MENOG

Middle East Network Operators Group

MENOG 18

Segment Routing

- Rasoul Mesghali CCIE#34938
- Vahid Tavajjohi

From HAMIM Corporation

Agenda

- Introduction
- Technology Overview
- Use Cases
- Closer look at the Control and Data Plane
- Traffic Protection
- Traffic engineering
- SRv6

MENOG 18

Introduction

MPLS Historical Perspective

MPLS “classic” (LDP and RSVP-TE) control-plane was too complex and lacked scalability.

LDP is redundant to the IGP and that it is better to distribute labels bound to IGP signaled prefixes in the IGP itself rather than using an independent protocol (LDP) to do it.

LDP-IGP synchronization issue, RFC 5443, RFC 6138

Overall, we would estimate that 10% of the SP market and likely 0% of the Enterprise market have used RSVP-TE and that among these deployments, the vast majority did it for FRR reasons.

The point is to look at traditional technology (LDP/RSVP_TE) applicability in IP networks in 2018. Does it fit the needs of modern IP networks?

MPLS Historical Perspective

In RSVP-TE and the classic MPLS TE The objective was to create circuits whose state would be signaled hop-by-hop along the circuit path. Bandwidth would be booked hop-by-hop. Each hop's state would be updated. The available bandwidth of each link would be flooded throughout the domain using IGP to enable distributed TE computation.

First, RSVP-TE is not ECMP-friendly.

Second, to accurately book the used bandwidth, RSVP-TE requires all the IP traffic to run within so-called "RSVP-TE tunnels". This leads to much complexity and lack of scale in practice.

1.network has enough capacity to accommodate without congestion

traffic engineering to avoid congestion is not needed. It seems obvious to write it but as we will see further, this is not the case for an RSVP-TE network.

2.In the rare cases where the traffic is larger than expected or a non-expected failure occurs, congestion occurs and a traffic engineering solution may be needed. We write "may" because it depends on the capacity planning process.

3.Some other operators may not tolerate even these rare congestions and then require a tactical traffic-engineering process.

A tactical traffic-engineering solution is a solution that is used only when needed.

An analogy would be that one needs to wear his raincoat and boots every day while it rains only a few days a year.

equired in the
SAMPLE TE
be switched

Goals and Requirements

- Make things easier for operators
  - Improve scale, simplify operations
  - Minimize introduction complexity/disruption
- Enhance service offering potential through programmability
- Leverage the efficient MPLS dataplane that we have today
  - Push, swap, pop
  - Maintain existing label structure
- Leverage all the services supported over MPLS
  - Explicit routing, FRR, VPNv4/6, VPLS, L2VPN, etc
- IPv6 dataplane a must, and should share parity with MPLS

Operators Ask For Drastic LDP/RSVP Improvement

- Simplicity
  - less protocols to operate
  - less protocol interactions to troubleshoot
  - avoid directed LDP sessions between core routers
  - deliver automated FRR for any topology
- Scale
  - avoid millions of labels in LDP database
  - avoid millions of TE LSP’s in the network
  - avoid millions of tunnels to configure

Operators Ask For A Network Model Optimized For Application Interaction

- Applications must be able to interact with the network
  - cloud based delivery
  - internet of everything
- Programmatic interfaces and Orchestration
  - Necessary but not sufficient
- The network must respond to application interaction
  - Rapidly-changing application requirements
  - Virtualization
  - Guaranteed SLA and Network Efficiency

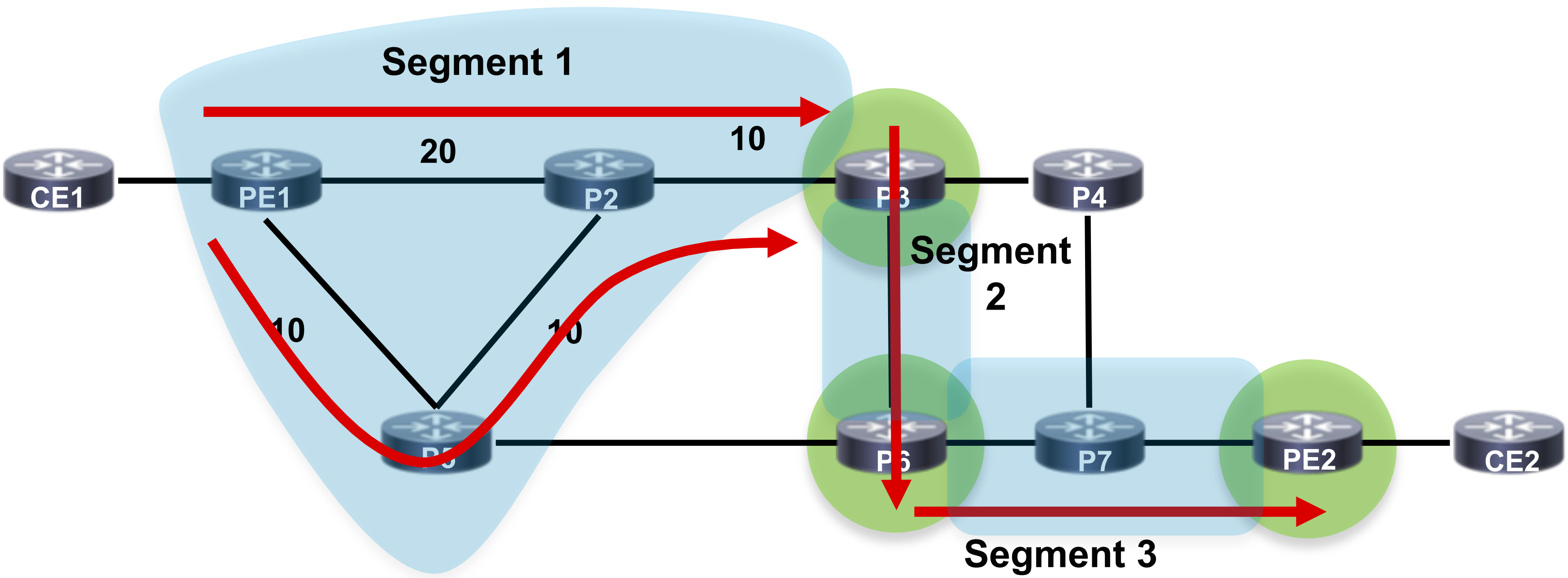
Segment Routing

- Simple to deploy and operate
  - Leverage MPLS services & hardware
  - straightforward ISIS/OSPF extension to distribute labels
  - LDP/RSVP not required
- Provide for optimum scalability, resiliency and virtualization
- SDN enabled
  - simple network, highly programmable
  - highly responsive

MENOG 18

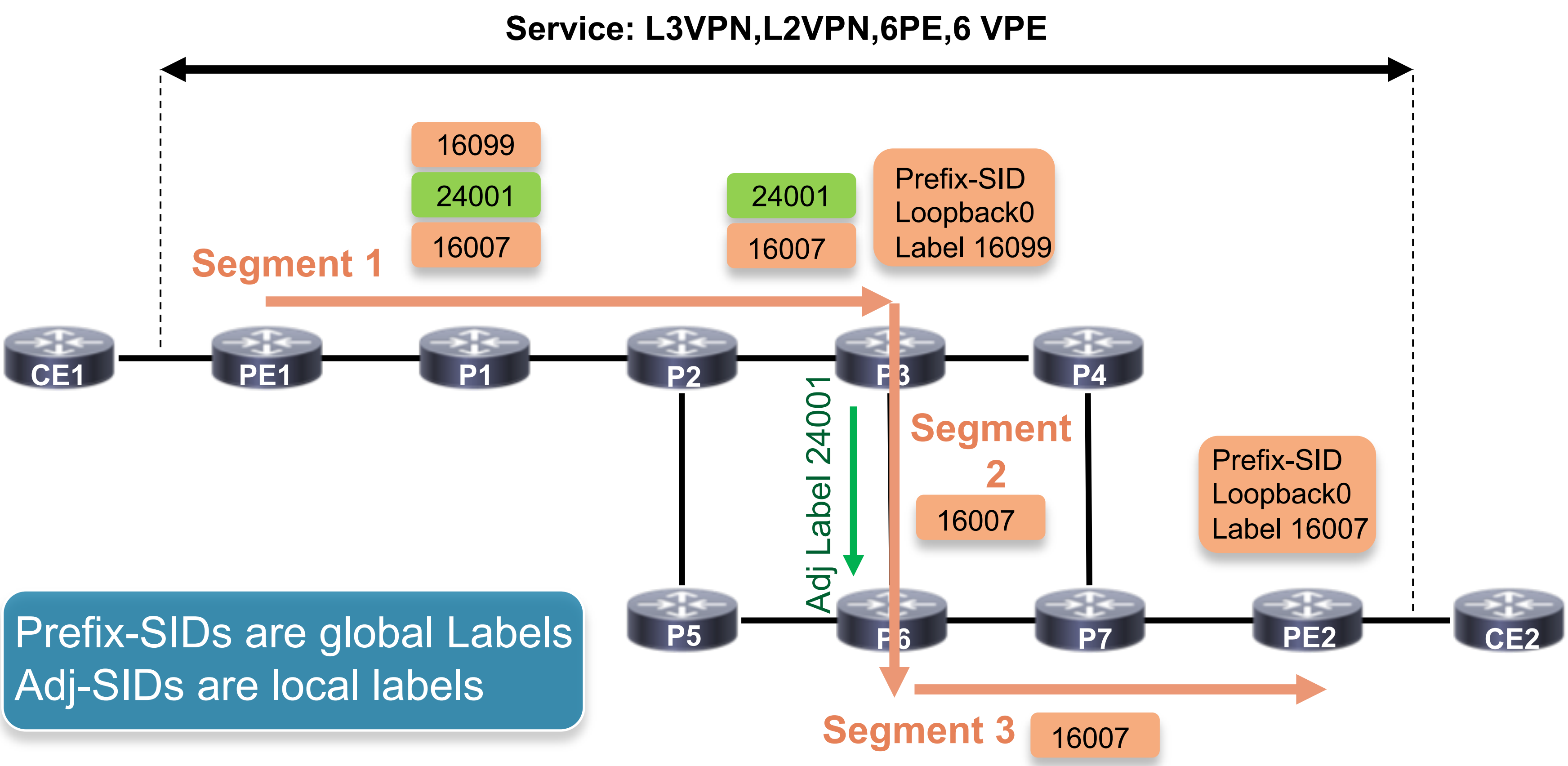
Technology Overview

What is the meaning of Segment Routing?



Default Cost is 100

SR in one Slide



Prefix-SIDs are global Labels
Adj-SIDs are local labels

Deviate from shortest path-Source Routing
Traffic Engineering based on SR

Default: PHP at each segment

Let’s take a closer look

- **Source Routing**

  the source chooses a path and encodes it in the packet header as an ordered list of segments

  • the rest of the network executes the encoded instructions (In Stack of labels/IPv6 EH)

- **Segment**: an identifier for any type of instruction

  - forwarding or service

- **Forwarding state (segment) is established by IGP**

  LDP and RSVP-TE are not required

  Agnostic to forwarding data plane: IPv6 or MPLS

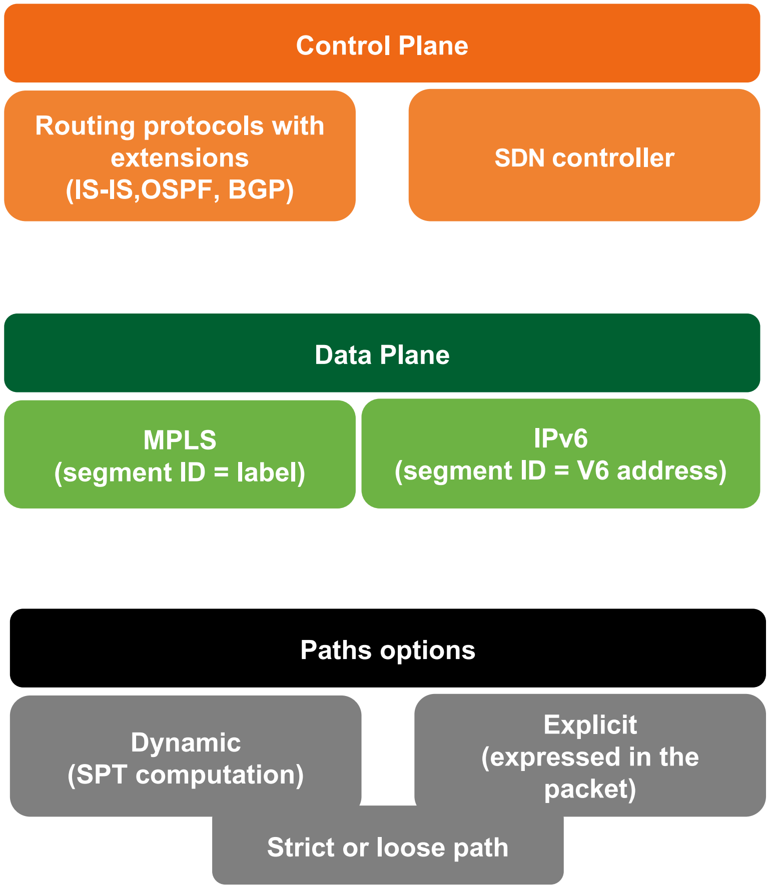
- **MPLS Data plane is leveraged without any modification**

  push, swap and pop: all that we need

  segment = label

Segment Routing – Overview

- **MPLS**: an ordered list of segments is represented as a stack of labels
- **IPv6:** an ordered list of segments is encoded in a routing extension header
- This presentation**: MPLS data plane**
  Segment → Label
  Basic building blocks distributed by the IGP or BGP



Global and Local Segments

- **Global Segment**

Any node in SR domain understands associated instruction
Each node in SR domain installs the associated instruction in its forwarding table
MPLS: global label value in Segment Routing Global Block (SRGB)

- **Local Segment**

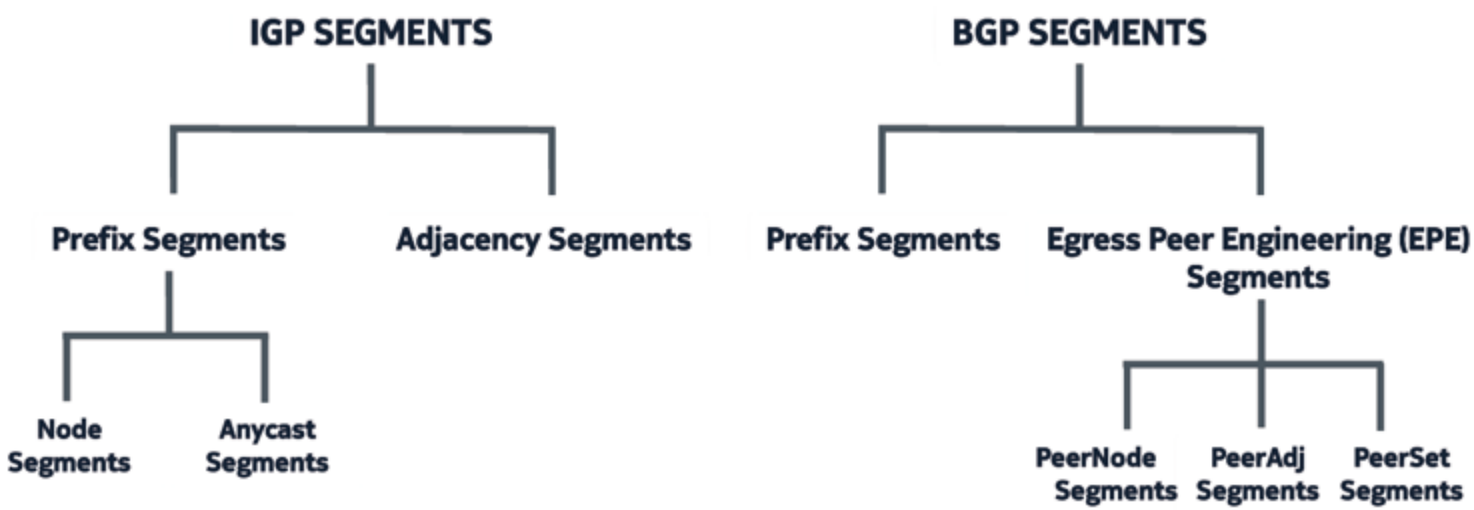
Only originating node understands associated instruction
MPLS: locally allocated label

Global Segments – Global Label Indexes

- **Global Segments always distributed as a label range**
  (SRGB) + Index
  Index must be unique in Segment Routing Domain

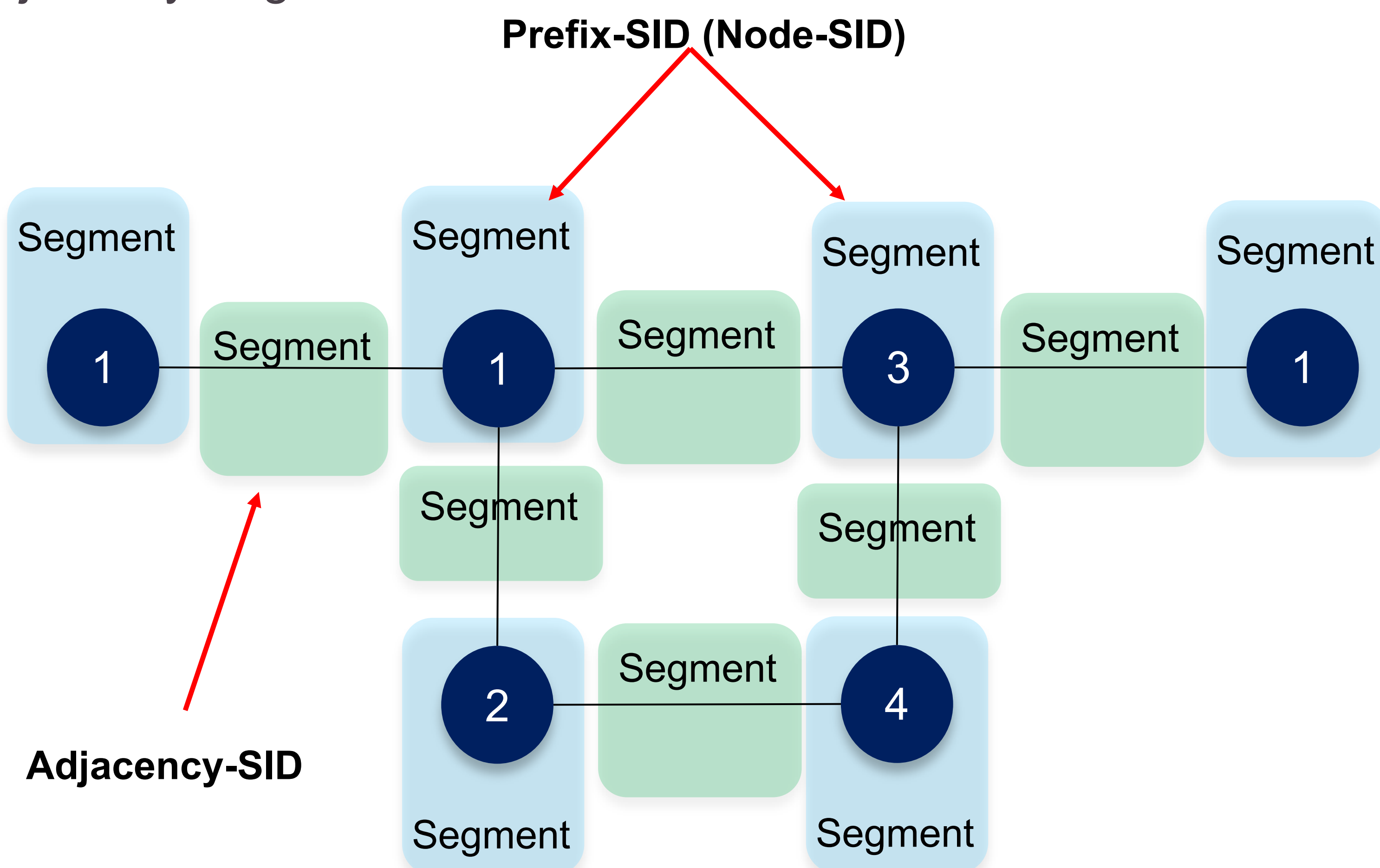
- **Best practice: same SRGB on all nodes**
  "Global model", requested by all operators
  Global Segments are global label values, simplifying network operations
  Default SRGB: 16,000 – 23,999
  Other vendors also use this label range

Types of Segment



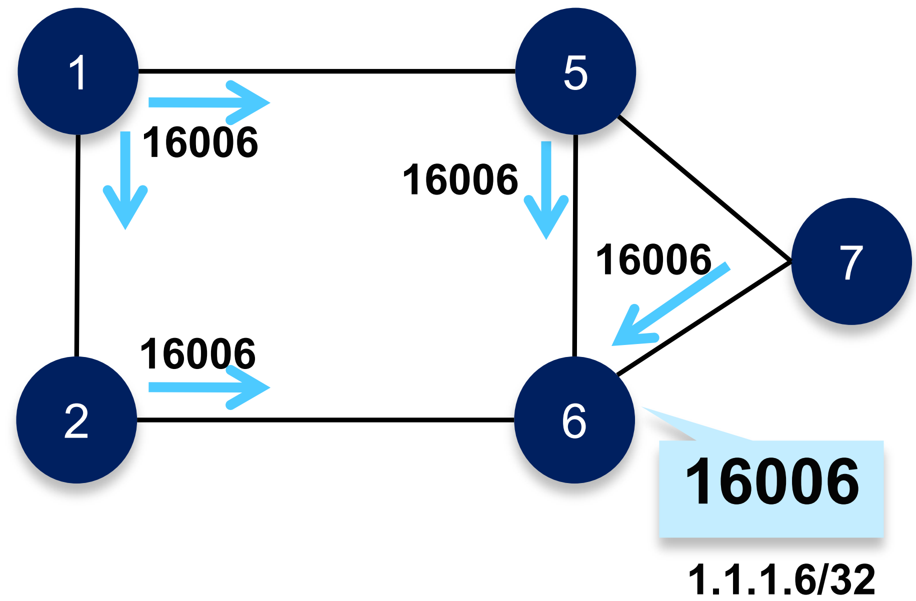
IGP Segment

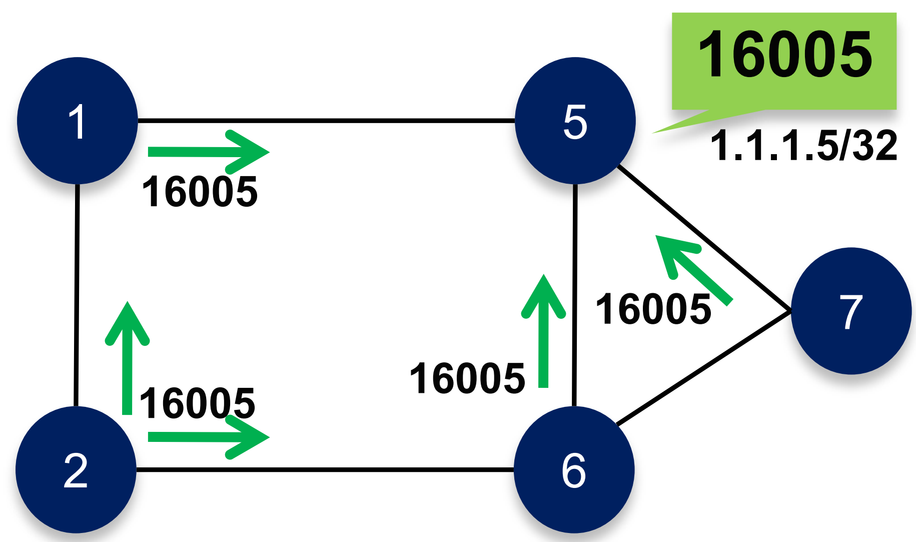
Two Basic building blocks distributed by IGP:
-Prefix Segment
-Adjacency Segment



IGP Prefix Segment Node-SID

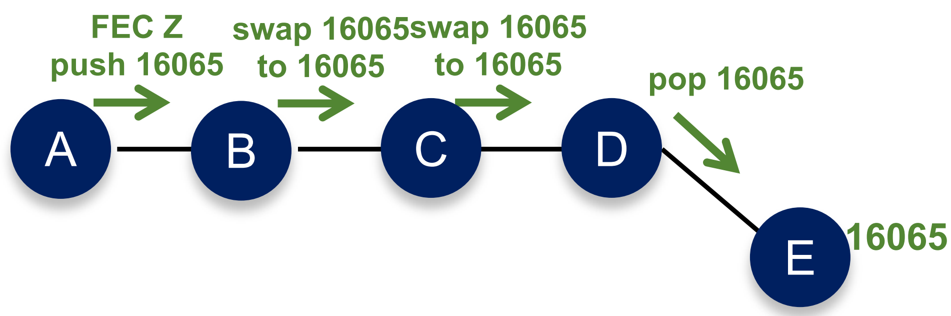
- Shortest-path to the IGP prefix
  - Equal Cost Multipath (ECMP)-aware
- Global Segment
- Label = 16000 + Index
  - Advertised as index
- Distributed by ISIS/OSPF





Default SRGB 16000-23,999

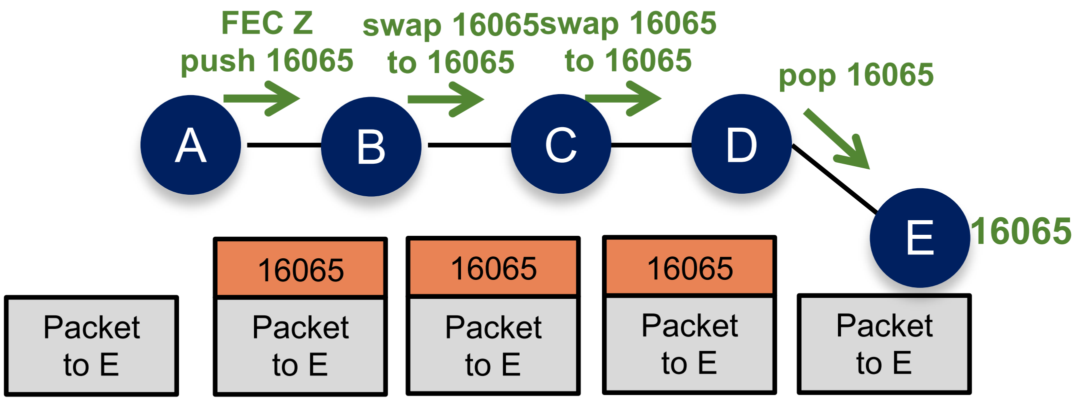
Node Segment



A packet injected anywhere with top label 16065 will reach E via shortest-path

- E advertises its node segment
  - Simple ISIS/OSPF sub-TLV extension
- All remote nodes install the node segment to E in the **MPLS Data Plane**

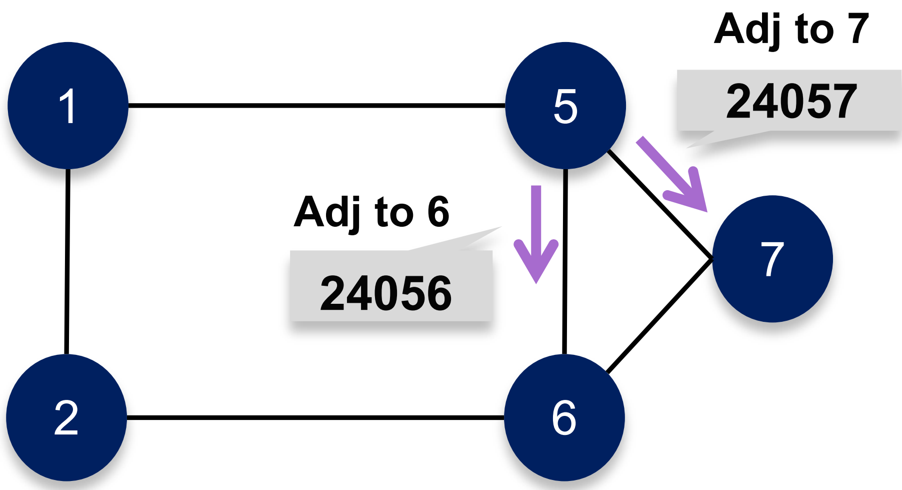
Node Segment



A packet injected anywhere with top label 16065 will reach E via shortest-path

- E advertises its node segment
  - simple ISIS sub-TLV extension and OSPF
- All remote nodes install the node segment to E in the MPLS dataplane

Adjacency Segment



A packet injected at node 5 with label **24056** is forced through datalink **5-6**

- C allocates a local label and forward on the IGP adjacency
- C advertises the adjacency **label**

  Distributed by OSPF/ISIS

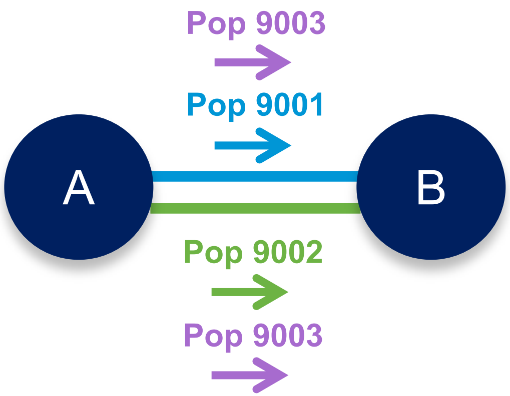
  simple sub-TLV extension

  [(https://datatracker.ietf.org/doc/draft-ietf-isis-segment-routing-extensions/)](https://datatracker.ietf.org/doc/draft-ietf-isis-segment-routing-extensions/)

  https://www.iana.org/assignments/isis-tlv-codepoints/isis-tlv-codepoints.xhtml

- C is the only node to install the adjacency segment in MPLS dataplane

Datalink and Bundle



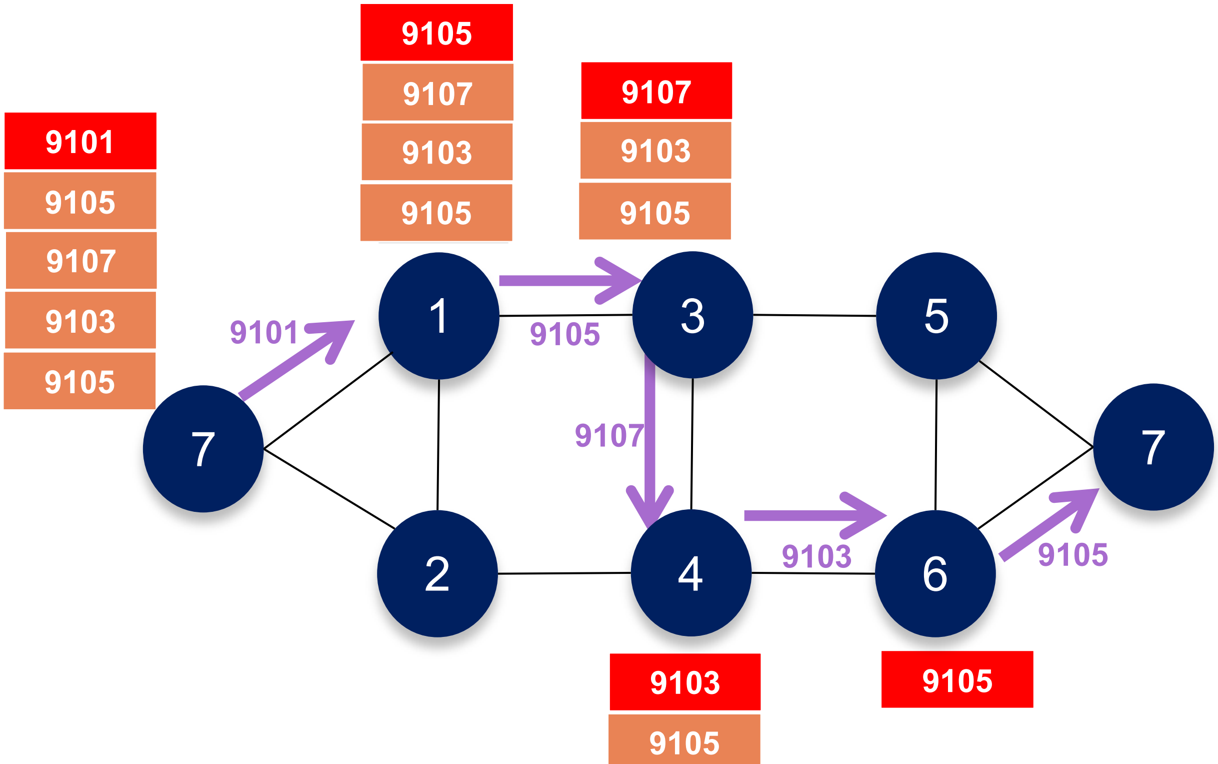
9001 switches on blue member

9002 switches on green member

9003 load-balances on any member of the adj

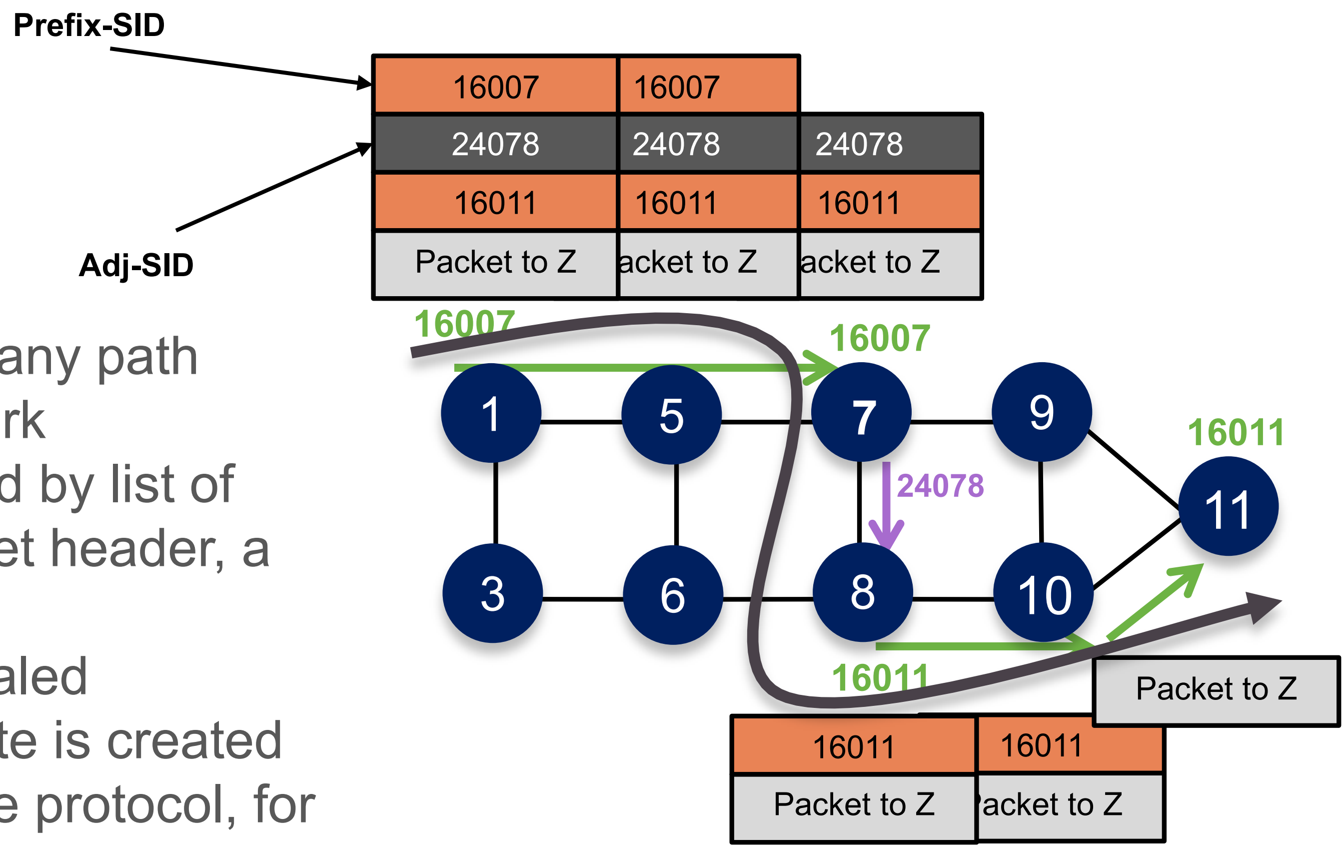
- Adjacency segment represents a specific datalink to an adjacent node
- Adjacency segment represents a set of datalinks to the adjacent node

A path with Adjacency Segments



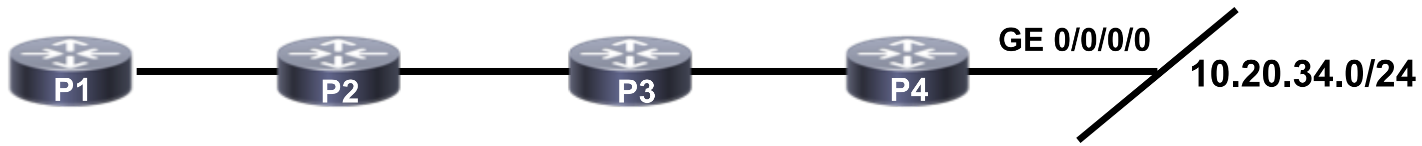
- Source routing along any explicit path
  - stack of adjacency labels
- SR provides for entire path control

Combining Segments



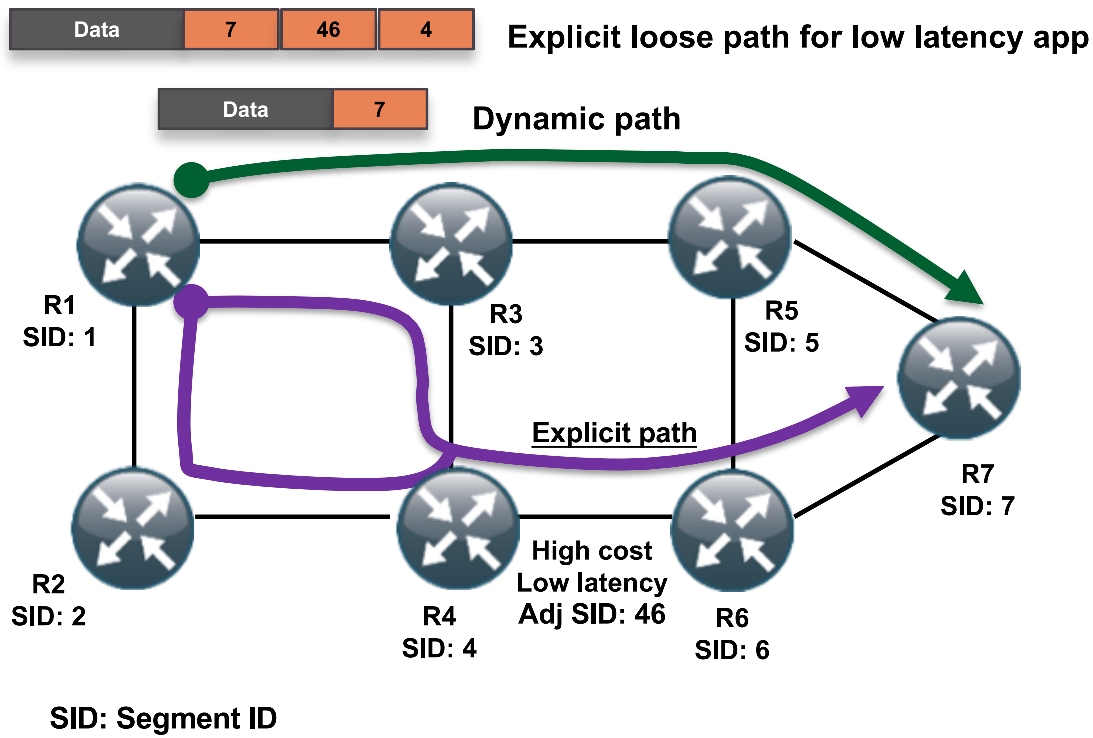
- Steer traffic on any path through the network
- Path is specified by list of segments in packet header, a stack of labels
- No path is signaled
- No per-flow state is created
- For IGP – single protocol, for BGP – AF LS

Labeling Which prefixes?



Prefix attached to P4	Outgoing label in CEF? Entry in LFIB?
Prefix-SID P4 (10.100.1.4/32)	**Y**
Prefix-SID P4 without Node flag (10.100.3.4/32)	**Y**
loopback prefix without prefix-sid (10.100.4.4/32)	**N**
link prefix connected to P4 (10.1.45.0/24)	**N**

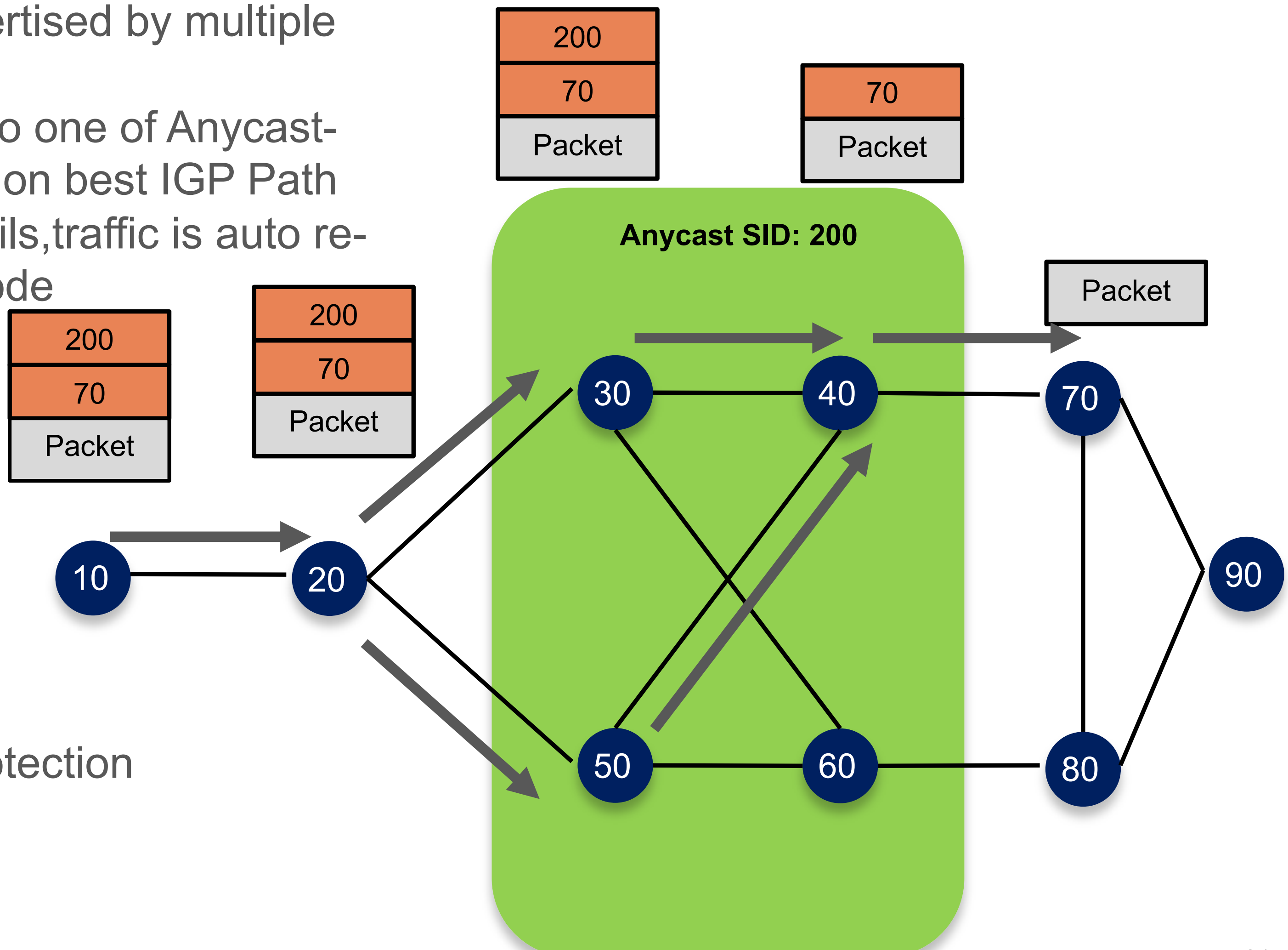
- So, this is the equivalent of LDP label prefix filtering: only assigning/advertising labels to /32 prefixes (loopback prefixes, used by service, (e.g. L3VPN), so BGP next hop IP addresses)
- Traffic to link prefixes is not labeled!



No LDP, no RSVP-TE

Any-Cast SID for Node Redundancy

- A group of Nodes share the same SID
- Work as a "Single" router, single Label
- Same Prefix advertised by multiple nodes
- traffic forwarded to one of Anycast-Prefix-SID based on best IGP Path
- if primary node fails,traffic is auto re-routed to other node

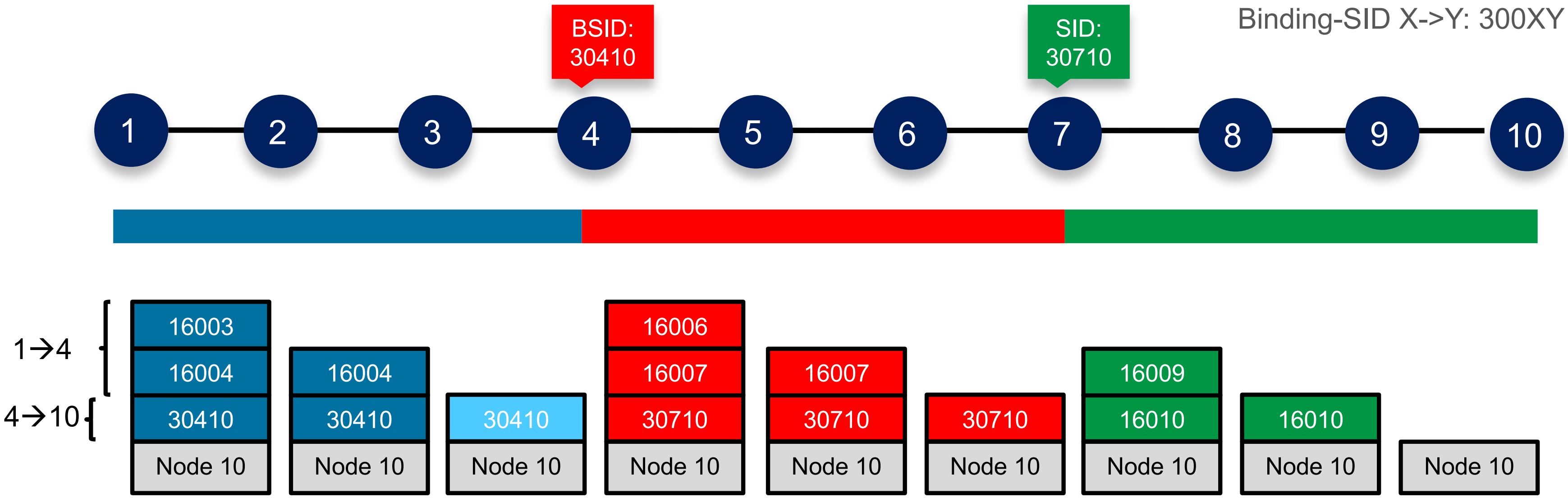


- **Application**

– ABR Protection
– Seamless MPLS
– ASBR inter-AS protection

Binding-SID

All Nodes SRGB [16000-23999]
Prefix-SID NodeX: 1600X
Binding-SID X->Y: 300XY

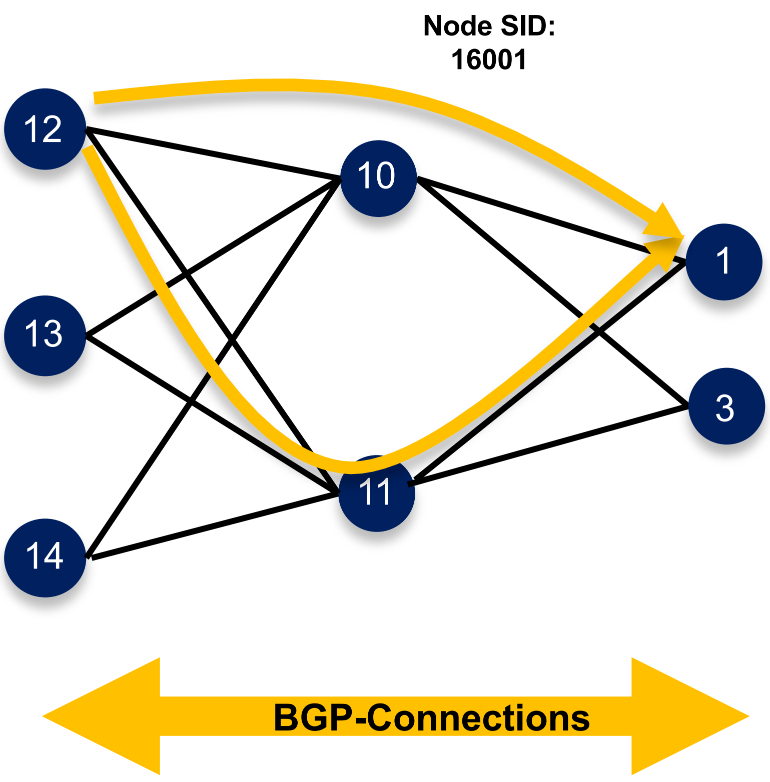


Binding-SIDs can be used in the following cases:

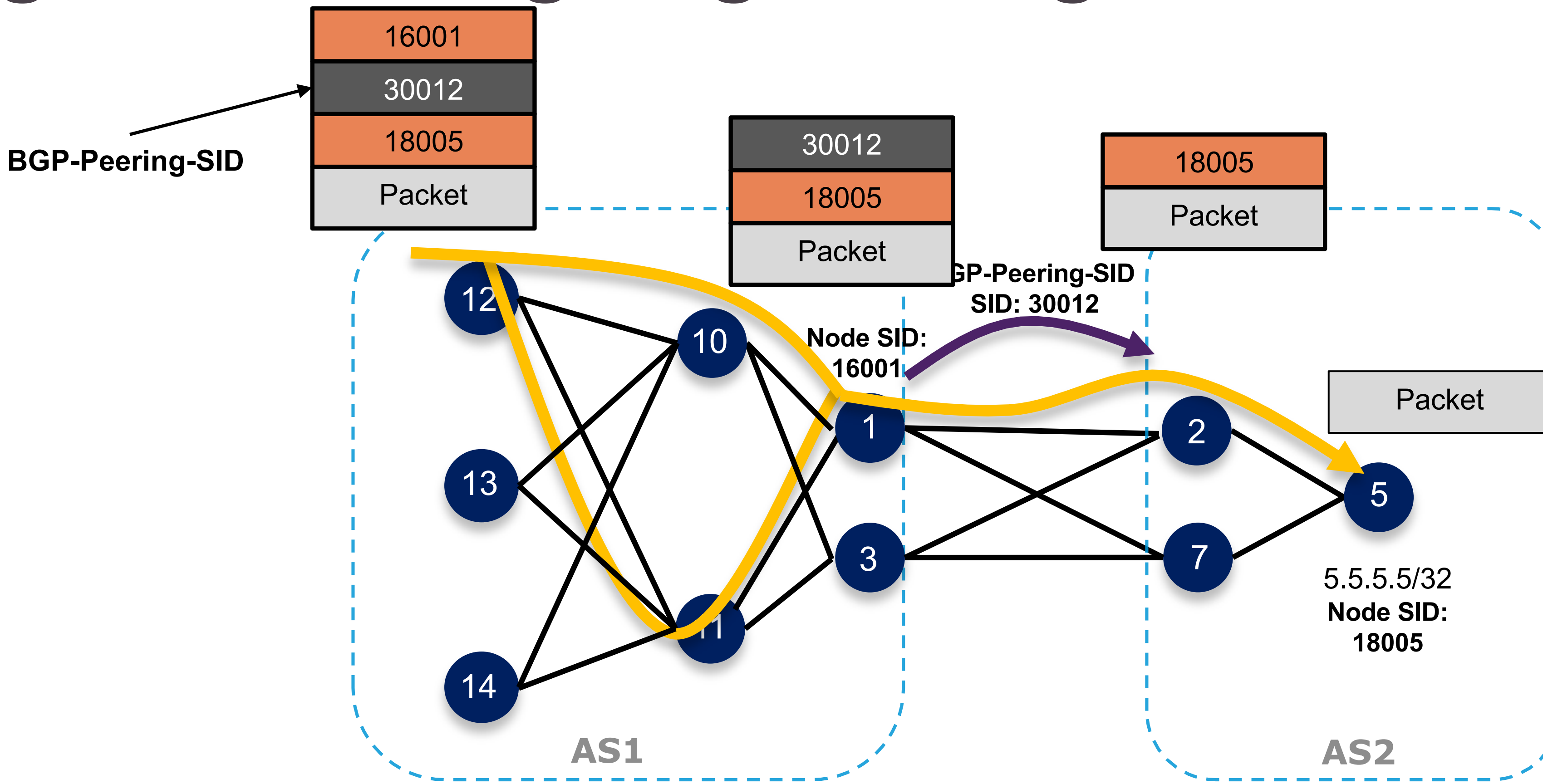
- Multi-Domain (inter-domain, inter-autonomous system)
- Large-Scale within a single domain
- Label stack compression
- BGP SR-TE Dynamic
- Stitching SR-TE Polices Using Binding SID

BGP Prefix Segment

- Shortest-Path to the BGP Prefix
- Global
- 16000 + Index
- Signaled by BGP



BGP Peering Segment
Egress Peering Engineering

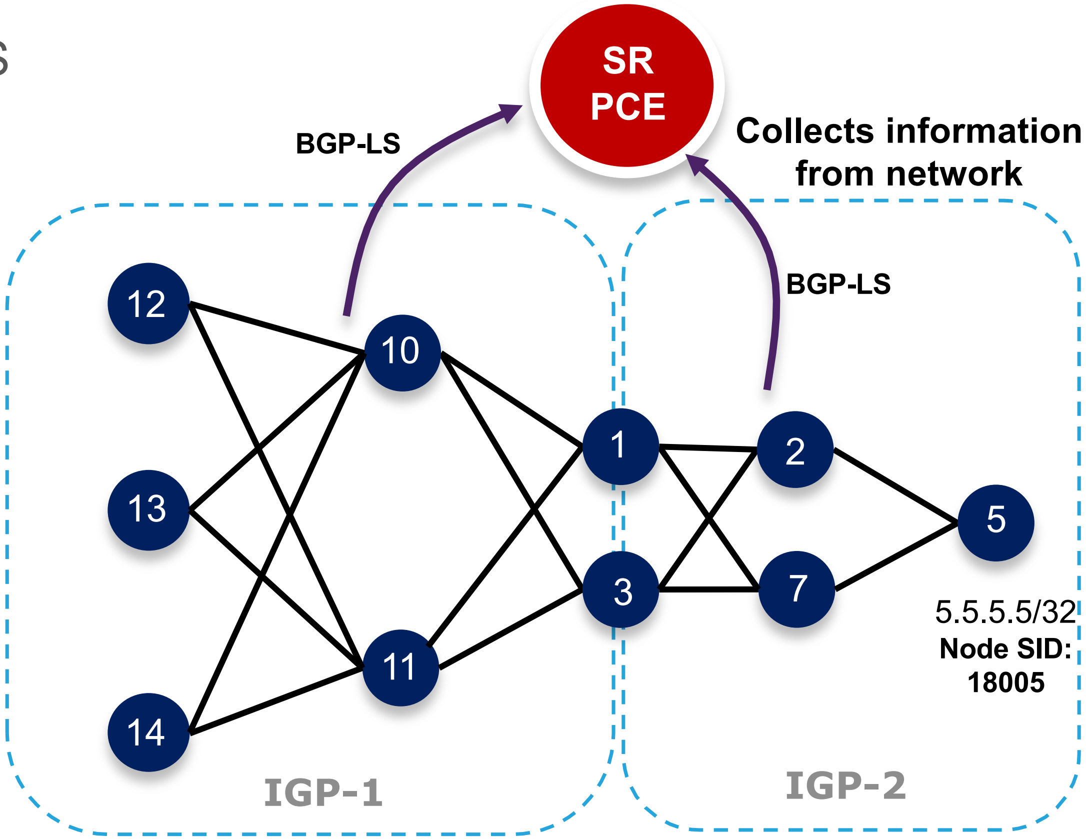


- Pop and Forward to the BGP Peer
- Local
- Signaled by BGP-LS (Topology Information) to the controller
- Local Segment- Like an adjacency SID external to the IGP
  Dynamically allocated but persistent

WAN Controller

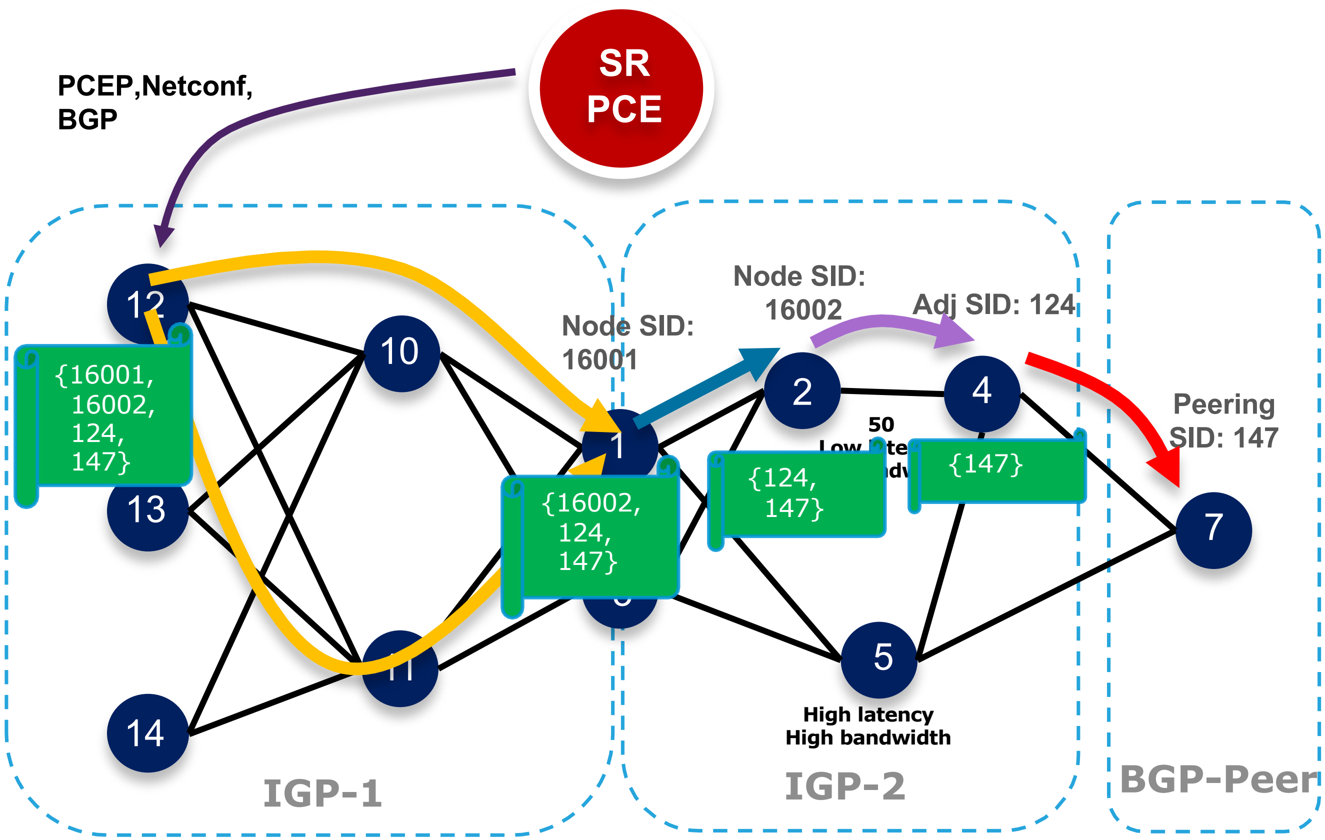
SR PCE Collects via BGP-LS

- IGP Segments
- BGP Segments
- Topology



An end-to-end path as a list of segment

- Controller learn the network topology and usage dynamically
- Controller calculate the optimized path for different applications: low latency, or high bandwidth
- Controller just program a list of the labels on the source routers. The rest of the network is not aware: no signaling, no state information → simple and Scalable



Default ISIS cost metric: 10

Segment Routing Value Proposition

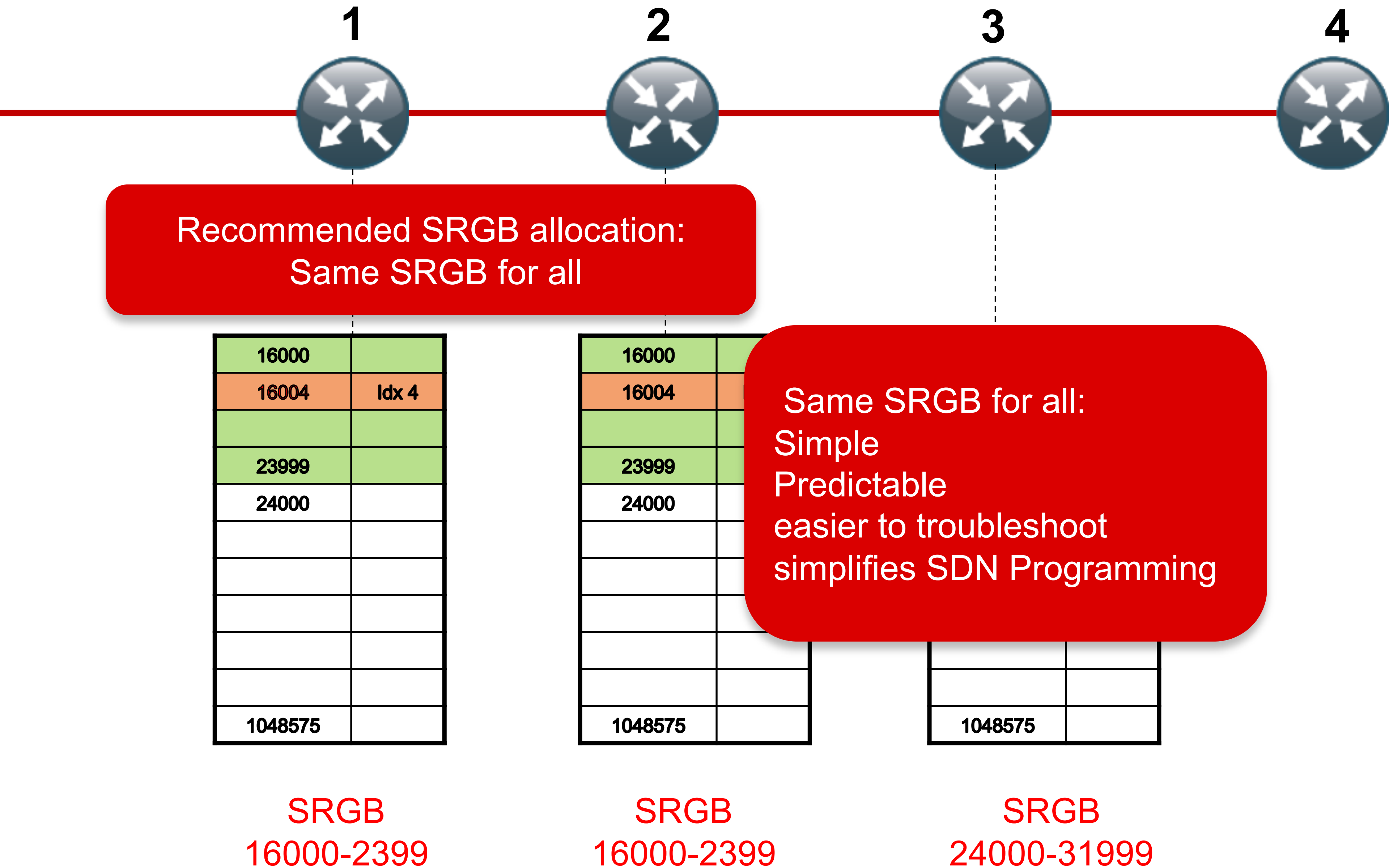
	MPLS for SDN (Segment Routng)	MPLS for TDM/IP (MPLS/uMPLS)
MPLS Transport Protocols	IGP	IGP + LDP
IGP/LDP synchronization	N/A	Problem to manage
50msec FRR	IGP	IGP + RSVP-TE
Extra TE states to support FRR	No extra state	Extra states to manage
Optimum backup path	Yes	No (SDH-alike)
ECMP-capability for TE	Yes	No
TE state only at headend	Yes	No (n^2 problem)
Seamless Interworking with classic MPLS and incremental deployment	Yes	N/A
Engineered for SDN	Yes	No

Segment Routing

Segment Routing Global Block

Segment Routing Global Block (SRGB)

- Segment Routing Global Block
  - Range of labels reserved for Segment Routing Global Segments
  - Default SRGB is 16,000 – 23,999
- A prefix-SID is advertised as a domain-wide unique index
- The Prefix-SID index points to a unique label within the SRGB
  - Index is zero based, i.e. first index = 0
  - Label = Prefix-SID index + SRGB base
  - E.g. Prefix 1.1.1.65/32 with prefix-SID index 65 gets label 16065
  - **index 65 --> SID is 16000 + 65 =16065**
- Multiple IGP instances can use the same SRGB or use different non-overlapping SRGBs


1
2
3
4
Recommended SRGB allocation:
Same SRGB for all
16000
16004
Idx 4
23999
24000
1048575
16000
16004
23999
24000
1048575
Same SRGB for all:
Simple
Predictable
easier to troubleshoot
simplifies SDN Programming
1048575
SRGB
16000-2399
SRGB
16000-2399
SRGB
24000-31999

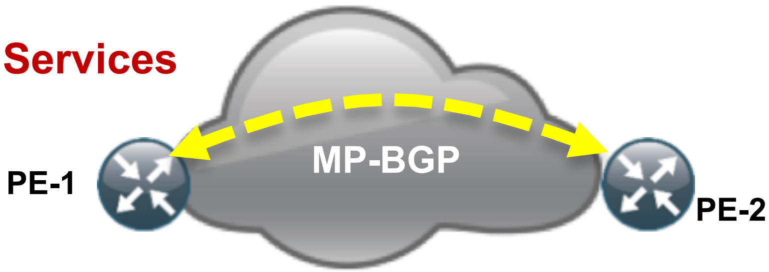
MENOG 18

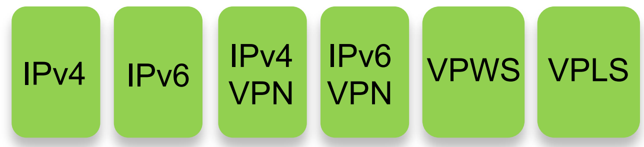
Segment Routing

IGP Control and Date Plane

MPLS Control and Forwarding Operation with Segment Routing

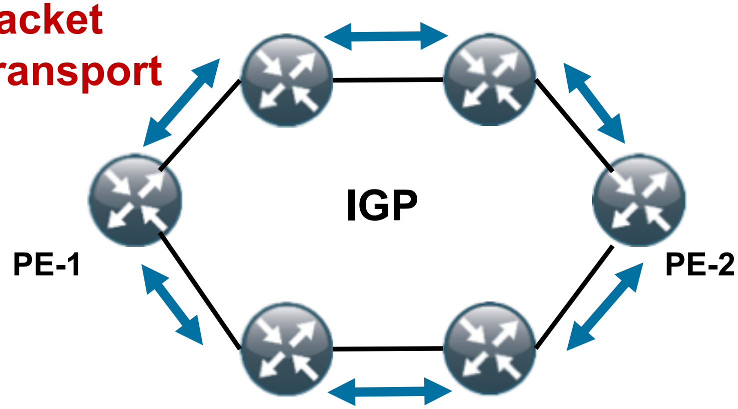
Services

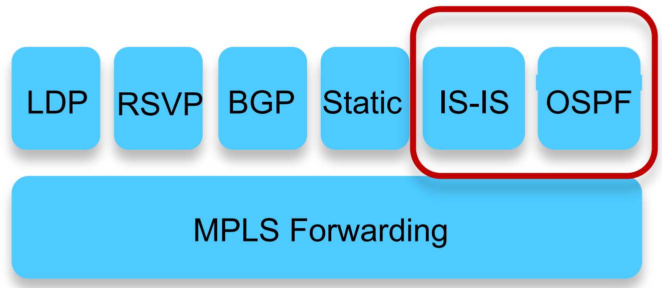




No changes to control or forwarding plane

Packet Transport





IGP label distribution for IPv4 and IPv6. Forwarding plane remains the same

SR IS-IS Control Plane overview

- IPv4 and IPv6 control plane
- Level 1, level 2 and multi-level routing
- Prefix Segment ID (Prefix-SID) for host prefixes on loopback interfaces
- Adjacency SIDs for adjacencies
- Prefix-to-SID mapping advertisements (mapping server)
- MPLS penultimate hop popping (PHP) and explicit-null label signaling

ISIS TLV Extensions

- SR for IS-IS introduces support for the following (sub-)TLVs:

– SR Capability sub-TLV (2)	IS-IS Router Capability TLV (242)
– Prefix-SID sub-TLV (3)	Extended IP reachability TLV (135)
– Prefix-SID sub-TLV (3)	IPv6 IP reachability TLV (236)
– Prefix-SID sub-TLV (3)	Multitopology IPv6 IP reachability TLV (237)
– Prefix-SID sub-TLV (3)	SID/Label Binding TLV (149)
– Adjacency-SID sub-TLV (31)	Extended IS Reachability TLV (22)
– LAN-Adjacency-SID sub-TLV (32)	Extended IS Reachability TLV (22)
– Adjacency-SID sub-TLV (31)	Multitopology IS Reachability TLV (222)
– LAN-Adjacency-SID sub-TLV (32)	Multitopology IS Reachability TLV (222)
– SID/Label Binding TLV (149)	

- Implementation based on draft-ietf-isis-segment-routing-extensions

SR OSPF Control Plane overview

SR OSPF Control Plane Overview
- OSPFv2 control plane
- Multi-area
- IPv4 Prefix Segment ID (Prefix-SID) for host prefixes on loopback interfaces
- Adjacency SIDs for adjacencies
- MPLS penultimate hop popping (PHP) and explicit-null label signaling

OSPF Extensions

- OSPF adds to the Router Information Opaque LSA (type 4):
  - SR-Algorithm TLV (8)
  - SID/Label Range TLV (9)
- OSPF defines new Opaque LSAs to advertise the SIDs
  - OSPFv2 Extended Prefix Opaque LSA (type 7)
    >OSPFv2 Extended Prefix TLV (1)
    - Prefix SID Sub-TLV (2)
  - OSPFv2 Extended Link Opaque LSA (type 8)
    >OSPFv2 Extended Link TLV (1)
    - Adj-SID Sub-TLV (2)
    - LAN Adj-SID Sub-TLV (3)
- Implementation is based on

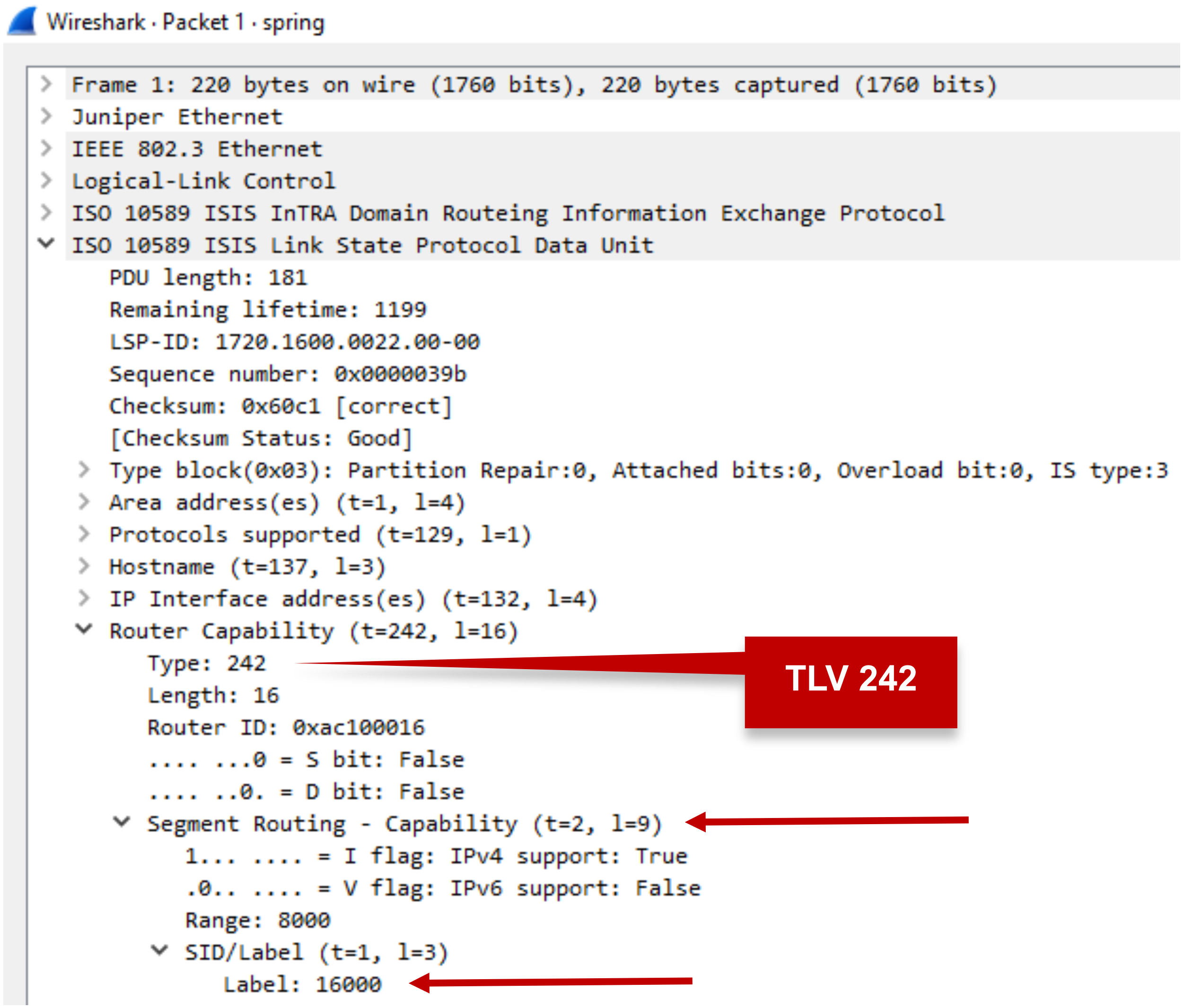
– draft-ietf-ospf-prefix-link-attr and draft-ietf-ospf-segment-routing-extensions

Wireshark · Packet 1 · spring

```
> Frame 1: 220 bytes on wire (1760 bits), 220 bytes captured (1760 bits)
> Juniper Ethernet
> IEEE 802.3 Ethernet
> Logical-Link Control
> ISO 10589 ISIS InTRA Domain Routeing Information Exchange Protocol
v ISO 10589 ISIS Link State Protocol Data Unit
    PDU length: 181
    Remaining lifetime: 1199
    LSP-ID: 1720.1600.0022.00-00
    Sequence number: 0x0000039b
    Checksum: 0x60c1 [correct]
    [Checksum Status: Good]
  > Type block(0x03): Partition Repair:0, Attached bits:0, Overload bit:0, IS type:3
  > Area address(es) (t=1, l=4)
  > Protocols supported (t=129, l=1)
  > Hostname (t=137, l=3)
  > IP Interface address(es) (t=132, l=4)
  > Router Capability (t=242, l=16)
  > Extended IS reachability (t=22, l=11)
  > Extended IS reachability (t=22, l=24)
  > Extended IS reachability (t=22, l=11)
  > Extended IS reachability (t=22, l=24)
  > Extended IP Reachability (t=135, l=36)
```

TLV 22

TLV 135


Wireshark · Packet 1 · spring
Frame 1: 220 bytes on wire (1760 bits), 220 bytes captured (1760 bits)
Juniper Ethernet
IEEE 802.3 Ethernet
Logical-Link Control
ISO 10589 ISIS InTRA Domain Routeing Information Exchange Protocol
ISO 10589 ISIS Link State Protocol Data Unit
PDU length: 181
Remaining lifetime: 1199
LSP-ID: 1720.1600.0022.00-00
Sequence number: 0x0000039b
Checksum: 0x60c1 [correct]
[Checksum Status: Good]
Type block(0x03): Partition Repair:0, Attached bits:0, Overload bit:0, IS type:3
Area address(es) (t=1, l=4)
Protocols supported (t=129, l=1)
Hostname (t=137, l=3)
IP Interface address(es) (t=132, l=4)
Router Capability (t=242, l=16)
Type: 242
Length: 16
Router ID: 0xac100016
....0 = S bit: False
.... ..0. = D bit: False
Segment Routing - Capability (t=2, l=9)
1... .... = I flag: IPv4 support: True
.0.. .... = V flag: IPv6 support: False
Range: 8000
SID/Label (t=1, l=3)
Label: 16000
TLV 242

```
Extended IP Reachability (t=135, l=36)
    Type: 135
    Length: 36
  Ext. IP Reachability
      Metric: 10
      0... .... = Distribution: Down
      .0.. .... = Sub-TLV: No
      ..01 1111 = Prefix Length: 31
      IPv4 prefix: 10.0.0.0
      no sub-TLVs present
  Ext. IP Reachability
      Metric: 10
      0... .... = Distribution: Down
      .0.. .... = Sub-TLV: No
      ..01 1111 = Prefix Length: 31
      IPv4 prefix: 10.0.0.4
      no sub-TLVs present
  Ext. IP Reachability
      Metric: 0
      0... .... = Distribution: Down
      .1.. .... = Sub-TLV: Yes
      ..10 0000 = Prefix Length: 32
      IPv4 prefix: 172.16.0.22
      SubCLV Length: 8
    subTLV: Prefix-SID (c=3, l=6)
        Code: Prefix-SID (3)
        Length: 6
      > Flags: 0x40, Node-SID
        Algorithm: Shortest Path First (SPF) (0)
        SID/Label/Index: 0x00000016
```

TLV 135

Sub-TLV 3 Prefix-SID

SID-Index 16

```
˅ Extended IS reachability (t=22, l=24)
    Type: 22
    Length: 24
  ˅ IS Neighbor: 1720.1600.0022.03
      IS neighbor ID: 1720.1600.0022.03
      Metric: 10
      SubCLV Length: 13
    ˅ subTLV: LAN-Adj-SID (c=32, l=11)
        Code: LAN-Adj-SID (32)
        Length: 11
      > Flags: 0x30, Value, Local Significance
        Weight: 0x00
        System-ID: 1720.1600.0011
        .... 0000 0101 1101 1100 0001 = SID/Label/Index: 24001
> Extended IS reachability (t=22, l=11)
˅ Extended IS reachability (t=22, l=24)
    Type: 22
    Length: 24
  ˅ IS Neighbor: 1720.1600.0022.01
      IS neighbor ID: 1720.1600.0022.01
      Metric: 10
      SubCLV Length: 13
    ˅ subTLV: LAN-Adj-SID (c=32, l=11)
        Code: LAN-Adj-SID (32)
        Length: 11
      > Flags: 0x30, Value, Local Significance
        Weight: 0x00
        System-ID: 1720.1600.0002
        .... 0000 0101 1101 1100 0000 = SID/Label/Index: 24000
```

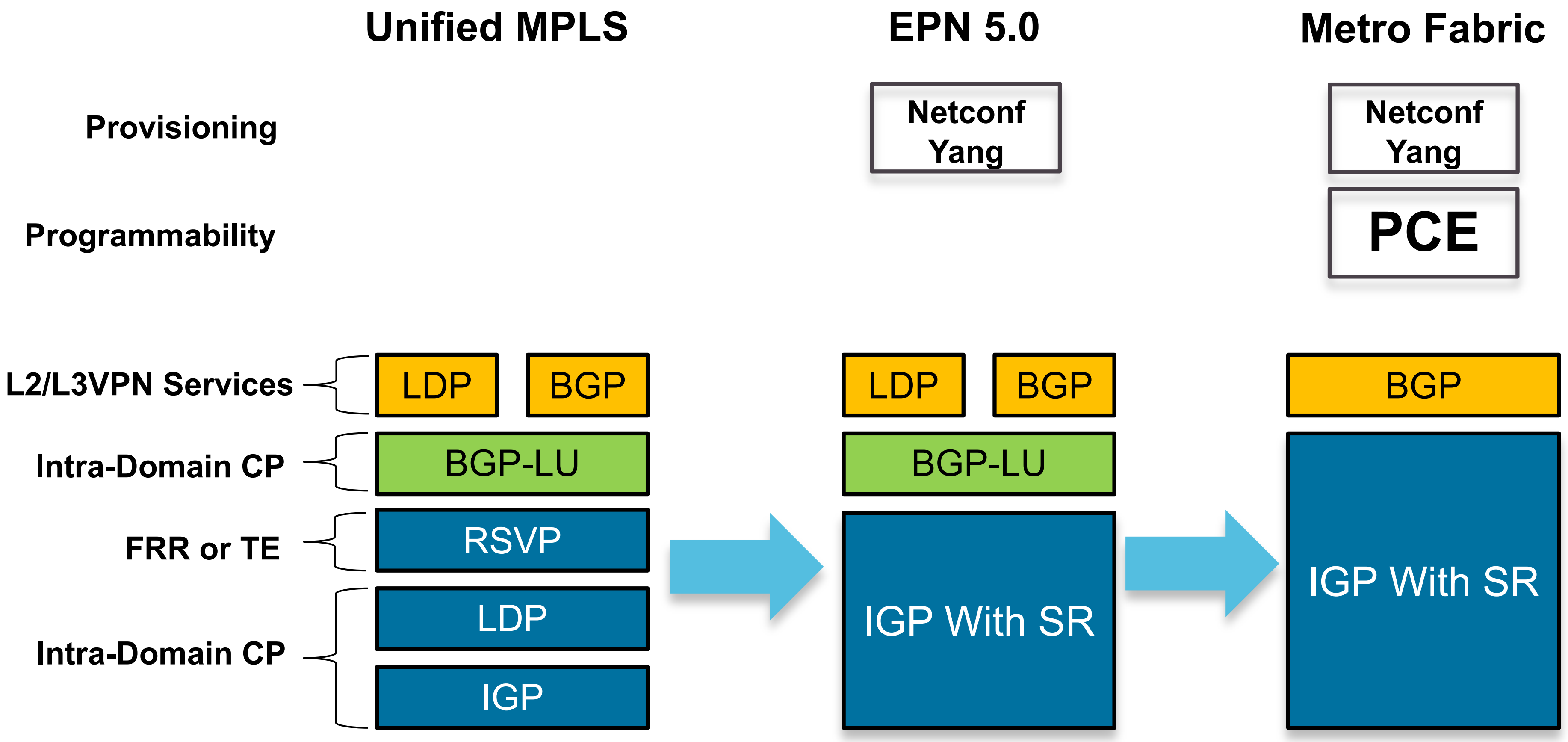
TLV 22

Sub-TLV 32
LAN-Adj-SID

LAN-Adj-SID
24001

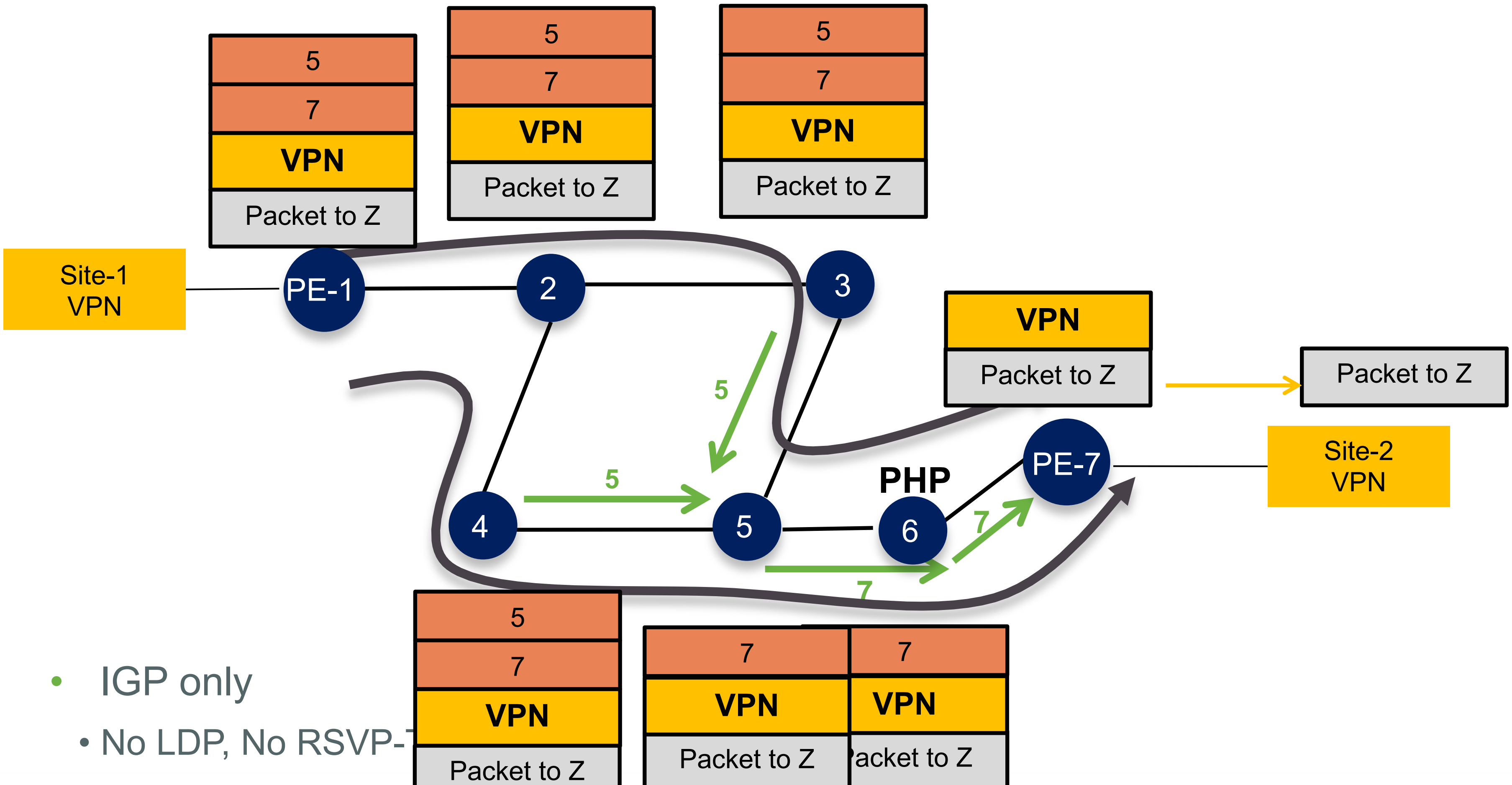
MENOG 18

Use Cases



Do More With Less

IPv4/v6 VPN/Service transport



- IGP only
  - No LDP, No RSVP-T
- ECMP multi-hop shortest-path

MENOG 18

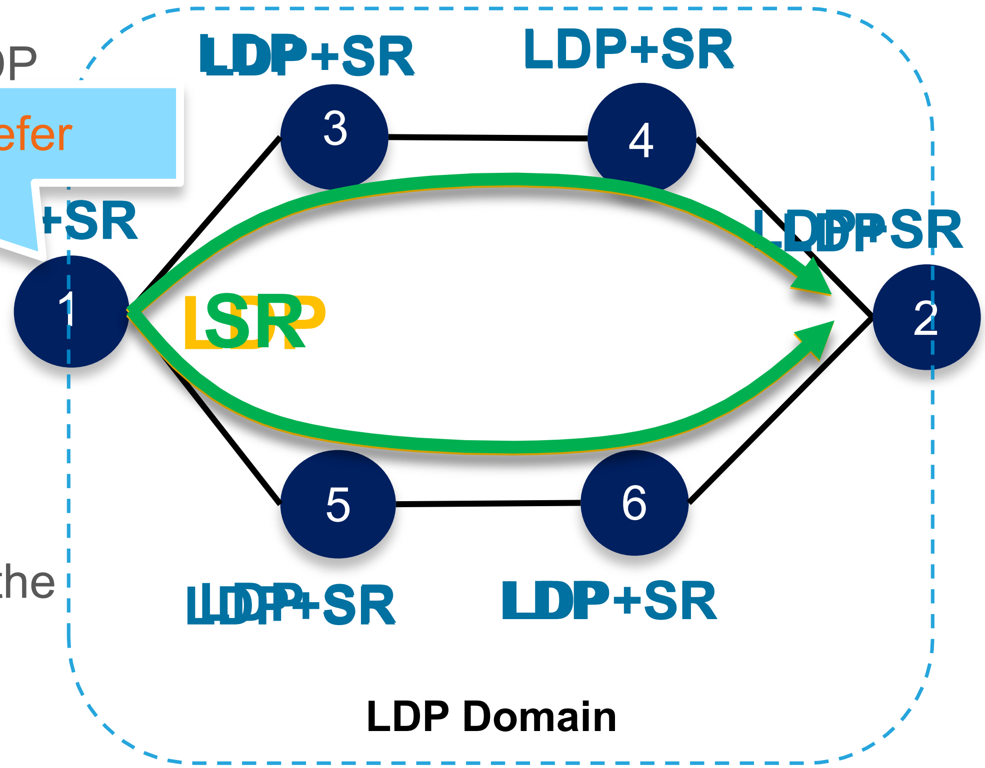
Internetworking With LDP

Simplest Migration: LDP to SR

Initial state: All nodes run LDP, not SR

Step1: All nodes are upgraded to SR
- in no particular order
- Default label imposition preference = LDP
- Leave c
  preferen

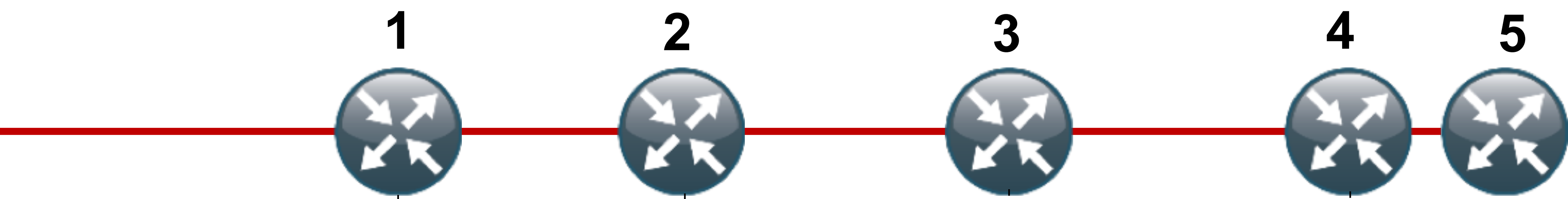
segment-routing mpls sr-prefer



Step2: All PEs are configured to prefer SR Label imposition
- in no particular order

Step3: LDP is removed from the nodes in the network
- in no particular order

Final State: All nodes run SR,Not LDP



segment-routing mpls sr-prefer

SRGB

Local/in	Out lbl
1600	
16005	16005
23999	
24000	
24002	24001

Local/in lbl	Out lbl
16000	
16005	24005
23999	
24000	
24001	32011
1048575	

Local/in lbl	Out lbl
16000	
23999	
24000	
24005	16005
31999	
32011	24003
1048575	

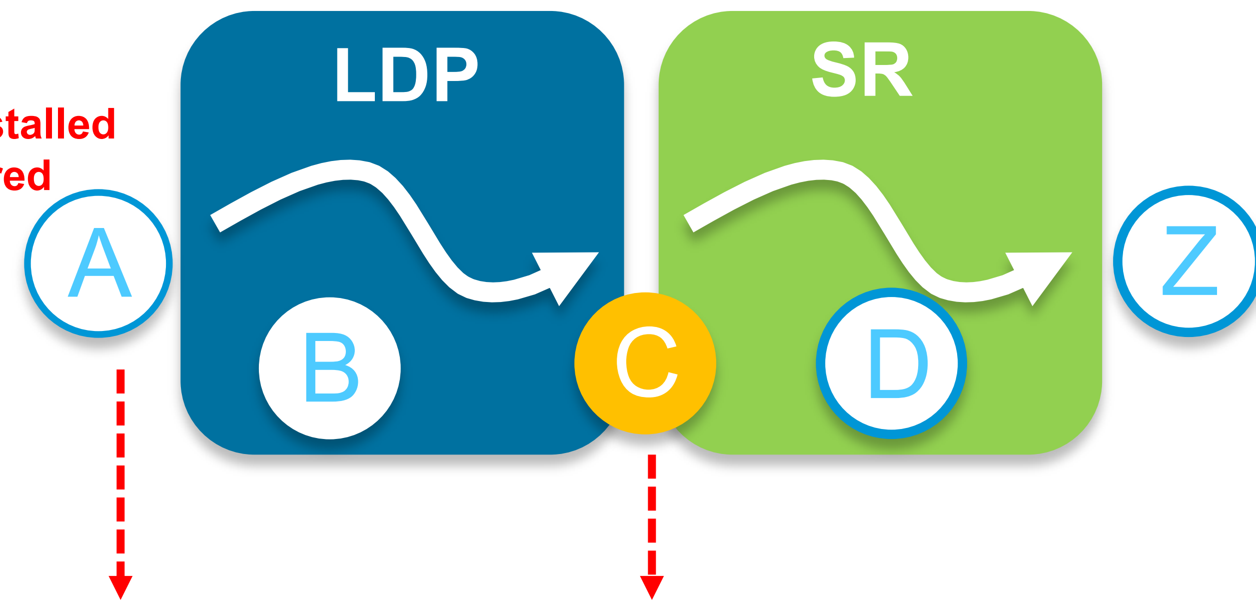
Local/in lbl	Out lbl
16000	
16005	pop
23999	
24000	
24003	pop
1048575	

segment-routing mpls (defualt)

LDP/SR Interworking - LDP to SR

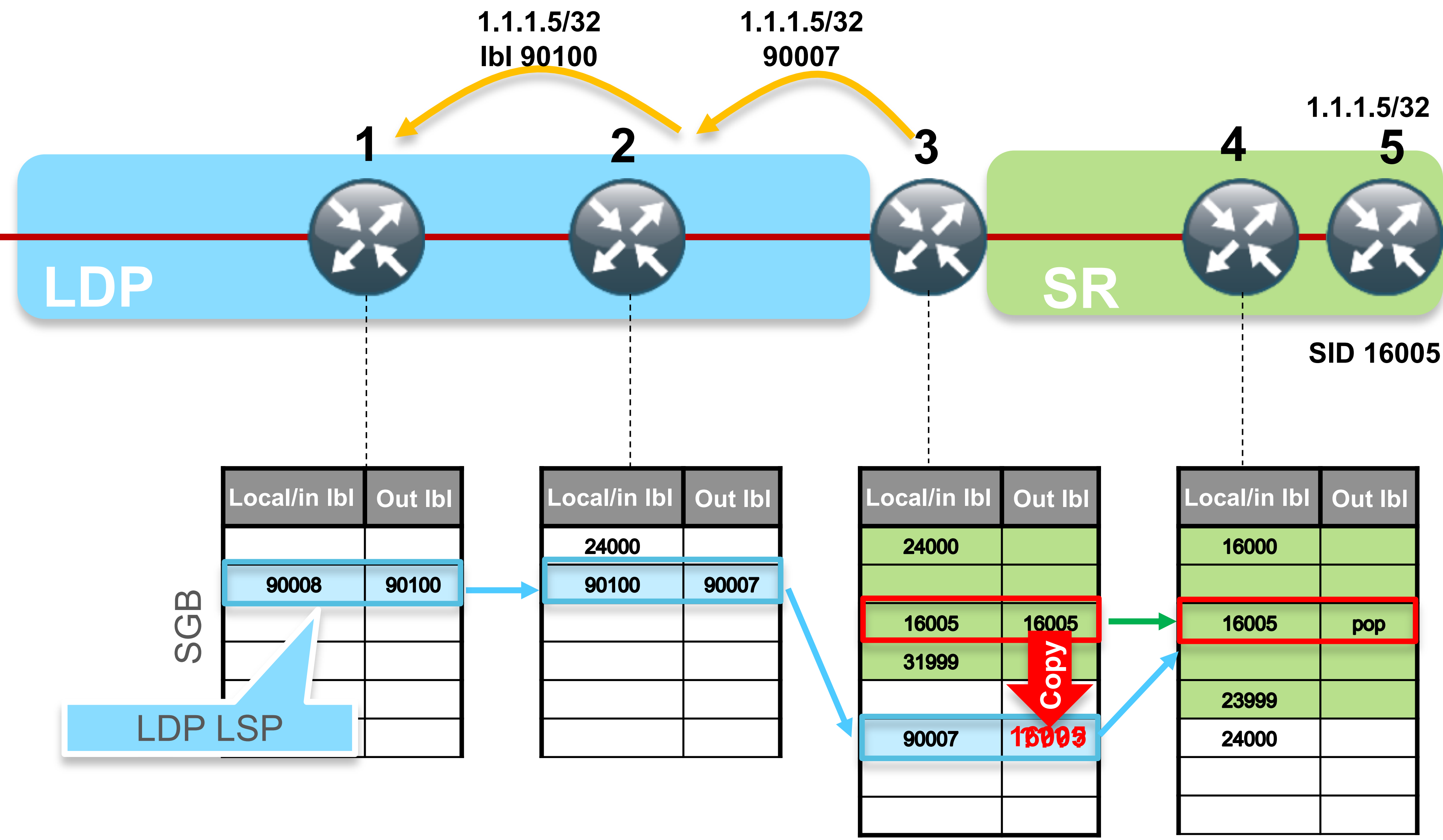
- When a node is LDP capable but its next-hop along the SPT to the destination is not LDP capable
  - no LDP outgoing label
- In this case, the LDP LSP is connected to the prefix segment
- C installs the following LDP-to-SR FIB entry:
- incoming label: label bound by LDP for FEC Z
- outgoing label: prefix segment bound to Z
- outgoing interface: D

- **This entry is derived and installed automatically , no config required**



Prefix	Out Label (LDP), Interface
Z	16, 0

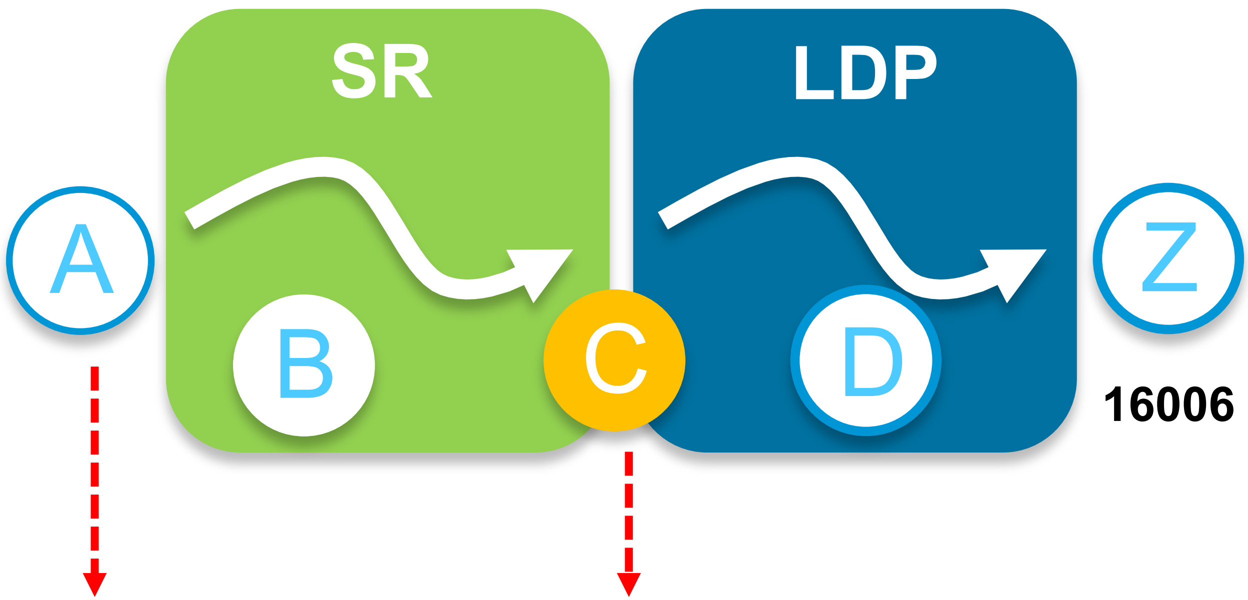
Input label(LDP)	Out Label (SID), Interface
32	16006, 1



1.1.1.5/32
lbl 90100
1.1.1.5/32
90007
1.1.1.5/32
1
2
3
4
5
LDP
SR
SID 16005
SGB
LDP LSP
Local/in lbl
Out lbl
90008
90100
Local/in lbl
Out lbl
24000
90100
90007
Local/in lbl
Out lbl
24000
16005
16005
31999
90007
Copy
Local/in lbl
Out lbl
16000
16005
pop
23999
24000

LDP/SR Interworking - SR to LDP

- When a node is SR capable but its next-hop along the SPT to the destination is not SR capable
- no SR outgoing label available
- In this case, the prefix segment is connected to the LDP LSP
- Any node on the SR/LDP border installs SR-to-LDP FIB entry(ies)

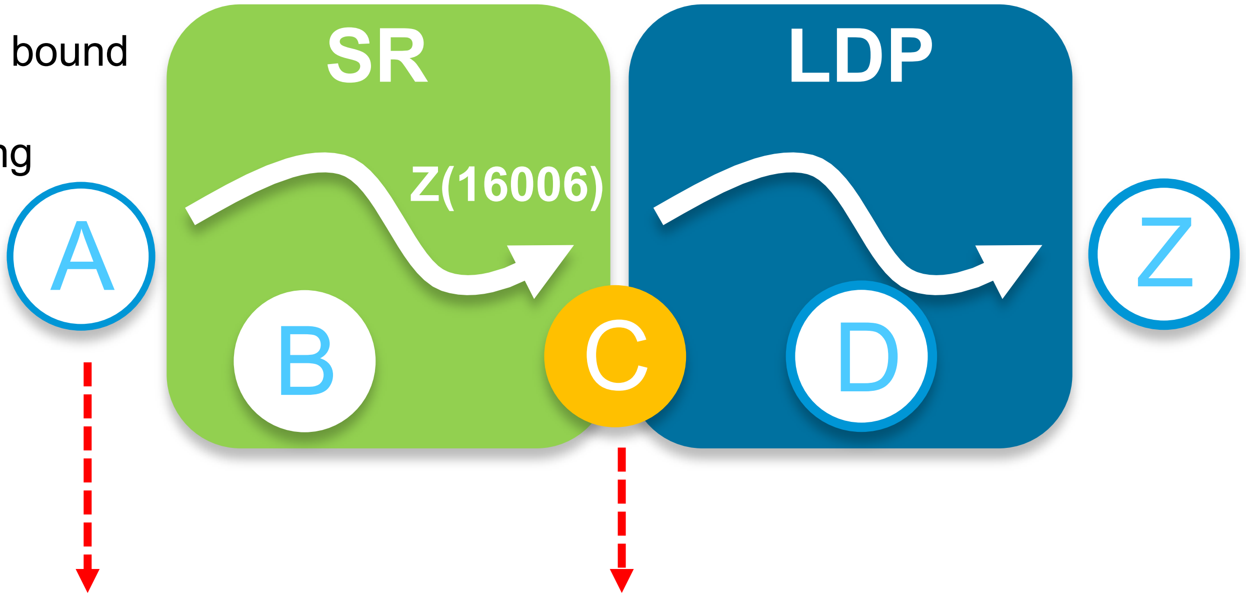


Prefix	Out Label (SID), Interface
Z	?, 0

Input Label(SID)	Out Label (LDP), Interface
?	16, 1

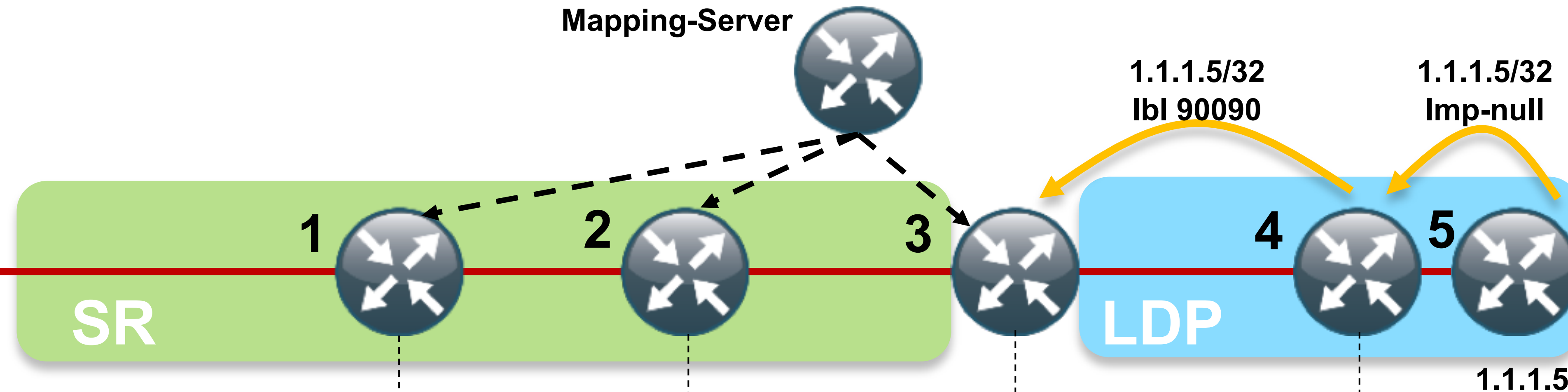
LDP/SR Interworking - Mapping Server

- A wants to send traffic to Z, but
  - Z is not SR-capable, Z does not advertise any prefixSID
  - → which label does A have to use?
- The **Mapping Server** advertises the SID mappings for the non-SR routers
  - for example, it advertises that Z is 16066
- A and B install a normal SR prefix segment for 16066
- C realizes that its next hop along the SPT to Z is not SR capable hence C installs an SR-to-LDP FIB entry
  - incoming label: prefix-SID bound to Z (16066)
  - outgoing label: LDP binding from D for FEC Z
- A sends a frame to Z with a single label: 16006



Prefix	Out Label (SID), Interface
Z	16006, 0

Input Label(SID)	Out Label (LDP), Interface
16006	16, 1



Local/in lbl	Out lbl
16000	
16005	16005
23999	

Local/in lbl	Out lbl
16000	
16005	pop
23999	

Local/in lbl	Out lbl
16000	
16005	90090 NA
23999	
90002	90090

Copy

?

Local/in lbl	Out lbl
90090	pop
90090	pop

MENOG 18

Traffic Protection

Classic Per-Prefix LFA – disadvantages

• Classic LFA has disadvantages:
– Incomplete coverage, topology dependent
– Not always providing most optimal backup path

→Topology Independent LFA (TI-LFA) solves these issues

Classic LFA Rules

General Theory - Rules

RFC 5286

Loop Free Alternate

Inequality 1: $D(N,D) < D(N,S) + D(S,D)$

“Path is loop-free because N’s best path is not through local router.”
Traffic sent to backup next hop is not sent back to S.

Downstream Path

Inequality 2: $D(N,D) < D(S,D)$

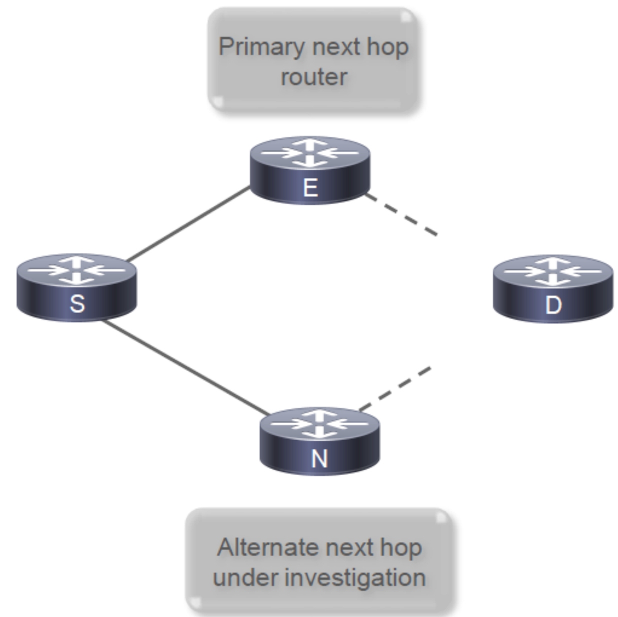
“Neighbor router is closer to the destination than local router.”
Loop-free is guaranteed even with multiple failures (if all repair-paths are downstream path).

Node protection

Inequality 3: $D(N,D) < D(N,E) + D(E,D)$

“N's path to D must not go through E.”
“The distance from the node N to the prefix via the primary next-hop is strictly greater than the optimum distance from the node N to the prefix.“

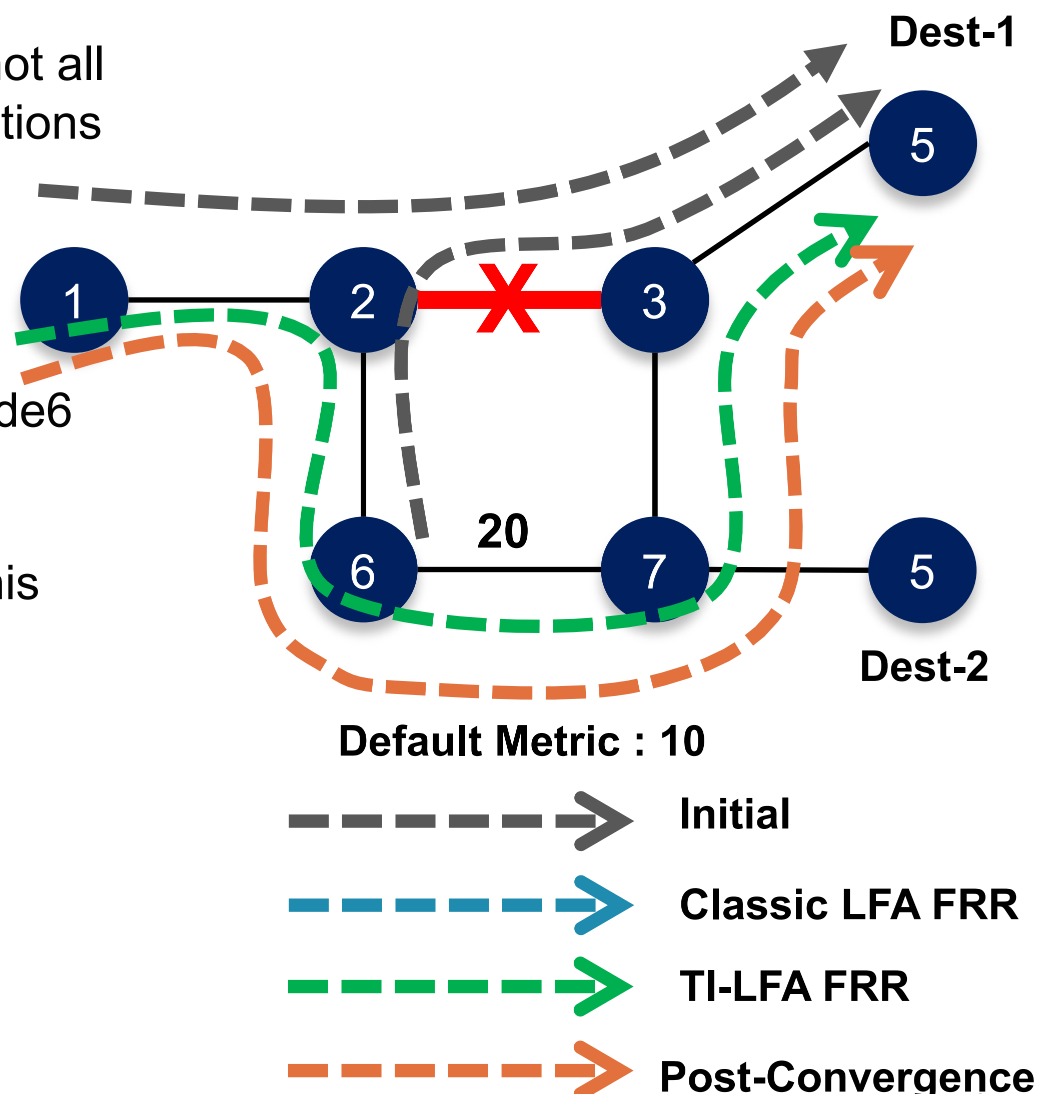
coverage



Classic LFA has partial coverage

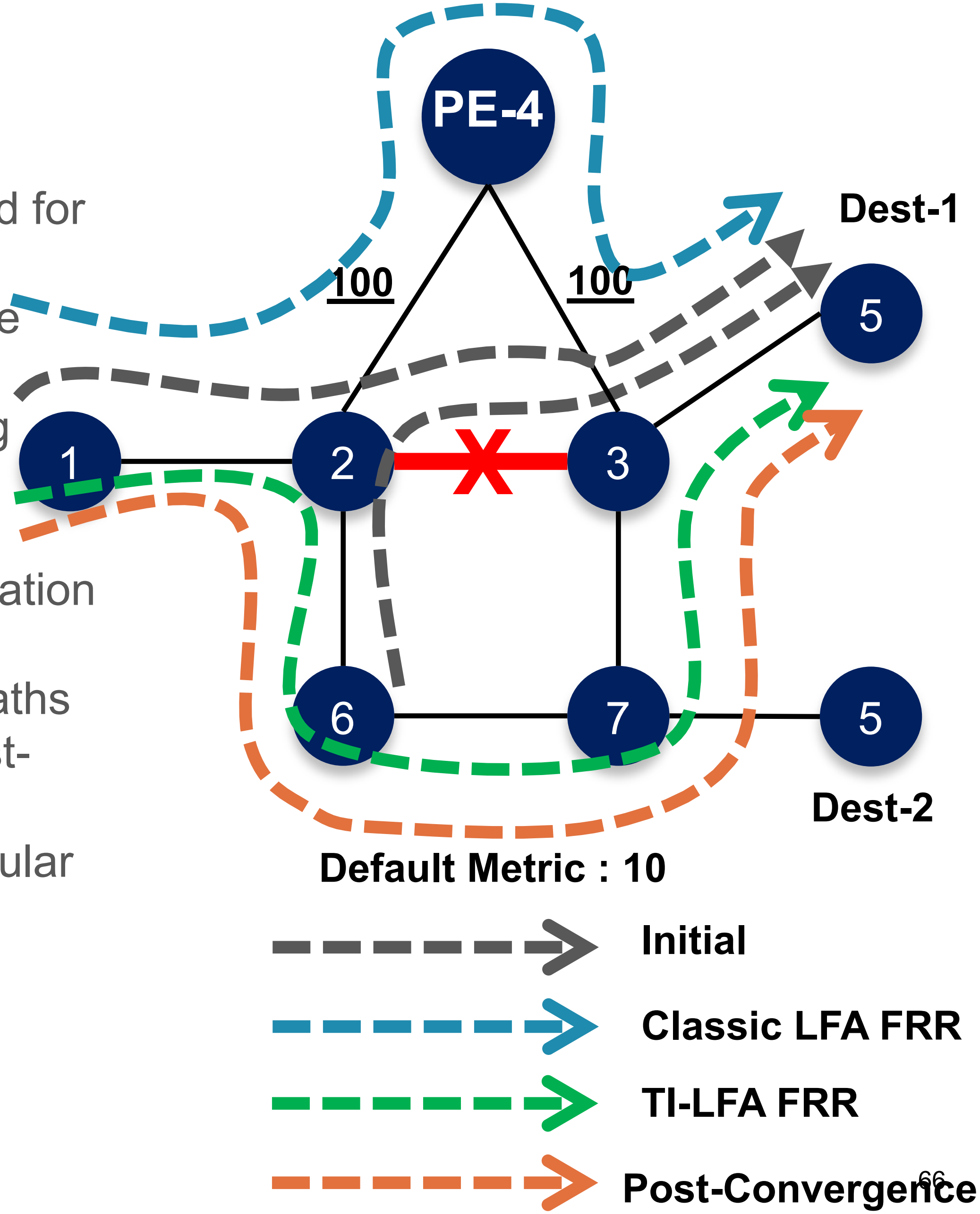
Classic LFA is topology dependent: not all topologies provide LFA for all destinations
– Depends on network topology and metrics
– E.g. Node6 is not an LFA for Dest1 (Node5) on
Node2, packets would loop since Node6 uses Node2
to reach Dest1 (Node5)
→Node2 does not have an LFA for this destination
(no → backup path in topology)

Topology Independent LFA (TI-LFA) provides 100% coverage



Classic LFA and suboptimal path

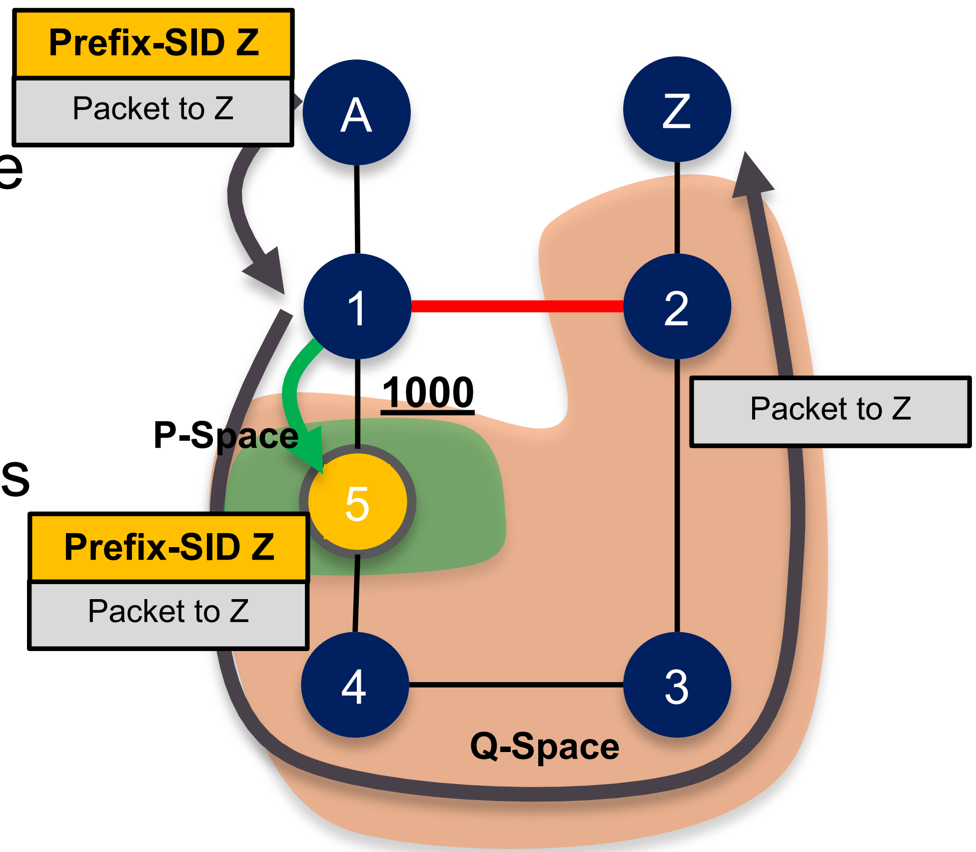
Classic LFA may provide a suboptimal
FRR backup
path:
– This backup path may not be planned for
capacity, e.g. P
node 2 would use PE4 to protect a core
link, while a
common planning rule is to avoid using
Edge nodes for
transit traffic
– Additional case specific LFA configuration
would be needed
to avoid selecting undesired backup paths
– Operator would prefer to use the post-
convergence path as
FRR backup path, aligned with the regular
IGP
convergence
→ TI-LFA uses the post-convergence
path as FRR backup path



TI-LFA – Zero-Segment Example

- TI-LFA for link R1R2 on R1
- Calculate LFA(s)

- Compute post-convergence SPT
- Encode post-convergence path in a SID-list
- In this example R1 forwards the packets towards R5

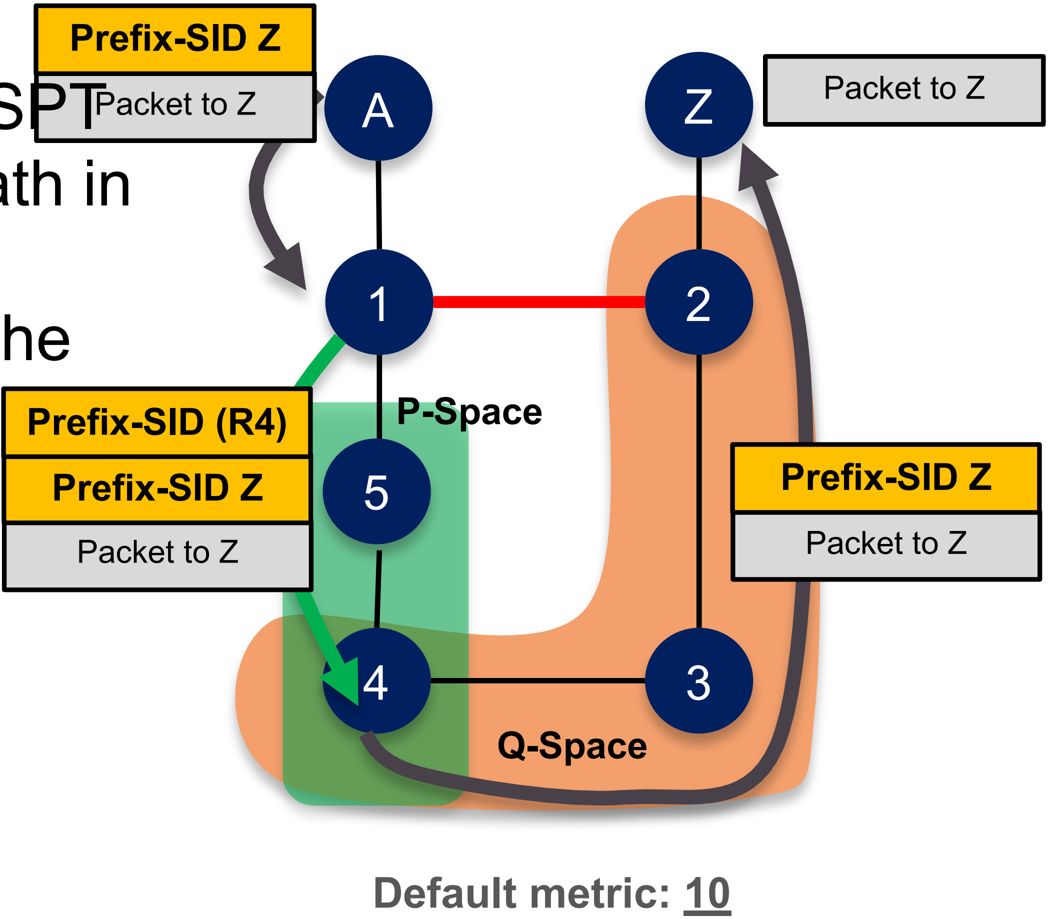


Default metric: 10

TI-LFA – Single-Segment Example

- TI-LFA for link R1R2 on R1

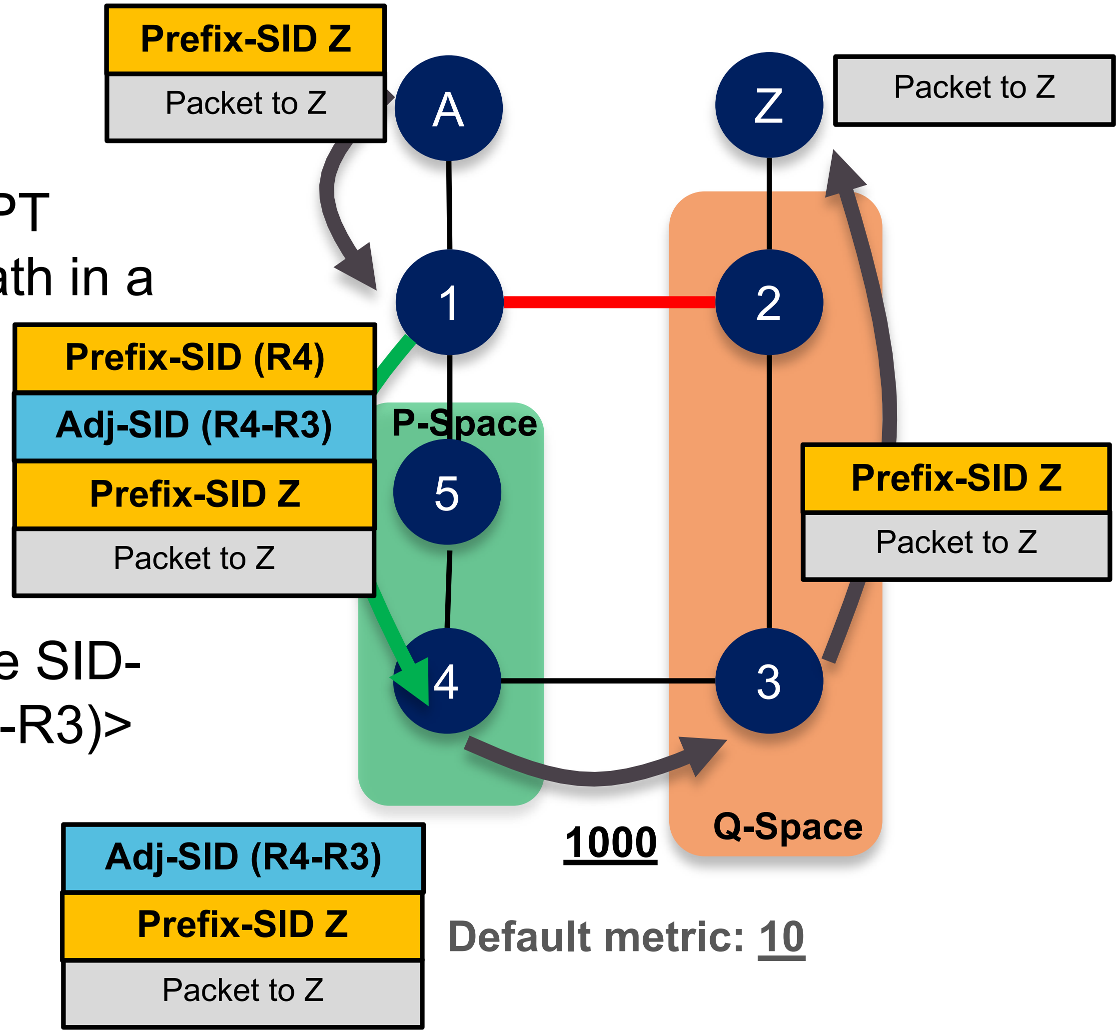
- Compute post-convergence SPT
- Encode post-convergence path in a SID-list
- In this example R1 imposes the SID-list <Prefix-SID(R4)> and sends packets towards R5



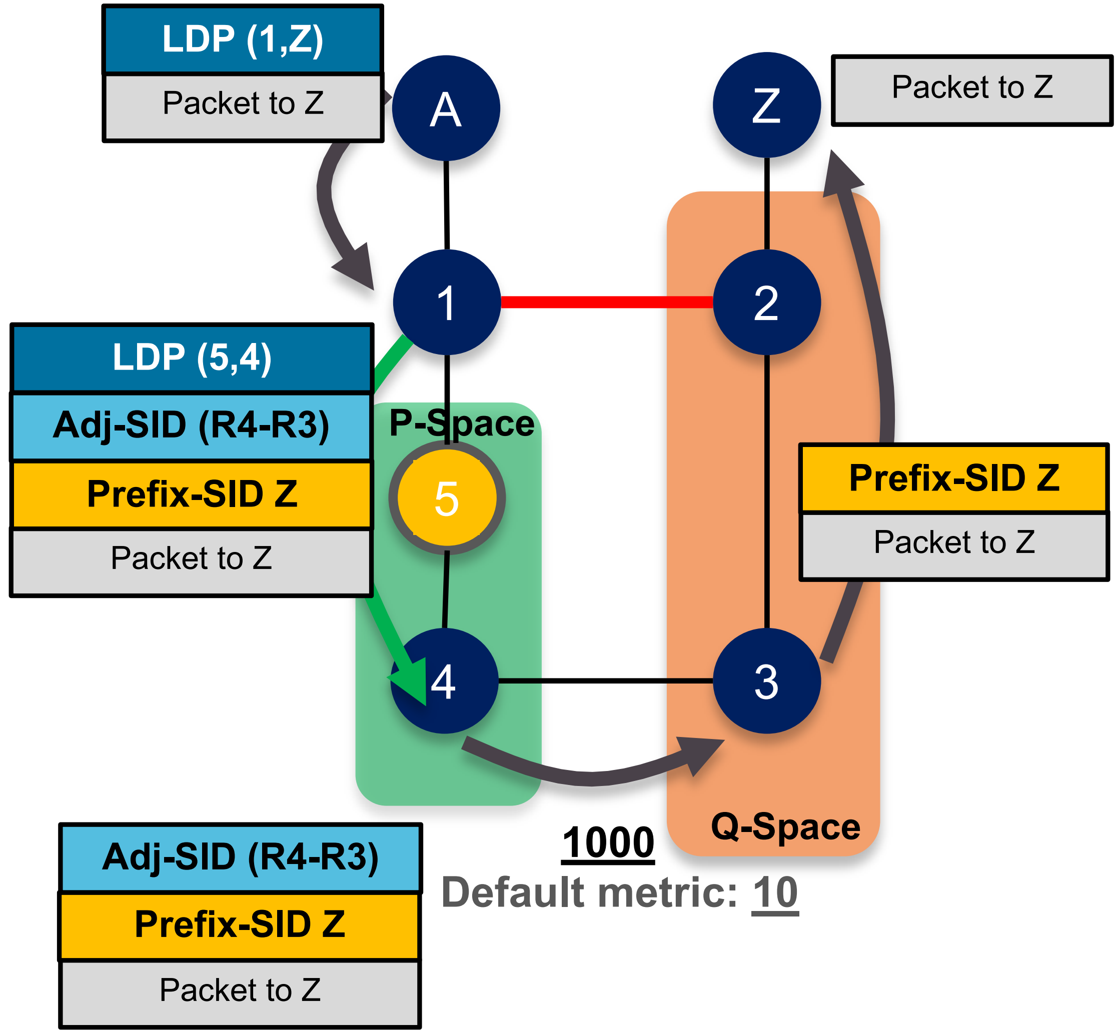
TI-LFA – Double-Segment Example

TI-LFA for link R1R2 on R1
- Compute post-convergence SPT
- Encode post-convergence path in a SID-list

- In this example R1 imposes the SID-list <Prefix-SID(R4), Adj-SID(R4-R3)> and sends packets towards R5



TI-LFA for LDP Traffic



MENOG 18

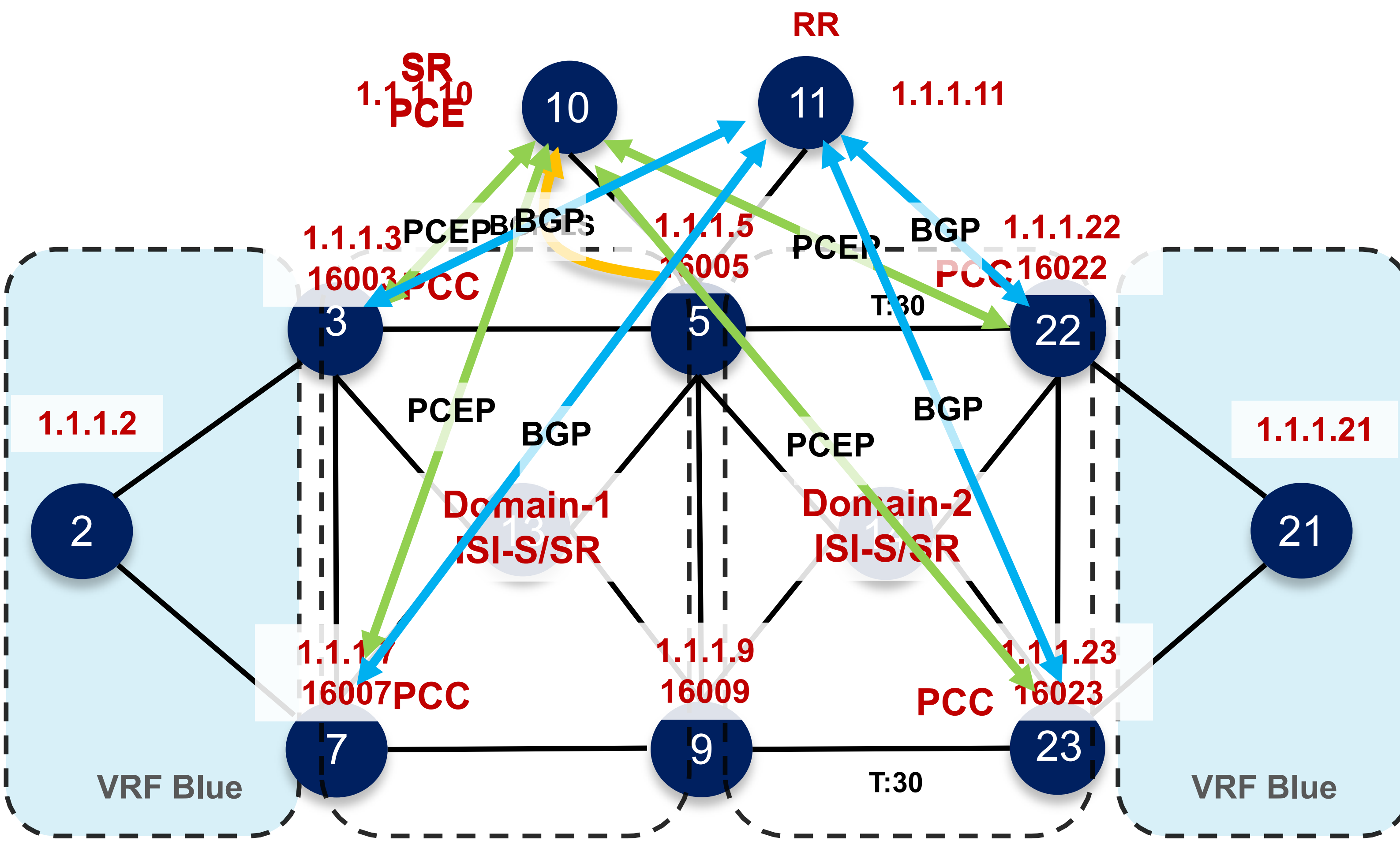
Traffic Engineering

RSVP-TE

- Little deployment and many issues
- Not scalable
  - Core states in k×n2
  - No inter-domain
- Complex configuration
  - Tunnel interfaces
- Complex steering
  - PBR, autoroute
- Does not support ECMP

SRTE

- Simple, Automated and Scalable
  - No core state: state in the packet header
  - No tunnel interface: "SR Policy"
  - No head-end a-priori configuration: on-demand policy instantiation
  - No head-end a-priori steering: automated steering
- Multi-Domain
  - SDN Controller for compute
  - Binding-SID (BSID) for scale
- Lots of Functionality
  - Designed with lead operators along their use-cases
- Provides explicit routing
- Supports constraint-based routing
- Supports centralized admission control
- No RSVP-TE to establish LSPs
- Uses existing ISIS / OSPF extensions to advertise link attributes
- Supports ECMP
- Disjoint Path

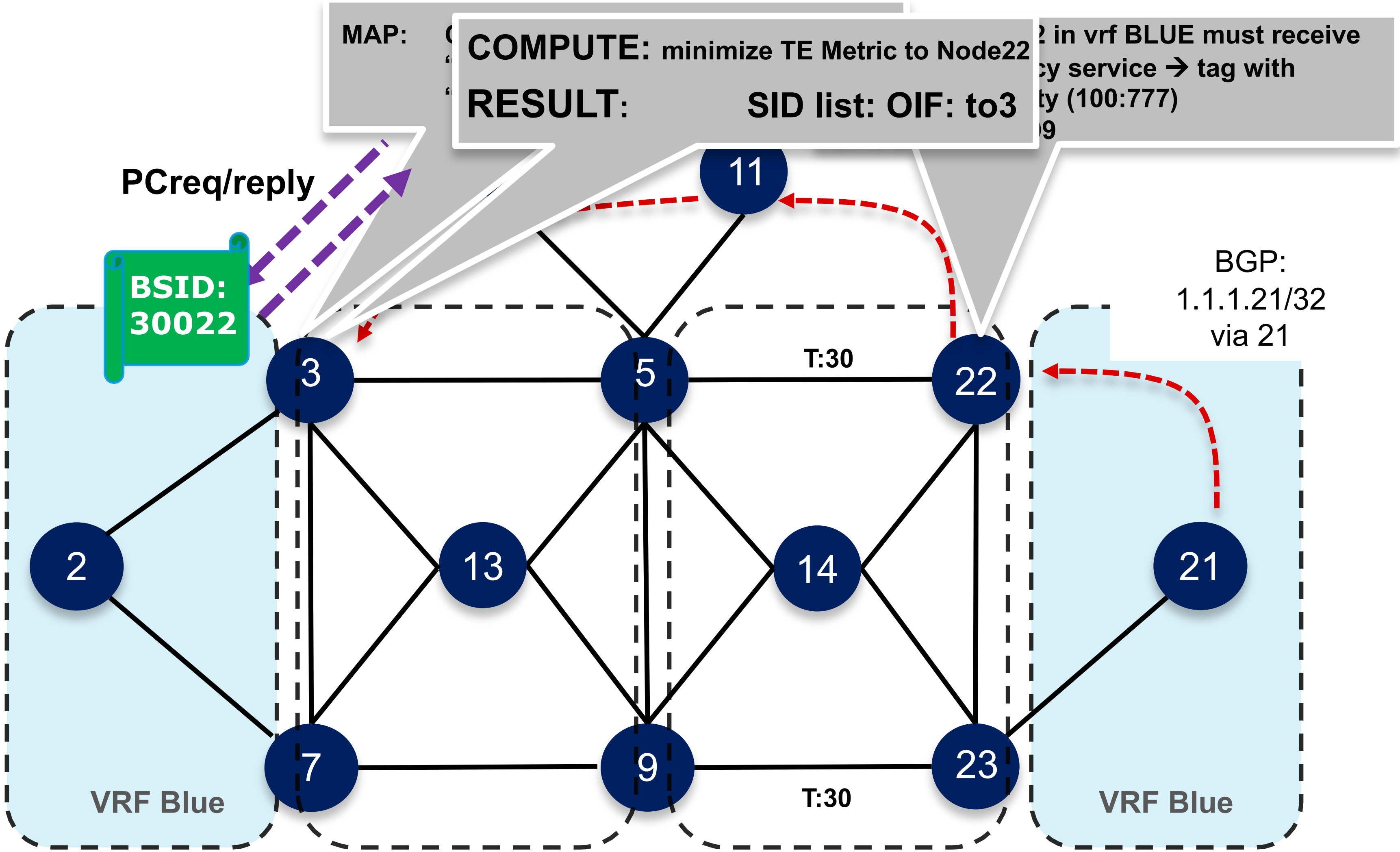


Router-id of NodeX: 1.1.1.X
Prefix-SID index of NodeX: X
Link address: 99.X.Y.X/24 with X < Y
Adj-SID XY: 240XY

Default IGP Metric: I:10
Default TE Metric: T:10
TE metric used to express latency

Domain-1
ISI-S/SR

Domain-2
ISI-S/SR



Automated Steering uses color extended communities and nexthop to match with the color and end-point of an SR Policy
E.g. BGP route 2/8 with nexthop 1.1.1.1 and color 100
will be steered into an SR Policy with color 100 and end-point 1.1.1.1
If no such SR Policy exists, it can be instantiated automatically (ODN)

MENOG 18

SRv6

SRv6 for underlay

SRv6 for Underlay

IPv6 for reach

Simplification, FRR, TE, SDN

Horrendous states scaling in k*N^2

Opportunity for further simplification

NSH for NFV

UDP+VxLAN Overlay

SRv6 for Underlay

IPv6 for reach

Additional Protocol and State

Additional Protocol just for tenant ID

Simplification, FRR, TE, SDN

- Multiplicity of protocols and states hinder network economics

- **IPV6 Header**
- Next Header (NH)
- Indicate what comes next

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|Version| Traffic Class |           Flow Label                  |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|         Payload Length        |  Next Header  |   Hop Limit   |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
|                         Source Address                        |
|                                                               |
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
|                      Destination Address                      |
|                                                               |
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

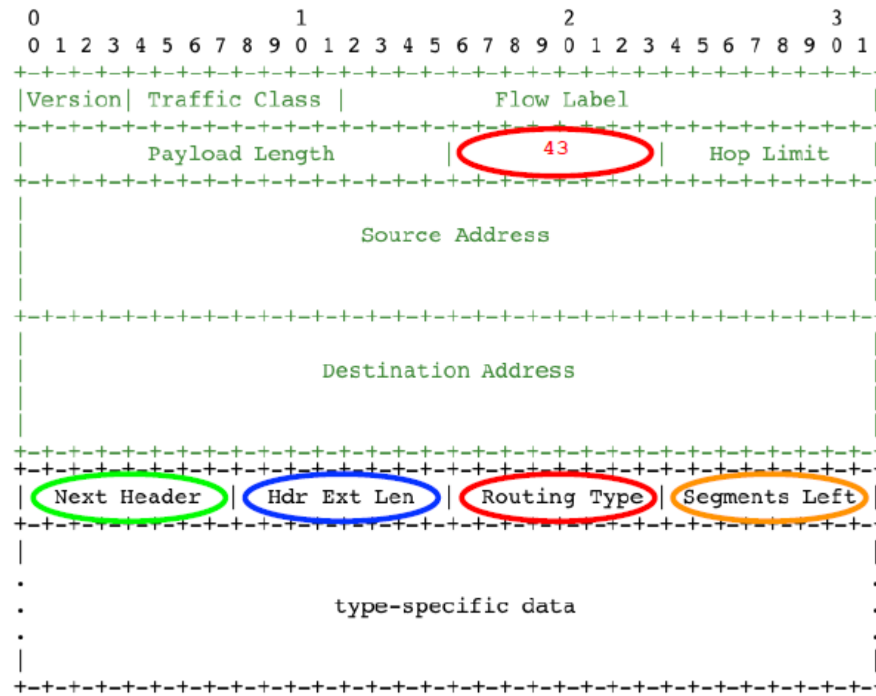
- **NH=IPv6**

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|Version| Traffic Class |           Flow Label                  |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|         Payload Length        |      41       |   Hop Limit   |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
|                         Source Address                        |
|                                                               |
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
|                      Destination Address                      |
|                                                               |
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

- **NH=IPv4**

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|Version| Traffic Class |           Flow Label                  |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|         Payload Length        |       4       |   Hop Limit   |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
|                         Source Address                        |
|                                                               |
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
|                      Destination Address                      |
|                                                               |
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

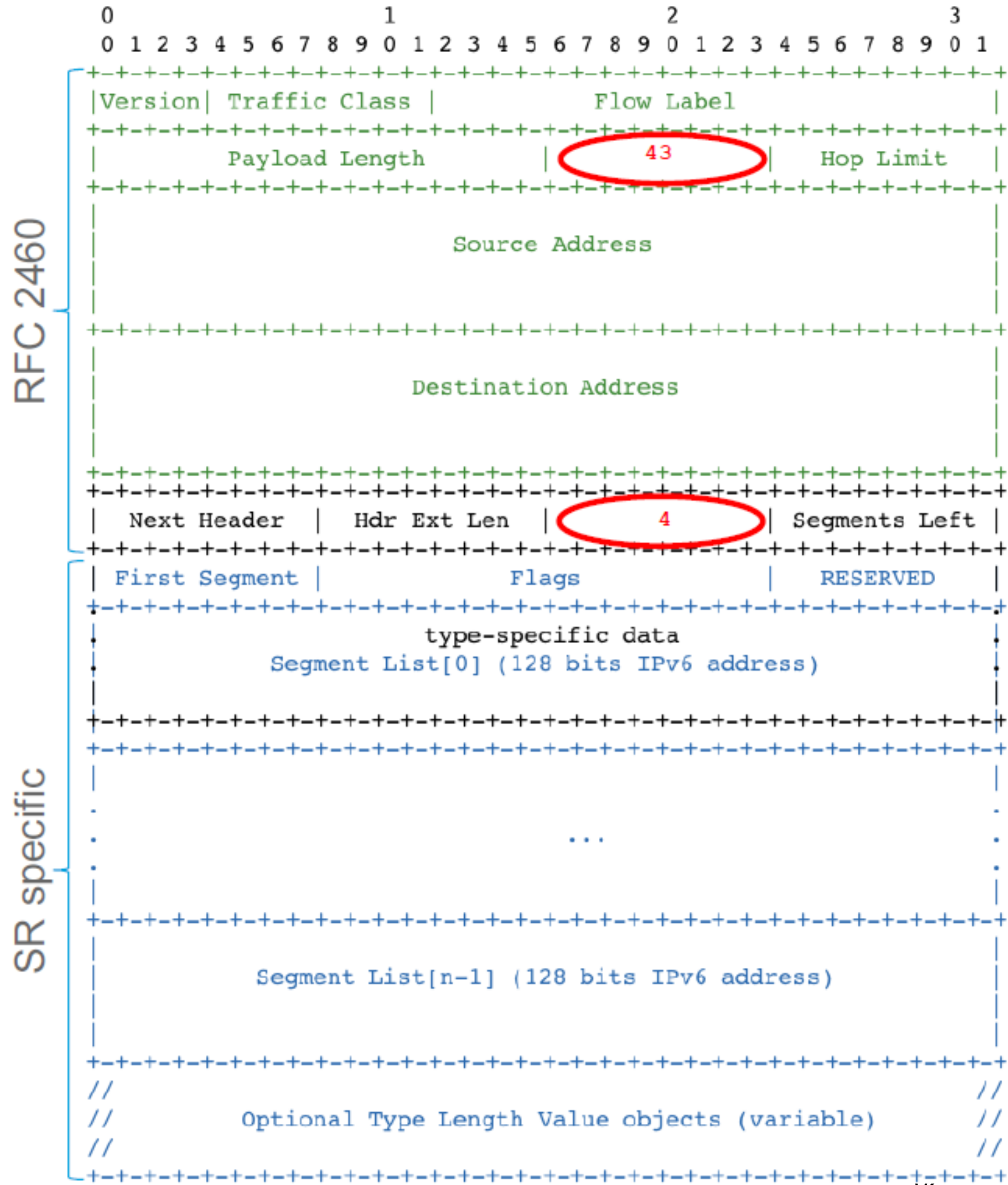
- **NH=Routing Extension**



- Generic routing extension header
  - Defined in RFC 2460
  - Next Header: UDP, TCP, IPv6…
  - Hdr Ext Len: Any IPv6 device can skip this header
  - Segments Left: Ignore extension header if equal to 0
- Routing Type field:
  - > 0 Source Route (deprecated since 2007)
  - > 1 Nimrod (deprecated since 2009)
  - > 2 Mobility (RFC 6275)
  - > 3 RPL Source Route (RFC 6554)
  - > 4 Segment Routing

- **NH=SRv6**

  NH=43,Type=4



Wireshark · Packet 5 · sr-header

```
> Frame 5: 182 bytes on wire (1456 bits), 182 bytes captured (1456 bits)
> Ethernet II, Src: 22:1a:95:d6:7a:23 (22:1a:95:d6:7a:23), Dst: 86:93:23:d3:37:8e (86:93:23:d3:37:8e)
v Internet Protocol Version 6, Src: fc00:42:0:1::2, Dst: fc00:2:0:5::1
    0110 .... = Version: 6
  v .... 0000 0000 .... .... .... .... .... = Traffic Class: 0x00 (DSCP: CS0, ECN: Not-ECT)
      .... 0000 00.. .... .... .... .... .... = Differentiated Services Codepoint: Default (0)
      .... .... ..00 .... .... .... .... .... = Explicit Congestion Notification: Not ECN-Capable Transport (0)
    .... .... .... 1111 1011 1011 0111 0100 = Flow Label: 0xfbb74
    Payload Length: 128
    Next Header: Routing Header for IPv6 (43)
    Hop Limit: 63
    Source: fc00:42:0:1::2
    Destination: fc00:2:0:5::1
    [Source GeoIP: Unknown]
    [Destination GeoIP: Unknown]
  v Routing Header for IPv6 (Segment Routing)
      Next Header: IPv6 (41)
      Length: 6
      [Length: 56 bytes]
      Type: Segment Routing (4)
      Segments Left: 2
      First segment: 2
    v Flags: 0x00
        0... .... = Unused: 0x0
        .0.. .... = Protected: False
        ..0. .... = OAM: False
        ...0 .... = Alert: Not Present
        .... 0... = HMAC: Not Present
        .... .000 = Unused: 0x0
      v [Expert Info (Note/Undecoded): Dissection for SRH TLVs not yet implemented]
          [Dissection for SRH TLVs not yet implemented]
          [Severity level: Note]
          [Group: Undecoded]
      Reserved: 0000
      Address[0]: fc00:2:0:6::1
      Address[1]: fc00:2:0:7::1 [next segment]
      Address[2]: fc00:2:0:5::1
    v [Segments in Traversal Order]
        Address[2]: fc00:2:0:5::1
        Address[1]: fc00:2:0:7::1 [next segment]
        Address[0]: fc00:2:0:6::1
> Internet Protocol Version 6, Src: fc00:2:0:1::1, Dst: fc00:2:0:2::1
> Transmission Control Protocol, Src Port: 8080, Dst Port: 43424, Seq: 1, Ack: 94, Len: 0
```

NH=43
Routing Extension

RT = 4

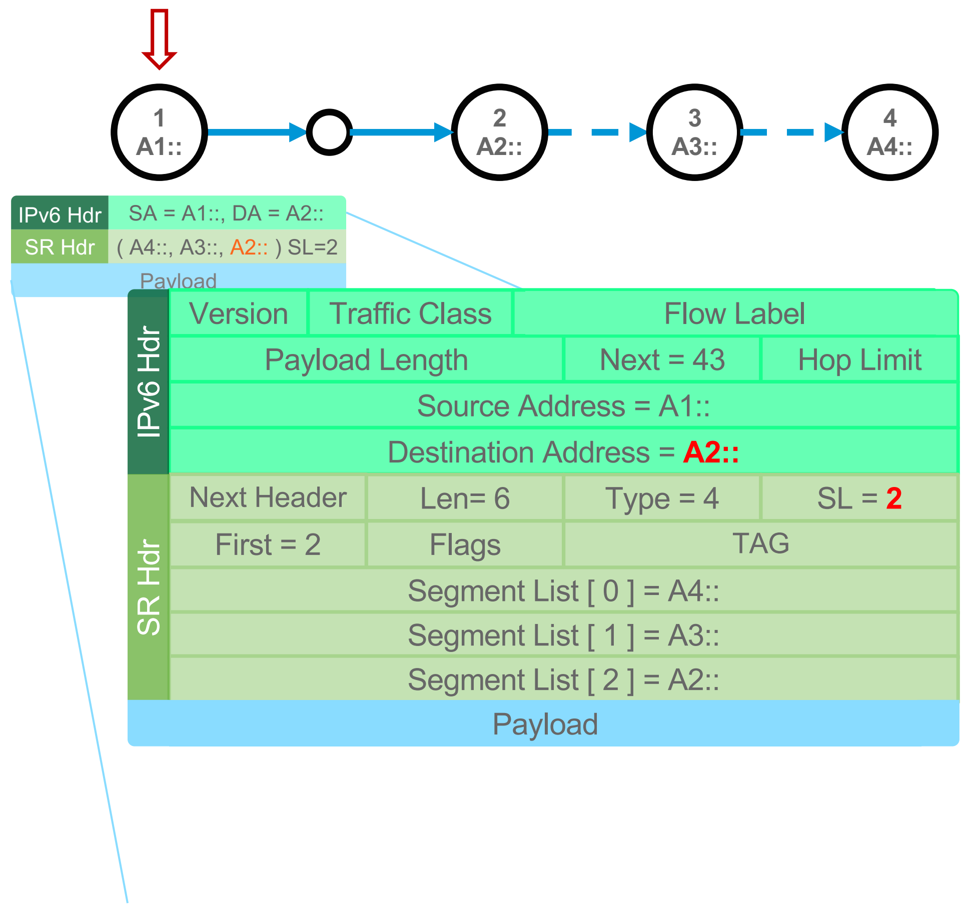
Segment-List

MENOG 18

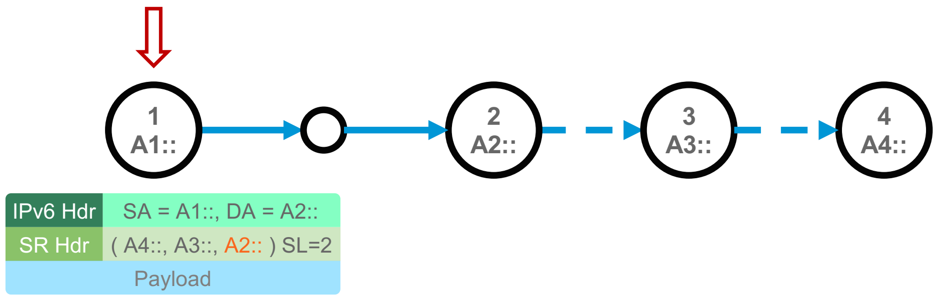
SRH Processing

Source Node

- Source node is SR-capable
- SR Header (SRH) is created with
  - Segment list in reversed order of the path
    - Segment List [0] is the LAST segment
    - Segment List [$n-1$] is the FIRST segment
  - Segments Left is set to $n-1$
  - First Segment is set to $n-1$
- IP DA is set to the first segment
- Packet is send according to the IP DA
  - Normal IPv6 forwarding



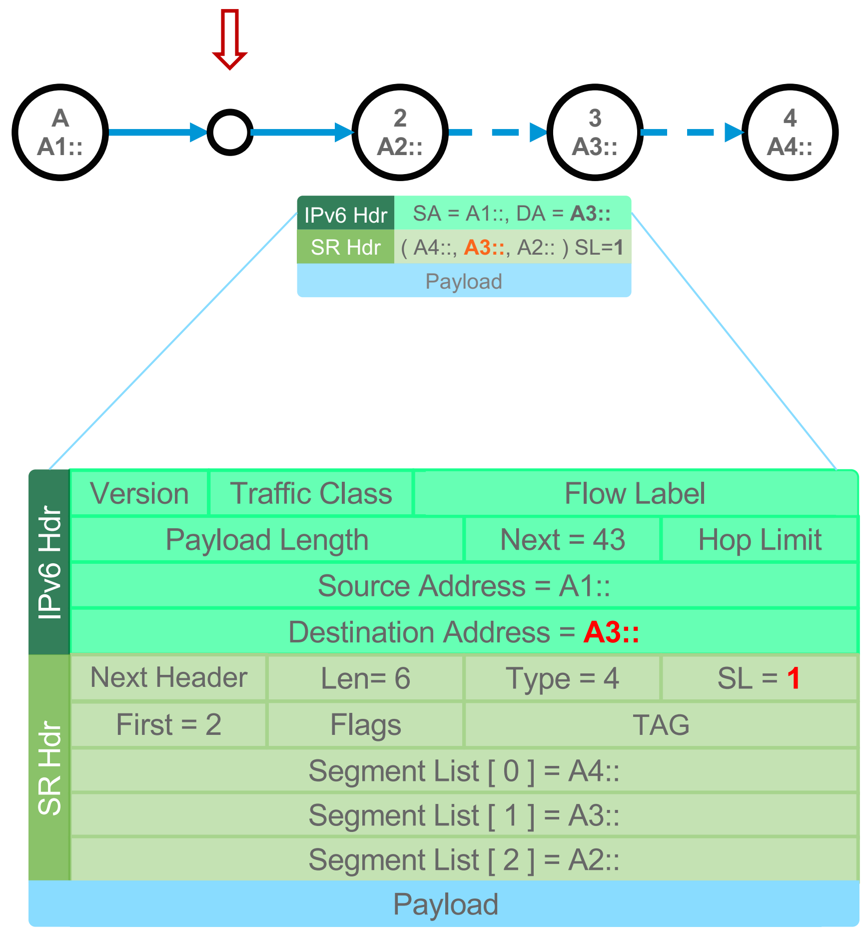
Non-SR Transit Node



- Plain IPv6 forwarding
- Solely based on IPv6 DA
- No SRH inspection or update

SR Segment Endpoints

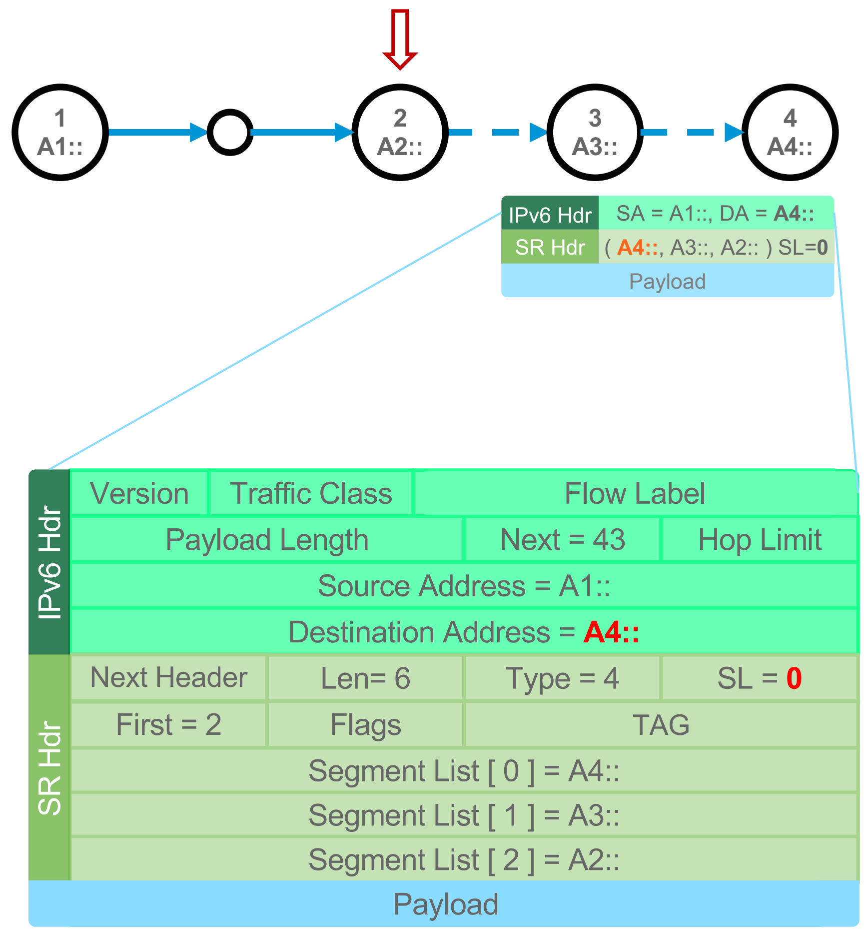
- SR Endpoints: SR-capable nodes whose address is in the IP DA
- SR Endpoints inspect the SRH and do:
  - IF Segments Left > 0, THEN
    - Decrement Segments Left (-1)
    - Update DA with Segment List [Segments Left]
    - Forward according to the new IP DA



SR Segment Endpoints

- SR Endpoints: SR-capable nodes whose address is in the IP DA
- SR Endpoints inspect the SRH and do:
  - IF Segments Left > 0, THEN
    - Decrement Segments Left (-1)
    - Update DA with Segment List [Segments Left]
    - Forward according to the new IP DA
  - ELSE (Segments Left = 0)
    - Remove the IP and SR header
    - Process the payload:
      - Inner IP: Lookup DA and forward
      - TCP / UDP: Send to socket
      - …

Standard IPv6 processing
The final destination does not have to be SR-capable.



Deployments around the world

- Bell in Canada
- Orange
- Microsoft
- SoftBank
- Alibaba
- Vodafone
- Comcast
- China Unicom

Deployments in IRAN

- IRAN TIC new Network is going to be implemented based on SR

Rasoul Mesghali : rasoul.mesghali@gmail.com

Vahid Tavajjohi : vahid.tavajjohi@gmail.com