Introduction to Segment Routing

A foundation for autonomous networking

Alberto Donzelli - Principal Solutions Engineer, Cisco
Cees de Gruijter - Network Domain Architect, Rijkswaterstaat NL
Michiel Koolen - Domain Architect, Rijkswaterstaat NL

BRKSP-2551

Agenda

- Introduction
- Standardization
- How SR works
- TI-LFA, Flex Algo
- Traffic Engineering
- SRv6 uSID overview
- SRv6 to the host
- Rijkswaterstaat SRv6
- Deployment and Interop
- Conclusions

Resiliency, Flexibility, Simplification

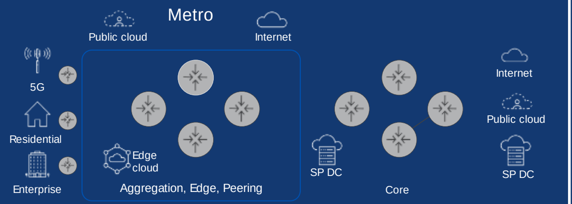
Economic value creation

One Network for all services

Self healing

AI/Autonomous networking

Service delivery is changing



- Quality of Experience is key
- Removing domain boundaries
- Flexible delivery points placement
- Always available

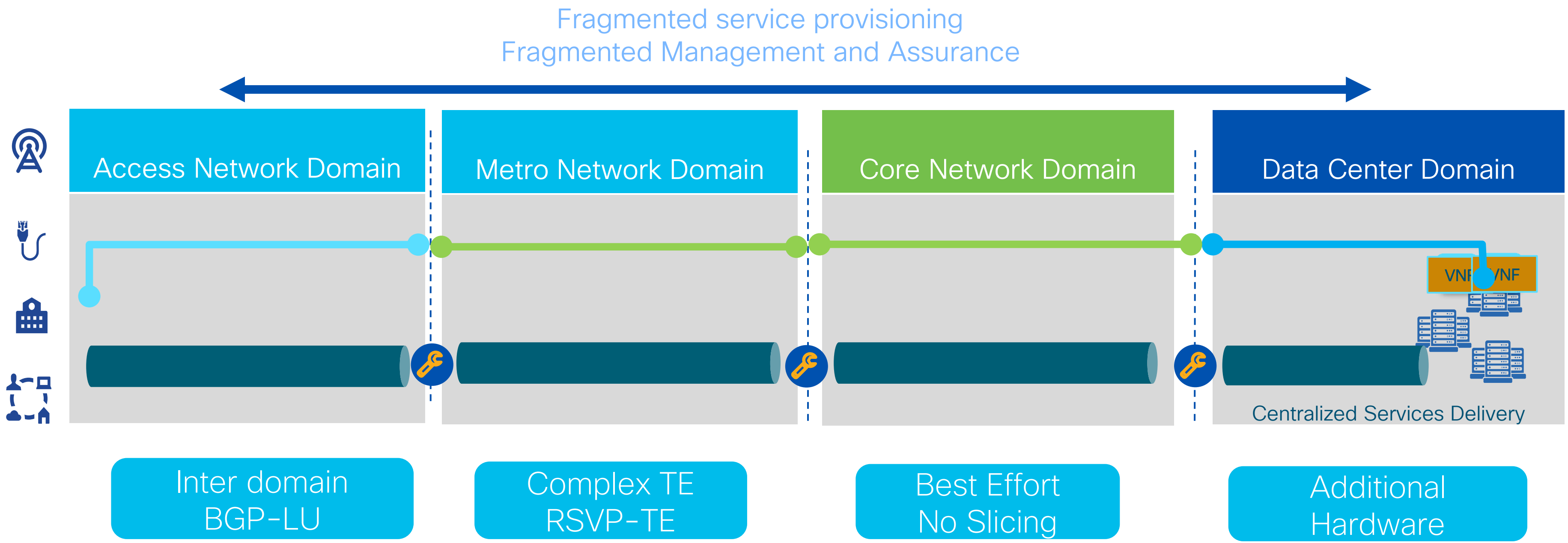
Programmable and resilient Fabric

SR Segment Routing

Connect Everything Everywhere

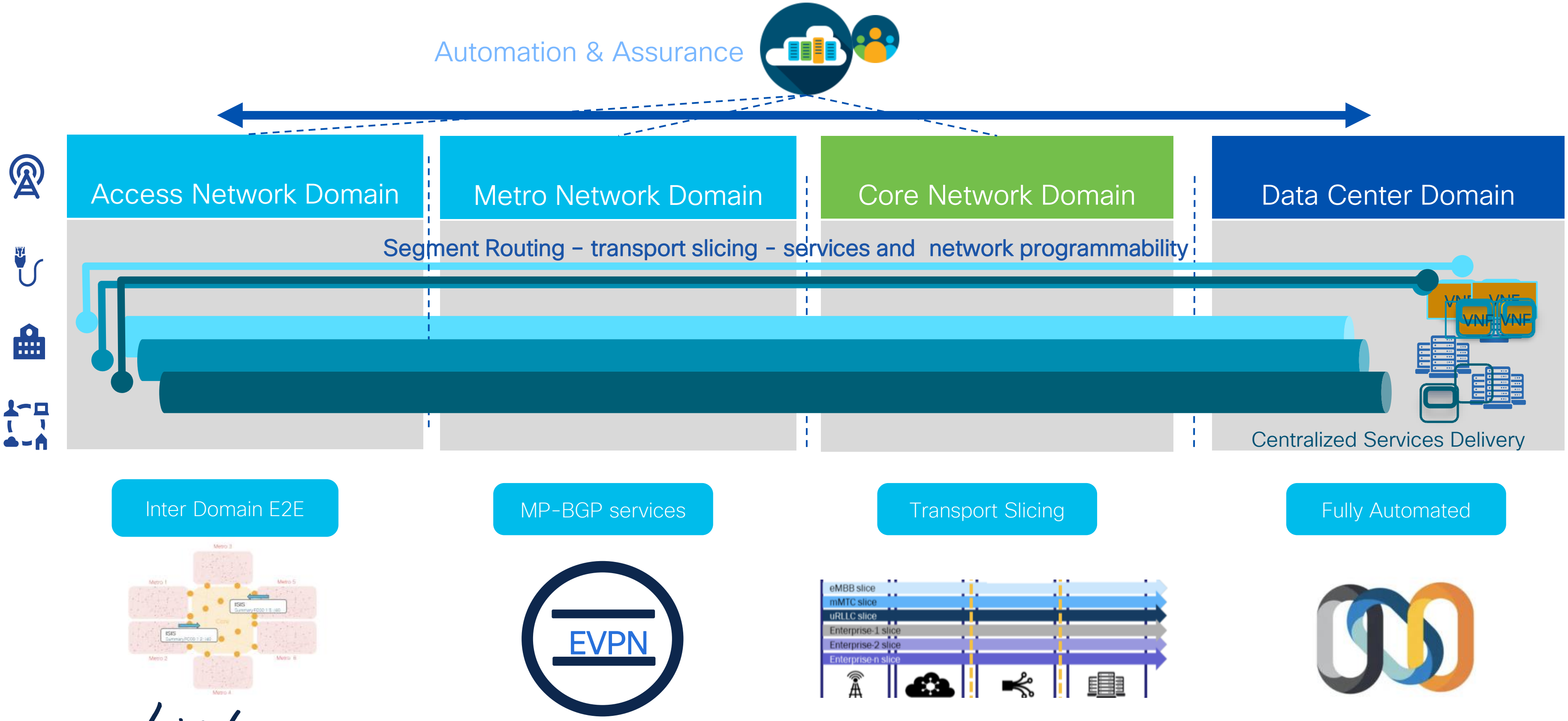
Traditional MPLS network Challenges

Inter domain connectivity, protocol complexity and limited SLA



The end goal

A multi-domain programmable fabric for service creation



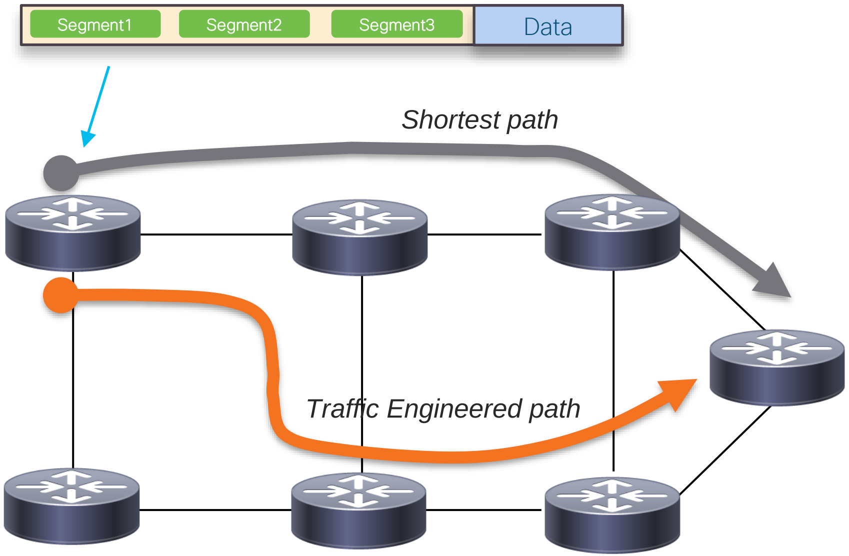
The path towards Autonomous Networks

"An autonomous network consists of a simplified network architecture, virtualized components, automating agents, and intelligent decision engines which present self-dynamic capabilities with the goal to create intelligent business and network operations based on the concept of closed-loop controls."

https://www.etsi.org/images/files/ETSIWhitePapers/etsi-wp-40-Autonomous-networks.pdf

Segment Routing

- Source Routing principle
  - Stateless IP fabric !!!
  - Path expressed in the packet



Data Plane

- MPLS (segment labels)
- IPv6 (IPv6 header + extension)

Control Plane

- Routing protocols with extensions (IS-IS, OSPF, BGP)
- SDN controller (BGP-LS, PCEP, NETCONF/YANG, gNMI)

Massive Protocol Symplification

Data Plane

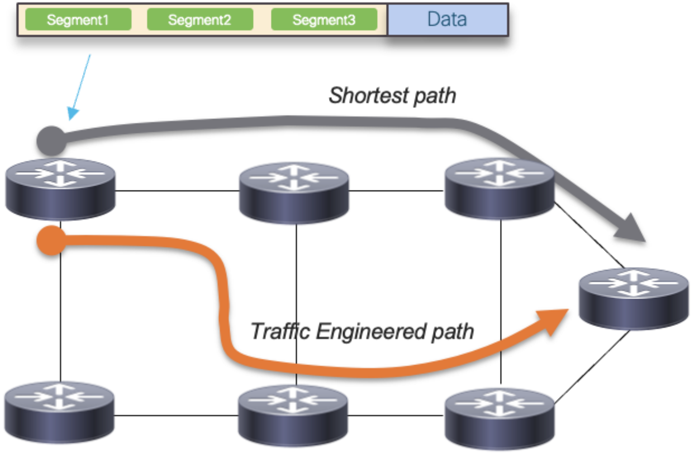
MPLS
(segment labels)

IPv6
(IPv6 header + extension)

Control Plane

Routing protocols with
extensions
(IS-IS, OSPF, BGP)

SDN controller
(BGP-LS, PCEP,
NETCONF/YANG, gNMI)



Before

L2/L3VPN Services	LDP, BGP
Inter-Domain Connectivity	BGP-LU
Protection FRR/TE	RSVP
Intra-Domain CP	LDP, IGP

After

L2/L3VPN EVPN	BGP
Inter-Domain Connectivity with SLA Traffic Engineering Protection FRR – TI-LFA Intra -Domain CP	IGP+ Segment Routing

One Architecture / Two Data-Plane possibilities

Segment Routing

SR-MPLS

- Instantiation of SR on the MPLS data plane
- A segment is encoded with an MPLS label

SRv6

- Instantiation of SR on the IPv6 data plane
- One or more segments are encoded within an IPv6 address

Standardization

Segment Routing Standardization IETF

- First RFC - 7855 (May 2016)

Active working groups

- IS-IS
- OSPF
- PCEP
- BGP
- IDR
- 6MAN

Strong Cisco Commitment and Leadership

Editor of	96% IETF RFCs
Co-author of	100% IETF RFCs

A comprehensive list @ www.segment-routing.net

Reference IETF drafts and RFCs

Architecture
- Segment Routing Architecture RFC 8402
- Segment Routing Policy Architecture RFC 9256

MPLS
- Segment Routing with MPLS data plane RFC 8660
- Segment Routing interworking with LDP RFC 8661
- SR-MPLS over IP RFC 8663

SRv6 Data Plane
- SRv6 Network Programming – RFC 8986
- IPv6 SR Header – RFC 8754
- Compressed SRv6 Segment List – WG Draft

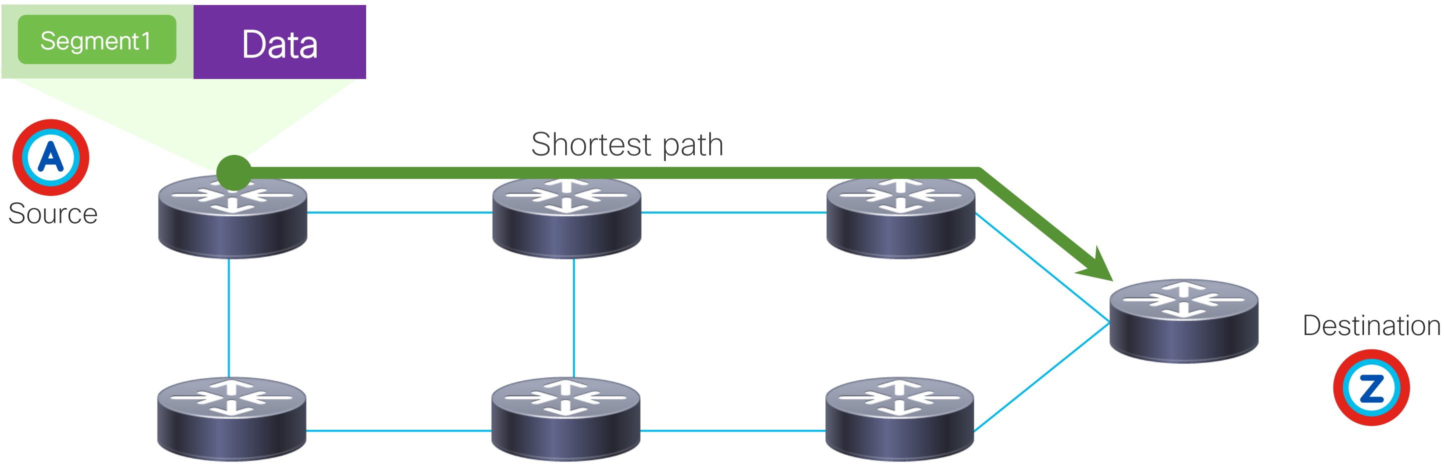
IS-IS
- IS-IS Extensions for Segment Routing RFC 8667
- IGP Flexible Algorithm RFC 9350
- IS-IS Traffic Engineering (TE) Metric Extensions RFC 7810
- SRv6 ISIS extensions – RFC 9352

OSPF
- OSPF Extensions for Segment Routing RFC 8665
- IGP Flexible Algorithm WG Document
- OSPF Traffic Engineering (TE) Metric Extensions RFC 7471

SR basic blocks

How does it work?

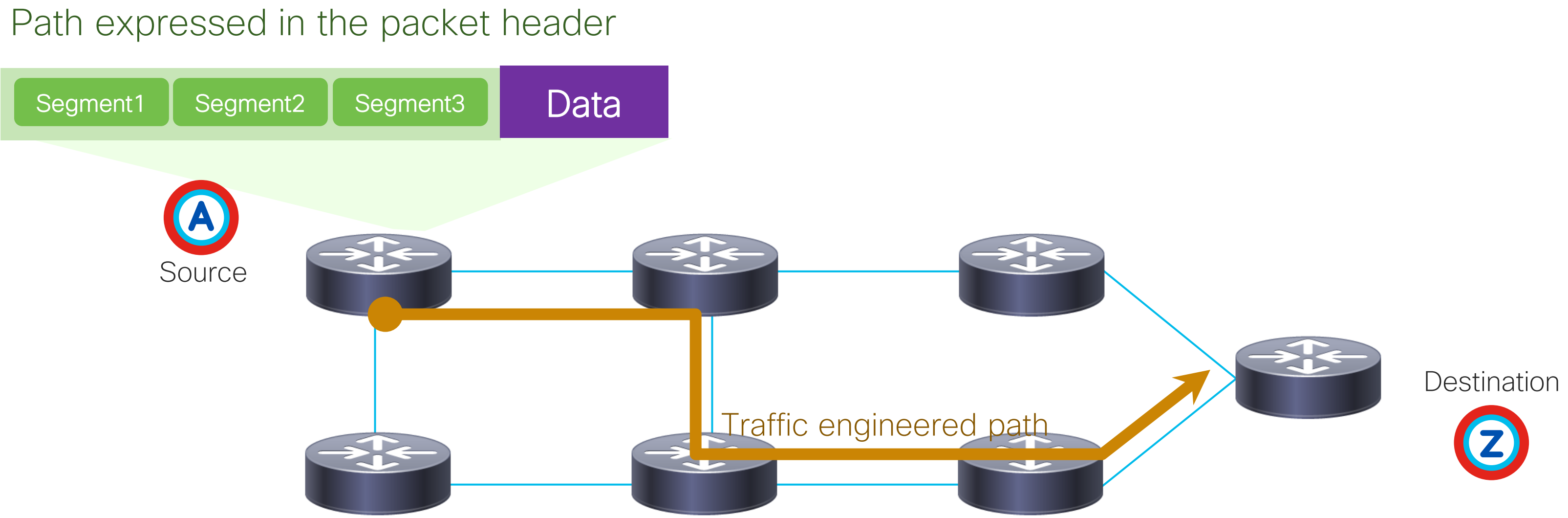
Path expressed in the packet header



- Segment: instruction a node executes on the incoming packet
  - SID → a segment identifier

How does it work?

- Segment list: an ordered set of segments



How does it work?

- SR-MPLS: the instantiation of SR on the MPLS data-plane
  - SID → an MPLS label associated with the segment
  - A SID list is expressed as a stack of MPLS labels

- SRv6: the instantiation of SR on the IPv6 data-plane
  - SID → an IPv6 address associated with the segment.
  - A SID list is encoded in the same IPv6 address/packet

Two type of segment categories

GLOBAL SEGMENT

- Segments learnt and programmed by all nodes in the SR domain
- SID is operator-assigned
- Example: node segment

LOCAL SEGMENT

- Segments learnt by all nodes in the SR domain but only programmed by the advertising node
- SID is dynamically allocated by router and option for operator-assigned
- Example: adjacency segment, peering segment

IGP Segments

Starting with SR MPLS examples
the same applies to SRv6, covered later

Why not to use the IGP to program MPLS labels? IGP segments

- Two basic building blocks distributed by IGP

- Prefix Segments
- Adjacency Segments

IGP Prefix Segment (Node Segment)

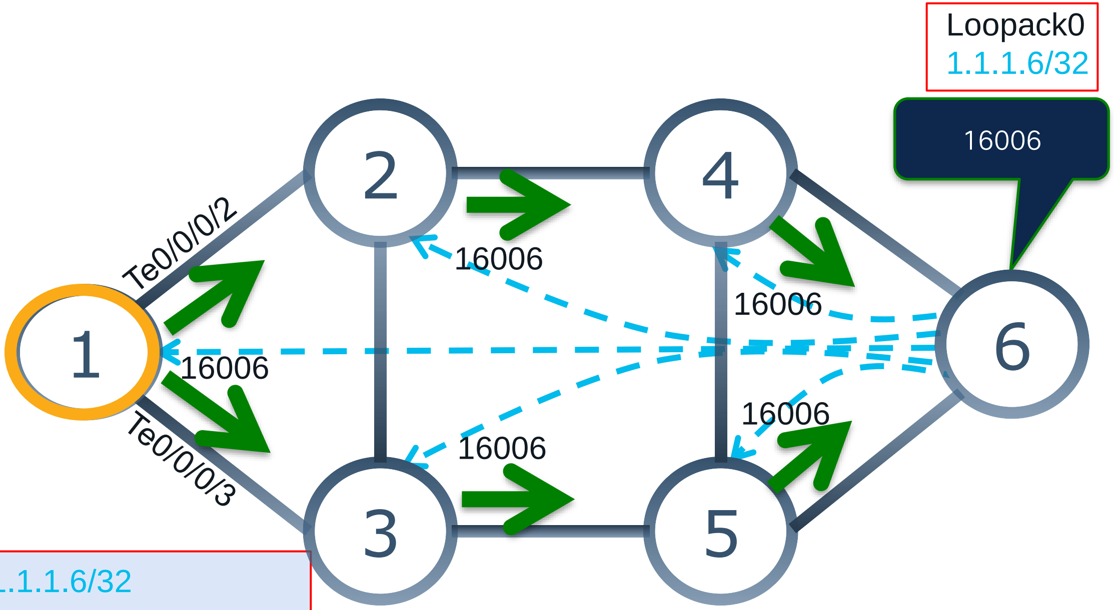
Shortest-path to the IGP prefix

Equal Cost MultiPath (ECMP)-aware

Label = 16000 + Index

Advertised as index

Distributed by ISIS/OSPF



Global Segment

RP/0/RP0/CPU0:Node-1#sh mpls forwarding prefix 1.1.1.6/32
Tue Jan 29 10:30:53.133 UTC

Local Label	Outgoing Label	Prefix or ID	Outgoing Interface	Next Hop	Bytes Switched
16006	16006	1.1.1.6/32	Te0/0/0/2	77.1.2.2	0
	16006	1.1.1.6/32	Te0/0/0/3	77.1.3.3	0

IGP Prefix Segment

Shortest-path to the IGP prefix

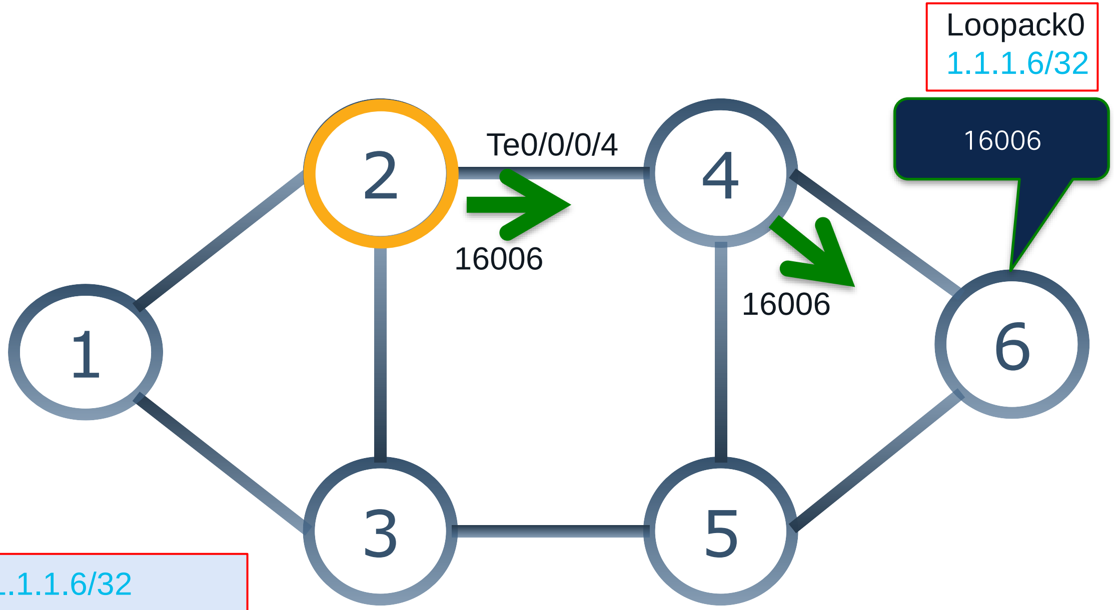
Global Segment

Equal Cost MultiPath (ECMP)-aware

Label = 16000 + Index

Advertised as index

Distributed by ISIS/OSPF



Global Segment

RP/0/RP0/CPU0:Node-2#sh mpls forwarding prefix 1.1.1.6/32
Tue Jan 29 10:30:53.133 UTC

Local Label	Outgoing Label	Prefix or ID	Outgoing Interface	Next Hop	Bytes Switched
------	-----------	-----------------	------------	---------------	------------
16006	16006	1.1.1.6/32	Te0/0/0/4	77.2.4.4	0

IGP Prefix Segment

Shortest-path to the IGP prefix

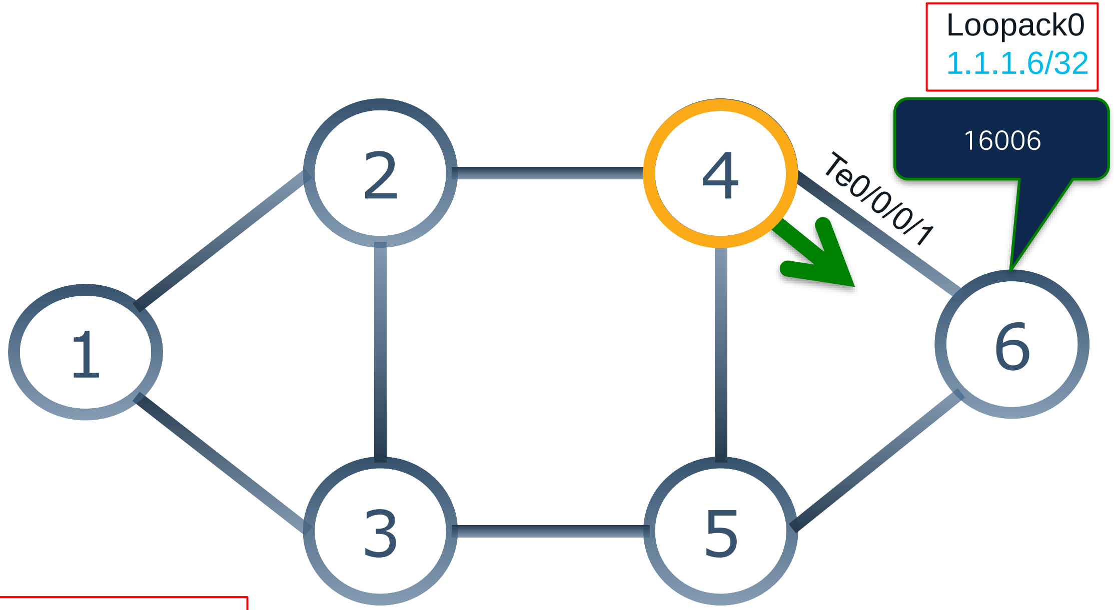
Global Segment

Equal Cost MultiPath (ECMP)-aware

Label = 16000 + Index

Advertised as index

Distributed by ISIS/OSPF



Global Segment

```
RP/0/RP0/CPU0:Node-3#sh mpls forwarding prefix 1.1.1.6/32
Tue Jan 29 10:30:53.133 UTC
Local  Outgoing   Prefix          Outgoing    Next Hop       Bytes
Label  Label      or ID           Interface                  Switched
------ ---------- --------------- ----------- -------------- ------------
16006  Pop        1.1.1.6/32      Te0/0/0/1   77.4.6.4       0
```

IGP Adjacency Segment

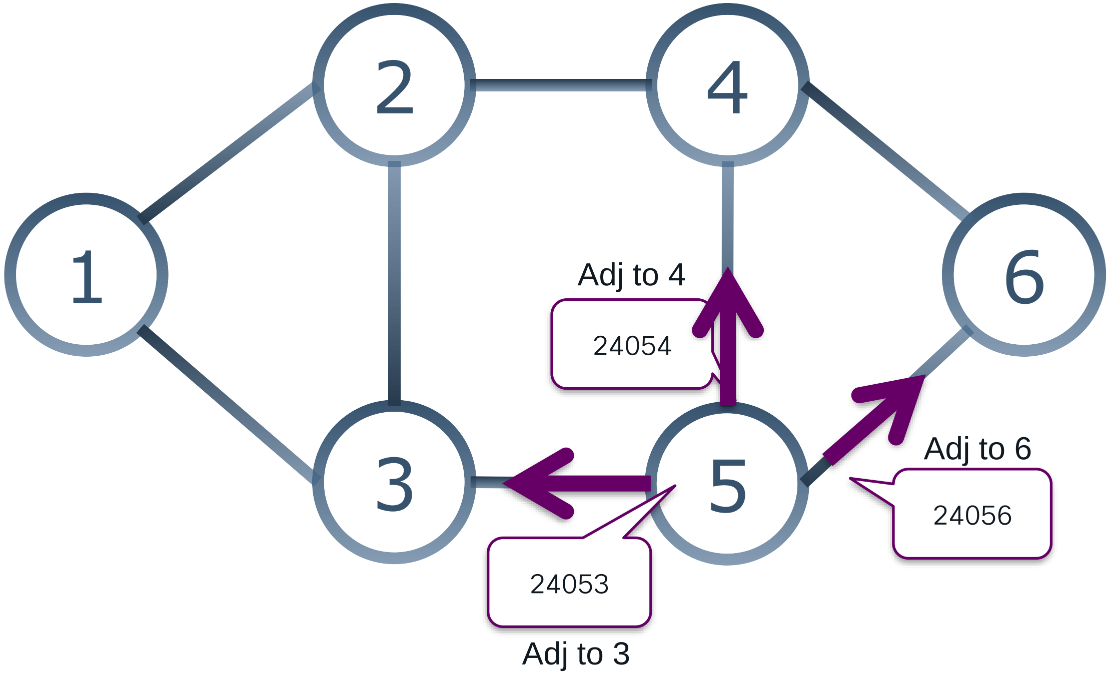
Forward on the IGP adjacency

Local Segment

Advertised as label value

Distributed by ISIS/OSPF

Label automatically allocated from the dynamic label pool



Combining IGP Segments

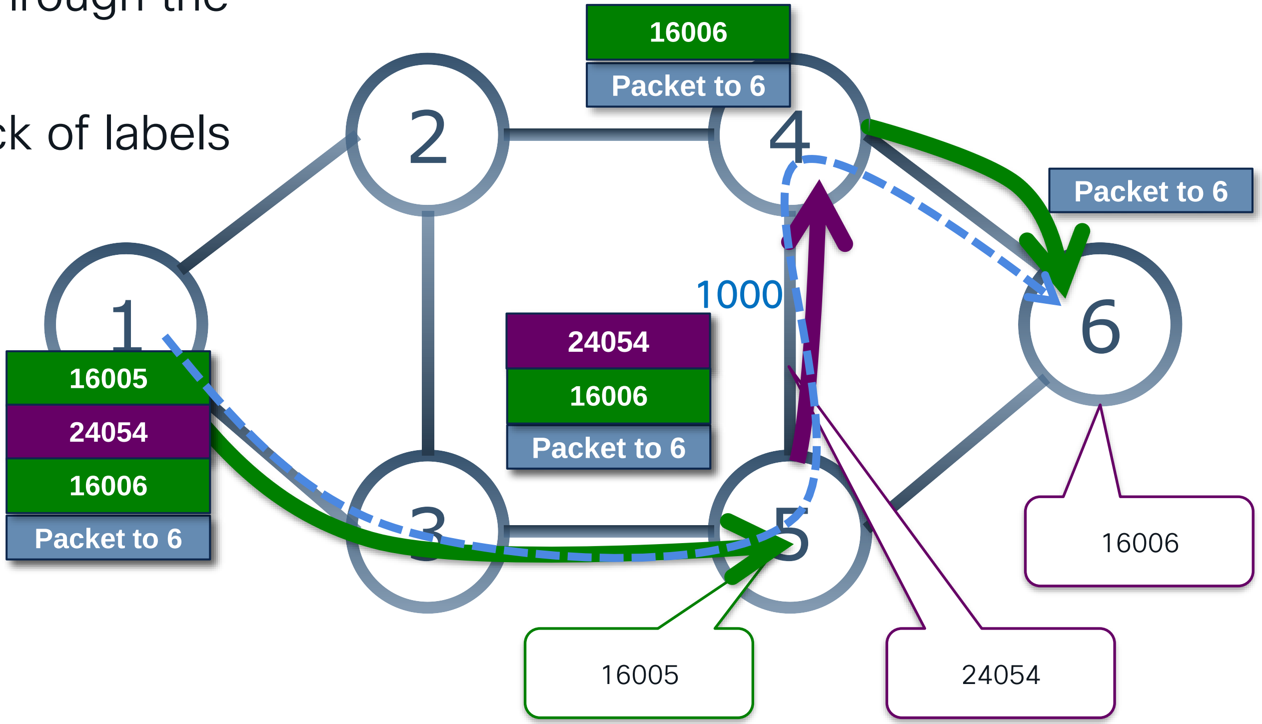
Steer traffic on any path through the network

Path is specified by a stack of labels

No path is signaled

Single protocol:

IS-IS or OSPF



What are the different type of Segments

IGP Prefix SID	GLOBAL segment representing an IGP prefix Forward packet along shortest-path (ECMP-aware) to reach the prefix associated with the segment
IGP Adjacency SID	LOCAL segment representing an IGP adjacency Forward packet over the interface where the adjacency is formed
IGP Anycast SID	An IGP-Prefix segment assign to an IGP prefix advertised by multiple routers (anycast prefix)
BGP Prefix SID	GLOBAL segment representing a BGP prefix Forward packet along best-path to reach the prefix associated with the segment
BGP Peering SID	LOCAL segment representing a BGP neighbor Forward packet over the interface where the neighbor is formed
Binding SID	LOCAL segment representing an SR traffic engineering Policy Forward packet along the path(s) of the associated SR Policy
Tree-SID	GLOBAL segment representing a Multicast Tree Replicate / Forward multicast packet to all receivers of the multicast group

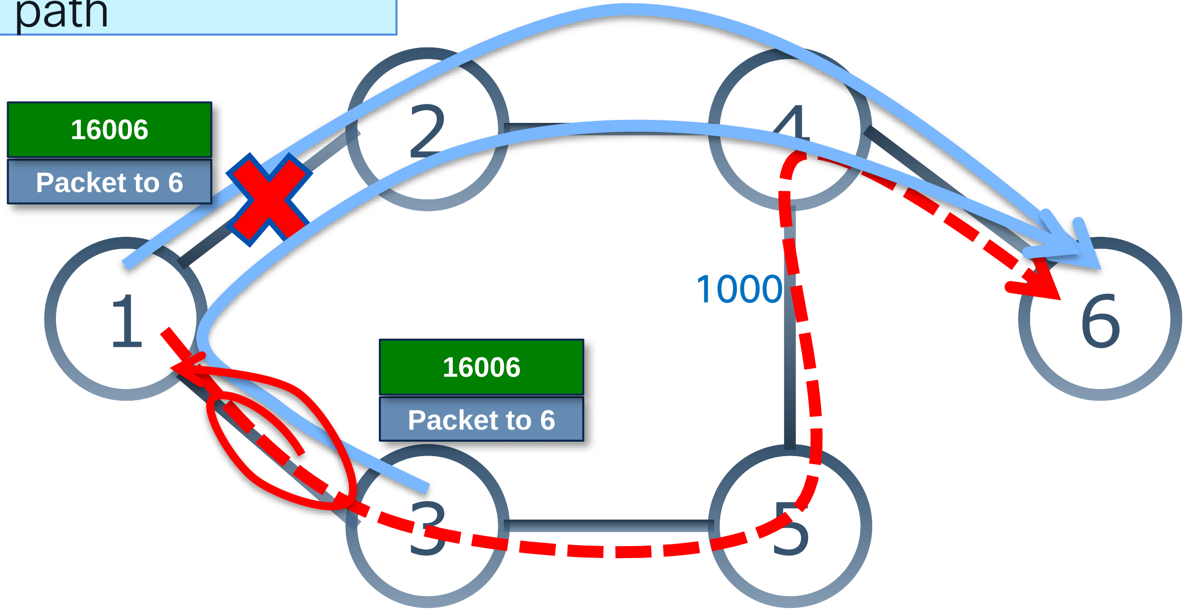
Global Segment

Local Segment

The IGP can compute and program any path
Topology Independent Loop Free Alternate (TI-LFA)

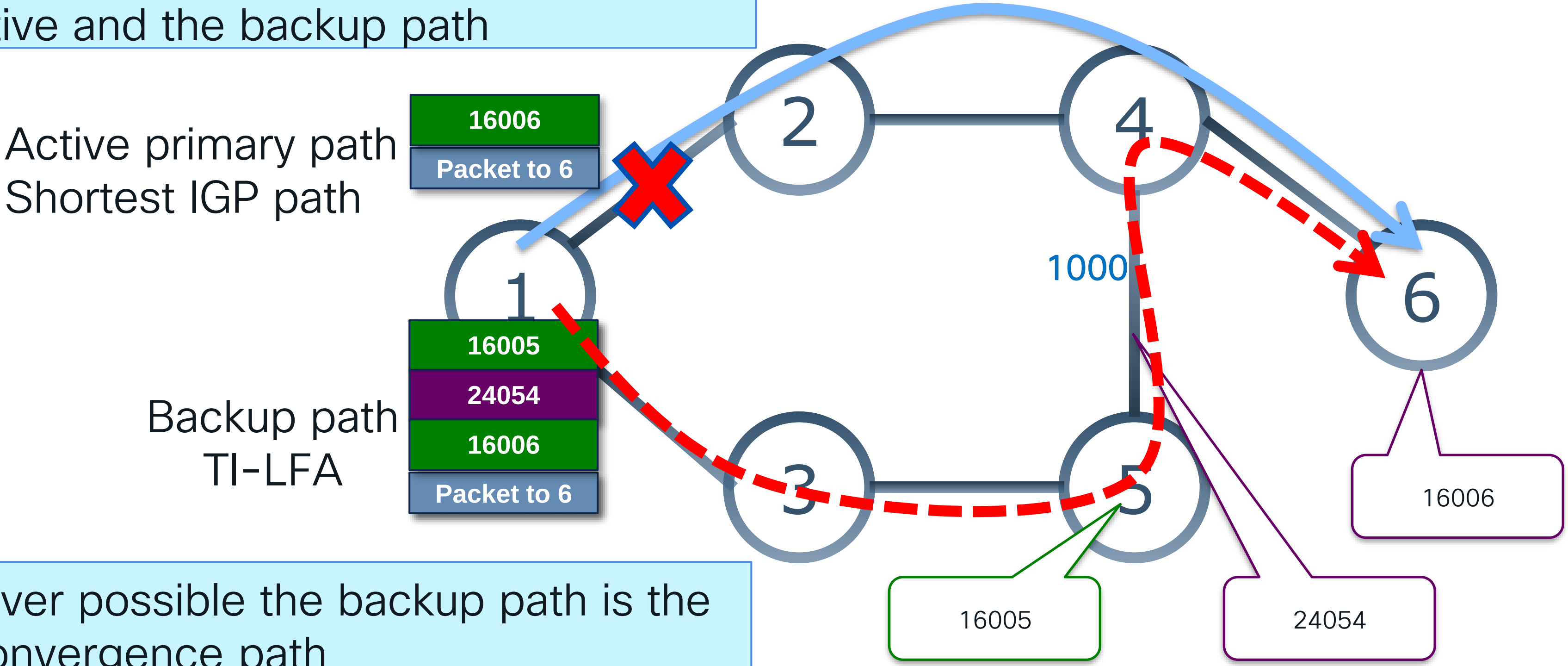
For every destination the IGP is computing the active and the backup path

Active primary path
Shortest IGP path



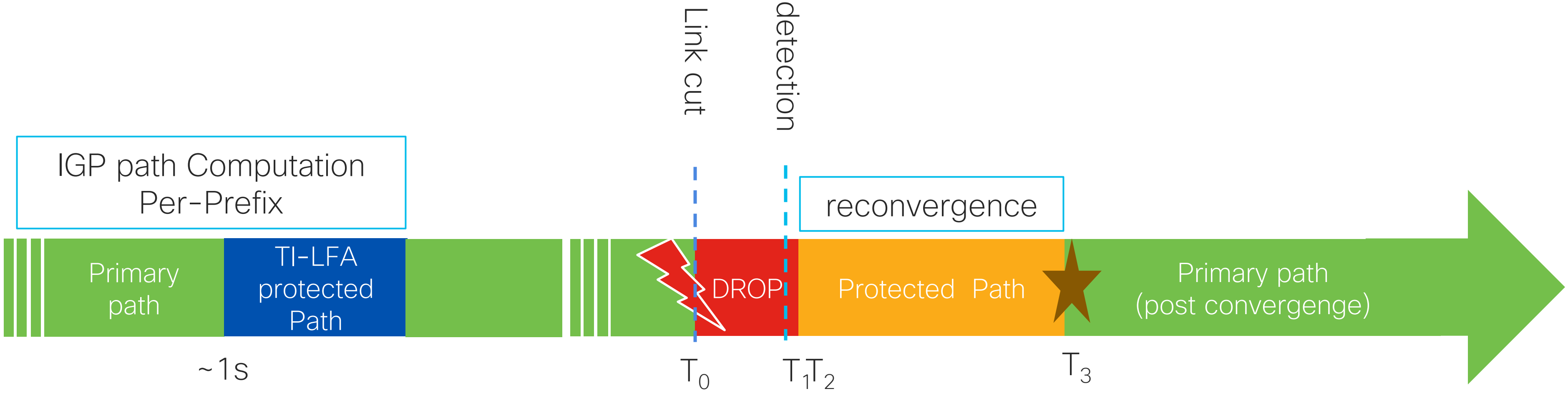
The IGP can compute and program any path
Topology Independent Loop Free Alternate (TI-LFA)

For every destination the IGP is computing the active and the backup path



Whenever possible the backup path is the post convergence path

TI-LFA – Timeline



$T_1 - T_0$ = time to detect the failure: from few ms (light down) ~15-30ms (BFD)
$T_2 - T_1$ = time to invalidate the impacted interface: few ms (Hierarchical FIB)
$T_2 - T_0 < 50$ms
$T_3 - T_1$ = time for IGP to re-converge, sub-second (~500ms)

TI-LFA protection Coverage

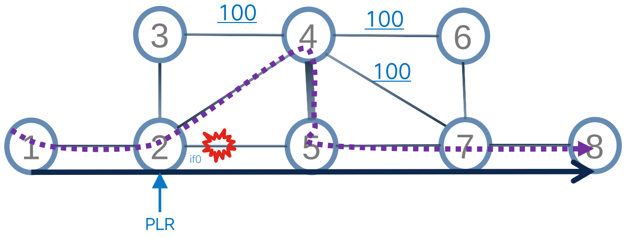
- Every prefix route is protected

```
RP/0/0/CPU0:XR-1#sh route 50.50.50.50
Routing entry for 50.50.50.50/32
  Known via "isis dc", distance 115, metric 20, labeled SR, type level-2
  Installed Feb  1 09:19:33.208 for 2d21h
  Routing Descriptor Blocks
    33.77.86.77, from 50.50.50.50, via TenGigE0/0/0/1, Backup (TI-LFA)
      Repair Node(s): 69.69.69.69
      Route metric is 40
    33.40.86.40, from 50.50.50.50, via TenGigE0/0/0/0, Protected
      Route metric is 20
  No advertising protos.
```

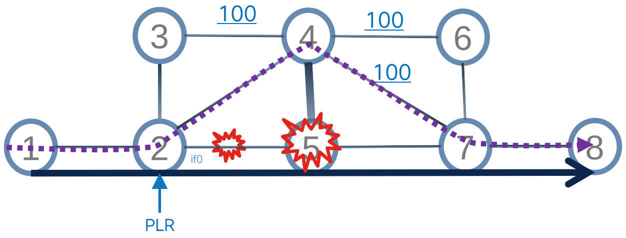
```
RP/0/0/CPU0:XR-1#show isis fast-reroute summary
IS-IS SR-AS-1 IPv4 Unicast FRR summary
                         Critical High     Medium   Low        Total
                         Priority Priority Priority Priority
Prefixes reachable in L2
 All paths protected    0          0         4         8         12
 Some paths protected    0          0         0         0         0
 Unprotected            0          0         0         0         0
 Protection coverage    0.00%      0.00%     100.00%   100.00%   100.00%
```

TI LFA FRR

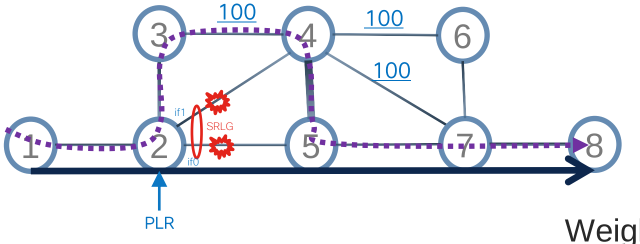
Link protection



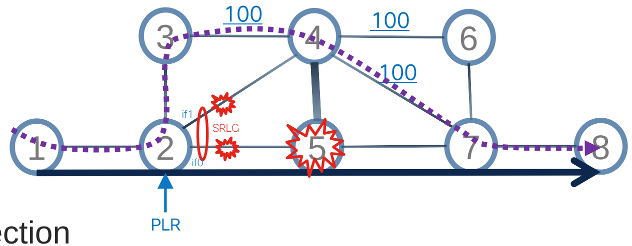
Node protection



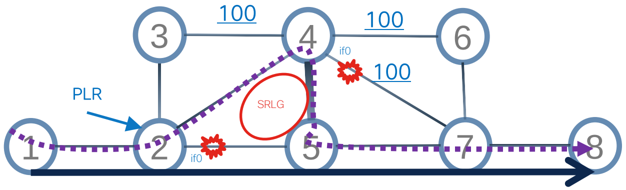
Local SRLG protection



Local SRLG + Node protection



Weighted Remote SRLG protection



Pre-convergence
TI-LFA/post-convergence

More power to the IGP Flexible Algorithm

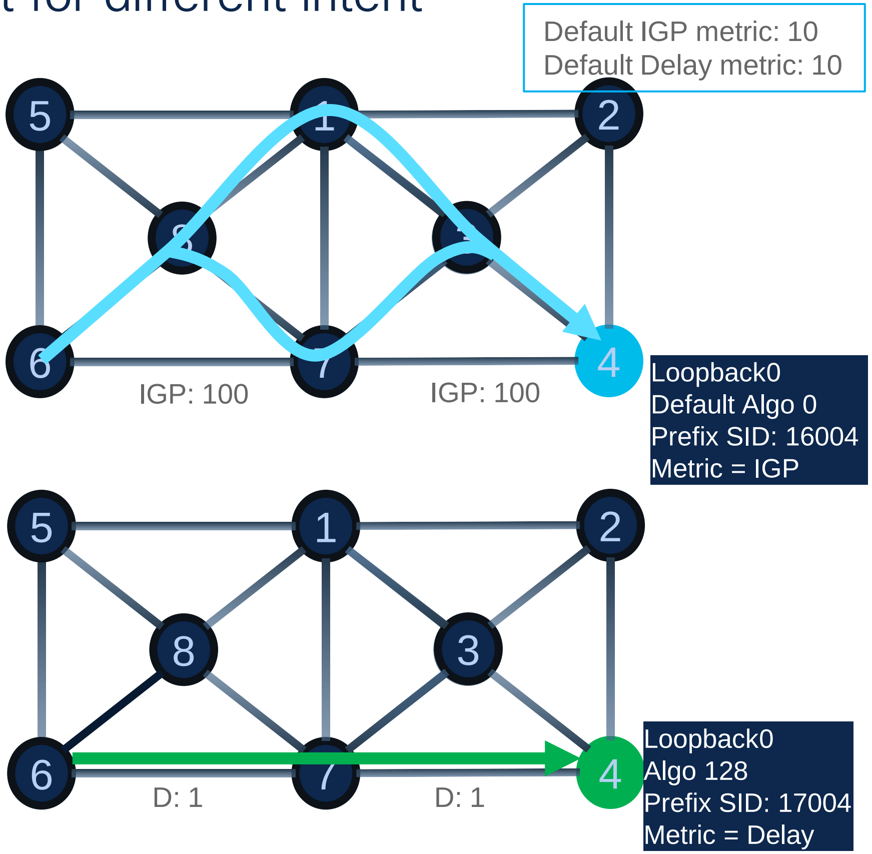
IGP Flexible Algorithm

Multiple Prefix SIDs for the same end-point for different intent

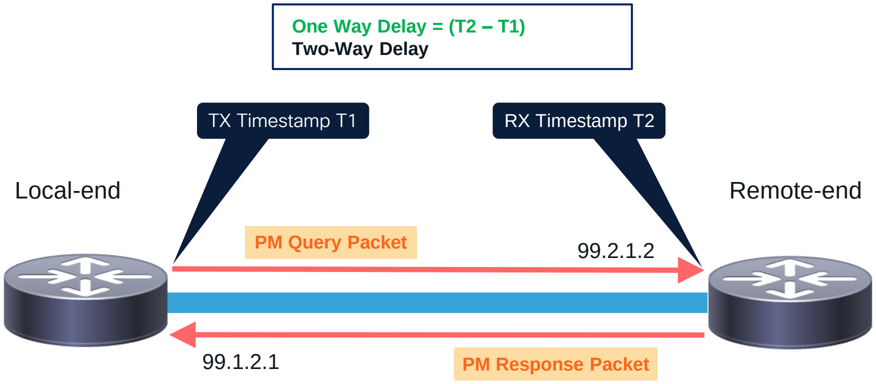
Operator-defined custom IGP algorithm leveraging dedicated Prefix-SIDs set

Example:

- Operator configure pref-SID 16004 associated to Loopback 0
- Operator defines Flex-Algo 128 as “minimize delay metric”
- Dedicated Prefix SID flex-algo 128 17004
- For each destination two different SIDs are installed in FIB



Link Delay Measurement



IOS-XR

```
performance-measurement
 interface TenGigE0/0/0/8
  delay-measurement
 !
 interface TenGigE0/0/0/9
  delay-measurement
!
delay-profile interfaces
  advertisement
   periodic
    minimum-change 200
    threshold 5
```

SR- PCE view

```
Link[0]: local address 99.1.2.1, remote address 99.2.1.2
  Local node:
    ISIS system ID: 0000.0000.6666 level-2 ASN: 64002
  Remote node:
    TE router ID: 5.5.5.5
    Host name: Napoli-5
    ISIS system ID: 0000.0000.5555 level-2 ASN: 64002
  Metric: IGP 10, TE 50,Delay 6000
  Bandwidth: Total 125000000, Reservable 0
  Adj SID: 24005 (protected) 24004 (unprotected)
  Excluded from CSPF: no
```

Multiple Prefix SIDs for the same end-point for different intent

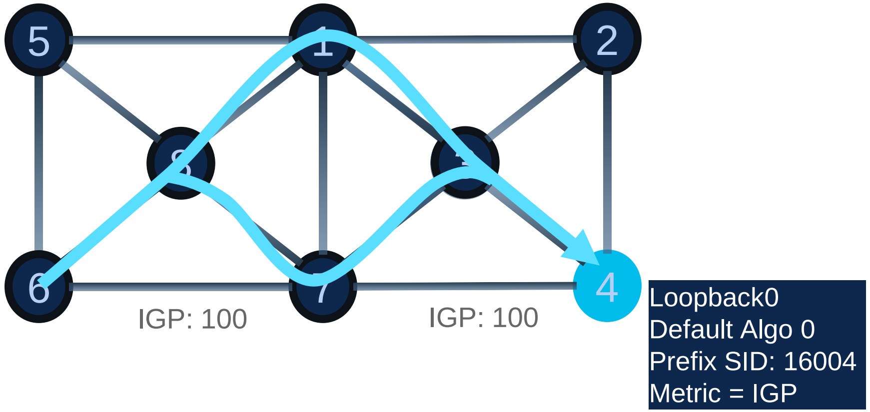
Default IGP metric: 10
Default Delay metric: 10

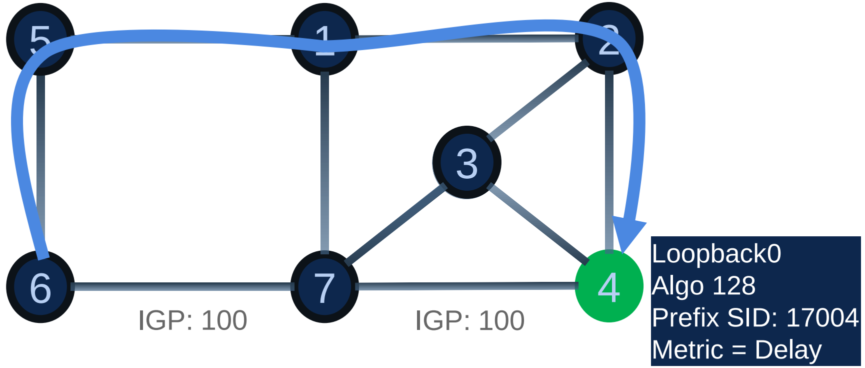
Flex Algo can be used also to build virtual topologies

Excluding Nodes

- Node is not participating in a flex Algo

- Excluding (including) Links
  - E.g. Only high bw links
  - E.g Only macsec links
  - E.g Plane A – Plane B
  - Done via link affinity exclusion/inclusion





Flex Algo «««*super powers*»»»

Automatically managed by the IGP protocol
with 100% self-healing capabilities.

One single SID even for complex intent
e.g. Low Latency, exclude/include affinity.

Protected path stays in Flex Algo virtual topology
TI-LFA aware

Traffic Engineering

SR Traffic Engineering (SRTE)

- The RSVP-TE tunnel Interface construct has been replaced
- The SR Policy the new construct
  - In SR there is *no tunnel anymore*, the ***policy*** is programmed **only** at the headend.
  - The newly created Policy architecture has been designed for simplicity, self healing and automation required in SDN and Autonomous Networking era.

HOW is policy instantiated?
- Local Configuration
- Controller instantiated
- On-demand (hint: by BGP / Service routes)

WHAT type of path?
- Explicit path
- Dynamic path

WHO computes a dynamic path?
- Distributed - Head-end
- Centralized - Controller

If Controller instantiated
WHAT protocol / mechanism is used to deploy?
- PCEP
- NETCONF
- gNMI API

SR Policy – configuration example

On Node1:

User-defined name

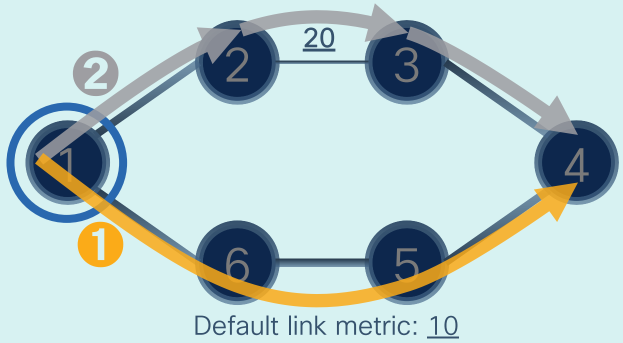
Color and End-point

Binding-SID

Candidate Paths

```
segment-routing
 traffic-eng
  policy POLICY1
   color 20 end-point ipv4 1.1.1.4
   binding-sid mpls 1000
   candidate-paths
1   preference 200
     dynamic
      metric type te
     constraints
      affinity
       exclude-any color red
    !
2   preference 100
     explicit segment-list SIDLIST1
  !
  segment-list name SIDLIST1
   index 10 mpls label 16002
   index 20 mpls label 30203
   index 30 mpls label 16004
```

```
segment-routing
 traffic-eng
  affinity-map
   color red bit-position 0
```



WECMP example

On Node1:

```
segment-routing
 traffic-eng
  policy POLICY1
   color 20 end-point ipv4 1.1.1.4
   binding-sid mpls 1000
   candidate-paths
    preference 200
     explicit segment-list SIDLIST1
      weight 1
     !
     explicit segment-list SIDLIST2
      weight 4
     !
  segment-list name SIDLIST1
   index 10 mpls label 16002
   index 20 mpls label 30203
   index 30 mpls label 16004
  !
  segment-list name SIDLIST2
   index 10 address ipv4 1.1.1.4
```

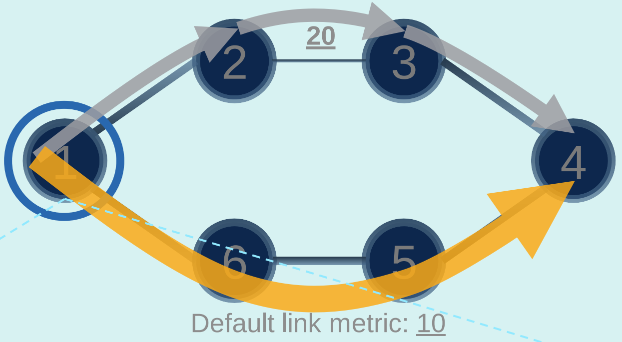
Path preference 200

Explicit SID-list1, Weight 1

Explicit SID-list2, Weight 4

SID-list1

SID-list2



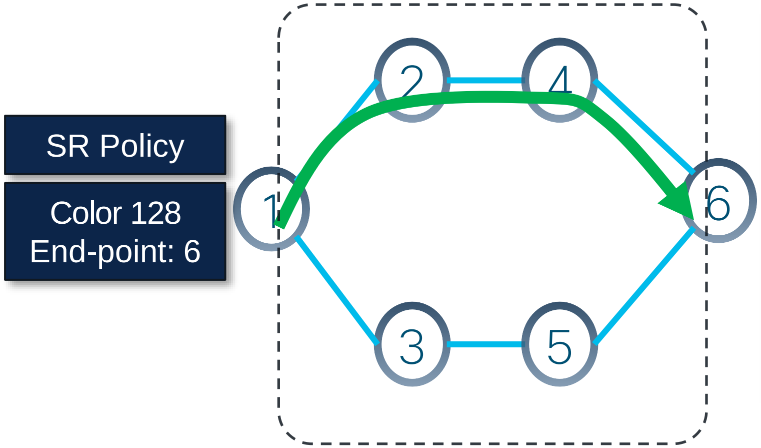
FIB @ head-end Node1
Incoming label: 1000
Action:pop and push <16002, 30203, 16004> (20%)
push <16004> (80%)

SR Policy Identification

An **SR Policy** is uniquely identified by end-point and color:

End-point: the destination of the SR Policy

Color: a numerical value to differentiate multiple SRTE Policies between the same pair of nodes with potentially different SLA.




```
segment-routing
  traffic-eng
    policy POLICY1
      color 128 end-point ipv4 1.1.1.6
       candidate-paths
        preference 100
          dynamic
            metric
              type latency
```

Automated Steering

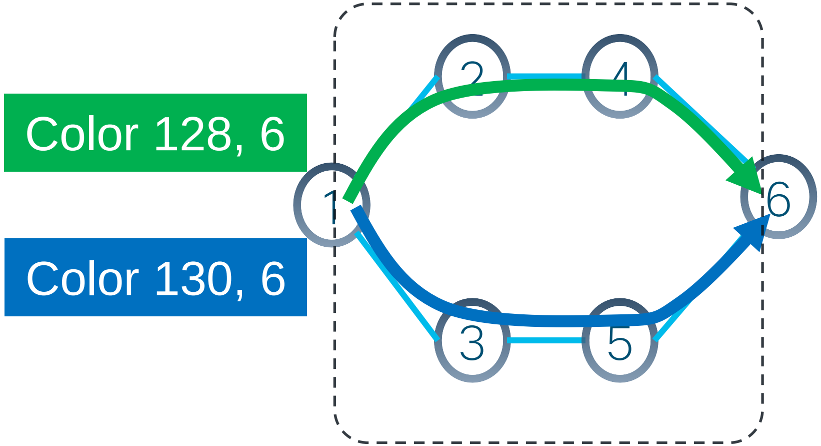
How to inject traffic into a Traffic Engineering Policy

- Traditional ways are complex to be configured and managed and often have performance impact
  (e.g. Policy Based Tunnel Selection PBTS)
- With Segment Routing steering traffic into a Traffic Engeneering policy is completely automated for BGP signaled services.

SR Policy Color

For the same source/end-point different colors for different SLA

- E.g Green = Low Latency and Blue = High Bandwidth
- SRTE Policy *Color* go hand in hand with BGP **Ext. Community *Color***
- Extended Community Color is specified in RFC 5512

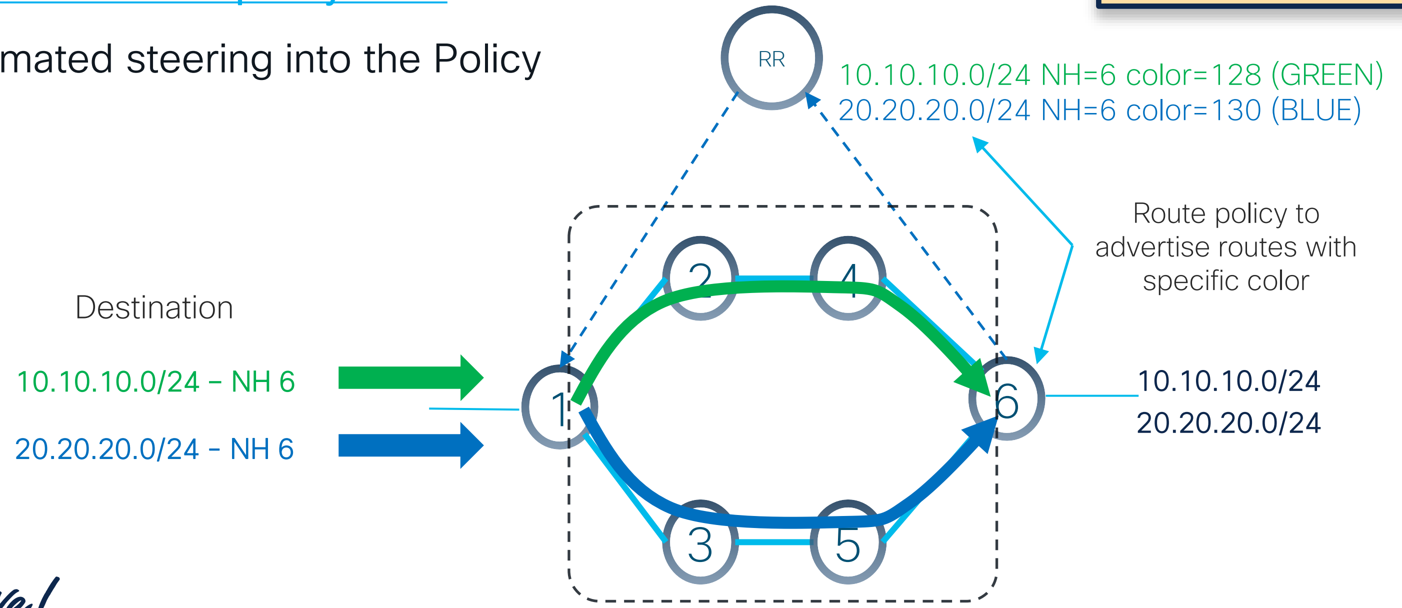


Segment Routing – Automated Steering (AS)

Steer traffic into SR Policy based on Next Hop BGP and Color

- BGP signaled routes (e.g. IPv4, IPv6, VPNv4, EVPN):
- End-pont = BGP Next Hop
- Route color = SR policy color
- Automated steering into the Policy

```
vrf 1234
 address-family ipv4 unicast
  import route-target
   3450:3450
  !
  export route-policy SET_COLOR_128_130
  export route-target
   3450:3450
```



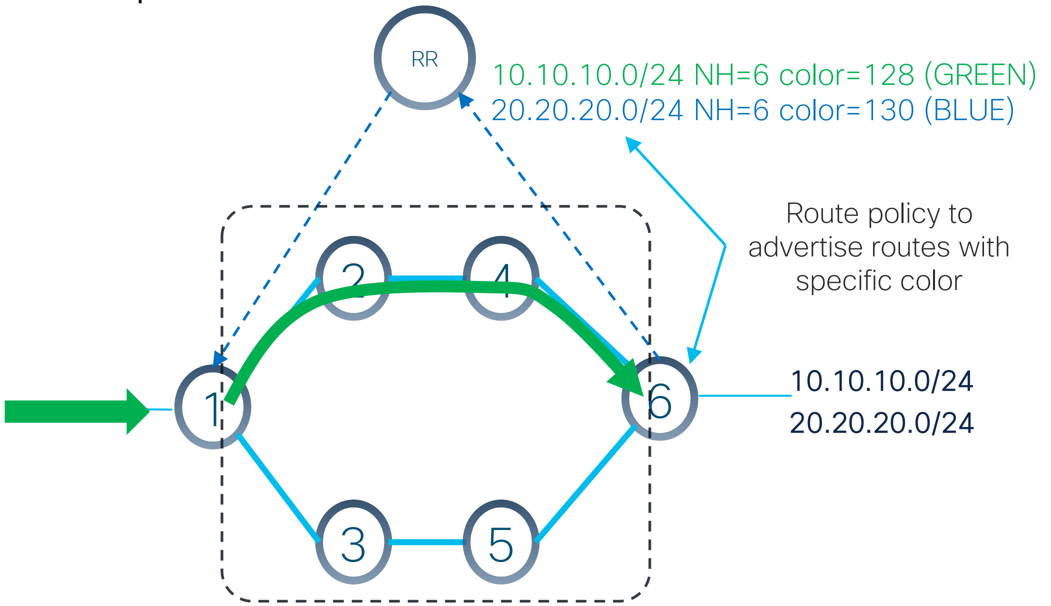
Segment Routing – ODN (+AS)

- Setup SRTE policy to the BGP NH On Demand

- BGP signaled routes (e.g. IPv4, IPv6, VPNv4, EVPN):
- End-pont = BGP Next Hop   color GREEN (128)
- No existing policy but ODN template defined

```
segment-routing
 traffic-eng
  on-demand color 128
   preference 100
   dynamic
    metric type latency
```

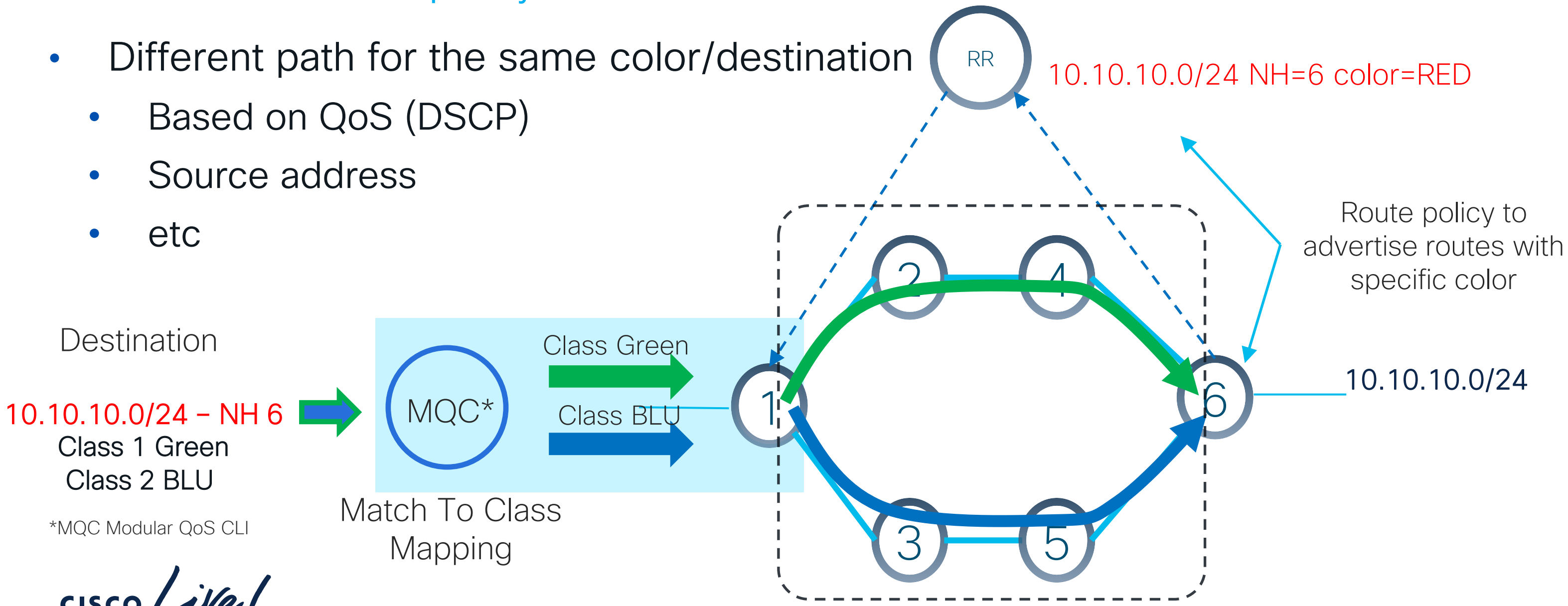
10.10.10.0/24 – color 128 NH 6



Per Flow Automated Steering (AS)

Steer traffic into SR Policy based on Destination – Color – DSCP

- BGP signaled routes (e.g. IPv4, IPv6, VPNv4, EVPN):
- End-pont = BGP Next Hop
- Route color = SR policy color
- Different path for the same color/destination
  - Based on QoS (DSCP)
  - Source address
  - etc



*MQC Modular QoS CLI

Other Steering mechanism

- **Preferred path**: for L2 services. The pseudowire of the L2 service is mapped over a SRTE policy (and not following the IGP path)
- **Static Route**: traffic towards specific route (or Next hop) will be steered over the policy
- **Autoroute include**: IGP shortcut – the IGP will use the policy as a preferred link between headend and tail-end of the policy
- **Color-Only Automated Steering -** is a traffic steering mechanism where a policy is created with given color, regardless of the endpoint.
- **Using Binding Segments -** using BSID to stitch SRTE policies

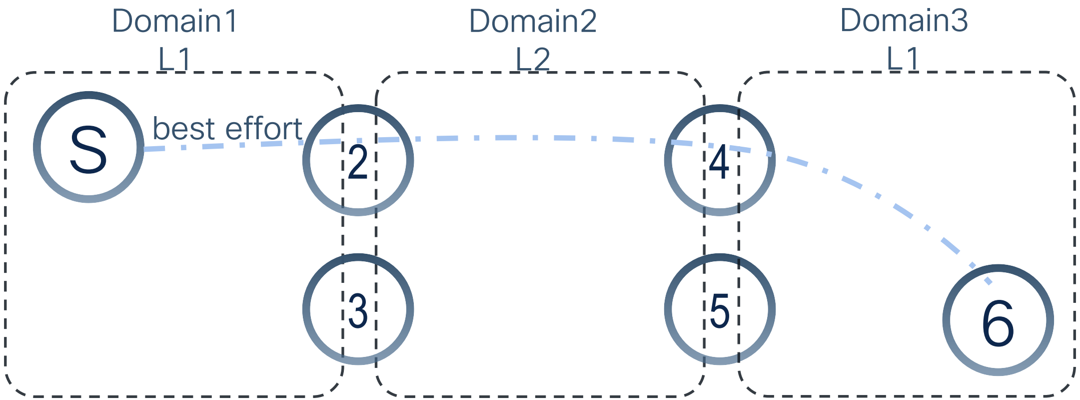
More info at : https://www.cisco.com/c/en/us/td/docs/iosxr/ncs5500/segment-routing/77x/b-segment-routing-cg-ncs5500-77x/configure-sr-te-policies.html#id_128905

Controller Based Advanced Use Cases

Inter domain with SLA

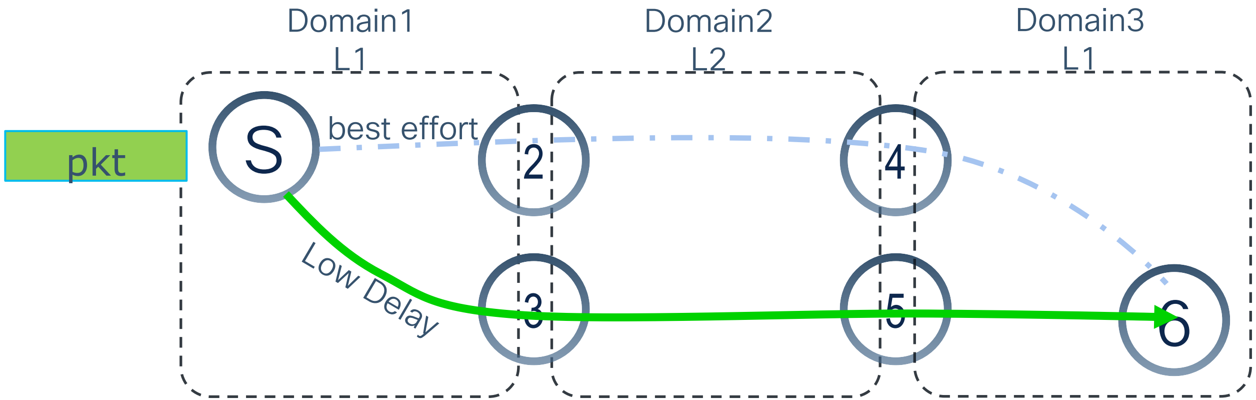
Crossing IGP borders

- With a stack of labels through border routers
- Source Based Routing: only ingress node need to be programmed
- This means all other nodes needs only to support basic SR forwarding
- Not only best effort connectivity!



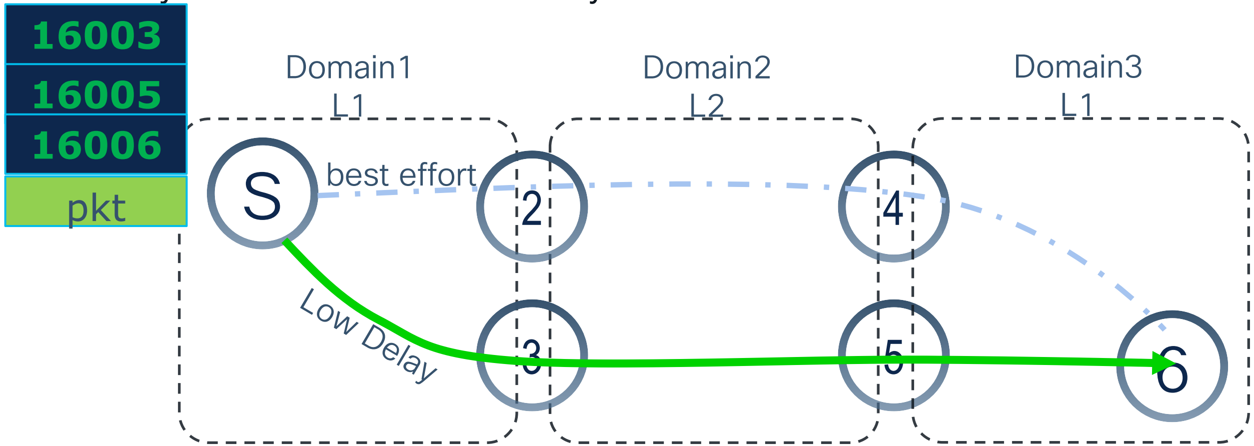
Crossing IGP borders

- With a stack of labels through border routers
- Source Based Routing: <u>only ingress</u> node need to be programmed
- This means all other nodes needs only to support basic SR forwarding
- Not only best effort connectivity!



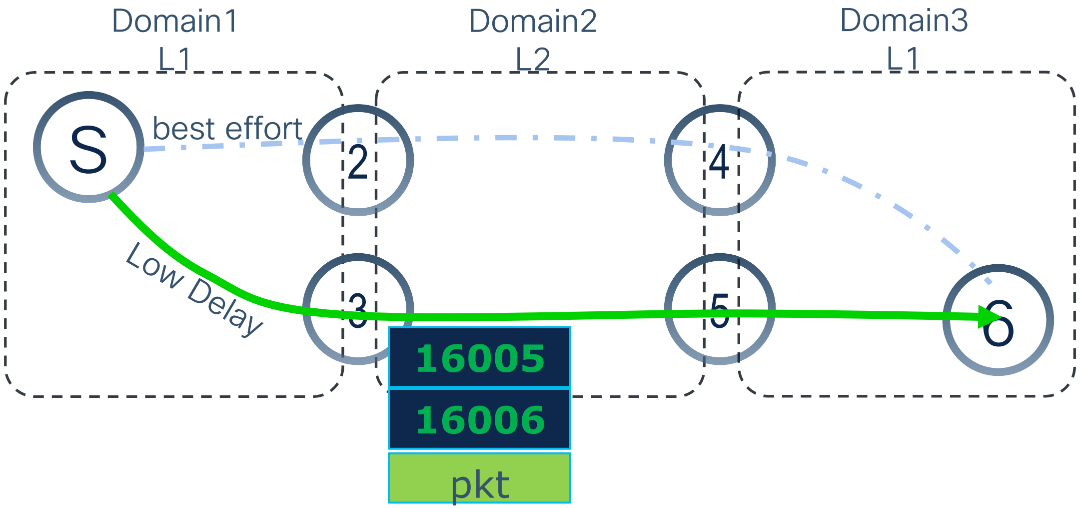
Crossing IGP borders

- With a stack of labels through border routers
- Source Based Routing: <u>only ingress</u> node need to be programmed
- This means all other nodes needs only to support basic SR forwarding
- Not only best effort connectivity!



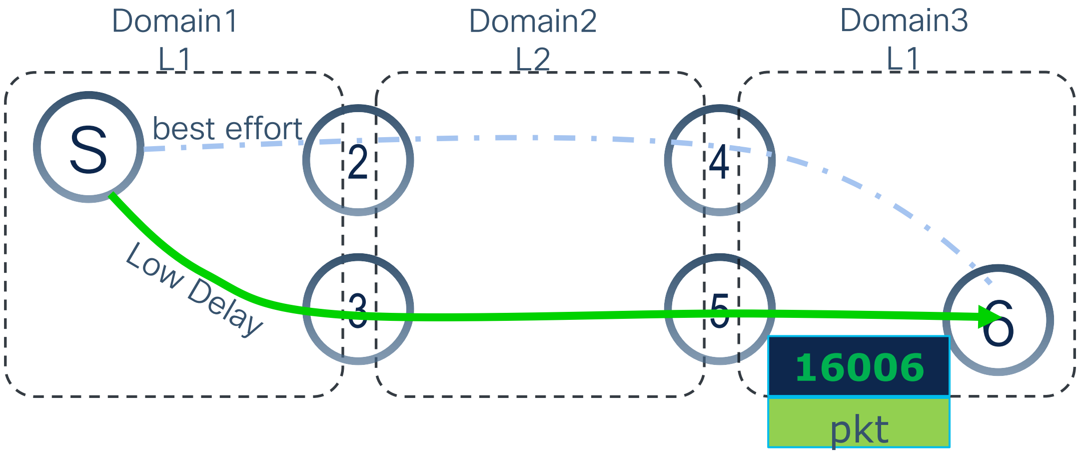
Crossing IGP borders

- With a stack of labels through border routers
- Source Based Routing: <u>only ingress</u> node need to be programmed
- This means all other nodes needs only to support basic SR forwarding
- Not only best effort connectivity!



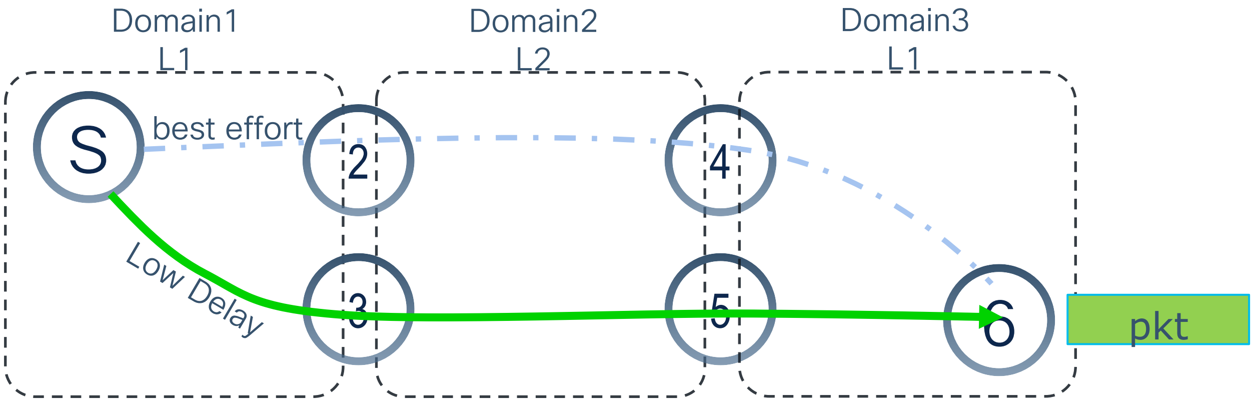
Crossing IGP borders

- With a stack of labels through border routers
- Source Based Routing: <u>only ingress</u> node need to be programmed
- This means all other nodes needs only to support basic SR forwarding
- Not only best effort connectivity!



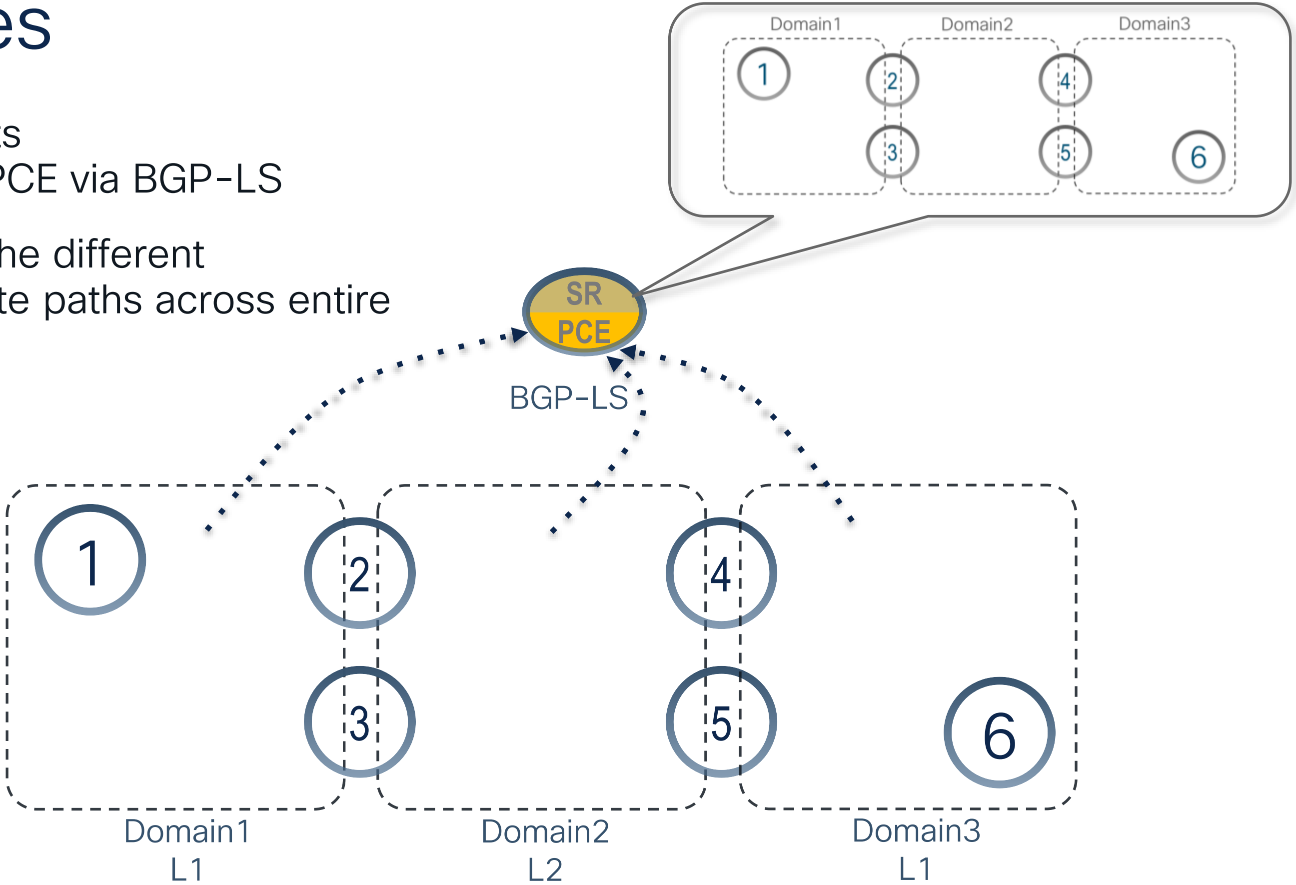
Crossing IGP borders

- With a stack of labels through border routers
- Source Based Routing: only ingress node need to be programmed
- This means all other nodes needs only to support basic SR forwarding
- Not only best effort connectivity!



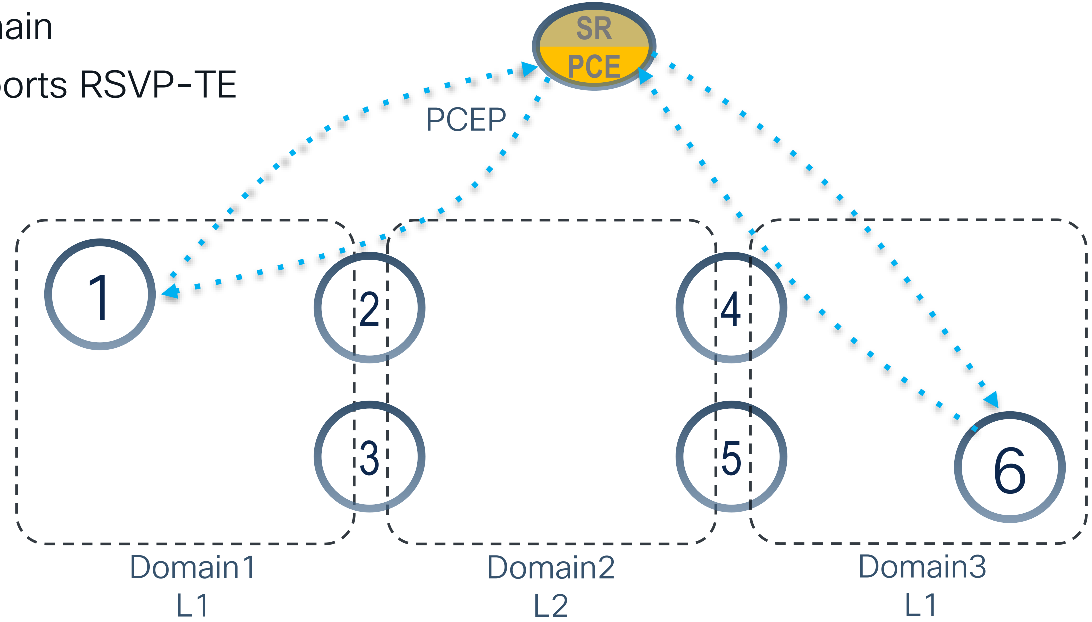
SR-PCE Receives & Combines Multiple Topologies

- Each domain feeds its topology to the SR-PCE via BGP-LS
- SR-PCE combines the different topologies to compute paths across entire topology



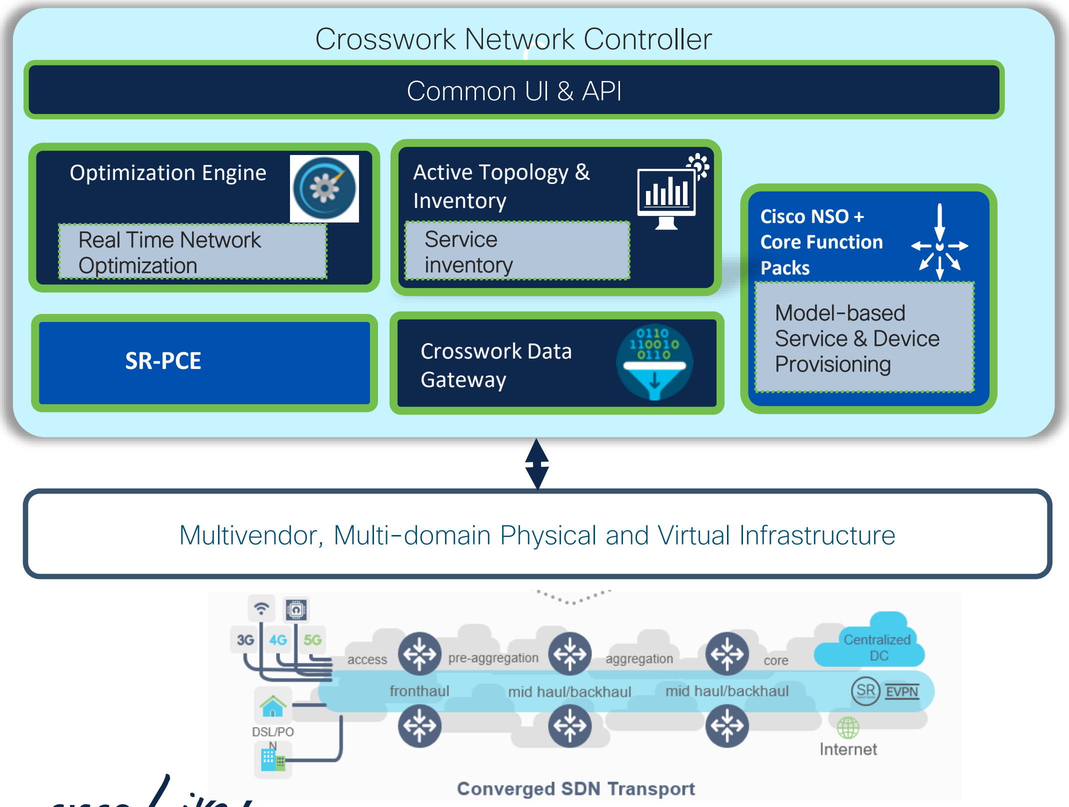
SR-PCE Receives & Combines Multiple Topologies

- SR-PCE is IOS-XR based stateful Path Computation Element (PCE)
  - PCEP session between SR-PCE and Headend nodes for centralized computation
  - Fundamentally Distributed (RR-like Deployment)
  - Multi Domain
  - Also supports RSVP-TE



Crosswork Network Controller (CNC)

Integrated solution for **deploying** and **operating IP transport** networks



Use Case	Description
Service Provisioning	Provision L2VPN & L3VPN services with transport intent
Intent-Oriented Transport Provisioning	Provision segment routing traffic-engineering policies for services with SLAs.
Bandwidth Optimization	Tactically optimize the network during times of congestion
Real time network optimization	Collect real-time performance information and optimize the network as needed to maintain the SLA
Topology & Inventory	Collect and expose information about network and services

SR innovations and use cases summary

Network Availability

Introduce seamlessly
Protect with automatic TI LFA FRR
Stabilize with microloop avoidance
Operate with advanced monitoring and blackhole detection
Monitor with SR Performance Measurement toolkit

Advanced Use Cases

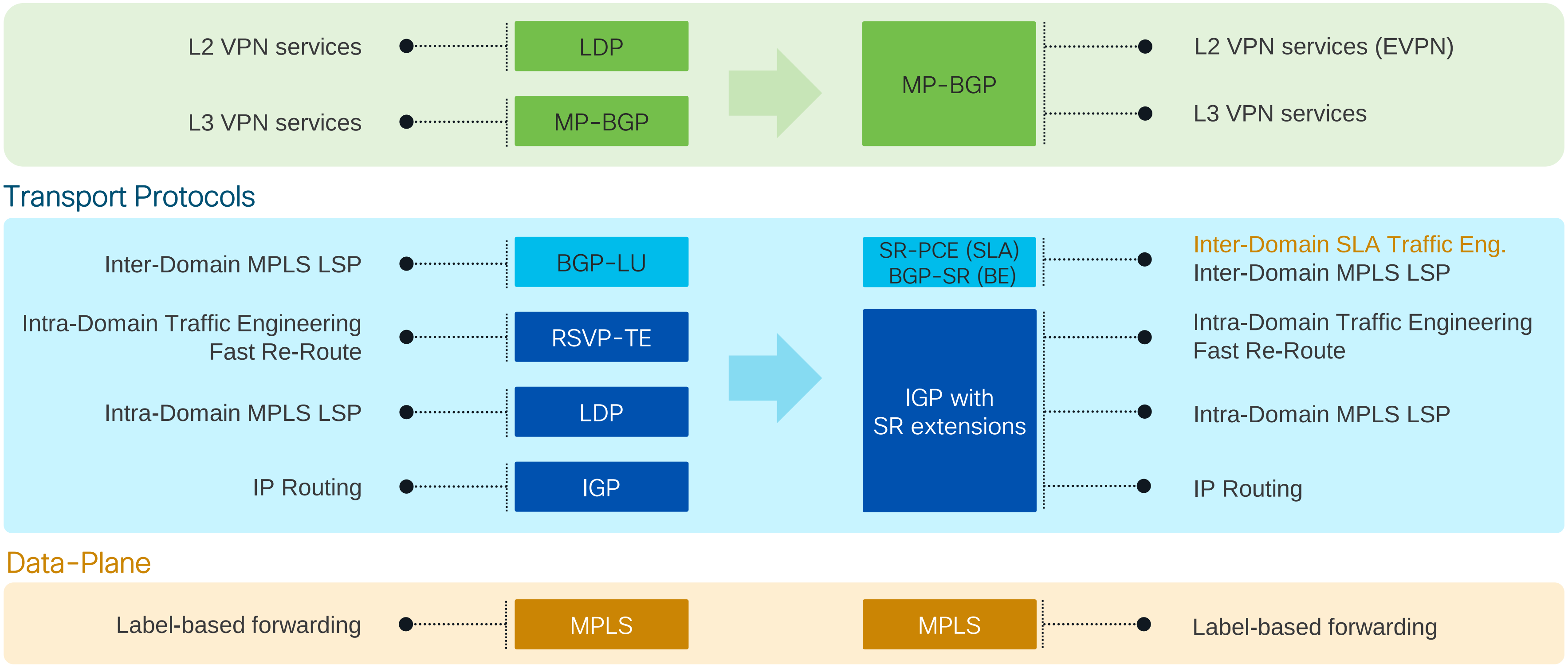
Path Disjointness (Multi-plane)
Real-Time Low Latency Services
Egress Peer Engineering (EPE)
Point-to-Multipoint delivery with Tree-SID
Bandwidth Optimization

Intent-Based Traffic Engineering

On-Demand Next-Hop (ODN) + Automated steering (AS)
Multi-plane Network Slicing using IGP Flex Algorithms
Multi-Domain intent with SR-PCE
Intent-Based Per-Flow Automated Steering
Circuit-Style SR Policies

Network Evolution

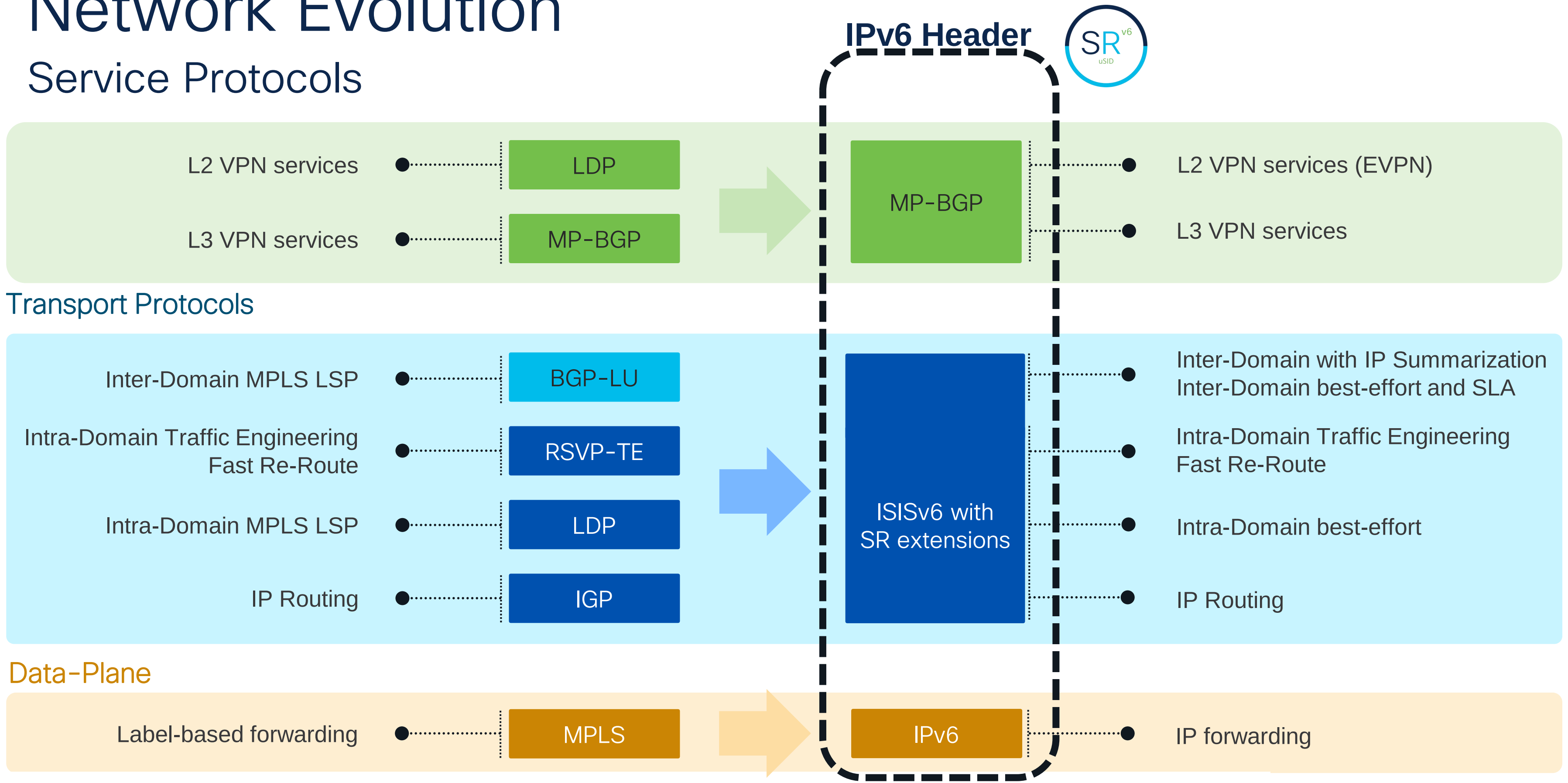
Service Protocols



LDP: Label Distribution Protocol, **MP-BGP**: Multi-protocol BGP, **BGP-LU**: BGP Labeled-Unicast, **PCE**: Path Computation Element, **RSVP-TE**: Reservation Protocol Traffic Engineering, **BE**: Best-Effort

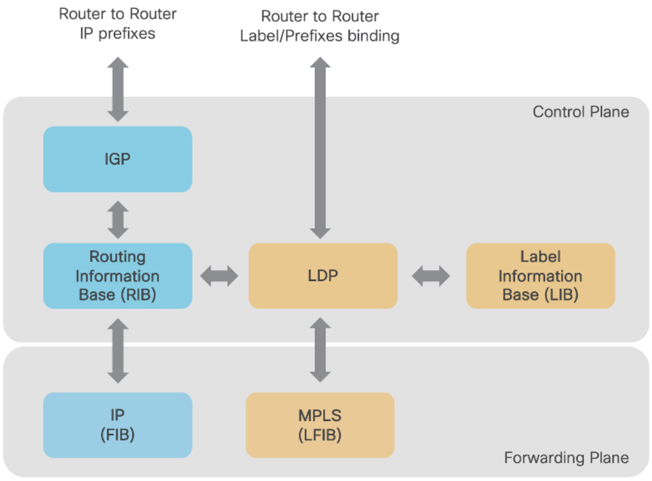
Network Evolution

Service Protocols



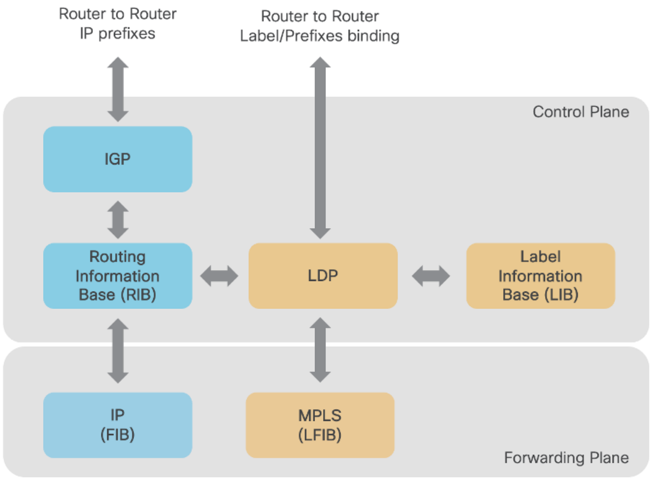
LDP: Label Distribution Protocol, **MP-BGP**: Multi-protocol BGP, **BGP-LU**: BGP Labeled-Unicast, **RSVP-TE**: Reservation Protocol Traffic Engineering

Segment Routing architecture simplification

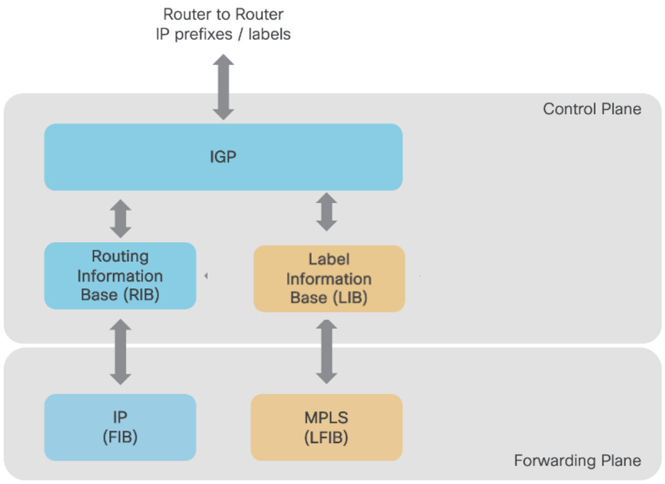


Routing Architecture
with MPLS

Segment Routing architecture simplification

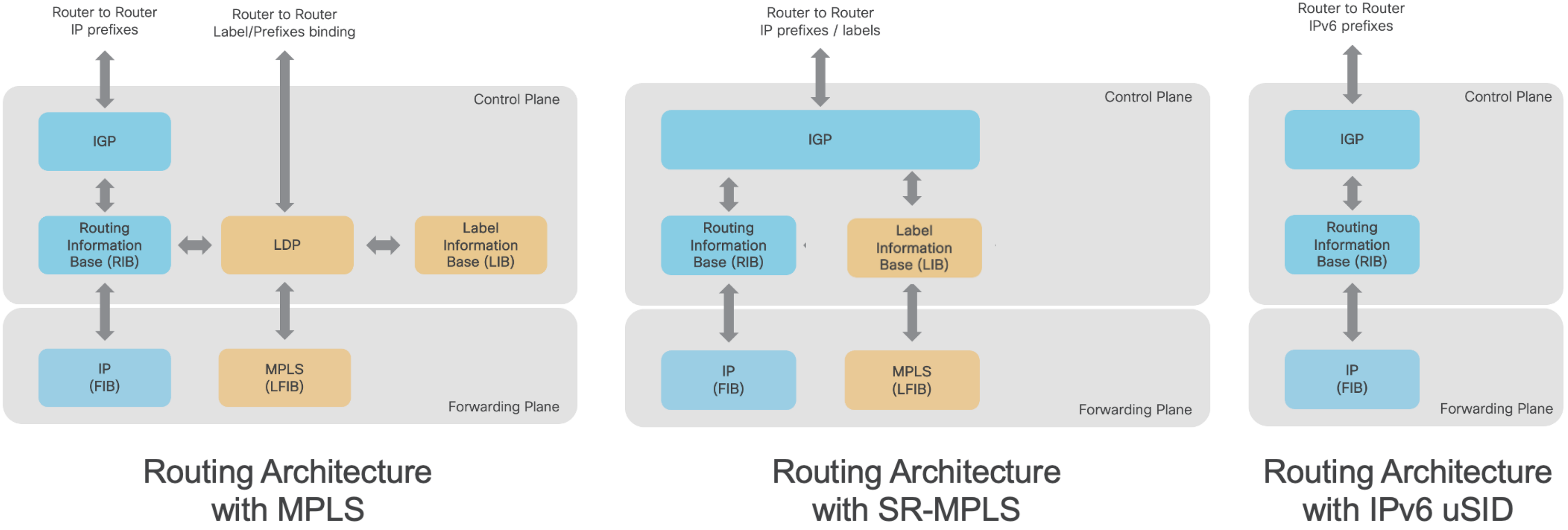


Routing Architecture with MPLS



Routing Architecture with SR-MPLS

Segment Routing architecture simplification



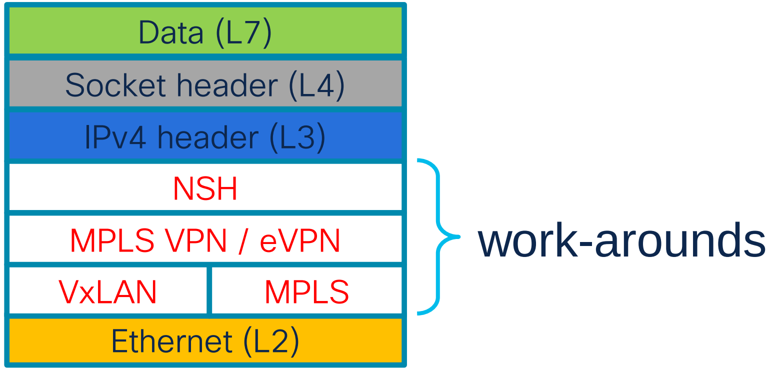
SRv6 uSID

Explained in few minutes...

IPv4 limitations & work-arounds

Network Functions	IPv4
Reachability	IPv4 Header
Engineered Load Balancing	MPLS Entropy Label, VxLAN UDP
VPN	MPLS VPN's, VxLAN
Traffic Engineering	RSVP-TE, SR-TE MPLS
Source Routing	SR-TE MPLS
Service Chaining	NSH

Address space 32-bit limitation
No optional header
IPv4 header doesn't support
- VPN
- Traffic Engineer
- Service Chaining
- Engineered Flow optimization
- Source-Routing



SRv6 Solution

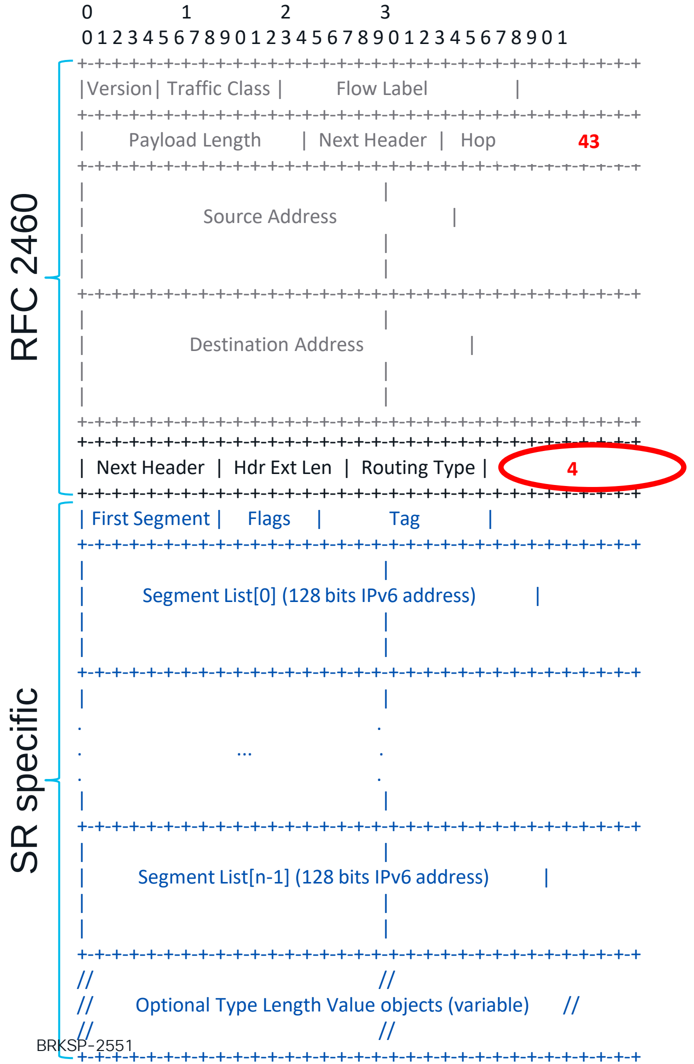
Network Functions	IPv6
Reachability	IPv6 Header
Engineered Load Balancing	IPv6 Header
VPN	IPv6 Header
Traffic Engineering	IPv6 Header
Source Routing	IPv6 Header
Service Chaining	IPv6 Header

IPv6 Address 128bits
IPv6 Flow Header
- Engineered Flow optimization

SRv6 Header
- Source-Routing
- Traffic Engineering
- VPN
- Service Chaining

IPv6 SR Header

- IPv6 header
  - Next header field: **43 → Routing**
- IPv6 Routing extension header
  - ***Generic header format defined in RFC 2460***
    - Next Header: IPv4, TCP, UDP, …
    - Hdr Ext Len: Any IPv6 device can skip this header
    - Segments Left: Ignore extension header if equal to 0
  - Specific data depends on 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*** *(RFC 8754)*



IPv6 uSID format

FDBB :BBBB = SRv6 uSID Block

32 bits here (but can be anything)

: 0100 : = SRV6 uSID (e.g. node uSID)

16 bits here (but can be anything)

SRV6 uSID Carrier

FDBB :BBBB : 0100 : 0200 : 0300 : 0400 : 0500 : 0000

FDBB :BBBB	: 0100	: 0200	: 0300	: 0400	: 0500	: 0000
SRv6 uSID Block	uSID 1	uSID 2	uSID 3	uSID 4	uSID 5	EoC 6

Locator advertised as /48 = uSID block + uNode

SRV6 Encapsulation

IPv6 header
```
SA:2001::1
DA:FDBB:BBBB:0:4:1:0:0:0
NH:RH
```

IPv6 extension header
```
Type:4(SRH)
NH:IPv4|SL:1
Segment List:
[0]: FDBB:BBBB:0:500:0:0:0:0
[1]: FDBB:BBBB:0:400:0:0:0:0
[2]: FDBB:BBBB:0:300:0:0:0:0
[3]: FDBB:BBBB:0:200:0:0:0:0
[4]: FDBB:BBBB:0:100:0:0:0:0
```

Service IP in IP encap
```
SA:7.5.4.3
DA:11.6.19.71
Port:UDP
```

```
UDP Header/Data
```

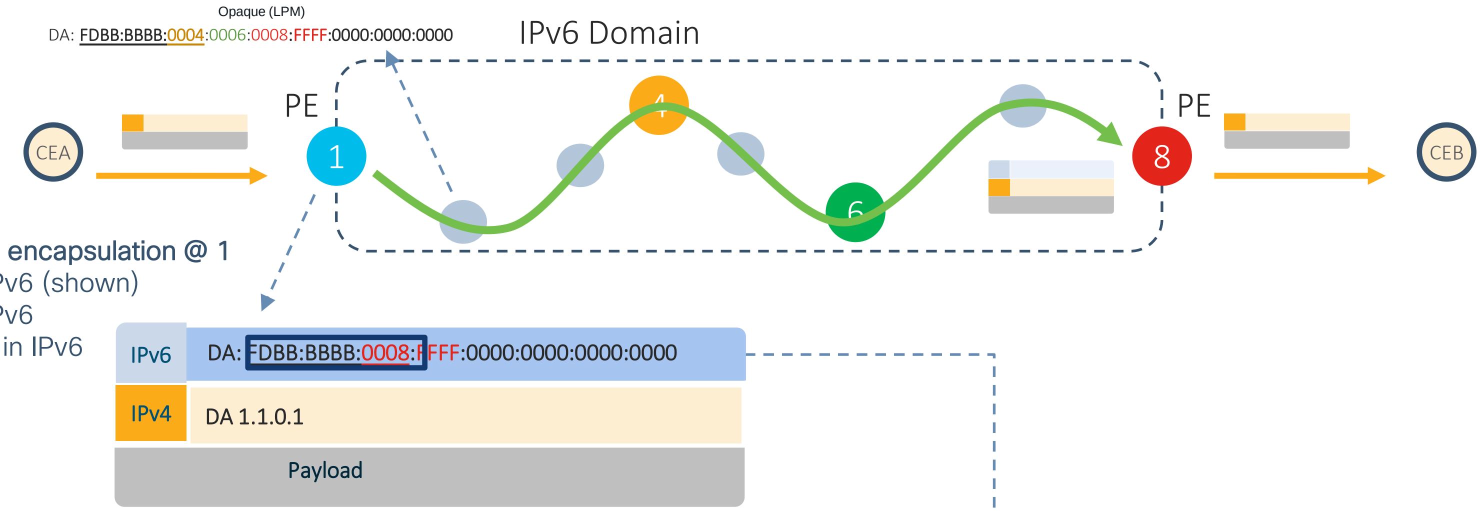
SRV6 uSID Encapsulation

IPv6 header
```
SA:2001::1
DA:FDBB:BBBB:100:200:300:400:500::
NH:Ipv4
```

Service IP in IP encap
```
SA:7.5.4.3
DA:11.6.19.71
Port:UDP
```

```
UDP Header/Data
```

SRv6 uSID – The ultra efficient Innovation



Simple IPv6 encapsulation @ 1

- IPv4 in IPv6 (shown)
- IPv6 in IPv6
- Ethernet in IPv6

A **source-routed path encoded in a single IPv6 address!**

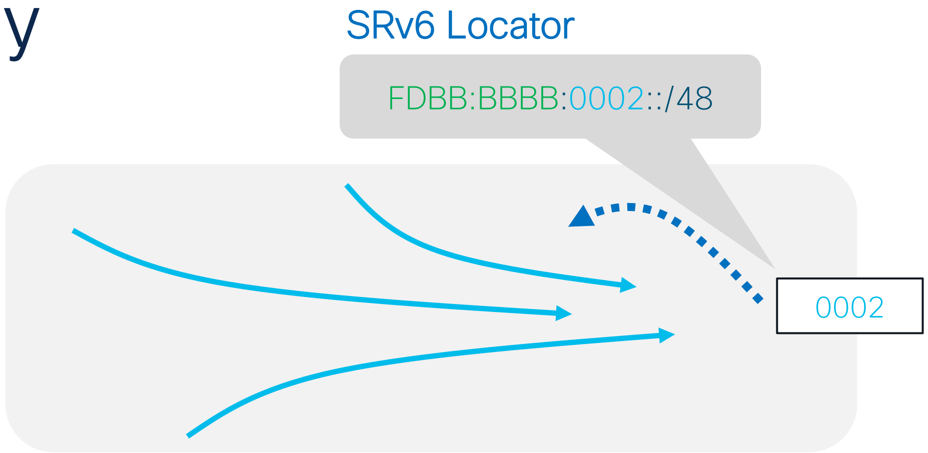
- follow igp shortest-path to node 4
- then shortest-path to node 6
- then shortest-path to node 8
- then decapsulate and lookup in VPN table FFFF

LPM – longest prefix match lookup

IPv6 uSID Block

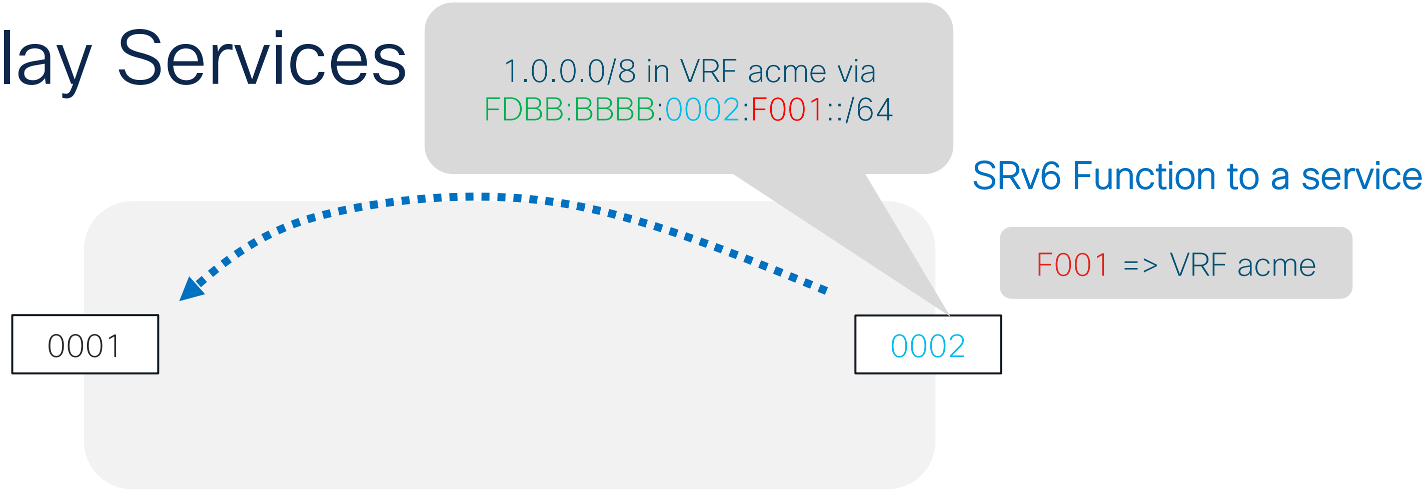
- Your network may have any pre-existing IPv6 address deployed
- uSID's are allocated from a new block
- All deployments allocate from FD/8 private block
- Let us assume: FDBB:BBBB/32 block is picked

ISIS Underlay



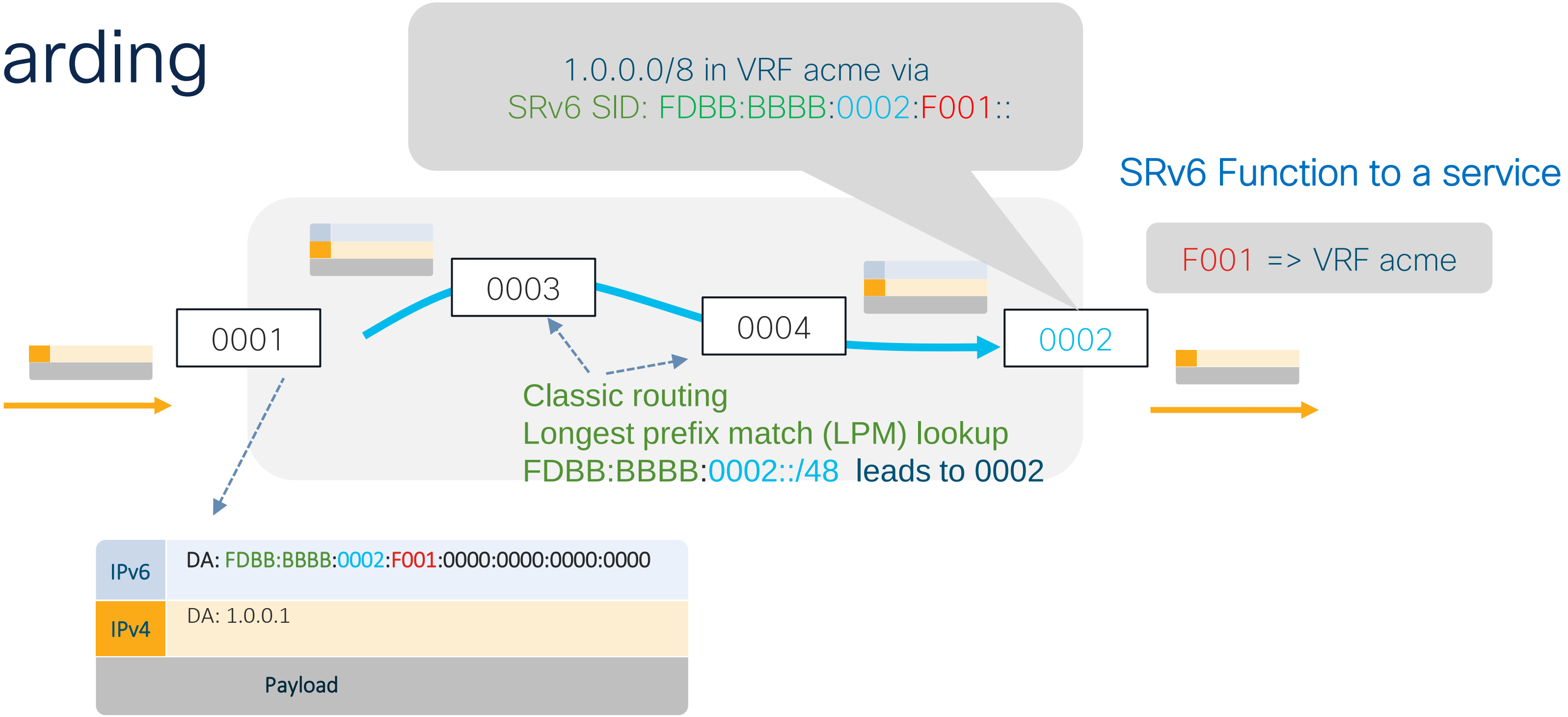
- ISIS is advertising locators as /48 IPv6 addresses
- All nodes are installing a route based on the IGP metrics
- Any packet to FDBB:BBBB:0002/48 follows the shortest path to 0002

BGP Overlay Services



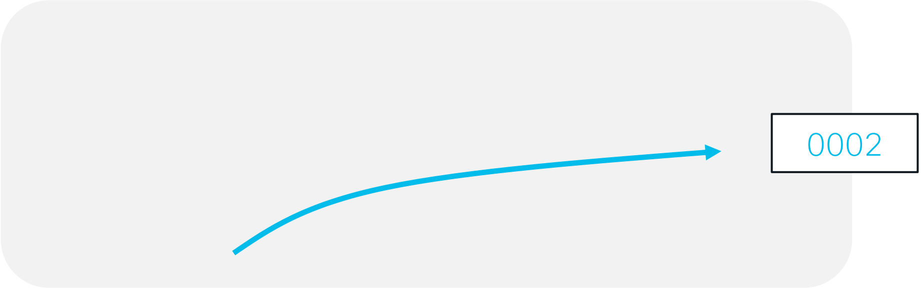
- BGP announces that 0002 has a local binding
  "F001" == "decaps and forward inner packet as per VRF acme"

Forwarding



- Simple IPv6 encapsulation at Ingress PE

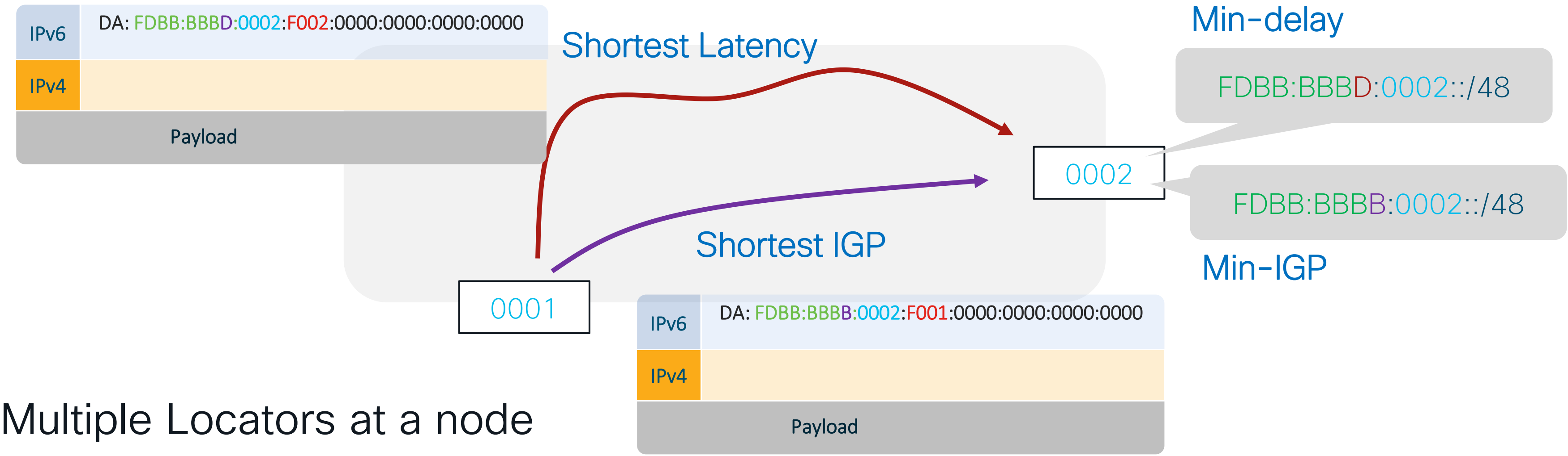
The shortest path is not always the best



- The shortest path is often optimized for lowest transport cost
- Alternative requirements
  - Lowest latency
  - Only via secured links
  - Avoid some geographies

IGP Flex-Algo

SRv6 Locators

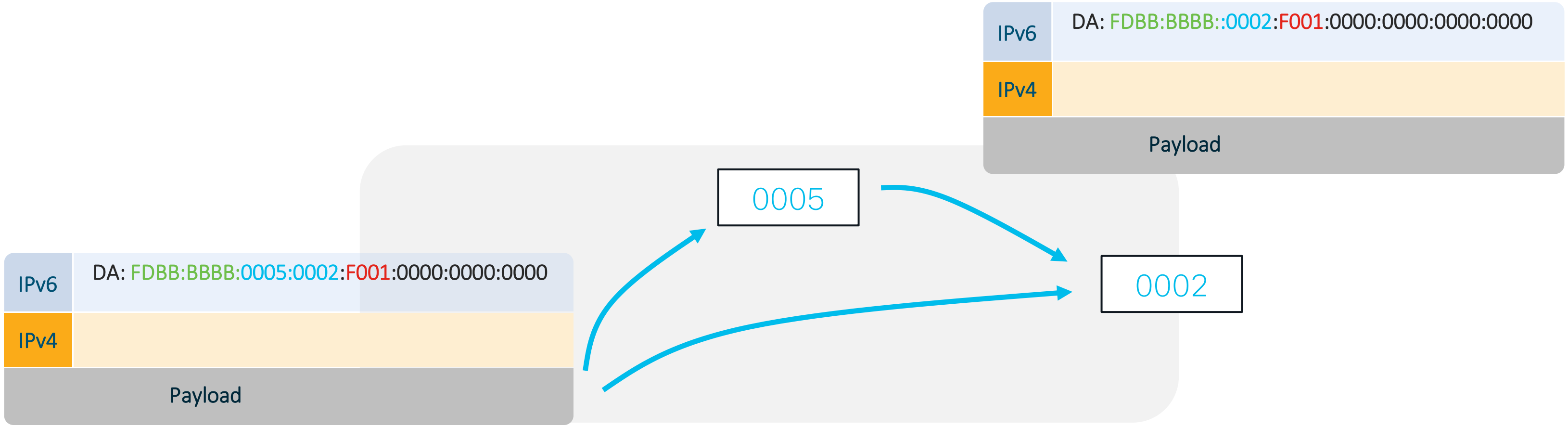


- Multiple Locators at a node
- Locators per intent
  - Min-IGP
  - Min-delay

Flex Algo Low Delay Locator

FDBB:BBBD:0002::/48

Humans combined paths

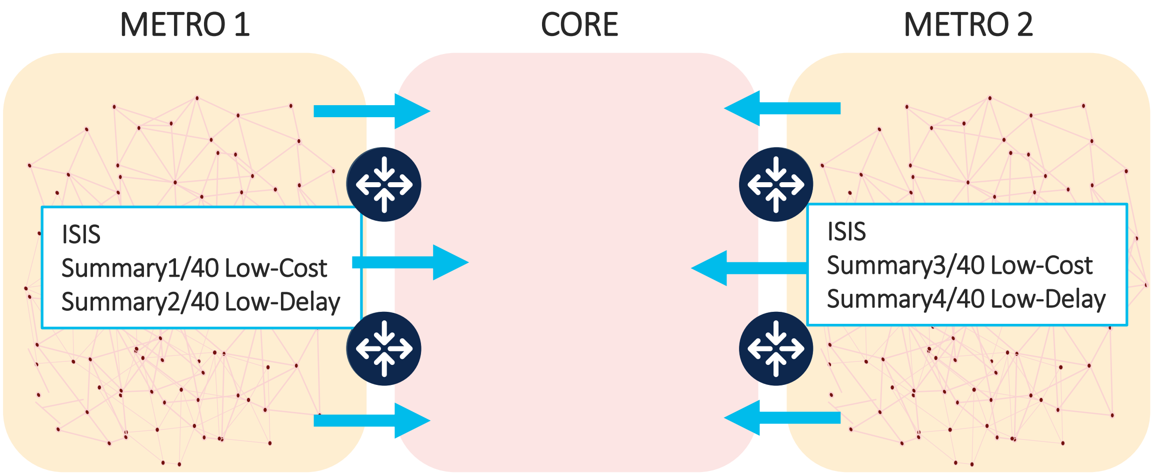


- Human analogy: I can sometimes get to my destination faster, if I take a detour when there is congestion on the highway.
- FDBB:BBBB:0005:0002::
  follows the shortest-path to 0005 and then to 0002

Ultra Scale

- Solely FDBB::/16 provides for 4 billions global locators
- Local uSID's of 32 bits provide for 4 billion local bindings
- We can finally get back to reachability with routing summarization
    - MPLS do not support routing summarization
      and hence need complex solutions like BGP-LU

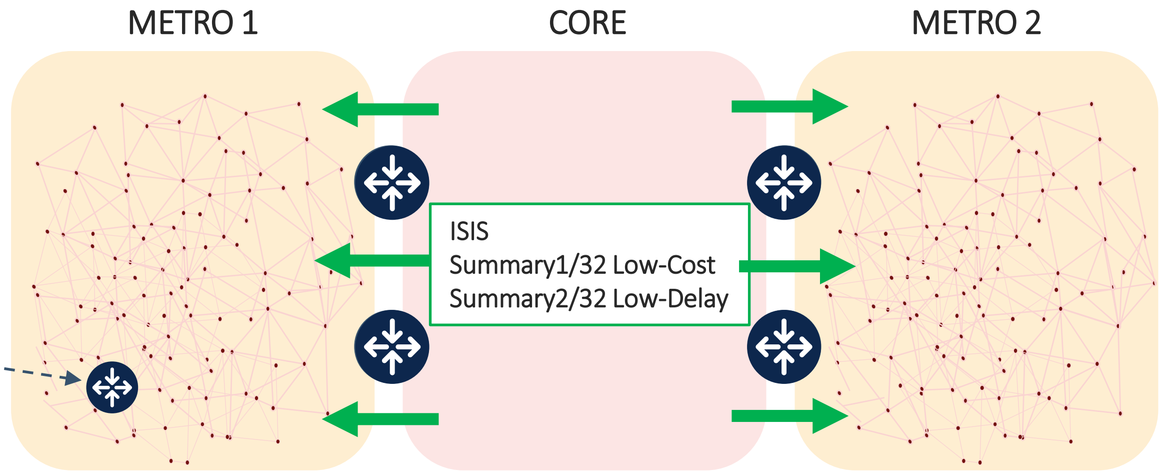
Ultimate Simplicity + Ultra Scale



- Simpler routing designs
- No BGP inter-AS Option A/B/C
- Back to basic IP routing and prefix summarization - thousands less IGP routes than with MPLS!

Ultimate Simplicity + Ultra Scale

A single entry in FIB to reach destinations in each slice outside Metro 1



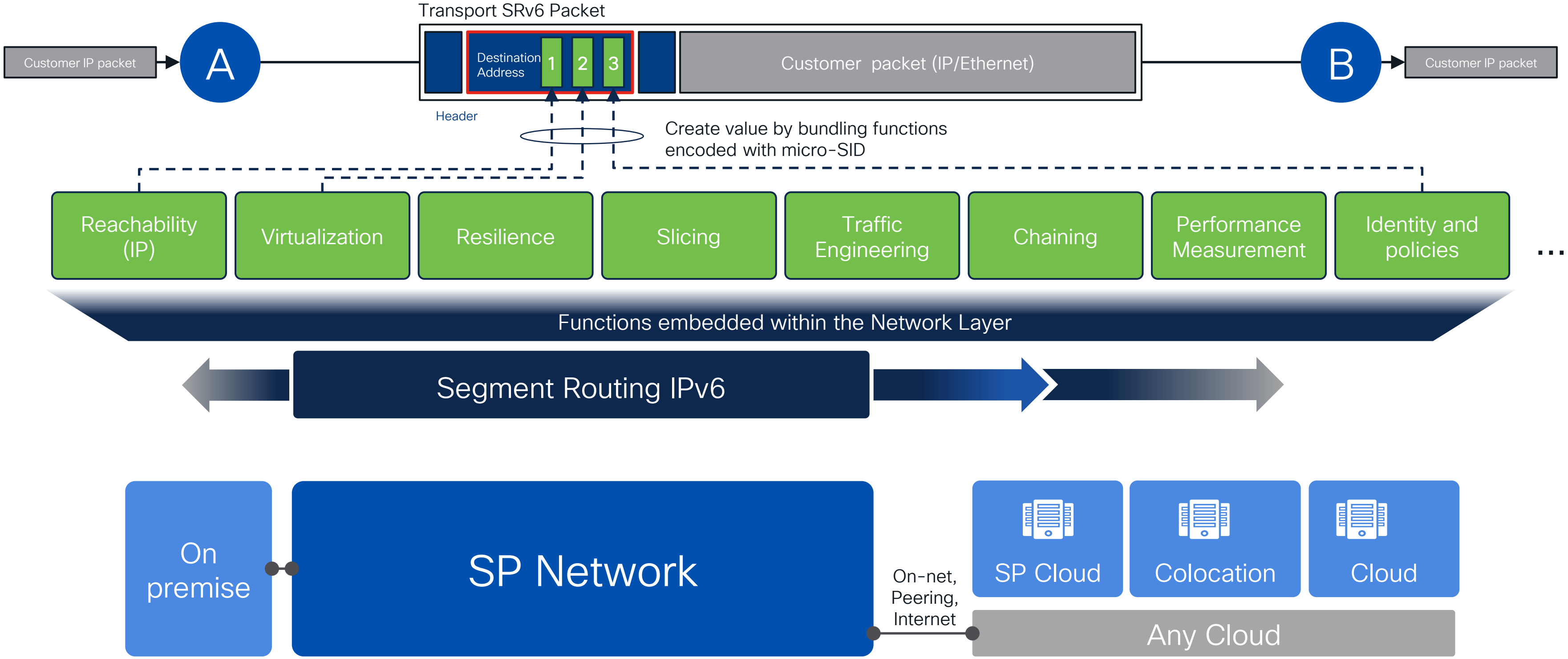
- End-to-End IGP Flex Algo Continuity (e.g. low-cost / low-delay)

SRv6 functions (Refer to : RFC 8986)

Codename		Behavior	
End	uN	Endpoint	[Node SID]
End.X	uA	Endpoint with Layer-3 cross-connect	[Adj SID]
End.B6.Insert	uB6.Insert	Endpoint bound to an SRv6 policy	[BSID]
End.B6.Encap	uB6.Encaps	Endpoint bound to an SRv6 encapsulation policy	[BSID]
End.DX6	uDX6	Endpoint with decapsulation and IPv6 cross-connect	[L3VPN Per-CE]
End.DX4	uDX4	Endpoint with decapsulation and IPv4 cross-connect	[L3VPN Per-CE]
End.DT6	uDT6	Endpoint with decapsulation and specific IPv6 table lookup	[L3VPN Per-VRF]
End.DT4	uDT4	Endpoint with decapsulation and specific IPv4 table lookup	[L3VPN Per-VRF]
End.DX2	uDX2	Endpoint with decapsulation and L2 cross-connect	[E-LINE]
End.DT2U/M	uDT2U/M	Endpoint with decapsulation and L2 unicast lookup / flooding	[E-LAN]
End.DTM	uDTM	Endpoint with decapsulation and MPLS table lookup	[Interworking]
H.Insert / H.Encaps		Headend with Insertion / Encapsulation of / into an SRv6 policy	[TiLFA]
H. Encaps.L2		H.Encaps Applied to Received L2 Frames	[L2 Port Mode]
H.Encaps.M		H.Encaps Applied to MPLS Label Stack	[Interworking]

SRV6 uSID as Service programming platform

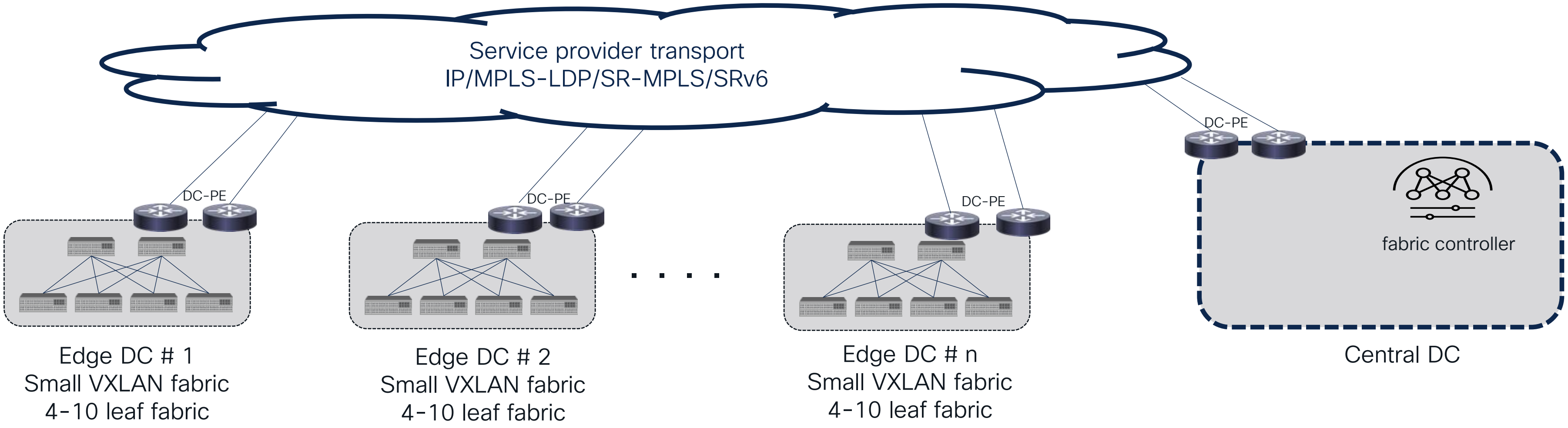
Deploy diverse functions where needed



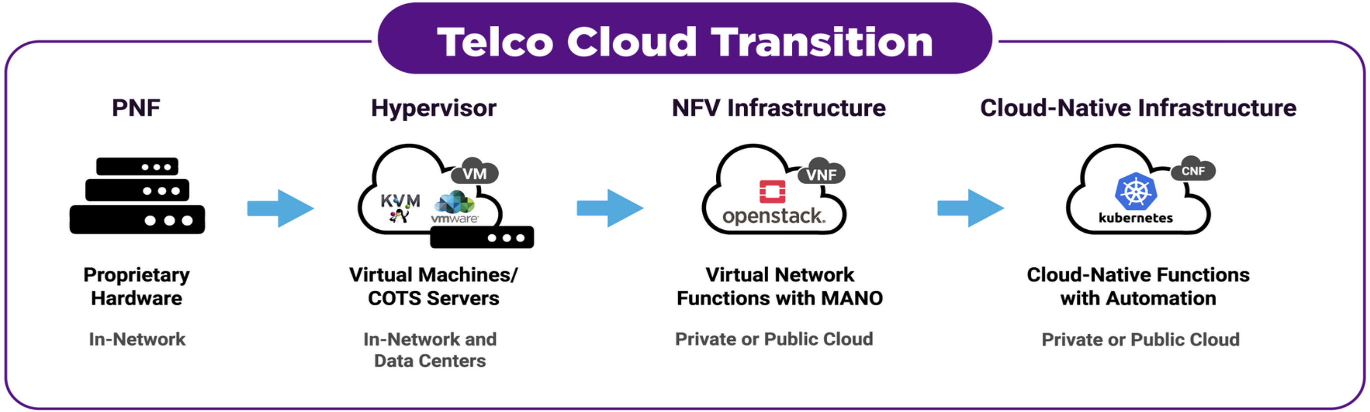
SRv6 to the host

Telco Cloud distributed DC - Present mode

- A smaller VXLAN fabric is required for relative bigger Edge DC sites
- VXLAN is only used within the DCs
- DC to transport handoff is required for all external communication (VXLAN to VRF mapping)
- FC is centrally hosted in one of the edge fabric or in a central DC can manage all edge DC sites



Potential for Simplification

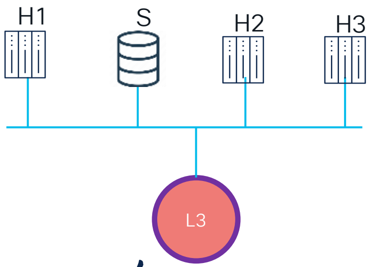


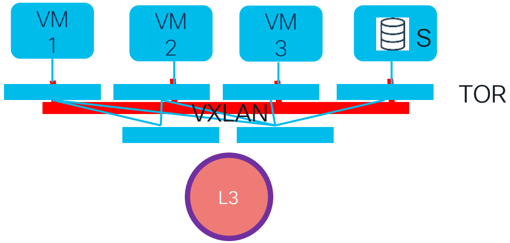
kubernetes

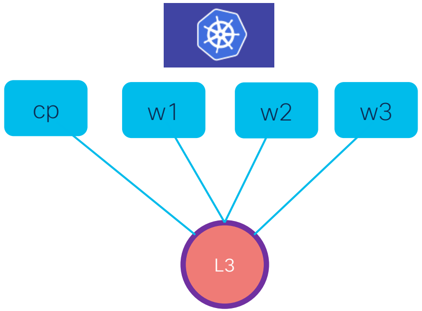
CNI

cilium

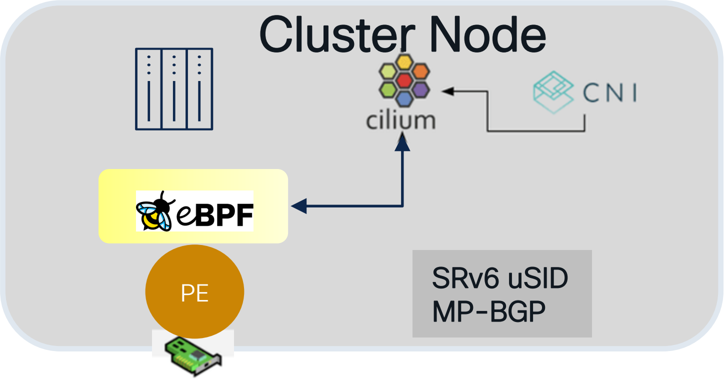
eBPF

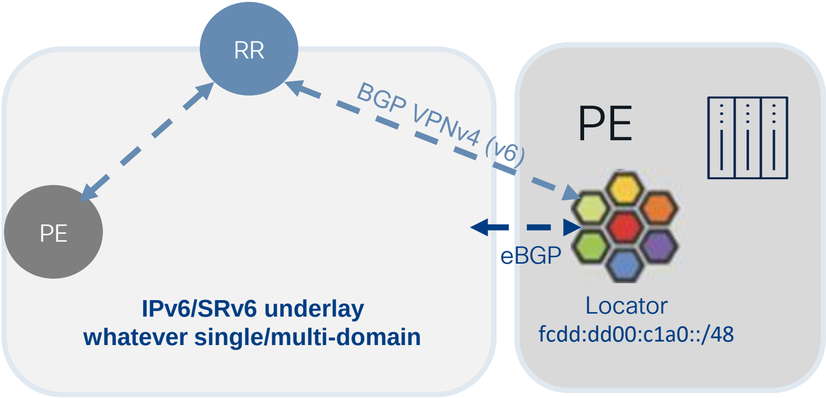


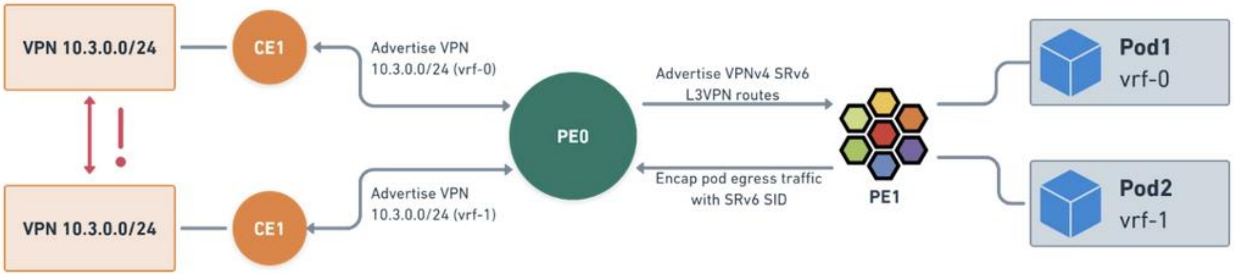




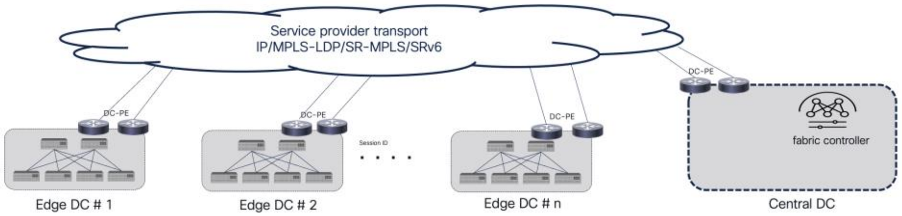
With SRv6 the compute node is becoming a PE

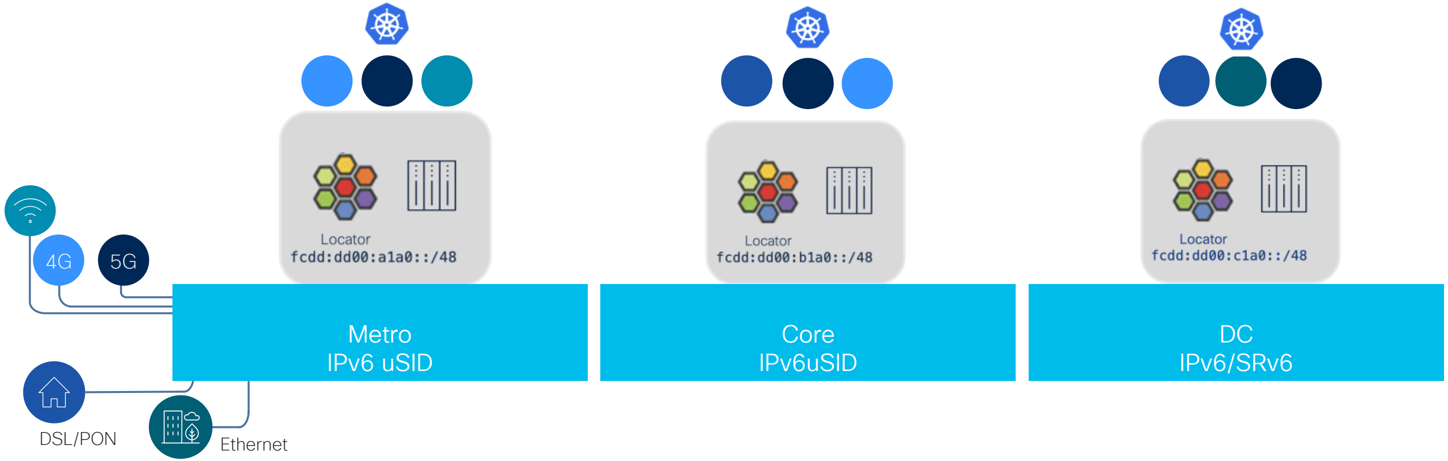






Potential Benefits



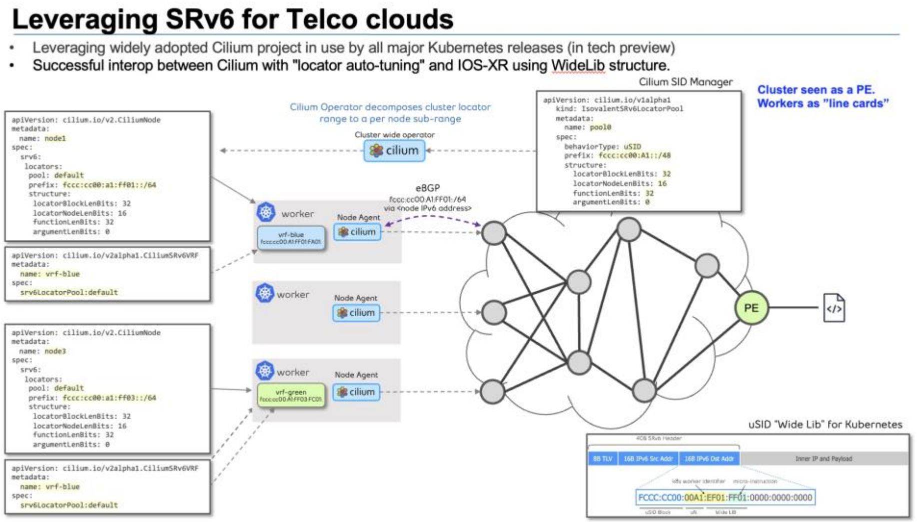


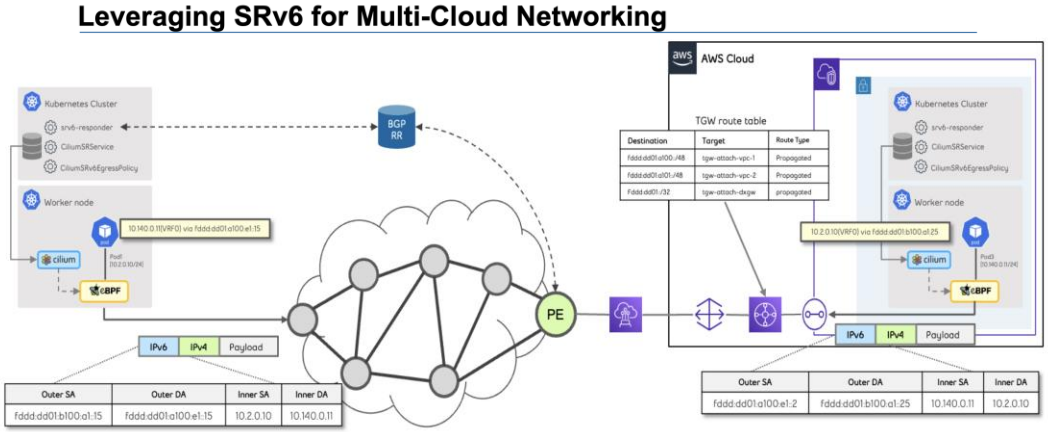
- No need of DCI
- No need of underlay L2 VXLAN
- No need of DC fabric in small sites
- Simplified networking via end 2 end VRF to the host
- Embedded security policies in Cilium
- Minimize service touch points

Bell Canada*

- One of the first SRv6 deployments
- Implementing Cilium SRv6 for Telco Cloud
- Working to extend it to public cloud
- Public recordings
  - [Presentation at CNCF]()
  - [Presentation at MPLS SD&AI]()

*Courtesy of Dan Bernier, Dir Technology Strategy, Bell Canada





Rijkswaterstaat SRv6

Business Case and Use Cases

Cees de Gruijter - Network domain architect, Rijkswaterstaat NL

Michiel Koolen - Domain Architect, Rijkswaterstaat NL

René van der Bilt - Network Architect, Royal KPN N.V

Rijkswaterstaat – what we do in 3 images

90.192 km2 water
45 km dunes
154 km dikes and dams
10 weirs
6 storm surge barriers

water system

3.462 km canal / river
92 lock complexes
128 lock chambers
325 (movable) bridges
44 Dynamic Information Panels
1416 cameras

waterways

7.372 km road
56 movable bridges
3100 video camera systems
320 GMS2 (road condition)
20.000 Speed loop pairs
17.000 signal sensors

roads

Rijkswaterstaat – Dutch government organization

- Nationwide OT-grade IP-network crucial for visibility, data collection, remote operation of objects, “smart” traffic, etc..
- ≈5000 km optical, MPLS backbone, 4 data center locations, regional traffic management centers.
- Evolution from “Enterprise” (demand/customer driven) to “Service provider” (standard services) model started in 2018.
- IT network must be made future proof, “automatable”
  - deterministic behavior,
  - autonomous functioning,
  - self healing.

RWS business case for Segment Routing

- Network **convergence** for OT: ≥1s interrupt can trigger Safety Protocols, SR-MPLS/SRv6 promises predictable fast convergence.
- **Stateless** property of SR crucial to deterministic network behavior. Minimal "unforeseen side effects" of automated changes.
  Cf. software industry best practice of stateless REST-APIs.
- **Reduce Cost** by replacing dedicated DWDM Infrastructure with Routed Optical Network (RON).
- SR Flex Algo **Traffic Engineering** can do the same as DWDM path protection, BUT with more flexibility!

Benefits of statelessness

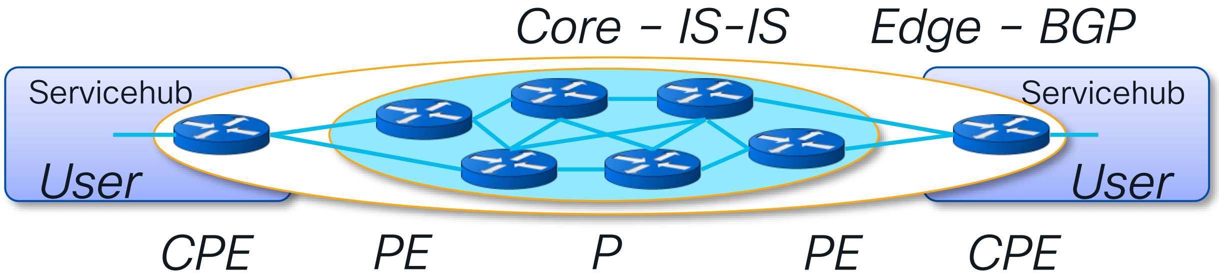
- State of the entire network is **deterministically** defined at any point in time. The network has no memory or history.
- Deterministic test conditions result in Changes **with no surprises or packet loss**. Very suitable for a Desired State automation approach.
- SR is stateless, because the SID is part of the data frame. Any “memory” is gone as soon as a packet leaves the network.
- SIDs truly behave as REST API calls to network routing plane!

Status of SR-MPLS/SRv6 implementation at RWS

- SRMPLS stack live when Backbone (P-PE) LCM in 2019.
- SR-MPLS signaling/control plane traffic for autonomous IP protection. Better network convergence. SR-MPLS at that time no business case for wider adoption over MPLS.
- SRv6 development finished and production tests in progress:
  - Multi-domain SRv6, Customer Edge is a separate uSID domain from the Backbone Core for security reasons.
  - SRv6 + Flex Algo together with DCO and RON for L2 Ethernet Private Line (EPL).
  - L2 EPL over SRv6 Transport replaces DWDM.
- Legacy DWDM network is to be discontinued in Q2 2025.

Multi domain SRv6 solves RWS specific challenges

- Huge part of the physical network is easily accessed (along public roads) – security implications!
- Multi domain SR: BGP (single control plane in our network) signals SIDs of connected PE-routers to CPE routers (co-operation with Alberto's team).
- Single domain SRv6 with IS-IS (link-state protocol) lacks security features for ≈900 servicehubs (+ 700 wireless)

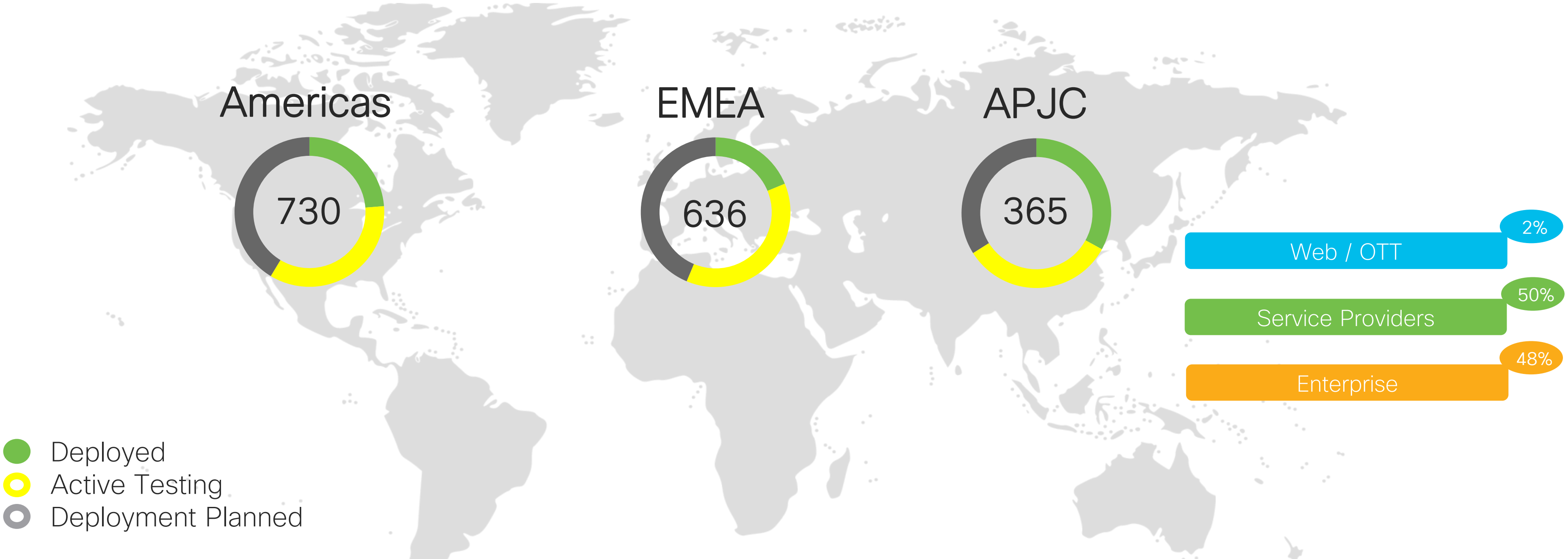


Other fun SRv6 stuff: Migration and Availability

- **Dual stack** IPv4+MPLS and IPv6+SR-MPLS within 1 "MPLS"-VPN, adds IPv6 to the backbone network without touching the IPv4 configurations or protocol stack.
- Switch between MPLS and SRv6 **without packet loss** and without changing our MPLS-VPN structure. Perfect fit with RWS chosen evolutionary development.
- **Increasing Availability** of LAN Ring topology: close LAN Rings along highway sections by emulating "Fiber path" via the IP Backbone using L2VPN services over SRv6 Transport. Huge cost savings on fiber cable installation.

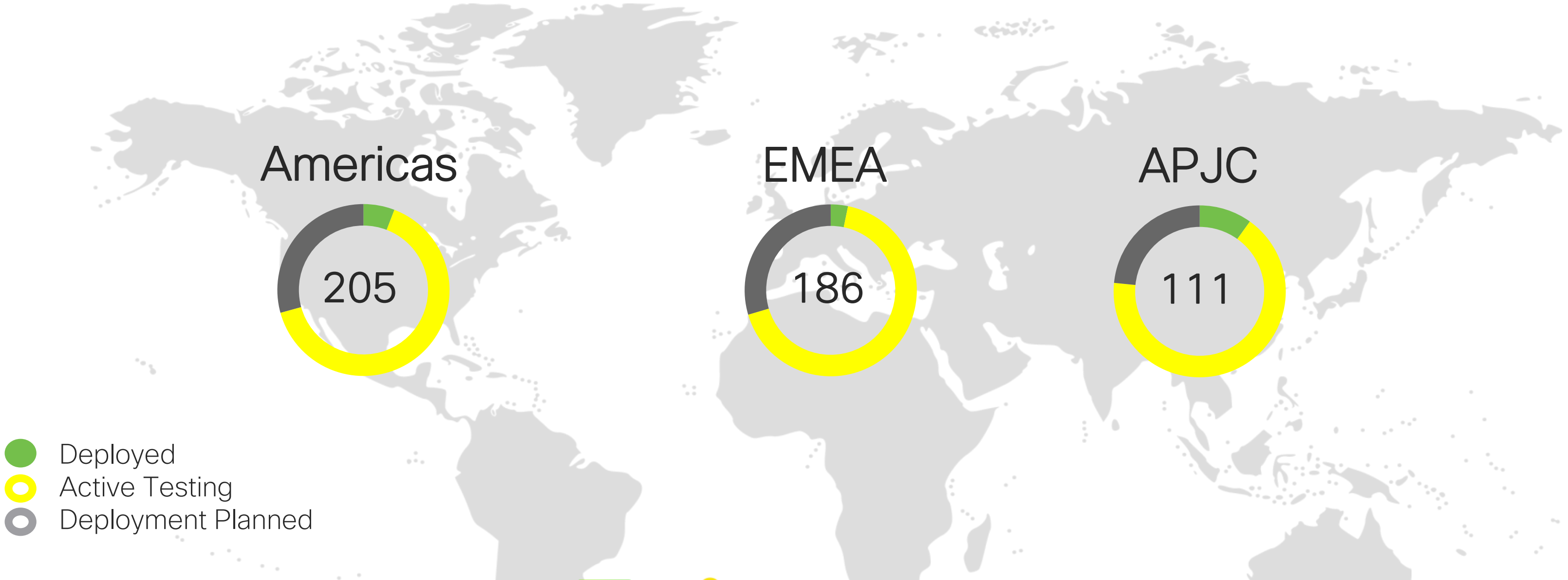
Deployments and Interoperability

From concept to standardization to deployment leadership



Bell, SoftBank, vodafone, Alibaba.com, Walmart, colt, South Slope, orange, Microsoft, Aussie Broadband, indosat ooredoo, altice, China unicom 中国联通, verizon, Telefonica, airtel, kt, eolo, etisalat, EQUINIX, Google, TELUS, docomo, usa gov, LINE, iliad, Rakuten

SRv6 … at Record-Speed



Rich SRv6 uSID Ecosystem

Network Equipment Manufacturers

CISCO, HUAWEI, NOKIA, ciena, JUNIPER NETWORKS, H3C, ERICSSON, ZTE, ARRCUS, ARISTA

Smart NIC / DPU

intel, NVIDIA

Open-Source Applications

Pyroute2, IPTABLES SERA, GoBGP, exa BGP, eBPF, cilium, FRRouting, TCPDUMP, WIRESHARK, SNORT, GoBMP

Open-Source Networking Stacks

fd.io, P4, OPEN Compute Project, SONiC, Open vSwitch

SR v6 uSID

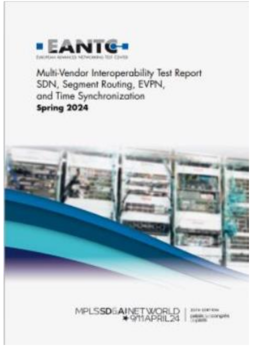
Merchant Silicon

CISCO Silicon One, BROADCOM, MARVELL, BAREFOOT NETWORKS (intel)

Partners

f5, kaloom, TREND MICRO, ENEA Qosmos Division, 6WIND, NoviFlow, KEYSIGHT TECHNOLOGIES, spirent Promise. Assured.

EANTC 2024 – Multi-Vendor Interop



Link

- Sixth year of public SRv6 interop tests
- uSID established as de-facto industry standard at EANTC 2024
  - *ALL* SRv6-related testing conducted exclusively with uSID
  - 10 vendors with 21 routers/switches – merchant (BRCM J/J+/J2) & custom-silicon

Cisco, Arista, Ciena, Nokia, ZTE, H3C
Huawei, Juniper Networks, Keysight Technologies, Ericsson
Spirent, Arrcus

2024

2023

Conclusions

IP is back and better than ever.

SR Segment Routing

Build anything

Simplified, scalable, and versatile networks that are self-sufficient

Self-sufficiency is standard

End-to-end connectivity with SLA
- From Host to Cloud through DC, Access, Metro, Core.
- No protocol conversion or gateways at domain boundaries

Any service, without any shim
- VPN, Slicing, Traffic Engineering, Green Routing, FRR, Host networking

Better scale, reliability, cost, and seamless deployment in Brownfield

Essential embedded assurance

Active probing between Fabric Edges **along all ECMP paths**

High-capacity probe generation and ingestion powered by Silicon One (14MPPS)

Continuous **routing monitoring**

Advanced analytics and intelligent service optimization driven by AI

Measure everything

IPM

SLA monitoring
Integrated
Performance
Measurement
Routing Analytics

Continue your education

Visit the World of Solutions for Demos

- Introduction to SRv6 uSID technology **BRKSPG-2203**
- Advanced Innovations in SRv6 uSID and IP measurements – **BRKSPG-319**
- Troubleshooting Segment Routing - **BRKSPG-3624**
- Segment Routing innovations in XE – **BRKENT-2520**

Continue your education

- Visit the Cisco Showcase for related demos
- Book your one-on-one Meet the Engineer meeting
- Attend the interactive education with DevNet, Capture the Flag, and Walk-in Labs
- Visit the On-Demand Library for more sessions at ciscolive.com/on-demand. Sessions from this event will be available from March 3.

Webex App

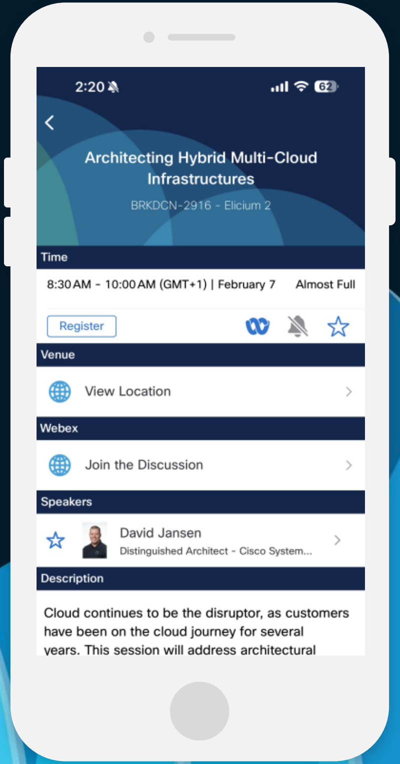
Questions?

Use the Webex app to chat with the speaker after the session

How

1. Find this session in the Cisco Events mobile app
2. Click “Join the Discussion”
3. Install the Webex app or go directly to the Webex space
4. Enter messages/questions in the Webex space

Webex spaces will be moderated by the speaker until February 28, 2025.



Fill Out Your Session Surveys

Participants who fill out a minimum of 4 session surveys and the overall event survey will get a unique Cisco Live t-shirt.

(from 11:30 on Thursday, while supplies last)

All surveys can be taken in the Cisco Events mobile app or by logging in to the Session Catalog and clicking the ‘Participant Dashboard’

Content Catalog

Cisco Live Amsterdam 2025 IPv6 Learning Map

Sunday–9th

TECIPV-2000 13:45
IPv6 in the Host and in the Local Network

Monday–10th

TECENT-2150 8:30
6+3=100! Use IPv6 and Python 3 to Transform how you do Networking

TECIPV-2001 8:45
IPv6 Beyond the Local Network

Tuesday–11th

BRKIPV-1007 8:00
Deploying Catalyst Center for IPv6 Networks

BRKSEC-2044 10:30
Secure Operations for an IPv6 Network

IBOIPV-2000 13:30
Sharing Experience on IPv6 Deployments

BRKSPG-2203 14:30
Introduction to SRv6 uSID Technology

BRKIPV-2191 16:30
IPv6:: It's Happening!

Wednesday–12th

BRKEWN-2834 8:00
IPv6-Enabled Wireless (Wi-Fi) Access: Design and Deployment Strategies

CTF-1001 10:15
IPv6: The Internet's best kept secret!

BRKIPV-2186 13:15
IPv6 Networking in a Cloud Native World

CISCOU-1038 14:45
IPv6 Groove: Get By with a Little Help from My Friends!

BRKENT-2008 13:00
Goodbye Legacy, the Move to an IPv6-Only Enterprise

BRKIPV-1616 16:00
IPv6 – What Do You Mean There Isn't a Broadcast?

Thursday–13th

IBOENT-2811 11:30
Everything You Wanted to Know about IPv6 but Were Afraid to Ask

IBOIPV-2000 13:30
Sharing Experience on IPv6 Deployments

BRKSPG-3198 14:15
Advanced Innovations in SRv6 uSID and IP Measurements

BRKOPS-2223 15:00
The Network of the Future is Here - Let's Automate your IPv6 deployment with Python!

BRKIPV-2228 17:00
The Automation Travel Guide for Your IPv6 Journey!

Friday –14th

BRKIPV-2418 9:00
Deploying IPv6 Routing Protocols: Specifics and Considerations

BRKENT-3340 11:00
The Hitchhiker's Guide to Troubleshooting IPv6

BRKENT-3002 11:15
IPv6 Security in the Local Area with First Hop Security

Walk in Labs

LABIPV-1639	IPv6 Foundations: A Dive into Basic Networking Concepts
LABIPV-2640	IPv6 Deep Dive: Beyond Basics to Brilliance
LABMPL-1201	SRv6 Basics
LABSP-2129	SRv6 Micro-Segment Basics
LABSP-3393	Implementing Segment Routing v6 (SRv6) Transport on NCS 55xx/5xx and Cisco 8000: Advanced

Instructor-led Labs

LTRIPV-2222	Implementing Future-Ready Networks - Deploy IOS XE IPv6 Configuration with Python!
LTRSPG-2212	SRv6 and Cloud-Native: A Platform for Network Service Innovation

Thank you

cisco Live!

GO **BEYOND**