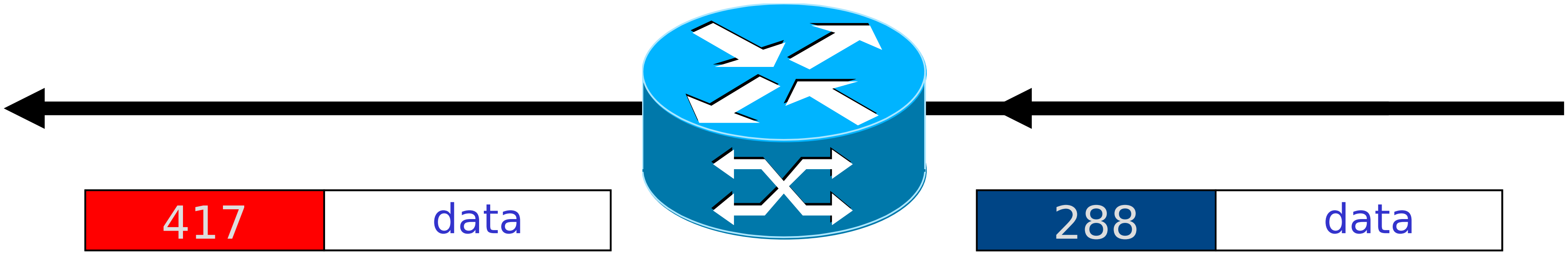
RFC 3032: MPLS Label Stack Encoding



- Label: Label Value, 20 bits
- Exp: Experimental, 3 bits
- S: Bottom of Stack, 1 bit
- TTL: Time to Live, 8 bits

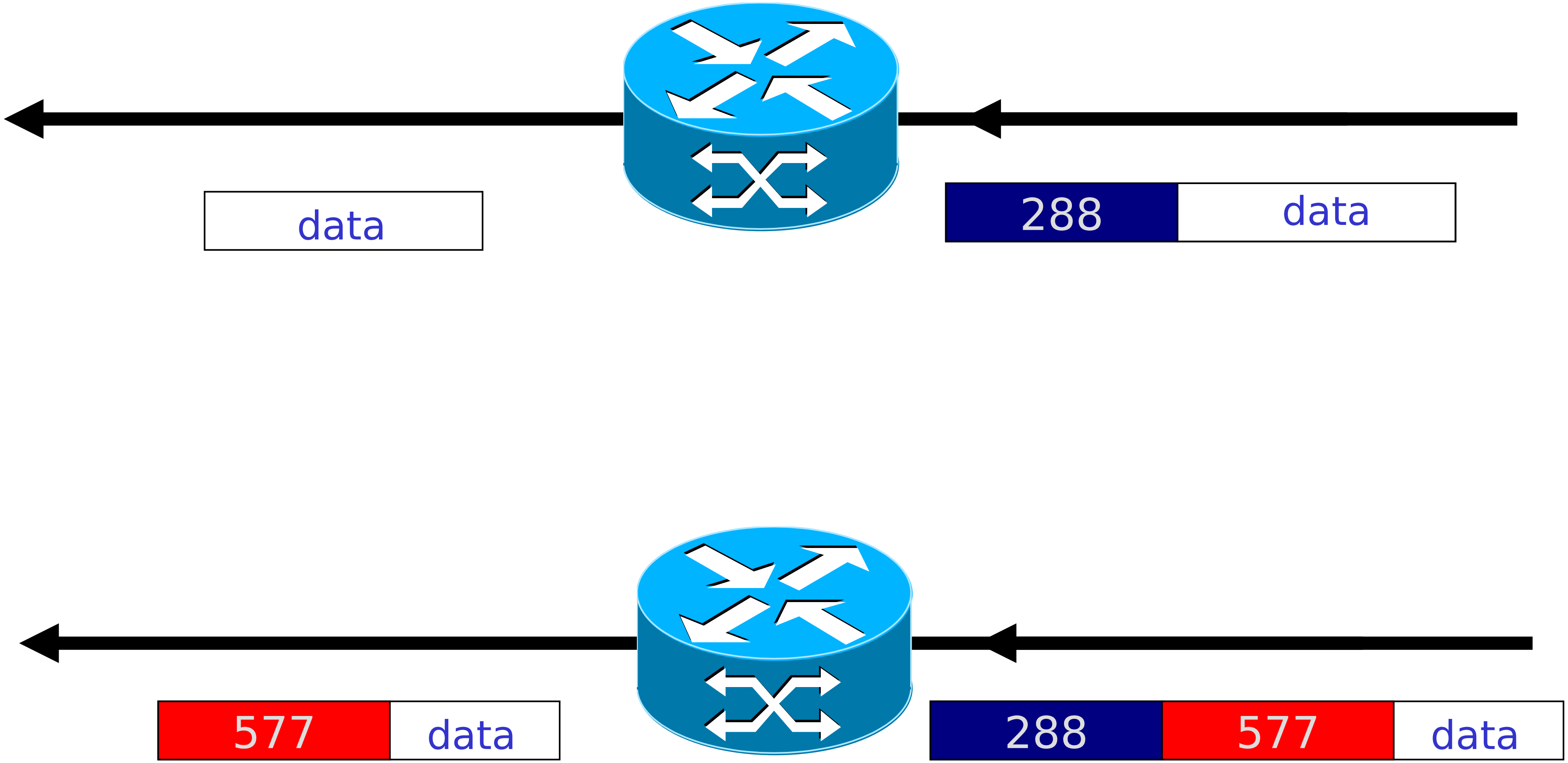
- Labels are arranged in a stack to support multiple services:
  - Inner labels are used to designate services, FECs, etc.
  - Outer label is used to switch the packets in MPLS core.
- Bottom of Stack (S) bit is set to one for the last entry in the label stack (i.e., for the bottom of the stack), and zero for all other labels.

Forwarding via Label Swapping

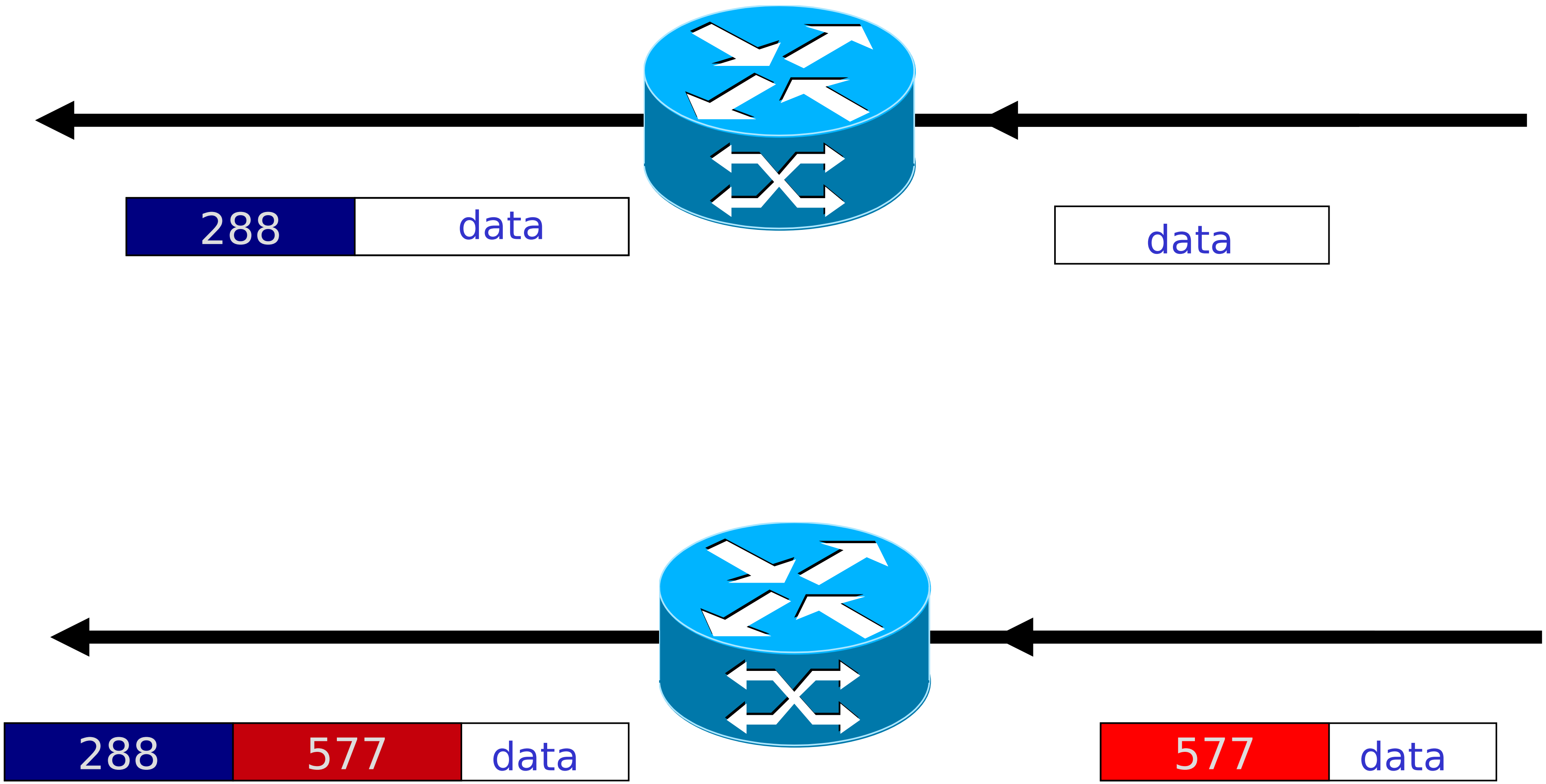


Labels are short, fixed-length values.

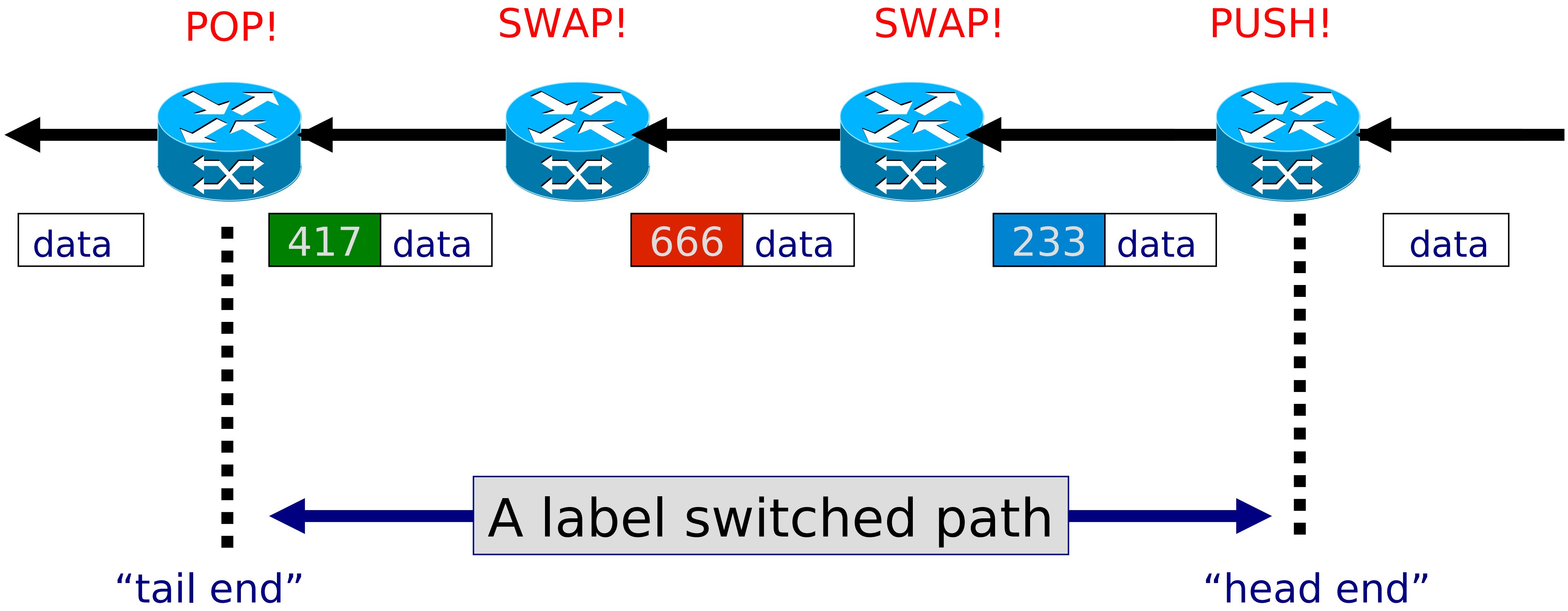
Popping Labels



Pushing Labels

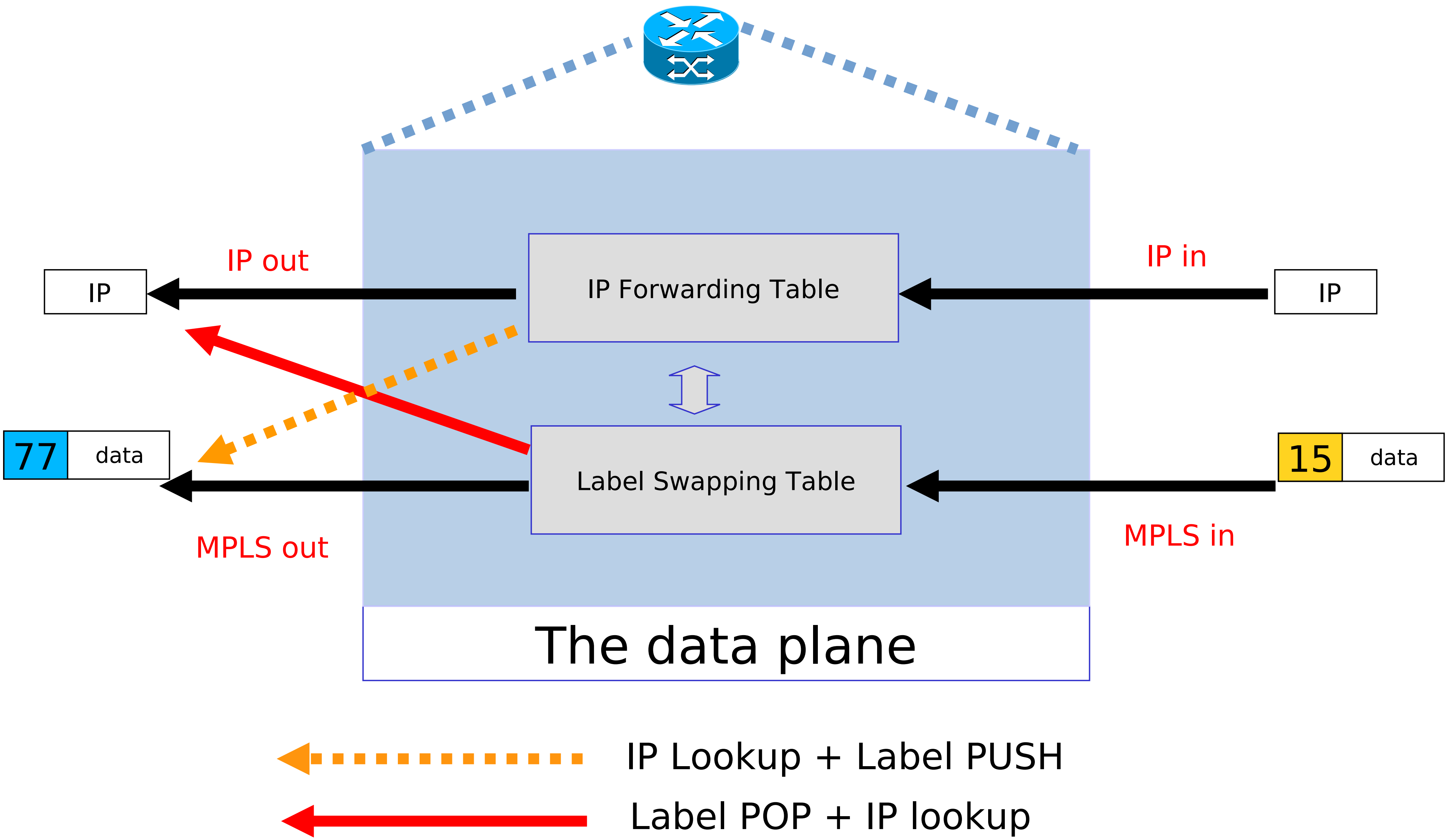


A Label Switched Path (LSP)

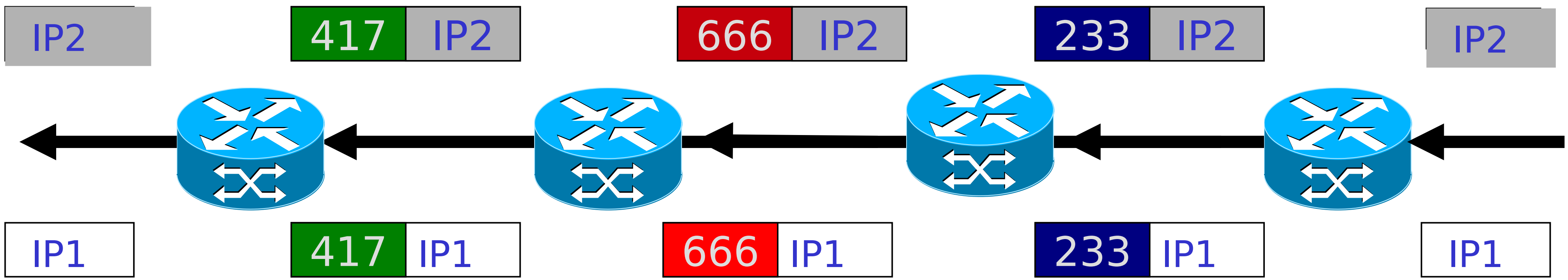


Often called an MPLS tunnel: payload headers are not
Inspected inside of an LSP. Payload could be MPLS …

Label Switched Router



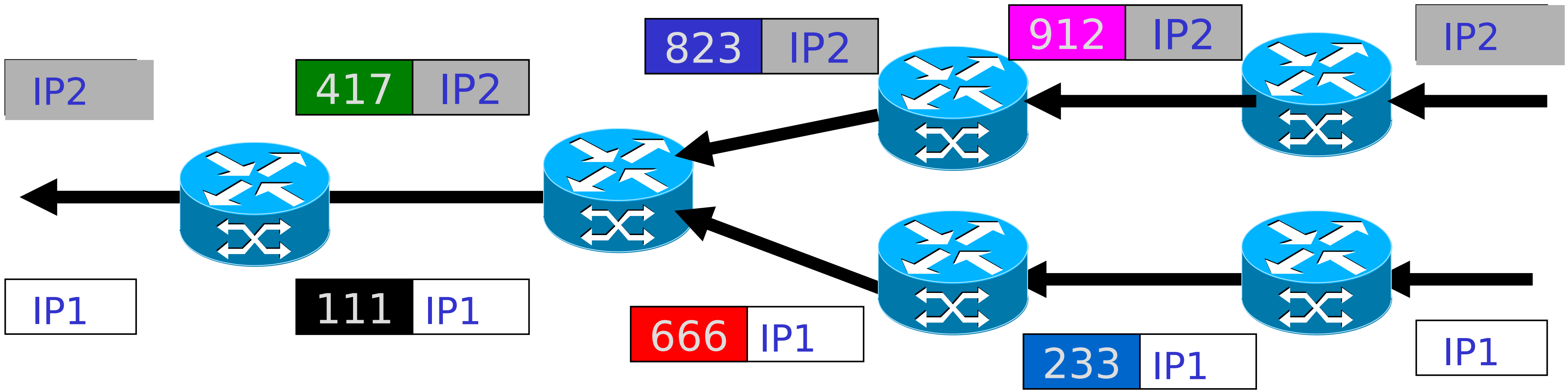
Forwarding Equivalence Class (FEC)

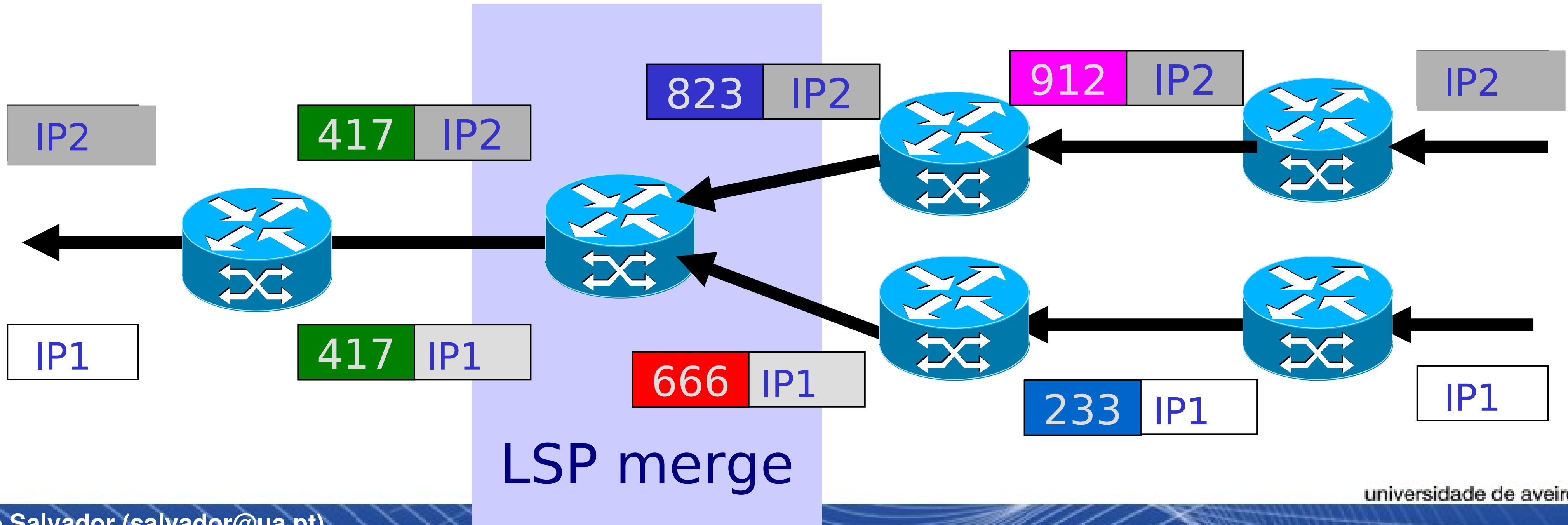


Packets IP1 and IP2 are forwarded in the same way --- they are in the same FEC.

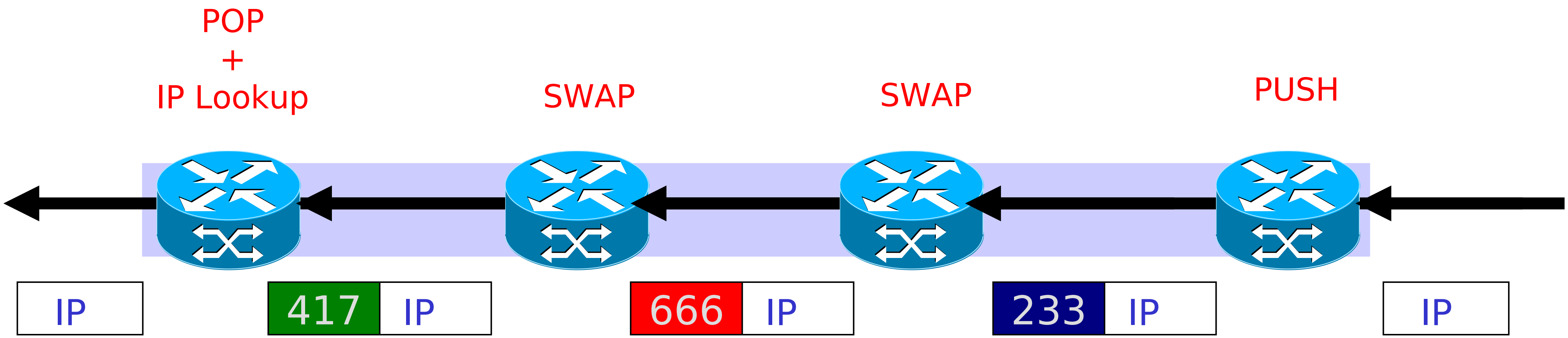
Network layer headers are not inspected inside an MPLS LSP. This means that inside of the tunnel the LSRs do not need full IP forwarding table.

LSP Merge

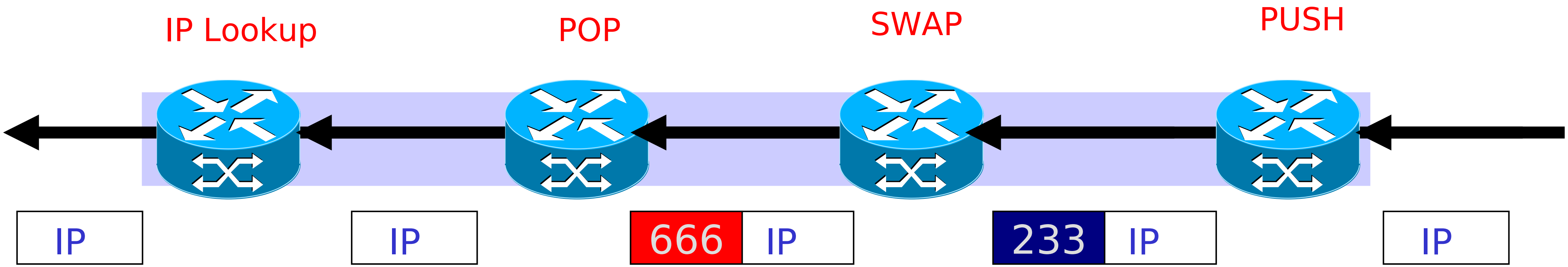

IP2
417 IP2
823 IP2
912 IP2
IP2
IP1
111 IP1
666 IP1
233 IP1
IP1


IP2
417 IP2
823 IP2
912 IP2
IP2
IP1
417 IP1
666 IP1
233 IP1
IP1
LSP merge

Penultimate Hop Popping (PHP)

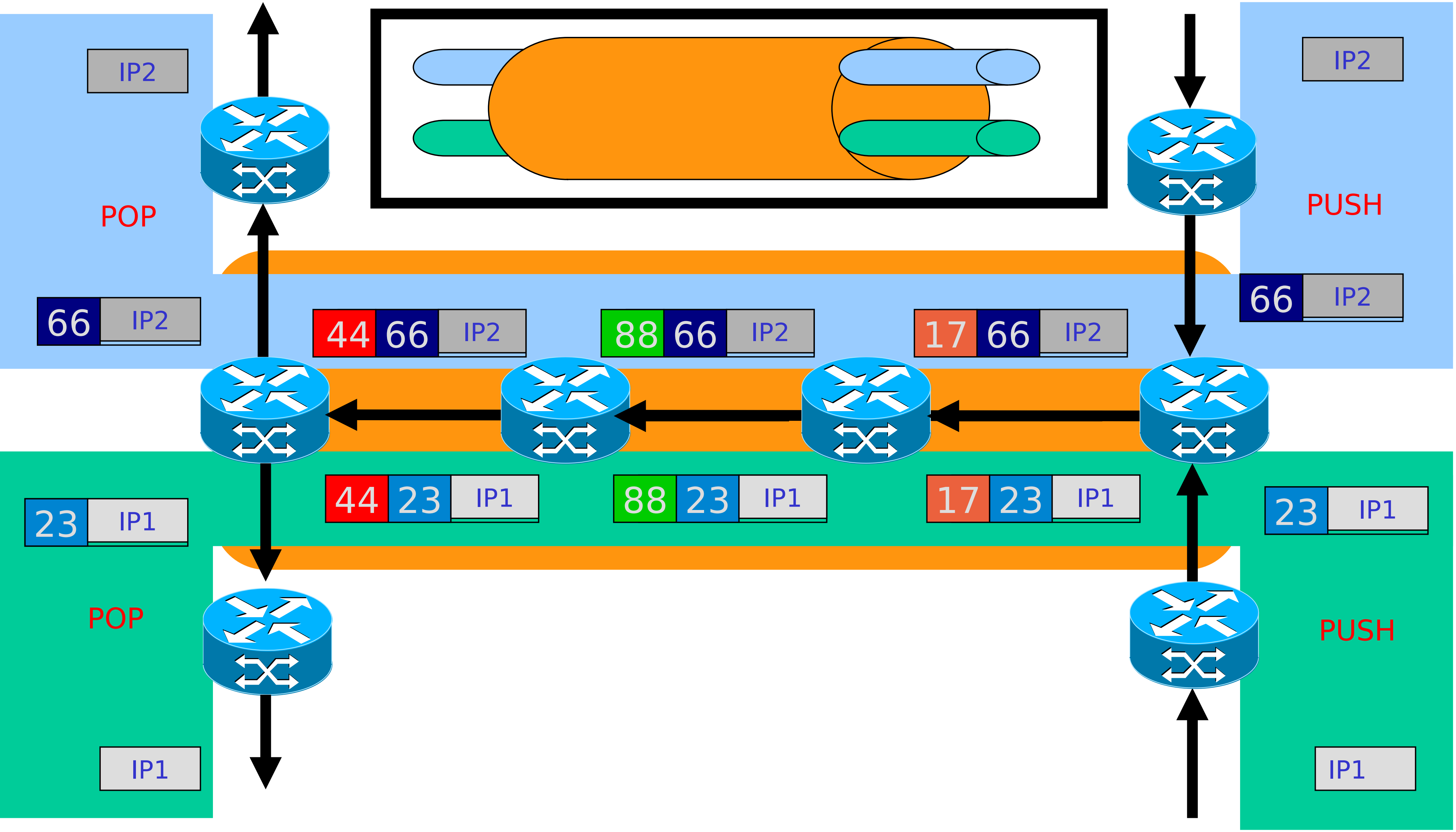


Without PHP



With PHP - Reduces Label Edge Router load

LSP Hierarchy via Label Stacking



Label Distribution Protocols

- Unconstrained routing
  - Label Distribution Protocol (LDP).
  - Path is chosen based on IGP shortest path.
- Constrained routing
  - Constrained by explicit path definition and/or performance requirements (e.g., available bandwidth).
  - Resource Reservation Protocol with Traffic Engineering (RSVP-TE).
    - Evolution of RSVP to support traffic engineering and label distribution.
  - Constrained based Routing LDP (CR-LDP).
    - Evolution of LDP to support constrained routing.
    - Deprecated!
- MPLS VPN scope
  - MP-BGP using address family VPN IPv4 and family specific MP_REACH_NLRI attribute.

Label Distribution Protocol (LDP)

RFC 5036: LDP Specification. (10/2007)

- Dynamic distribution of label binding information.
- LSR discovery.
- Reliable transport with TCP.
- Incremental maintenance of label swapping tables (only deltas are exchanged).
- Designed to be extensible with Type-Length-Value (TLV) coding of messages.
- Modes of behavior that are negotiated during session initialization
  - Label distribution control (ordered or independent).
  - Label retention (liberal or conservative).
  - Label advertisement (unsolicited or on-demand).

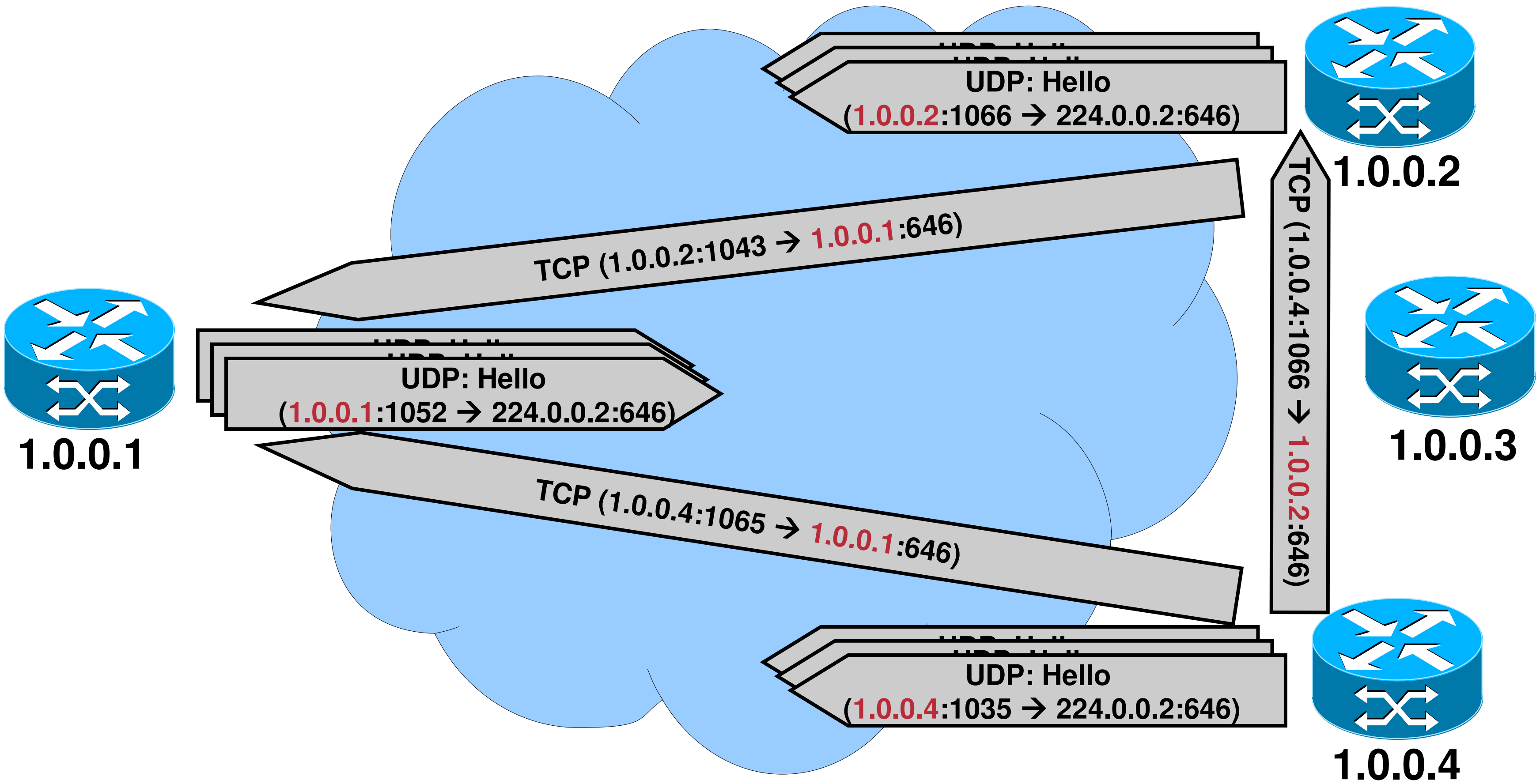
LDP Messages

- Discovery messages
  - Announce and maintain the presence of an LSR in a network.
  - **Hello Messages** (UDP) sent to “all-routers” multicast address.
  - Once neighbor is discovered, a LDP session is established over TCP.
- Session messages
  - Establish (**Initialization Message**) and maintain (**KeepAlive Message**) sessions between LDP peers.
- Advertisement messages
  - When a new LDP session is initialized and before sending label information an LSR advertises its interface addresses with one or more **Address Messages**.
  - An LSR withdraw previously advertised interface addresses with **Address Withdraw Messages**.
  - Create, change, and delete label mappings for FECs.
    - **Label Mapping, Label Request, Label Abort Request, Label Withdraw,** and **Label Release Messages.**
- Notification messages
  - Provide advisory information and to signal error information.

LDP Session Establishment

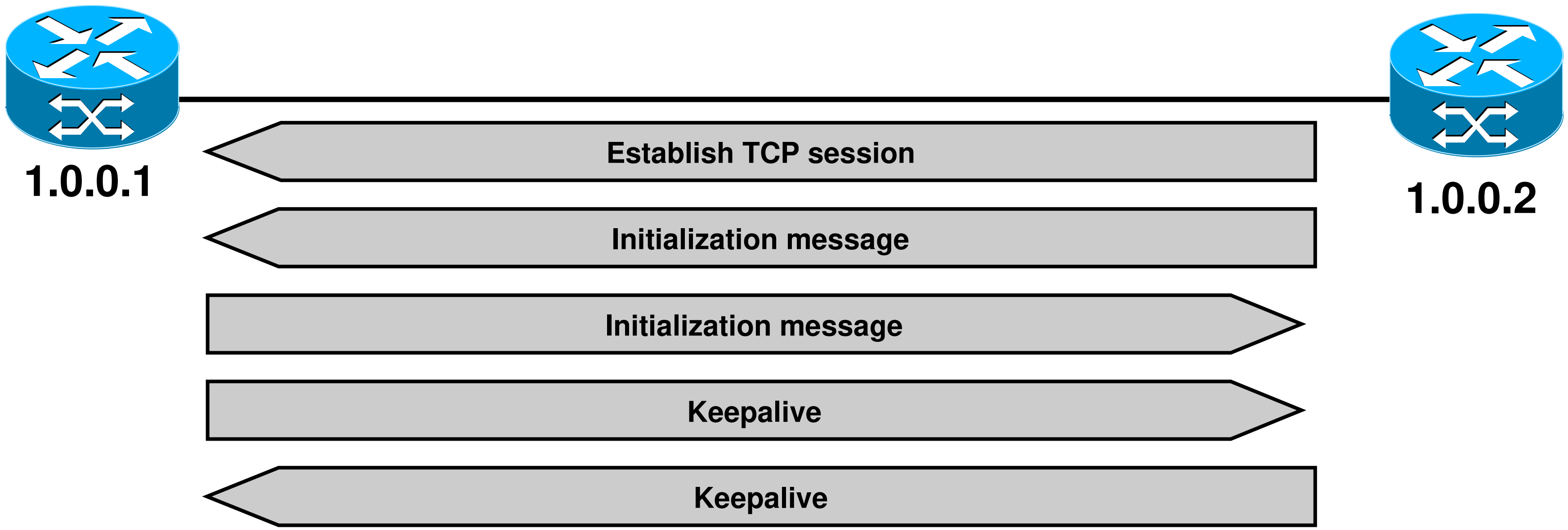
- Hello messages (UDP) are periodically sent on all interfaces enabled for MPLS to a “all-routers” multicast address (224.0.0.2).
- If there is another router on that interface it will respond by trying to establish a LDP/TCP session with the source of the hello messages.
- Both TCP and UDP messages use well-known LDP port number 646.

LDP Neighbor Discovery



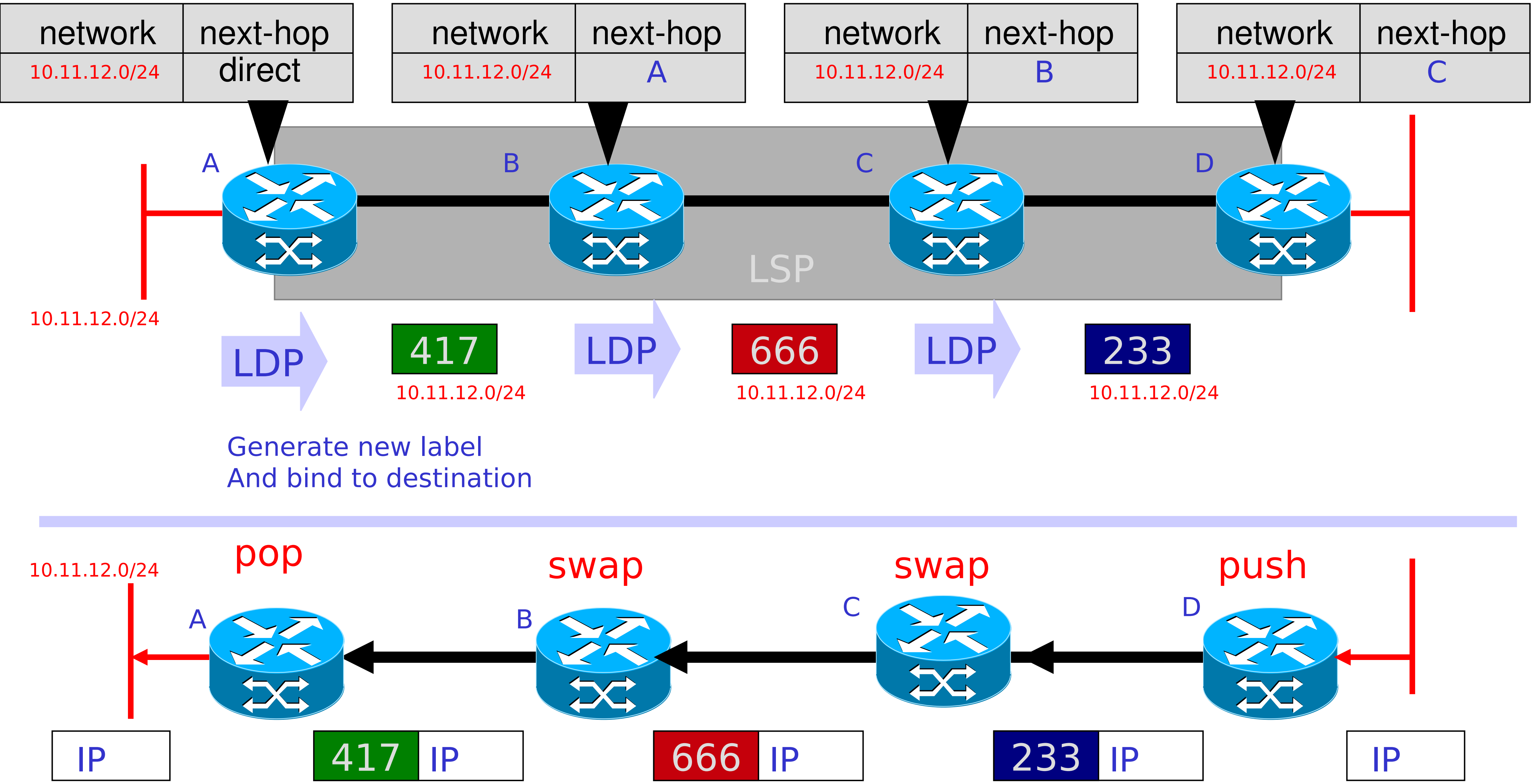
- LDP Session is started by the router with higher IP address.

LDP Session Negotiation



- Peers first exchange initialization messages.
- The session is ready to exchange label mappings after receiving the first keepalive.
  - Keepalives are resent periodically to maintain the LDP/TCP session active.

LDP and Hop-by-Hop routing



Constraint Based Routing

Basic components

1. Specify path constraints
2. Extend topology database to include resource and constraint information
3. Find paths that do not violate constraints and optimize some metric
4. Signal to reserve resources along path
5. Set up LSP along path (with explicit route)
6. Map ingress traffic to the appropriate LSPs

Extend Link State Protocols (IS-IS, OSPF)

Problem here: OSPF areas hide information for scalability. So these extensions work best only within an area…

Extend RSVP or LDP or both!

Problem here: what is the "correct" resource model for IP services?

Note: (3) could be offline,
or online (perhaps an extension to OSPF)

Resource Reservation + Label Distribution

Two competing approaches:

Add label distribution and explicit routes to a resource reservation protocol

RSVP-TE

+

RSVP

CR-LDP

+

LDP

Add explicit routes and resource reservation to a label distribution protocol

CR-LDP
RFC 3212: Constraint-Based LSP Setup using LDP

RSVP-TE:
RFC 3209: RSVP-TE: Extensions to RSVP for LSP Tunnels

As of February 2003, the IETF MPLS working group deprecated CR-LDP and decided to focus purely on RSVP-TE.

RFC 3468: The Multiprotocol Label Switching (MPLS) Working Group decision on MPLS signaling protocols

Resource Reservation Protocol with Traffic Engineering (RSVP-TE)

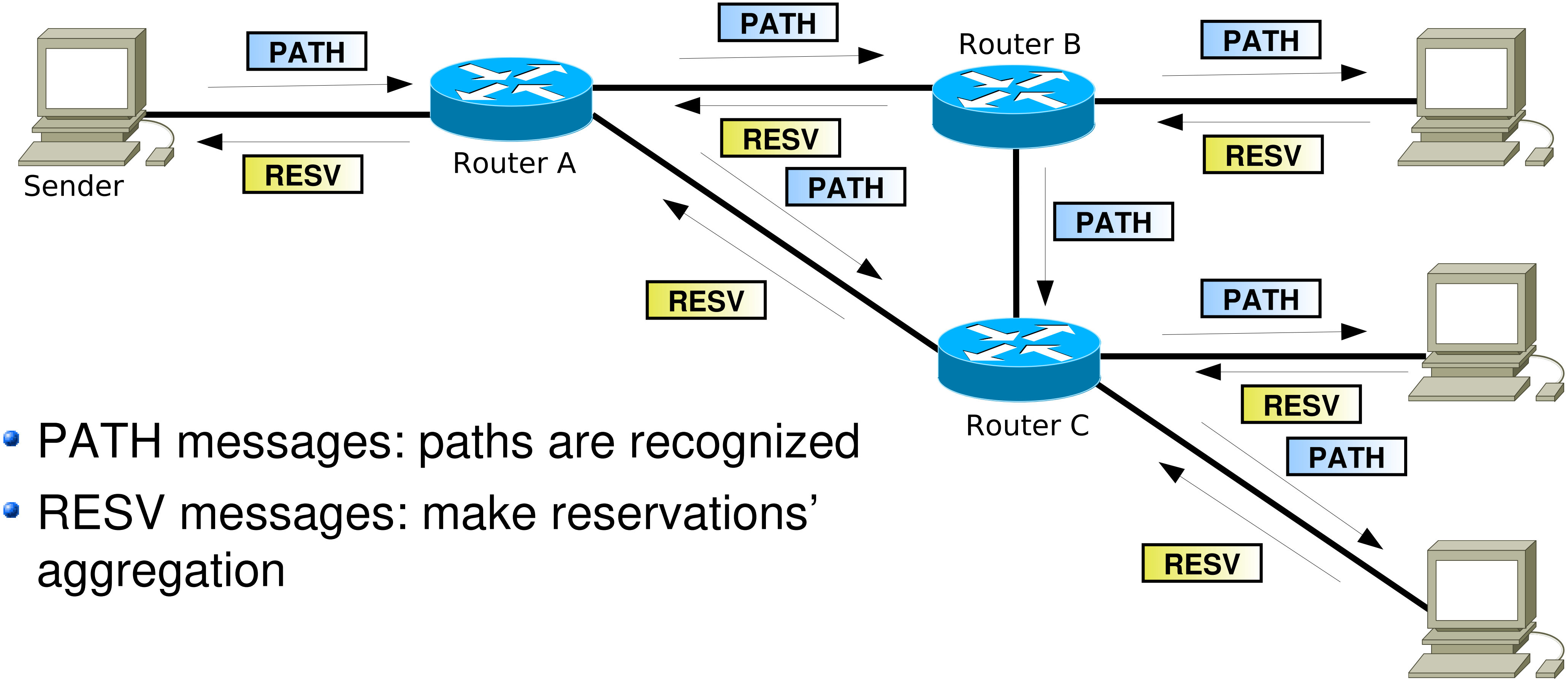
RFC 3209: RSVP-TE: Extensions to RSVP for LSP Tunnels. (12/2001)
RFC 5151: Resource Reservation Protocol-Traffic Engineering (RSVP-TE) Extensions. (2/2008)

- Evolution of RSVP.
- To map traffic flows onto the physical network topology through label switched paths, requires resource and constraint network information.
  - Provided by Extend Link State Protocols (IS-IS or OSPF with TE extensions).
    - RFC 3630: Traffic Engineering (TE) Extensions to OSPF Version 2. (9/2003)
    - RFC 5305: IS-IS Extensions for Traffic Engineering. (10/2008)

ReSerVation Protocol (RSVP)

- The resource ReSerVation Protocol (RSVP) was developed to communicate resource needs between hosts and network devices (RFC 2205-2215)
- RSVP allows:
  - The source do describe the characteristics of the IP packets flow.
  - Destinations to describe the reservation they want.
  - Routers to know how to process the packets flow in order to fulfill the requested reservation.
- Encapsulated on IP; protocol type = 46 (0x2E)
- Signaling is based on the exchange of PATH and RESV messages.
  - PATH announces the traffic characteristics at the sender.
  - RESV achieves reservations that were initiated by the receivers.
  - If the reservation is not possible, a RESV ERR message is sent.
- The routers reservation states have to be periodically refreshed (soft states).
- RSVP defines a "Session" to be a data flow with a particular destination and transport-layer protocol.
  - RSVP treats each session independently.

RSVP Signaling

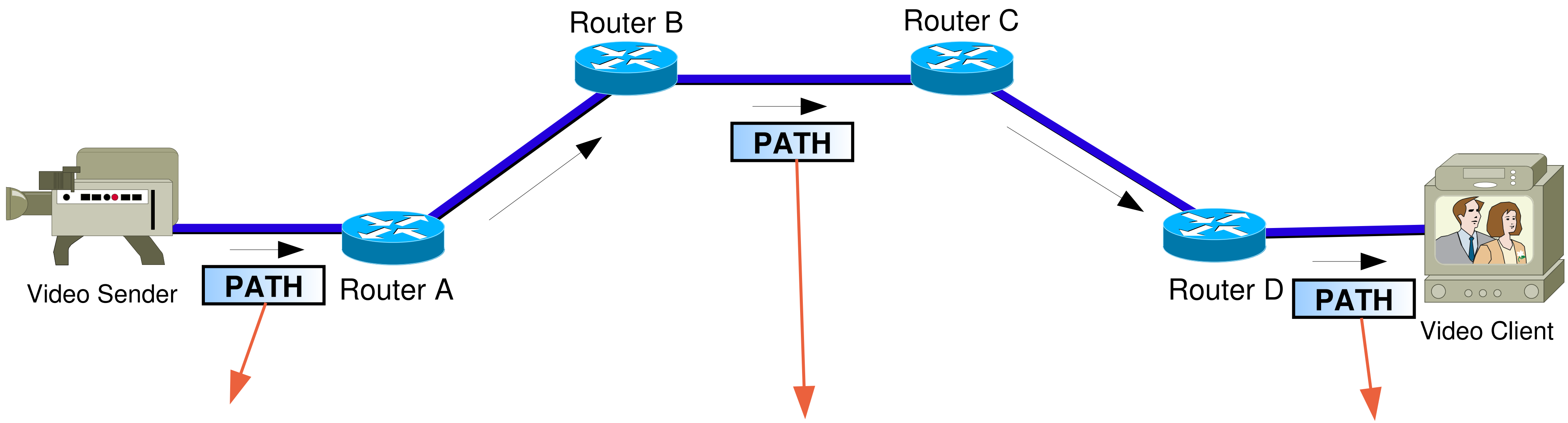


- PATH messages: paths are recognized
- RESV messages: make reservations' aggregation

RSVP messages

- PATH (*Type* = 0x01)
  - Tspec ("flow traffic specification"): contains the parameters that describe the traffic source based on the "Token Bucket" model
- RESV (*Type* = 0x02)
  - Tspec: the same that was received on the PATH message
  - FilterSpec ("*filter specification*"): contains the flow descriptor that enables routers to identify packets belonging to this reservation (source address, destination address, protocol type, source port number, destination port number, any combination of these parameters)
  - Rspec ("*flow reservation specification*"): contains the parameters describing the reservation that the receiver wants to become supported
    - Rspec is specified if the receiver wants a service of the "*guaranteed service*" type; when it is not specified, it means that the receiver wants a service of the "*controlled load*" type

RSVP PATH (Example)

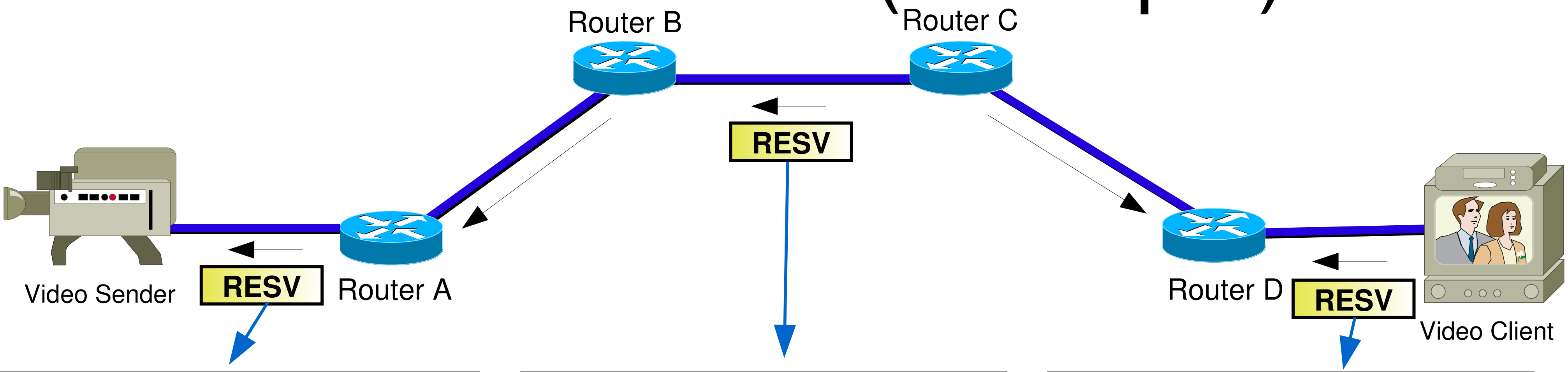


Vs.: 4	iHL: 5	Service	Total Length: 60	
Identification			Flg	Fragment Offset
Time to Live		Protocol: 46	Header Checksum	
Source Address: **Video Server**				
Destination Address: **Video Client**				
1	0	Type: 1	Checksum	
Send_TTL		0	Message Length: 40	
SESSION Length.: 12			Class Nº: 1	Class Type: 1
Destination Address: **Video Client**				
Protocol ID		Flags	Destination port	
RSVP_HOP Length. : 12			Class Nº: 3	Class Type: 1
Last Hop Address: **Video Server**				
Logical Interface Handle of the last node (LIH)				
TIME_VALUES Length: 8			Class Nº: 5	Class Type: 1
Update Period (ms)				

Vs.: 4	iHL: 5	Service	Total Length: 60	
Identification			Flg	Fragment Offset
Time to Live		Protocol: 46	Header Checksum	
Source Address: **Video Server**				
Destination Address: **Video Client**				
1	0	Type: 1	Checksum	
Send_TTL		0	Message Length: 40	
SESSION Length: 12			Class Nº: 1	Class Type: 1
Destination Address: **Video Client**				
Protocol ID		Flags	Destination Port	
RSVP_HOP Length: 12			Class Nº: 3	Class Type: 1
Last Hop Address: **Router B**				
Logical Interface Handle of the last node (LIH)				
TIME_VALUES Length: 8			Class Nº: 5	Class Type: 1
Update Period (ms)				

Vs.: 4	iHL: 5	Service	Total Length: 60	
Identification			Flg	Fragment Offset
Time to Live		Protocol: 46	Header Checksum	
Source Address: **Video Server**				
Destination Address: **Video Client**				
1	0	Type: 1	Checksum	
Send_TTL		0	Message Length: 40	
SESSION Length: 12			Class Nº: 1	Class Type: 1
Destination Address: **Video Client**				
Protocol ID		Flags	Destination Port	
RSVP_HOP Length: 12			Class Nº: 3	Class Type: 1
Last Hop Address: **Router D**				
Logical Interface Handle of the last node (LIH)				
TIME_VALUES Length: 8			Class Nº: 5	Class Type: 1
Update Period (ms)				

RSVP RESV (Example)



Vs.: 4	iHL: 5	Service	Total Length	
Identification			Flg	Fragment Offset
Time to Live		Protocol: 46	Header Checksum	
Source Address: **Router A**				
Destination Address: **Video Server**				
1	0	Type: 2	Checksum	
Send_TTL		0	Message Length	
SESSION Length: 12			Class Nº: 1	Class Type: 1
Destination Address: **Video Client**				
Protocol Id		Flags	Destination protocol port	
RSVP_HOP Length: 12			Class Nº: 3	Class Type: 1
Address of the last node: **Router A**				
Logical Interface Handle of the last node (LIH)				
TIME_VALUES Length: 8			Class Nº: 5	Class Type: 1
Update period (ms)				
STYLE Object Length : 8			Class Nº: 8	Class Type: 1
Flags		Style Option Vector: 0x00000A (FF)		
FLOWSPEC Length			Class Nº: 9	Class Type
FLOWSPEC object contents				
FILTER_SPEC Length: 12			Class Nº: 10	Class Type: 1
Source Address: Video Server				
Reserved		Reserved	Source protocol port	

Vs.: 4	iHL: 5	Service	Total Length	
Identification			Flg	Fragment Offset
Time to Live		Protocol: 46	Header Checksum	
Source Address: **Router C**				
Destination Address: **Router B**				
1	0	Type: 2	Checksum	
Send_TTL		0	Message Length	
SESSION Length: 12			Class Nº: 1	Class Type: 1
Destination Address: **Video Client**				
Protocol Id		Flags	Destination protocol port	
RSVP_HOP Length: 12			Class Nº: 3	Class Type: 1
Address of the last node: **Router C**				
Logical Interface Handle of the last node (LIH)				
TIME_VALUES Length: 8			Class Nº: 5	Class Type: 1
Update period (ms)				
STYLE Object Length : 8			Class Nº: 8	Class Type: 1
Flags		Style Option Vector: 0x00000A (FF)		
FLOWSPEC Length			Class Nº: 9	Class Type
FLOWSPEC object contents				
FILTER_SPEC Length: 12			Class Nº: 10	Class Type: 1
Source Address: Video Server				
Reserved		Reserved	Source protocol port	

Vs.: 4	iHL: 5	Service	Total Length	
Identification			Flg	Fragment Offset
Time to Live		Protocol: 46	Header Checksum	
Source Address: **Video Client**				
Destination Address: **R outer D**				
1	0	Type: 2	Checksum	
Send_TTL		0	Message Length	
SESSION Length: 12			Class Nº: 1	Class Type: 1
Destination Address: **Video Client**				
Protocol Id		Flags	Destination protocol port	
RSVP_HOP Length: 12			Class Nº: 3	Class Type: 1
Address of the last node: **Video Client**				
Logical Interface Handle of the last node (LIH)				
TIME_VALUES Length: 8			Class Nº: 5	Class Type: 1
Update period (ms)				
STYLE Object Length : 8			Class Nº: 8	Class Type: 1
Flags		Style Option Vector: 0x00000A (FF)		
FLOWSPEC Length			Class Nº: 9	Class Type
FLOWSPEC object contents				
FILTER_SPEC Length: 12			Class Nº: 10	Class Type: 1
Source Address: Video Server				
Reserved		Reserved	Source protocol port	

Extensions to RSVP for LSP Tunnels

- The SENDER_TEMPLATE (or FILTER_SPEC) object together with the SESSION object uniquely identifies an LSP tunnel (flow).
- LSP Tunnel related new objects
  - Explicit Route
    - Carried in PATH and contains a series of variable-length data items called sub-objects.
    - Possible sub-objects: IPv4 prefix, IPv6 prefix, and autonomous system number.
  - Label Request
    - Carried in PATH requesting a label for a specific tunnel/flow.
    - Request cab be without label range, with an ATM label range, or with an Frame Relay label range.
  - Label
    - Carried in RESV messages and contain a single label for a specific tunnel/flow.
  - Record Route
    - Carried in PATH and RESV, used to collect detailed path information and useful for loop detection and diagnostics.
  - Session Attribute
    - Carried in PATH, used to define the type and name of the session/tunnel/flow, also used to define priority values.
- LSP Tunnel related new object types
  - Session object new types
    - LSP_TUNNEL_IPv4 and LSP_TUNNEL_IPv6
  - Sender Template object new types
    - LSP_TUNNEL_IPv4 and LSP_TUNNEL_IPv6
  - Filter Specification object new types
    - LSP_TUNNEL_IPv4 and LSP_TUNNEL_IPv6

RSVP-TE PATH and RESV (example)

```
Resource ReserVation Protocol (RSVP): PATH Message. SESSION: IPv4-LSP
▷ RSVP Header. PATH Message.
▷ SESSION: IPv4-LSP, Destination 192.2.0.11, Tunnel ID 2, Ext ID c002000a.
▷ HOP: IPv4, 200.10.2.10
▶ TIME VALUES: 30000 ms
▷ EXPLICIT ROUTE: IPv4 200.10.2.2, IPv4 200.2.11.2, IPv4 200.2.11.11,
▷ LABEL REQUEST: Basic: L3PID: IP (0x0800)
▷ SESSION ATTRIBUTE: SetupPrio 7, HoldPrio 7, SE Style,  [RA_t2]
▷ SENDER TEMPLATE: IPv4-LSP, Tunnel Source: 192.2.0.10, LSP ID: 8.
▷ SENDER TSPEC: IntServ, Token Bucket, 18750 bytes/sec.
▷ ADSPEC
```

```
▽ Resource ReserVation Protocol (RSVP): RESV Message. SESSION: IPv4-LSP
  ▷ RSVP Header. RESV Message.
  ▷ SESSION: IPv4-LSP, Destination 192.2.0.11, Tunnel ID 2, Ext ID c002000a.
  ▷ HOP: IPv4, 200.10.2.2
  ▷ TIME VALUES: 30000 ms
  ▷ STYLE: Shared-Explicit (18)
  ▷ FLOWSPEC: Controlled Load: Token Bucket, 18750 bytes/sec.
  ▷ FILTERSPEC: IPv4-LSP, Tunnel Source: 192.2.0.10, LSP ID: 8.
  ▷ LABEL: 19
```

Traffic Engineering Extensions to OSPF

- RFC 3630: Traffic Engineering (TE) Extensions to OSPF Version 2. (9/2003)
- OSPF Traffic Engineering (TE) extensions are used to advertise TE Link State Advertisements (TE-LSAs) containing information about TE-enabled links.
  - Traffic Engineering LSA is a type 10 Opaque LSAs, which have an area flooding scope.
- TE-LSA contains one of two possible top-level Type Length Values (TLVs)
    - **Router Address**: specifies a stable IP address of the advertising router that is always reachable if there is any connectivity to it; this is typically implemented as a "loopback address";
    - **Link**: describes a single link with a a set of sub-TLVs (Link type, Link ID, Local interface IP address, Remote interface IP address, Traffic engineering metric, Maximum bandwidth, Maximum reservable bandwidth, Unreserved bandwidth, and Administrative group.
- The information made available by these extensions can be used to build an extended link state database
  - Can be used to:
    - Monitoring the extended link attributes;
    - Local constraint-based source routing;
    - Global traffic engineering.

OSPF-TE Opaque Area Database

- Router Address TLV

```
LS age: 250
  Options: (No TOS-capability, DC)
  LS Type: Opaque Area Link
  Link State ID: 1.0.0.0
  Opaque Type: 1
  Opaque ID: 0
  Advertising Router: 192.2.0.2
  LS Seq Number: 80000001
  Checksum: 0xDACD
  Length: 28
  Fragment number : 0

    MPLS TE router ID : 192.2.0.2
    Number of Links : 0
```

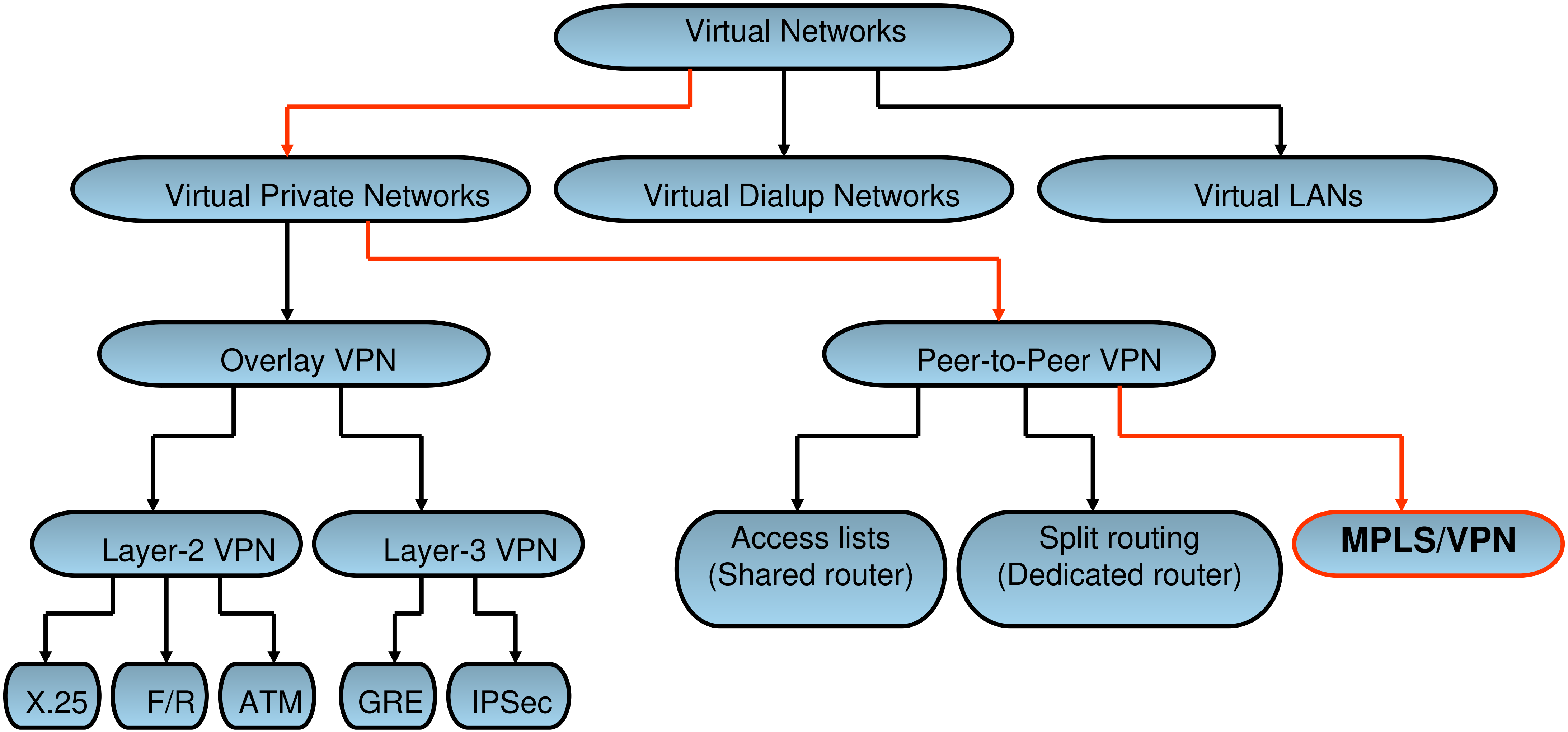
- Link TLV

```
LS age: 246
 Options: (No TOS-capability, DC)
 LS Type: Opaque Area Link
 Link State ID: 1.0.0.2
 Opaque Type: 1
 Opaque ID: 2
 Advertising Router: 192.2.0.2
 LS Seq Number: 80000001
 Checksum: 0x2FBB
 Length: 124
 Fragment number : 2

   Link connected to Broadcast network
     Link ID : 200.1.2.2
     Interface Address : 200.1.2.2
     Admin Metric : 1
     Maximum bandwidth : 12500000
     Maximum reservable bandwidth : 64000
     Number of Priority : 8
     Priority 0 : 64000        Priority 1 : 64000
     Priority 2 : 64000        Priority 3 : 64000
     Priority 4 : 64000        Priority 5 : 64000
     Priority 6 : 64000        Priority 7 : 64000
     Affinity Bit : 0x0
     IGP Metric : 1
   Number of Links : 1
```

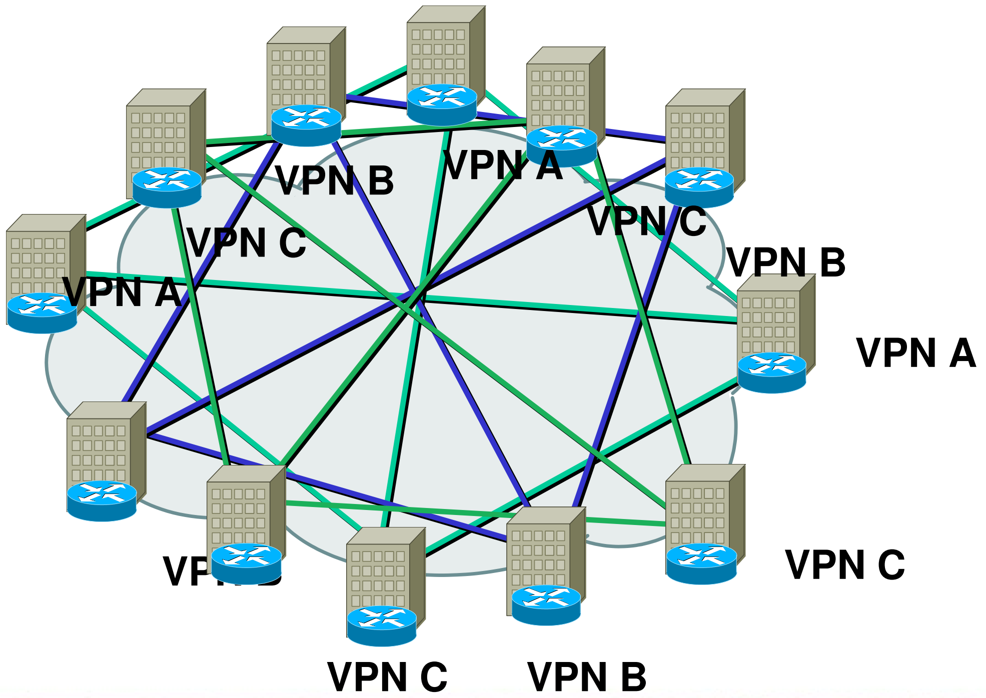
MPLS Layer 3 VPNs

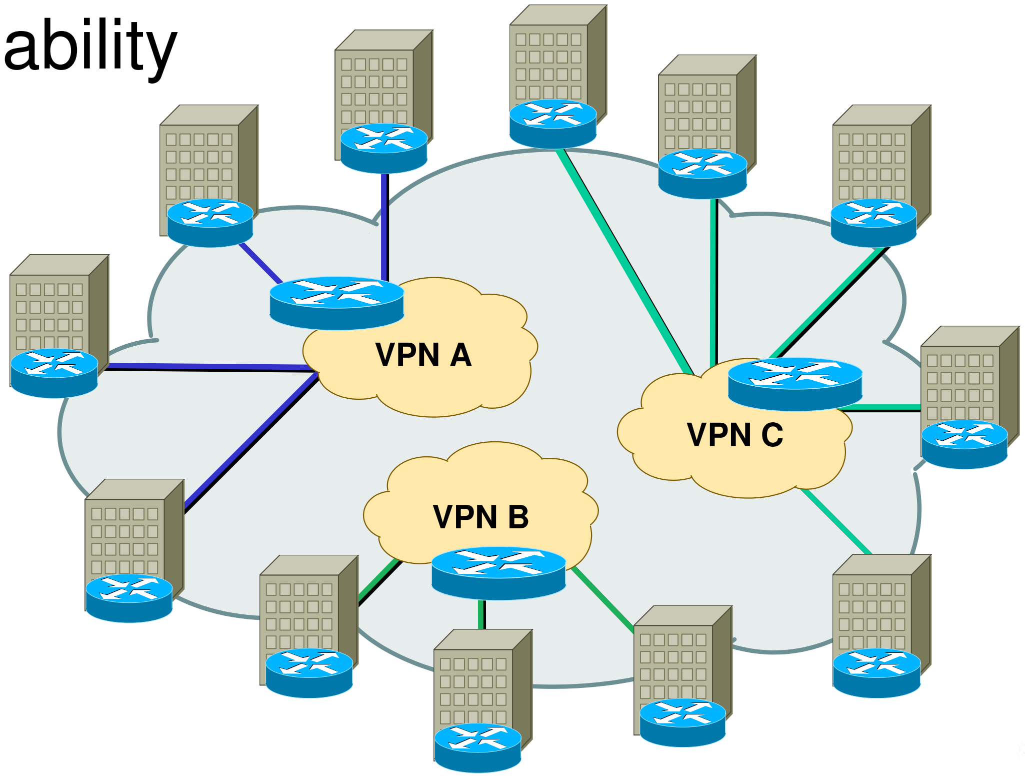
Virtual Network Models



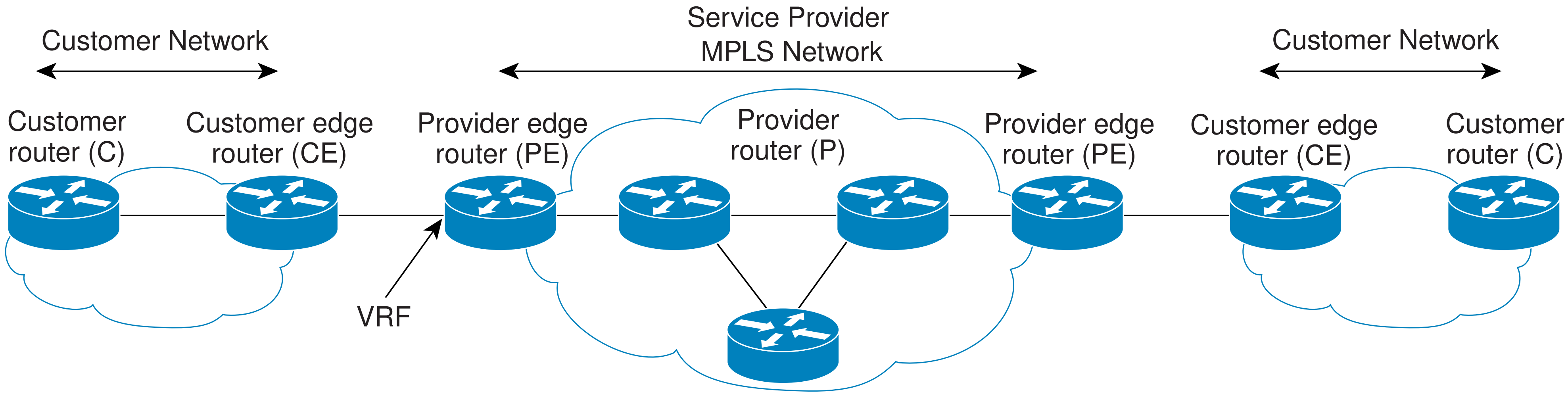
MPLS L3 VPNs using BGP (RFC2547)

- End user perspective
  - Virtual Private IP service.
  - Simple routing – just point default to provider.
  - Full site-site connectivity without the usual drawbacks (routing complexity, scaling, configuration, cost).
- Major benefit for provider – scalability





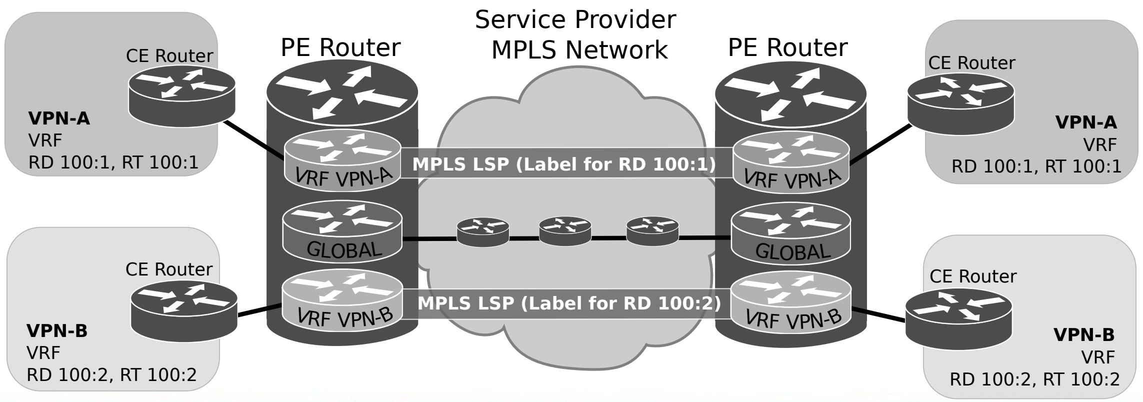
MPLS VPN Terminology



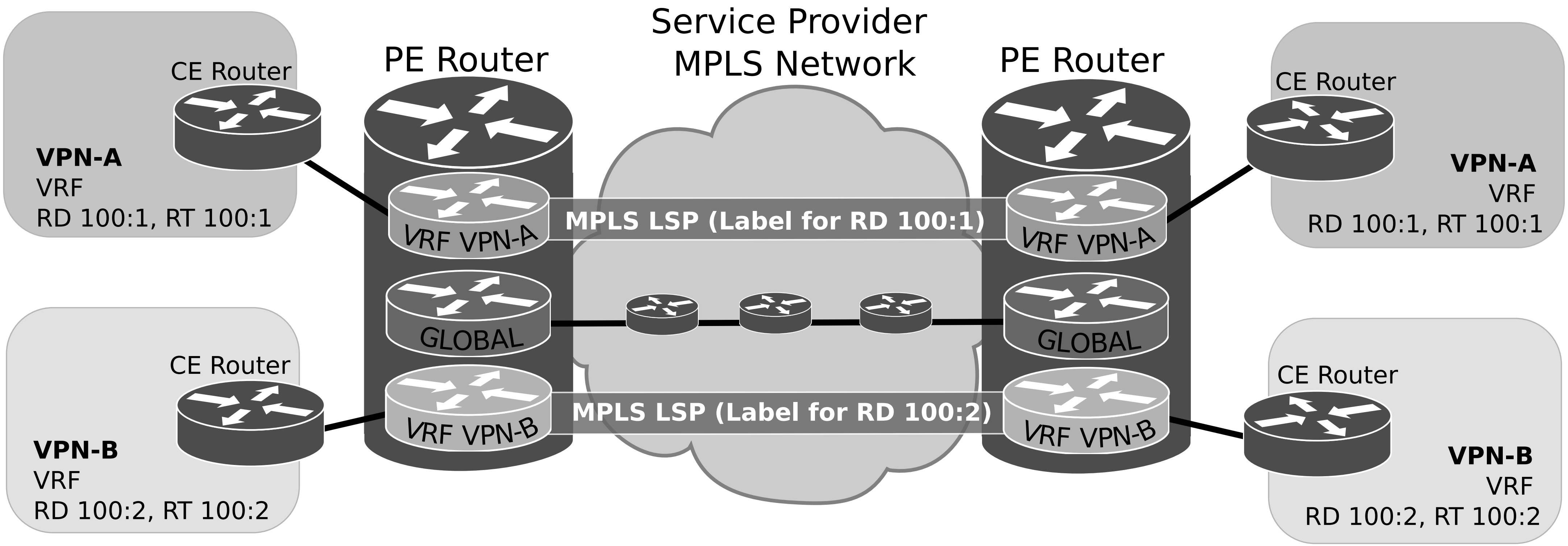
- Customer router (C) is connected only to other customer devices.
- Customer Edge (CE) router peers at Layer 3 to the Provider Edge (PE).
  - The PE-CE Interface runs either a dynamic routing protocol (eBGP, RIPv2, EIGRP, or OSPF) or has static routing (Static, Connected).
- Provider (P) router, resides in the core of the provider network.
  - Participates in the control plane for customer prefixes. The P router is also referred to as a Label Switch Router (LSR), in reference to its primary role in the core of the network, performing label switching/swapping of MPLS traffic.
- Provider Edge (PE) router, sits at the edge of the MPLS SP network.
  - In an MPLS VPN context, separate VRF routing tables are allocated for each user group.
  - Contains a global routing table for routes in the core SP infrastructure.
  - The PE is sometimes referred to as a Label Edge Router (LER) or Edge Label Switch Router (ELSR) in reference to its role at the edge of the MPLS cloud, performing label imposition and disposition.

Virtual Routing and Forwarding (VRF)

- Virtual Routing and Forwarding (VRF) instance, is separate from the global routing table that exists on PE routers.
- PE routers maintain separate routing tables:
  - Global routing table
    - Contains all PE and P routes (perhaps BGP).
    - Populated by the VPN backbone IGP .
  - VRF table
    - Routing and forwarding table associated with one or more directly connected sites (CE routers).
    - VRF is associated with any type of interface, whether logical or physical (e.g. sub/virtual/tunnel) .
    - Interfaces may share the same VRF if the connected sites share the same routing information.
    - Routes are injected into the VRF from the CE-PE routing protocols for that VRF and any MP-BGP announcements that match the defined VRF.



MPLS-VPN & VRF


Service Provider
MPLS Network
PE Router
PE Router
CE Router
VPN-A
VRF
RD 100:1, RT 100:1
CE Router
VPN-B
VRF
RD 100:2, RT 100:2
VRF VPN-A
GLOBAL
VRF VPN-B
MPLS LSP (Label for RD 100:1)
MPLS LSP (Label for RD 100:2)
VRF VPN-A
GLOBAL
VRF VPN-B
CE Router
VPN-A
VRF
RD 100:1, RT 100:1
CE Router
VPN-B
VRF
RD 100:2, RT 100:2

Route Distinguisher

- To differentiate 10.0.0.0/8 in VPN-A from 10.0.0.0/8 in VPN-B.
  - 64-bit quantity.
- Configured as ASN:YY or IPADDR:YY.
  - Almost everybody uses ASN.
- Purely to make a route unique.
  - Unique route is now RD:Ipaddr (96 bits) plus a mask on the IPAddr portion.
  - So customers don't see each others routes.

```
!
ip vrf VPN-A
rd 100:1
route-target export 100:1
route-target import 100:1
```

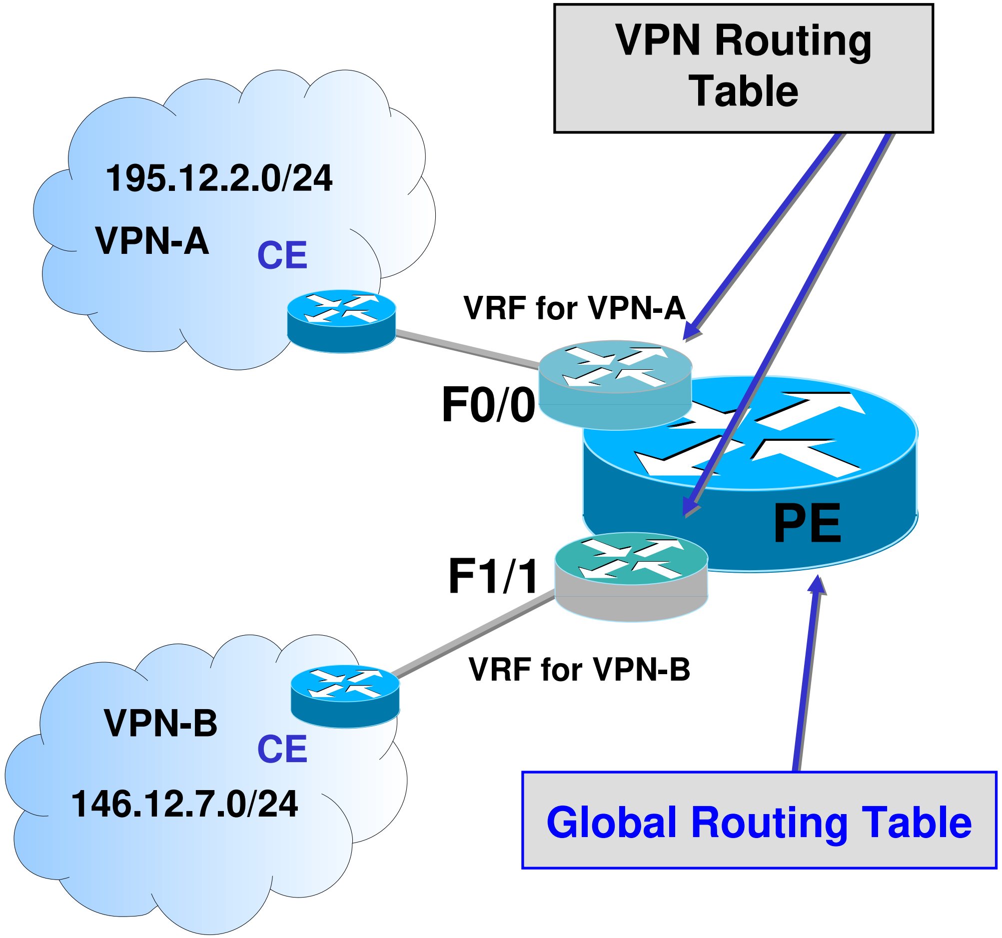
Route Target

- Creates or adds to a list of VPN extended communities used to determine which routes are imported by a VRF.
- To control policy about who sees what routes.
- 64-bit quantity (2 bytes type, 6 bytes value).
- Carried as an extended community.
  - Typically written as ASN:YY.
- Each VRF 'imports' and 'exports' one or more RTs.
  - Exported RTs are carried in VPNv4 BGP.
  - Imported RTs are local to the box.
- A VRF PE that imports an RT installs that route in that VRF routing table.
- Allows the interconnection of different VLAN by importing/exporting other VPN routes (other RTs).
  - (Private) Routes should not conflict!

```
!
ip vrf VPN-A
rd 100:1
route-target export 100:1
route-target import 100:1
```

VRF Interface Definition

- Define a unique VRF for interface F0/0.
- Define a unique VRF for interface F1/1
  - Packets will never go between interfaces F0/0 and F1/1.
  - Unless Each other RT are imported.
- Uses VPNv4 to exchange VRF routing information between PE's.



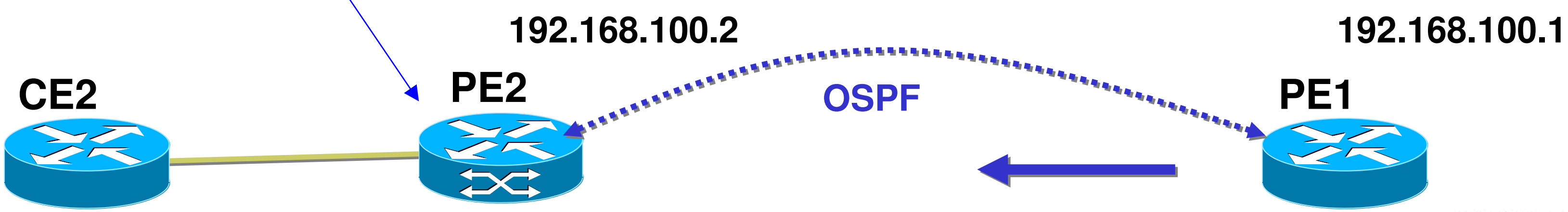
PE Router – Global Routing Table Output

```
PE2#sh ip route

Gateway of last resort is not set

C    192.168.1.0/24 is directly connected, Ethernet0/0
     192.168.100.0/32 is subnetted, 3 subnets
O       192.168.100.1 [110/11] via 192.168.1.1, 00:04:27, Ethernet0/0
C       192.168.100.2 is directly connected, Loopback0
O       192.168.100.3 [110/11] via 192.168.1.3, 00:04:27, Ethernet0/0
```

Routes from PE1's Global Routing Table

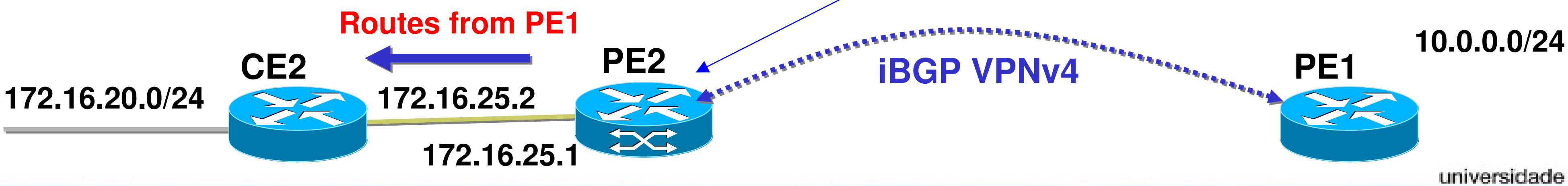


PE Router – VRF Routing Table Output

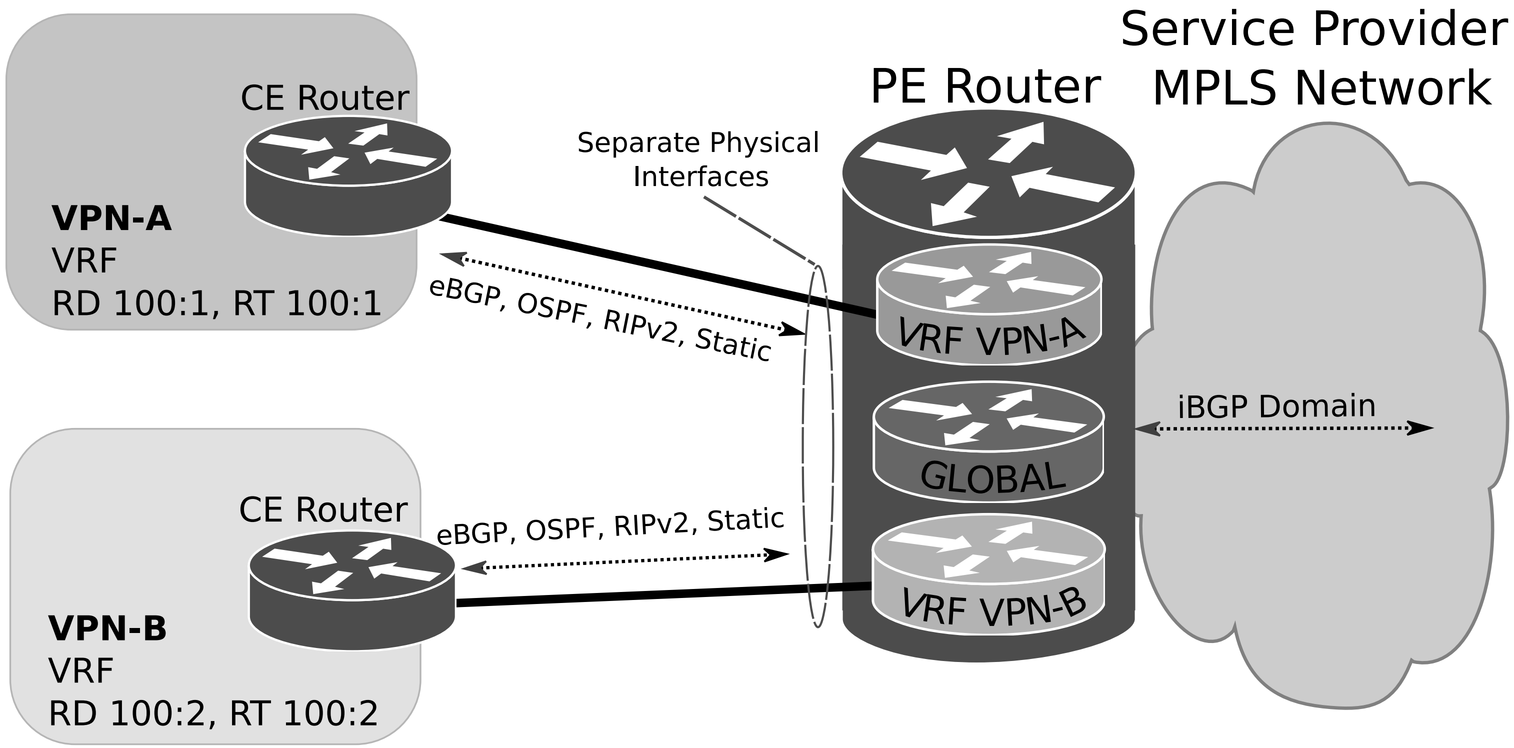
```
PE2#sh ip route vrf RED
Routing Table: RED

Gateway of last resort is 192.168.100.1 to network 0.0.0.0

     172.16.0.0/16 is variably subnetted, 8 subnets, 3 masks
C       172.16.25.0/30 is directly connected, Serial4/0
C       172.16.25.2/32 is directly connected, Serial4/0
B       172.16.20.0/24 [20/0] via 172.16.25.2, 00:07:04
     10.0.0.0/24 is subnetted, 1 subnets
B       10.0.0.0 [200/307200] via 192.168.100.1, 00:06:28
B*   0.0.0.0/0 [200/0] via 192.168.100.1, 00:07:03
```



VRF Route Population



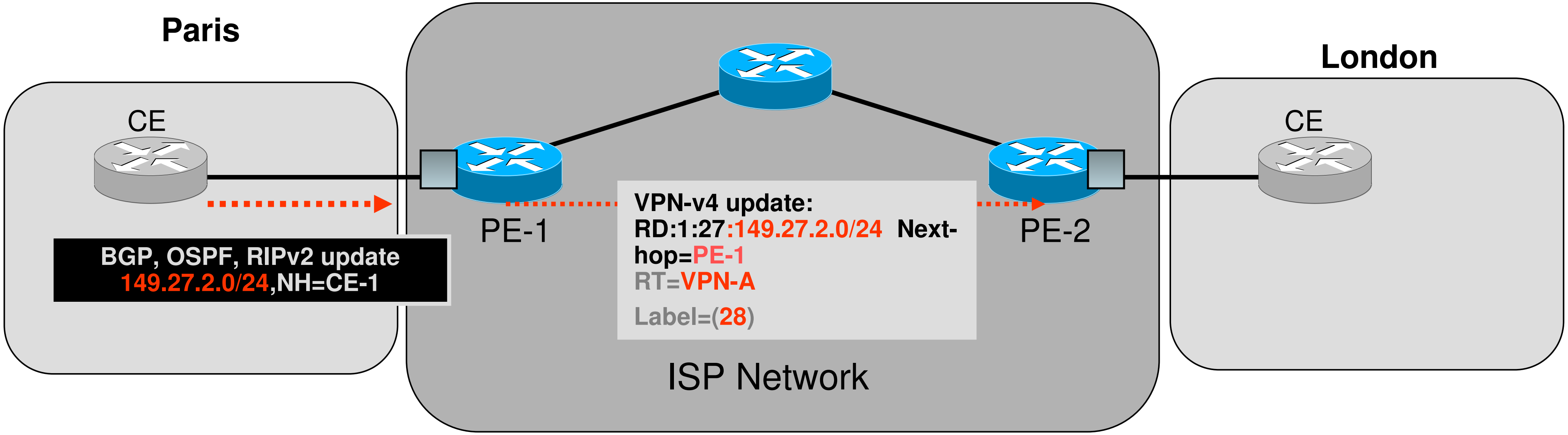
- VRF is populated locally through PE and CE routing protocol exchange.
  - EBGP, OSPF, RIPv2, and Static routing.
  - “Connected” is also supported.
- Separate routing context for each VRF.
  - Routing protocol context (e.g., MP-BGP).
  - Separate process (e.g., OSPF).

Carrying VPN Routes in BGP

- Need some way to get the VRF routing information off the PE and to other Pes.
- This is done with MP-BGP.
- Additions to MP-BGP to carry MPLS-VPN info:
    - Route Target (RT) sent in EXTENED_CIMMUNITY attribute.
    - MP_REACH_NLRI attribute for Labeled VPN IPv4 (VPNv4) address family,
        - VPN IPv4 network.
        - Route Distinguisher (RD).
        - MPLS Label.

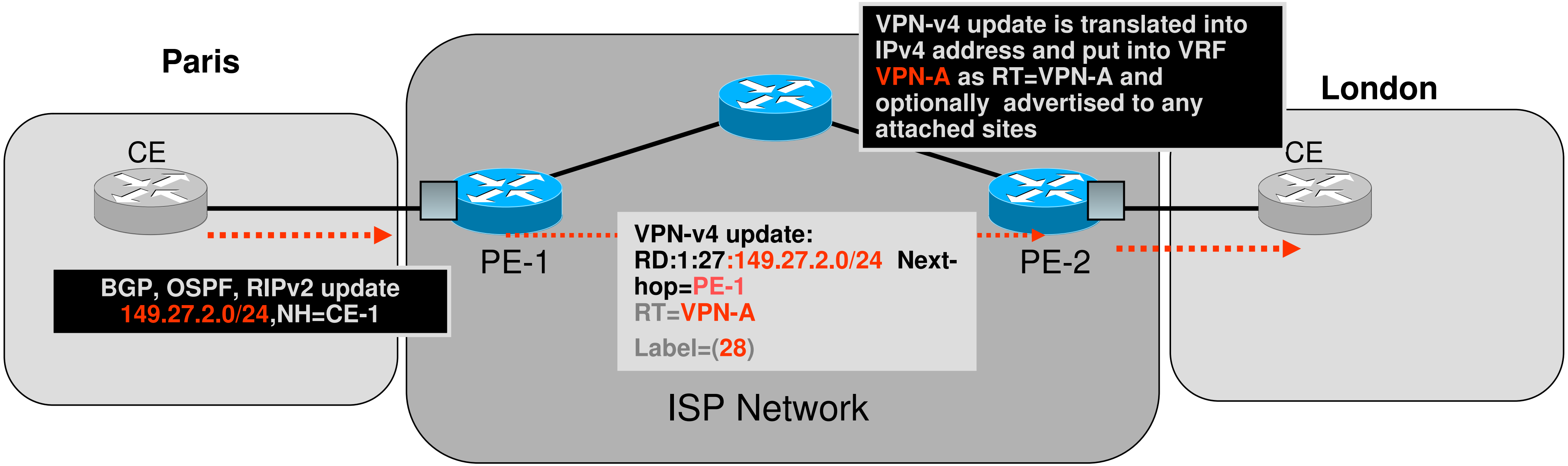
```
Border Gateway Protocol - UPDATE Message
  Marker: ffffffffffffffffffffffffffffffff
  Length: 91
  Type: UPDATE Message (2)
  Withdrawn Routes Length: 0
  Total Path Attribute Length: 68
▽ Path attributes
  ▷ Path Attribut - ORIGIN: INCOMPLETE
  ▷ Path Attribut - AS_PATH: empty
  ▷ Path Attribut - MULTI_EXIT_DISC: 0
  ▷ Path Attribut - LOCAL_PREF: 100
  ▽ Path Attribut - EXTENDED_COMMUNITIES
    ▷ Flags: 0xc0: Optional, Transitive, Complete
      Type Code: EXTENDED_COMMUNITIES (16)
      Length: 8
    ▽ Carried extended communities: (1 community)
      ▷ Community Transitive Two-Octet AS Route Target: 200:1
  ▽ Path Attribut - MP_REACH_NLRI
    ▷ Flags: 0x80: Optional, Non-transitive, Complete
      Type Code: MP_REACH_NLRI (14)
      Length: 33
      Address family: IPv4 (1)
      Subsequent address family identifier: Labeled VPN Unicast (128)
    ▷ Next hop network address (12 bytes)
      Subnetwork points of attachment: 0
    ▽ Network layer reachability information (16 bytes)
      ▷ Label Stack=24 (bottom) RD=200:1, IPv4=192.1.1.0/25
```

VRF Population of MP-BGP



- PE routers translate into VPN-V4 route
- Assigns an RD and RT based on configuration
- Re-writes Next-Hop attribute (to PE loopback)
- Assigns a label based on VRF and/or interface
- Sends MP-BGP update to all PE neighbors

VRF Population of MP-BGP

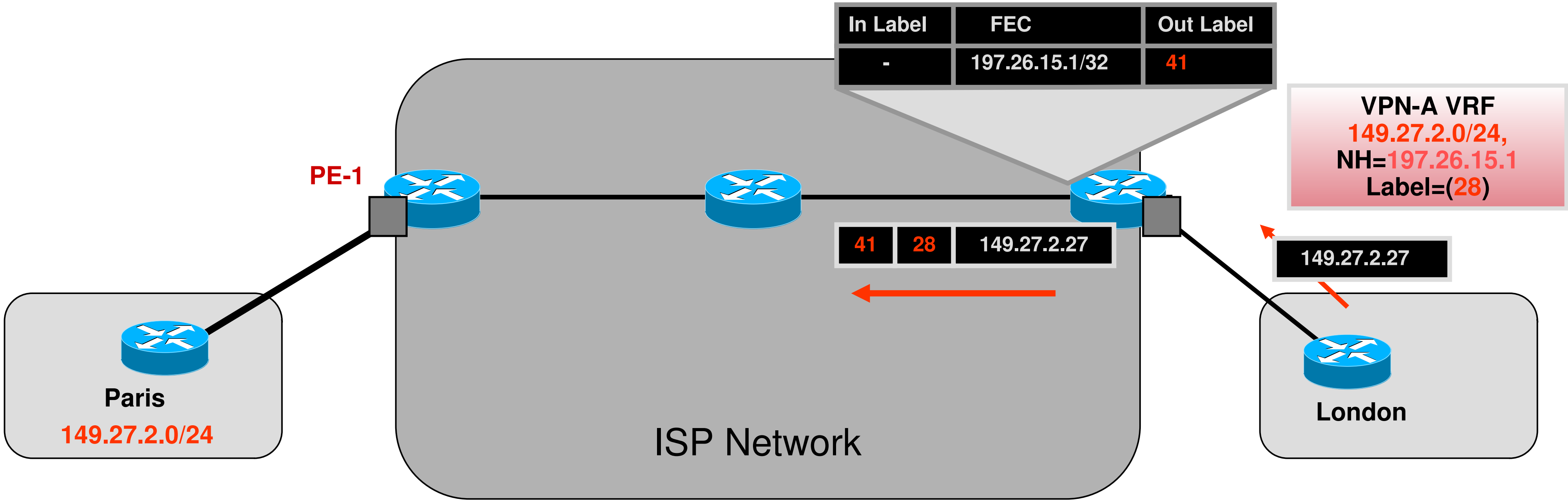


- Receiving PE routers translate to IPv4
  - Insert the route into the VRF identified by the RT attribute (based on PE configuration)
- The label associated to the VPN-V4 address will be set on packets forwarded towards the destination

MPLS/VPN Packet Forwarding

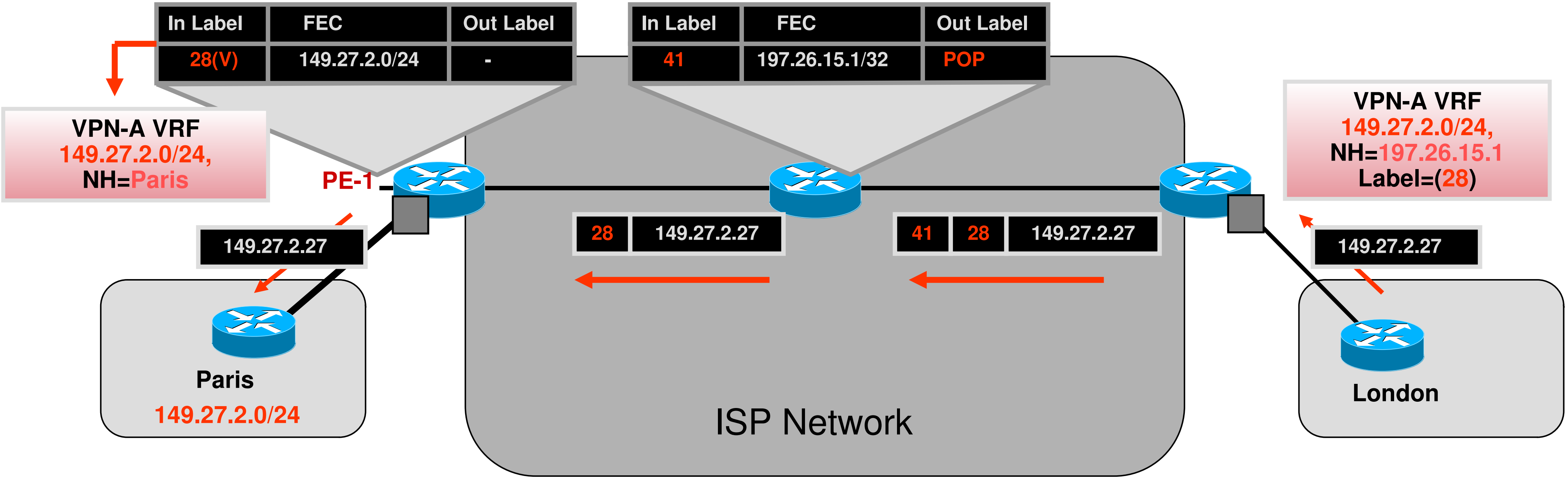
- **Between PE and CE, regular IP packets (currently)**
- **Within the provider network—label stack**
  - **Outer label: “get this packet to the egress PE”**
  - **Inner label: “get this packet to the egress CE”**
- **MPLS nodes forward packets based on TOP label!!!**
  - any subsequent labels are ignored
- **Penultimate Hop Popping procedures used one hop prior to egress PE router (shown in example)**

MPLS/VPN Packet Forwarding



- Ingress PE receives normal IP packets
- PE router performs IP Longest Match from VPN FIB (Forwarding Table), finds iBGP next-hop and imposes a stack of labels <IGP, VPN>

MPLS/VPN Packet Forwarding



- Penultimate PE router removes the IGP label
  - Penultimate Hop Popping procedures (implicit-null label)
- Egress PE router uses the VPN label to select which VPN/CE to forward the packet to
- VPN label is removed and the packet is routed toward the VPN site

Things to Note

- Core does not run VPNv4 BGP!
  - Same principle can be used to run a BGP-free core for an IP network,
- CE does not know it's in an MPLS-VPN!
- Outer label is from LDP/RSVP (Core LSP).
  - Getting packet to egress PE is mutually independent to MPLS-VPN.
- Inner label is from MP-BGP (VPN LSP).
  - Inner label is there so the egress PE can have the same network in multiple VRFs.