SRV6 – DEPLOYMENT & USECASES

Aditya Kaul

Principal Solution Architect – Juniper Professional Services

XL axiata

Juniper Networks | Driven by Experience™

AGENDA

- SRv6 Industry Outlook
  - Quick Update
- SRv6 Deployment @ XL Axiata
  - SRv6 Design
  - Deployment Challenges
  - Future Goals

JUNIPER NETWORKS

SRV6 INDUSTRY OUTLOOK QUICK UPDATE

JUNIPER NETWORKS

SRv6: What's all the hype about ? (vs MPLS)

Operational Simplicity: Stateless, Programmable, Single data plane across DC/Metro/Core, BGP Free Transport (BGP-LU isn't required for scaling the networks, DCI GW not required)

High Availability: Reliable link/node protection with TI-LFA, Load-balancing with IPv6 flow label

Scalability and Efficiency: IPv6 scale, Locator summarization, Longest Prefix Match (just like Internet routing), Shim Less vs VxLAN/MPLS Label

Seamless Integration: IPv6 data plane, Non-SRv6 routers co-existence, can also leverage 6PE MPLS in absence of IPv6 Core, Seamless & Better Migration than MPLS → SR-MPLS

Application-aware routing: Applications can program SRv6 and define constraints as most of the end Hosts & Applications support IPv6

How to choose between SR variants ?

SPS WHO RUN MPLS NETWORKS AND LIKE IT

Today's Typical Deployments
(core, edge, transit)

SR-MPLS v4 Infra

Accelerate Deployments
(leverage existing MPLS)

SR-MPLS v6 (SRo6)

IPv6 Infra
(Just works)

SPS WHO WANT TO MIGRATE AWAY FROM MPLS

Today's Typical Deployments
(5G x-haul, Metro, DC, Scaled Core &Edge)

SRv6

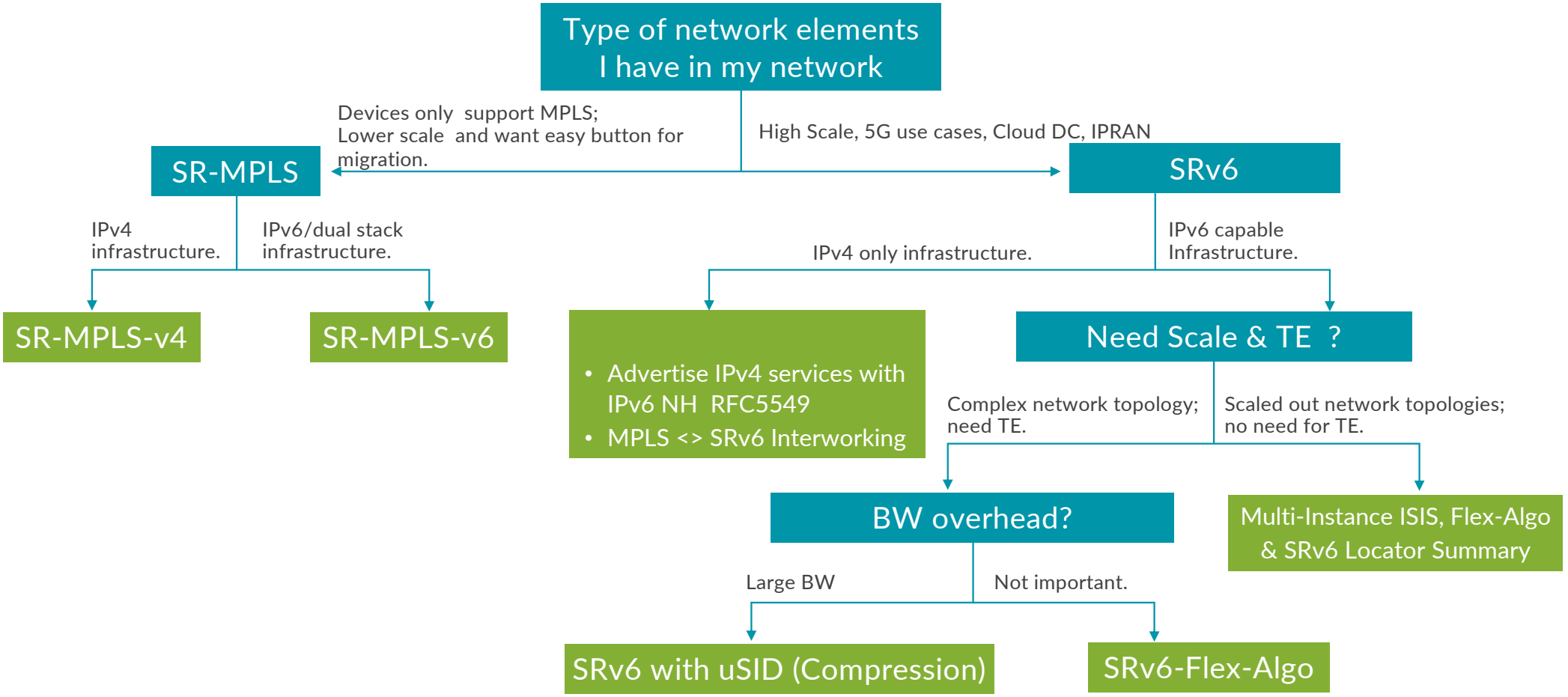
Co-Existence
Support

SRv6 uSID

Better Scale , Economics &
Easier Migration

Freedom of choice for transport technology

Keep IT simple…



SRv6: Industry Standards & Evolution

Ready to consume

TI-LFA

Prefix Summarization

Flexible Algorithm

VPN services

SR Policy

OAM

SRv6-MPLS Interworking

Multicast: BIERin6, SR-P2MP

SRH Compression: CSID Flavors–NEXT-C-SID (uSID)

Still cooking

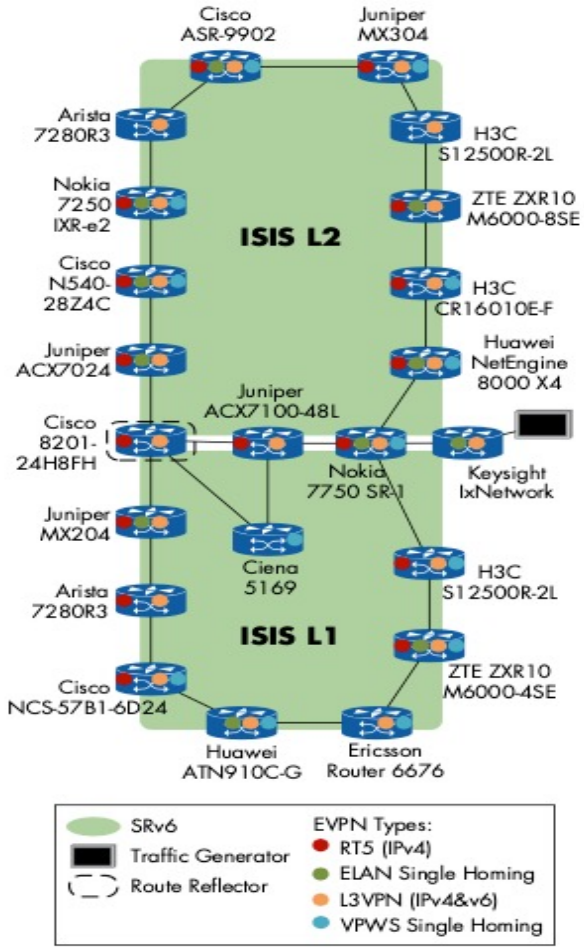
SRH Compression: CSID Flavors–NEXT-C-SID & REPLACE-C-SID Interop Inter-Op/Multiple data-plane (https://datatracker.ietf.org/doc/draft-ietf-spring-srv6-srh-compression/)

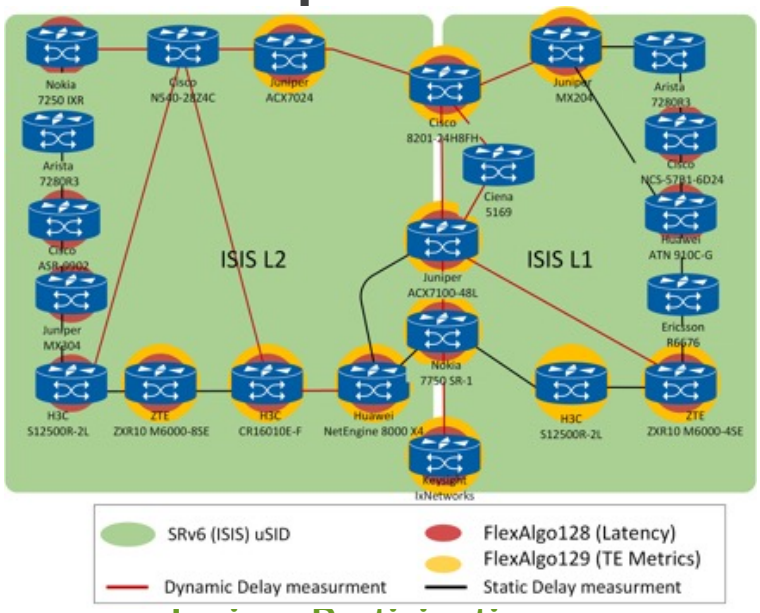
Multicast: Replication Segment (https://www.ietf.org/archive/id/draft-ietf-spring-sr-replication-segment-16.txt)

Analytics & Telemetry: Sensors, Flows

SRv6 Multi-Vendor Interop 2024

Arista	Cisco	Microchip	ZTE
Arrcus	Huawei	Nokia	Ericsson
Calnex	Juniper	Ribbon	
Ciena	Keysight	Spirent	





Juniper Participation

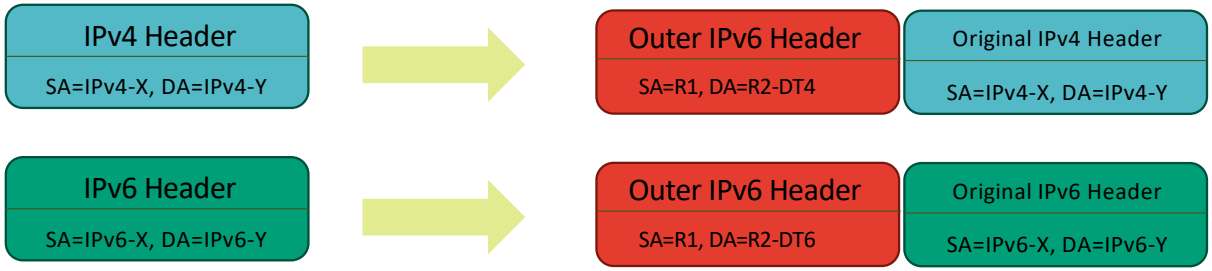
ACX5448-M	MX304
ACX7024	PTX10004
ACX71000-48L	QFX10002-72Q
MX10008	QFX5110-48S
MX204	QFX5120-32C
MPC10E	QFX5120-48Y
Paragon Insights	QFX5210-64C
Paragon Pathfinder	

2024 SRv6 uSID

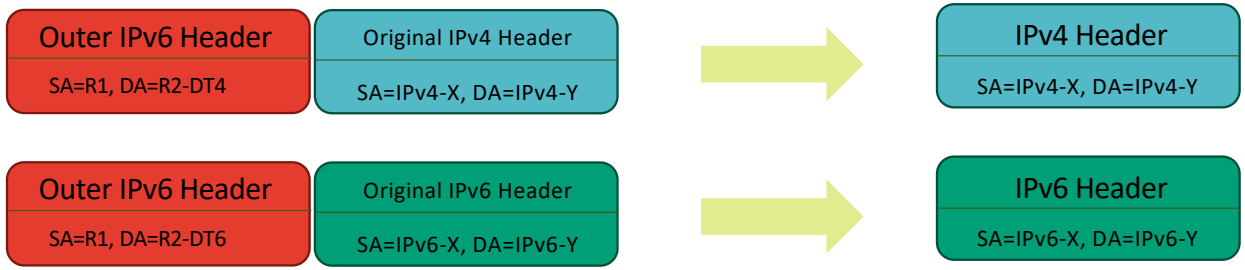
- SRv6 Micro Sid (uSID):
  - GRT (uDT4, uDT6, uDT46)
  - L3VPN (uDT4, uDT6, uDT46)
  - EVPN E-Line
  - EVPN E-LAN
  - EVPN T5
  - SRv6 locator summarization
  - SRv6 transposition
  - SRv6 TE policies with explicit paths
  - SRv6 Flex-Algo
    - Dynamic latency metric
    - Admin groups
    - FA constraints (max latency, min bandwidth, reverse affinity)

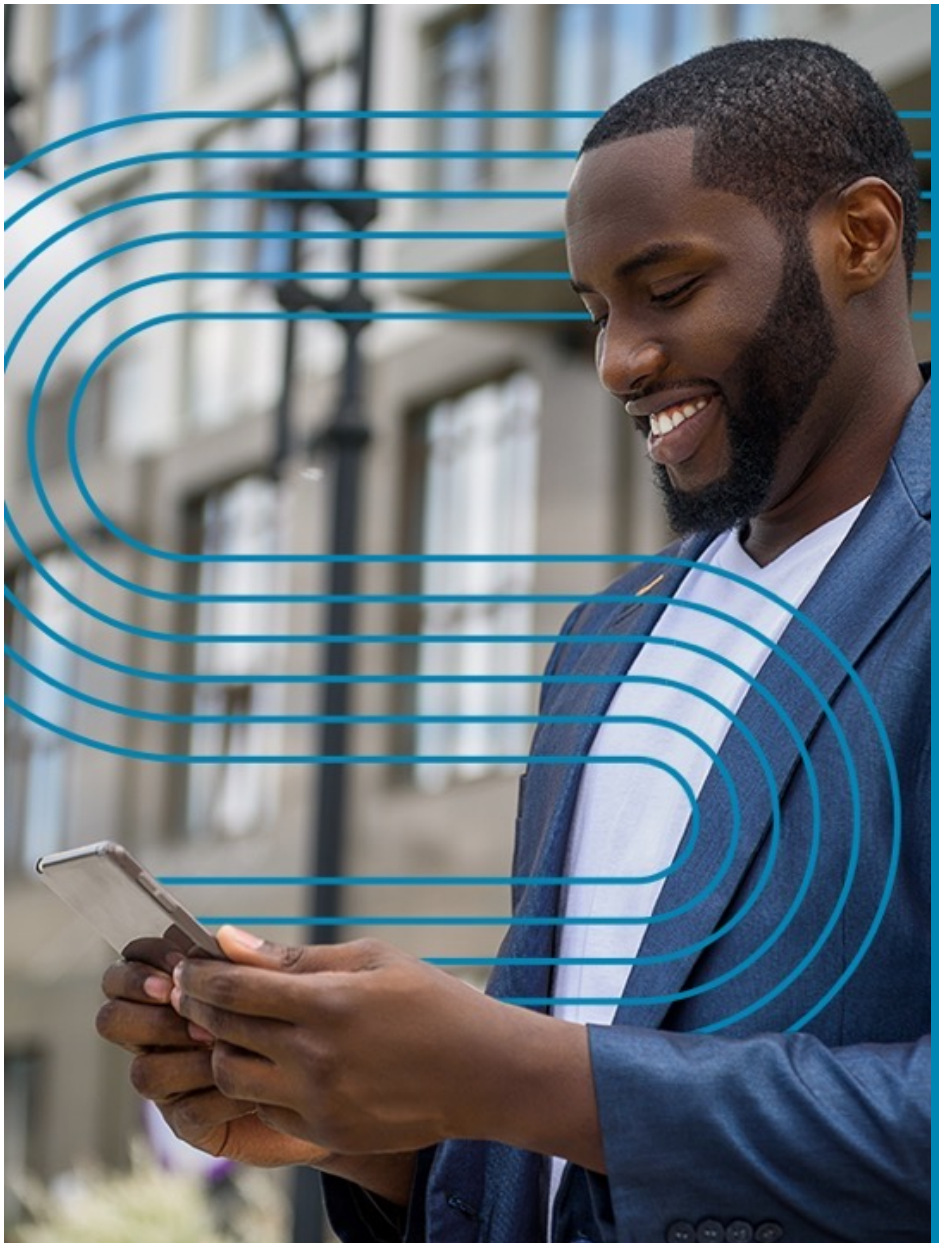
SRv6 SRH-less Operations Summary

- Ingress (for L3 services)
  - One IPv6 encapsulation



- Transit
  - standard IPv6 forwarding, based on SRv6 locator in destination IPv6 address
- Egress
  - one IPv6 decapsulation
  - subsequent lookup in specific table (VRF) and forwarding accordingly

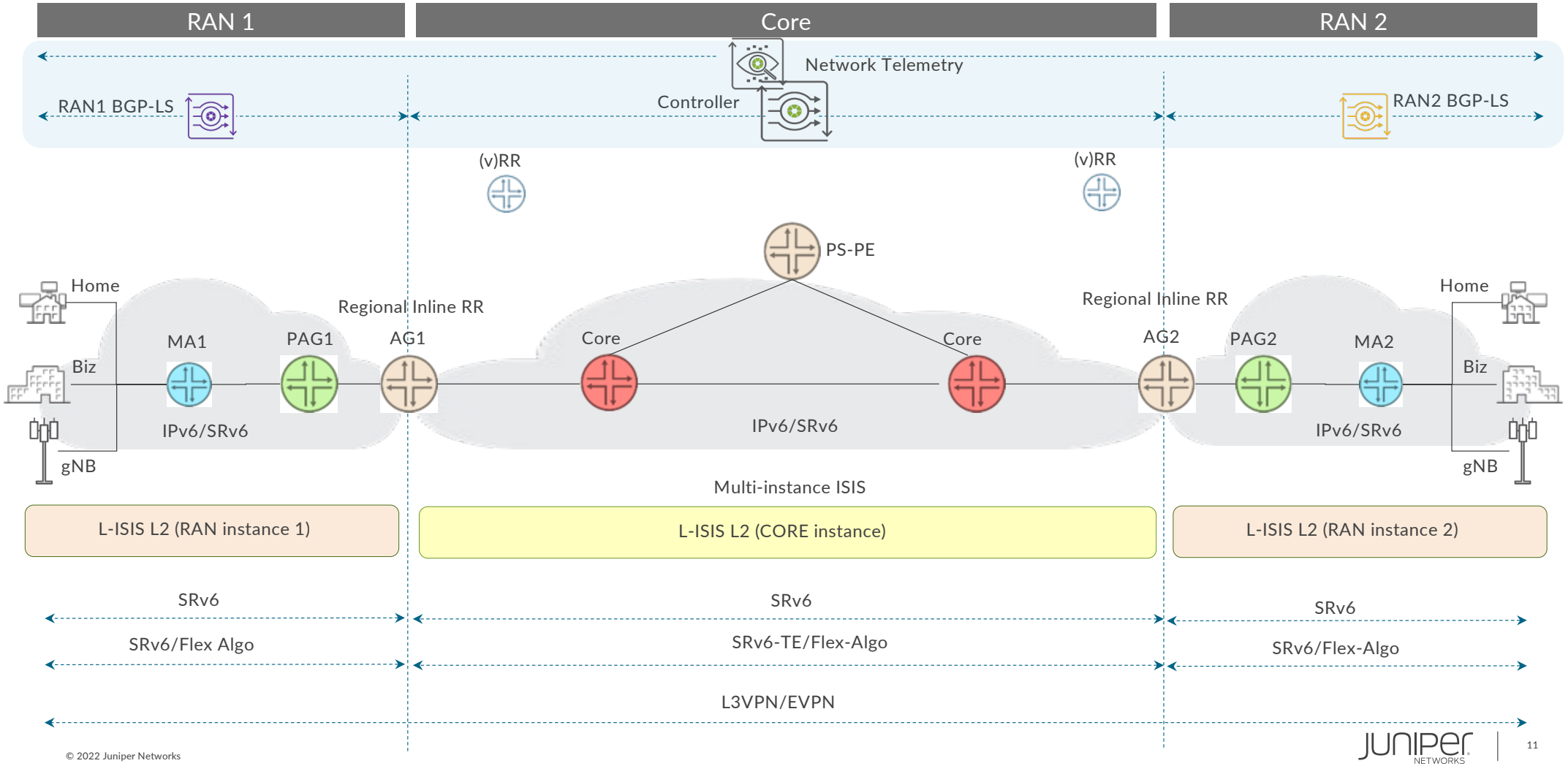




SRV6 UPDATE
XL AXIATA

JUNIPER NETWORKS

High-level Transport Architecture



LOCATOR ADDRESSING DESIGN

Locator Design

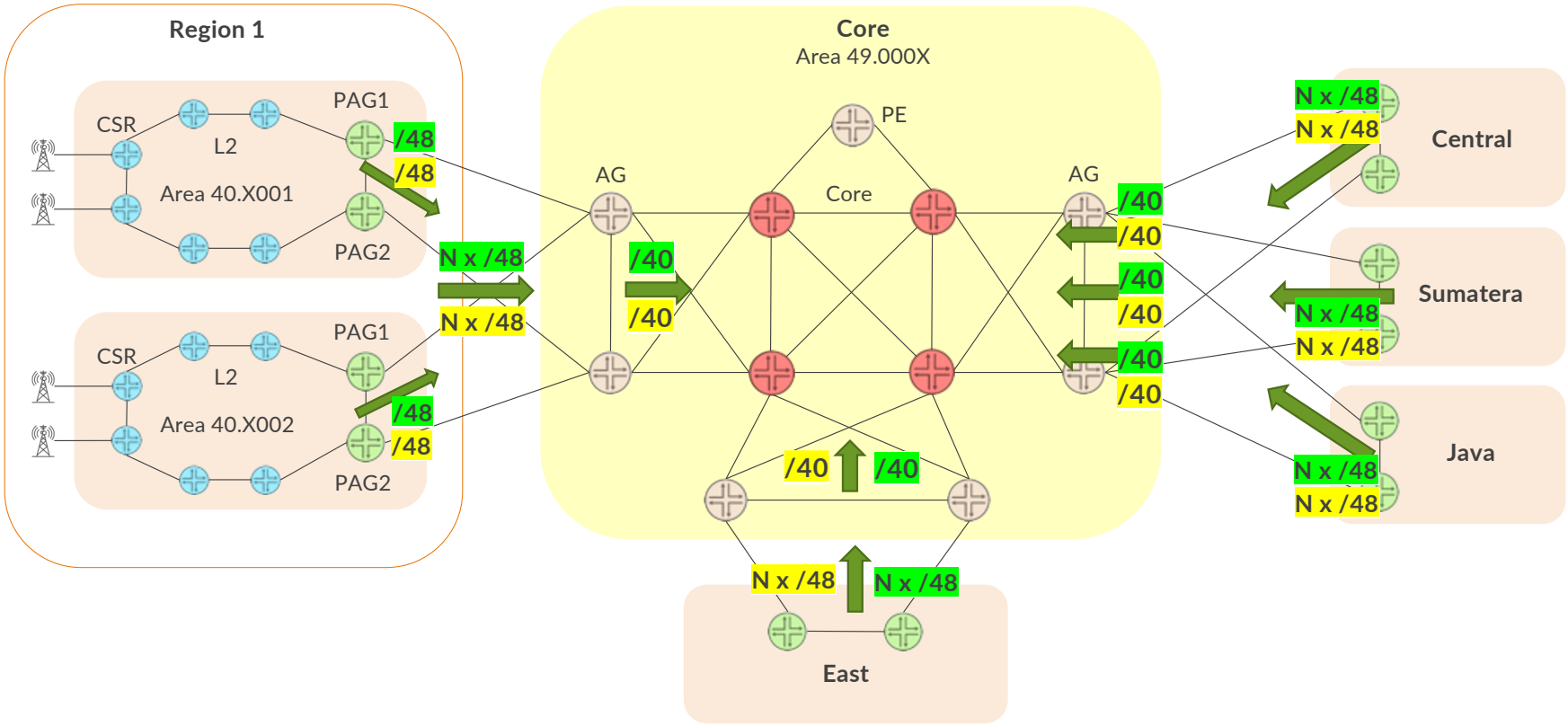
IPBB

Locator Block (28bit) for cross-region			Node ID (20bit) for cross-region			
Locator Block (32bit)				Node ID (16bit)		
Prefix	Reserved	Flex Algo	Region	Sub Region	Reserved	NE ID
20 bits	4 bits	4 bits	4 bits	4 bits	4 bits	8 bits
FC00:A	0	0	0	0-F	0-F	00-FF
Fixed	16 instances	16 instances	16 instances	16 instances	16 instances	256 instances

IPRAN

Locator Block (28bit) for cross-region			Node ID (20bit) for cross-region			
Locator Block (32bit)				Node ID (16bit)		
Prefix	Reserved	Flex Algo	Region	PAG Ring	Sub Ring	NE ID
20 bits	4 bits	4 bits	4 bits	8 bits	4 bits	4 bits
FC00:A	0	0	2-F	00-FF	0-F	0-F
Fixed	16 instances	16 instances	16 instances	256 instances	16 instances	16 instances

Prefix & Base Locator Summarization



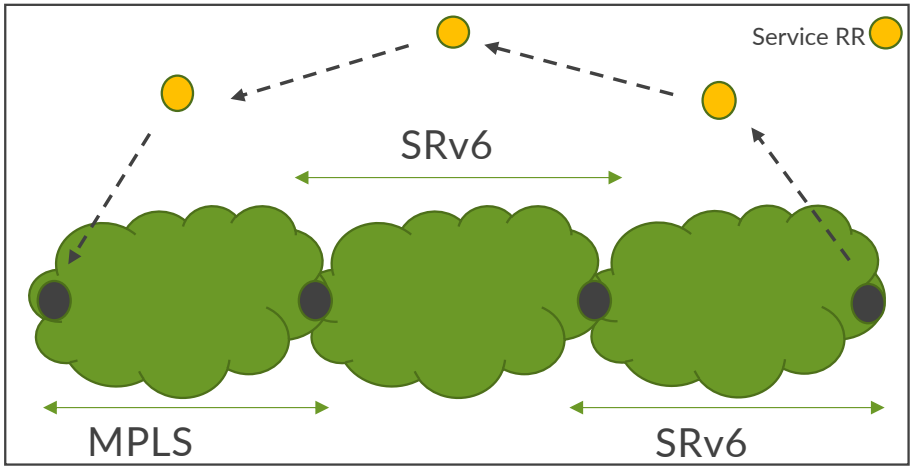
SRv6 <-> MPLS Interworking for Seamless SRv6 Deployment & Co-Existence

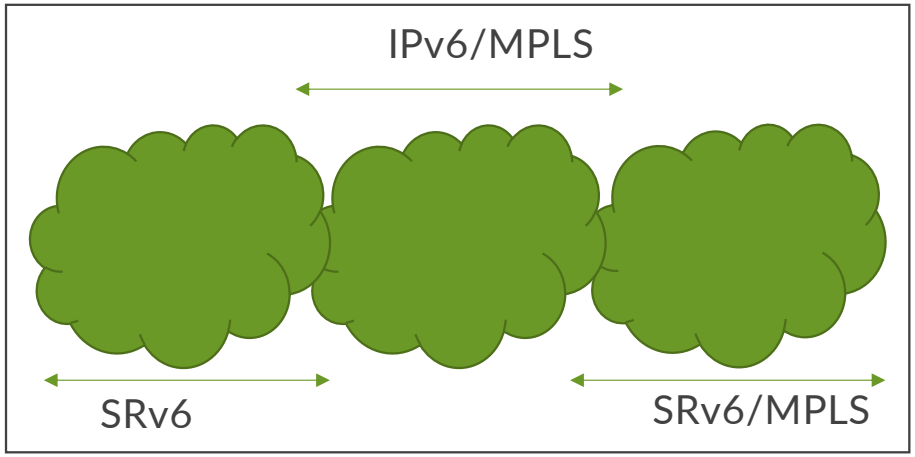
Transport scenarios

- Provide mechanisms for SRv6 domains to talk to SRv6 Domains Over IPv6 Networks
- SRv6 ABRs to advertise SRv6 Aggregate locators across the SRv6 network
- SRv6 domain can service constraints using color/flex-algo/SRv6-TE.

Gateway Interworking scenarios

- Provides service routes interworking at SRv6/MPLS ABR/GW routers
- SRv6 SID to Label mapping for both SRv6 to MPLS and MPLS to SRv6
- Option-AB and Option-B scenarios for SRv6 ←→ SRv6





Proposed SRv6 Evolution approach

1. Start with the IPv6: Dual Stack IPv6 in Core and IPRAN
2. Greenfield deployment of new SRv6 rings in the RAN regions and SRv6<>MPLS Interworking at AGG
3. Co-Existence of SRv6 and MPLS in Core, Connect new services to the SRv6 plane
4. Introduce SRv6-TE in Core and Flex-Algo in Core and SRv6 RAN
5. De-commission IP/MPLS in the Core but continue SRv6<> MPLS Interworking at AGG to connect legacy IP/MPLS RAN

CSR-SRv6 <-> PS-PE (4G L3VPN)
SRv6 Services in Production Network

SRv6/MPLS RAN <-> IPBB (SRv6)

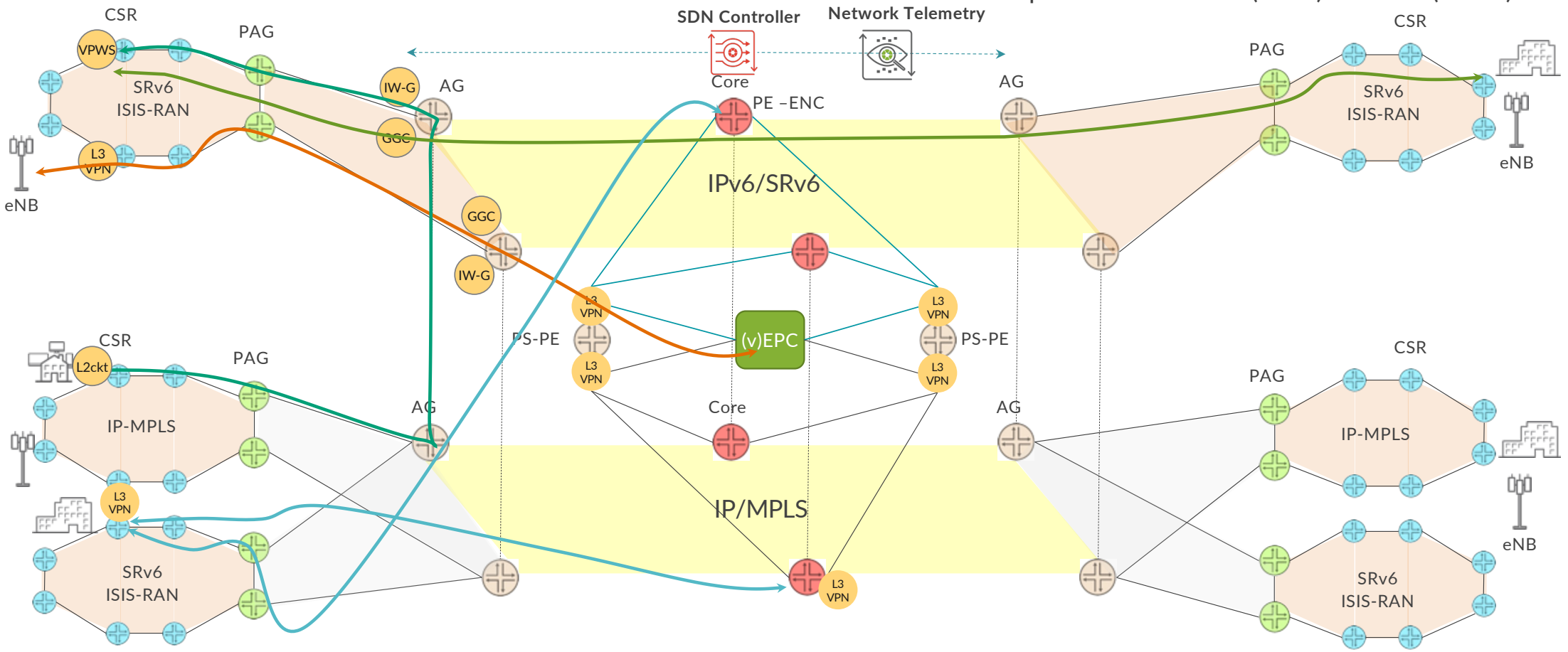
Mobile: eNB (CSR-SRv6) <-> vEPC (PS-PE-MPLS)

Fttx VPWS: CSR (SRv6) <-> PS-PE (MPLS)

Enterprise-EVPN : CSR (SRv6) <-> CSR(SRv6)

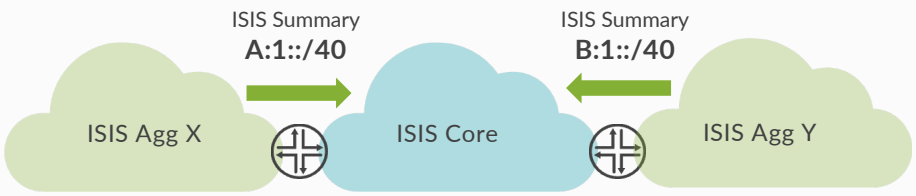
Enterprise-L3VPN : CSR (SRv6) <-> PE-ENC (SRv6)

Enterprise-L3VPN : CSR (SRv6) <-> CSR (MPLS)

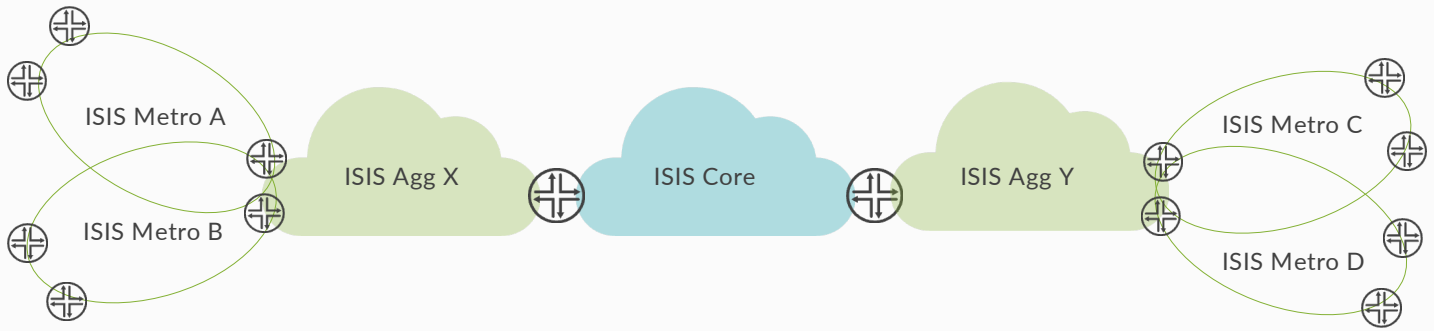


Key Architecture Enablers

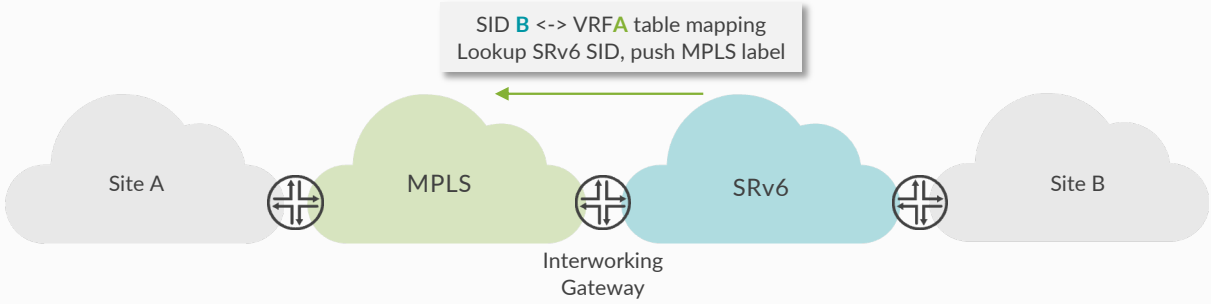
Locator Summarization



Multi-instance ISIS

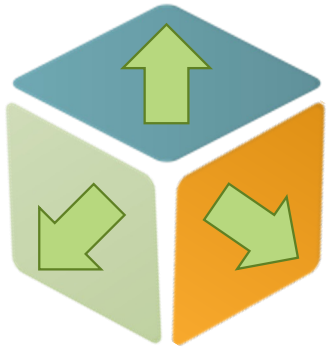


SRv6-MPLS Interworking



Multi Vendor SRv6 IOT

Key *"considerations"* for a successful multi-vendor SRv6 deployment



3 "P" Approach

1 *Locator Addressing*

- Plan Addressing based on GUA or ULA – Design Pre-requisite
  - IPv6 Infrastructure (GUA) and SRv6 Locator (ULA) as a potential option
- SRv6 Locator Block Length – Plan ahead based on the vendor mix
  - Different Vendors might have a preference based on their compression solution
  - Next-C-SID prefers 16, 32 and **48**
  - Replace-C-SID prefers 48, 56, **64**, 72 and 80

2 *Services*

- Global Internet, L3VPN and EVPN services over SRv6 transport (SRv6 BGP based Overlay services RFC9252)
  - SRv6 SID Structure sub-sub-TLV
    - Plan, verify and validate the use of sub-sub TLV for L3 & L2 services
- SRv6 OAM – Plan for Standardized OAM toolkit which'll work across vendors

3 *Interworking with MPLS*

- Plan for Seamless Interworking with existing MPLS Transport & Services
  - Co-Existence with MPLS or the use of single IPv6 data plane
  - SRv6 <> MPLS L3VPN Interworking Gateway functionality
  - SRv6 <> MPLS Interdomain Solution

Challenges deploying SRv6 Design in a Multi Vendor Environment

XL Axiata has deployed Multi-Vendor SRv6 Network across Huawei, Cisco, ZTE & Juniper platforms

Lack of SRv6 Skills

Specialized Skills compared to existing MPLS skills

Different terminology vs MPLS

Distinct best practices, different than MPLS

… per vendor

Complex Brownfield Deployment & IOT

Long Design Validation cycle for Multi Vendor IOT

Hybrid Model, MPLS in some sites and SRv6 at new sites

Business Applications across hybrid ecosystem

… prevents optimal service delivery

Inconsistent Ops during Co-existence

Multiple data plane (MPLS & SRv6)

Troubleshooting overhead in Hybrid state

Siloed visibility in some cases due to absence of consistent SRv6 OAM

… hinder IT operations performance

Platform Limitations Capabilities

Different capabilities across different vendors (SRv6 compression)

Some vendors were not IETF compliant

Software Roadmap variations across different platforms & vendors

… limits problem-solving

Controlling costs

Extended deployment schedule due to feature gaps

Multiple cost reports per vendor/region

TTM for new products constrained due to SRv6 Migration

Additional Opex

… leads to unpredictable costs

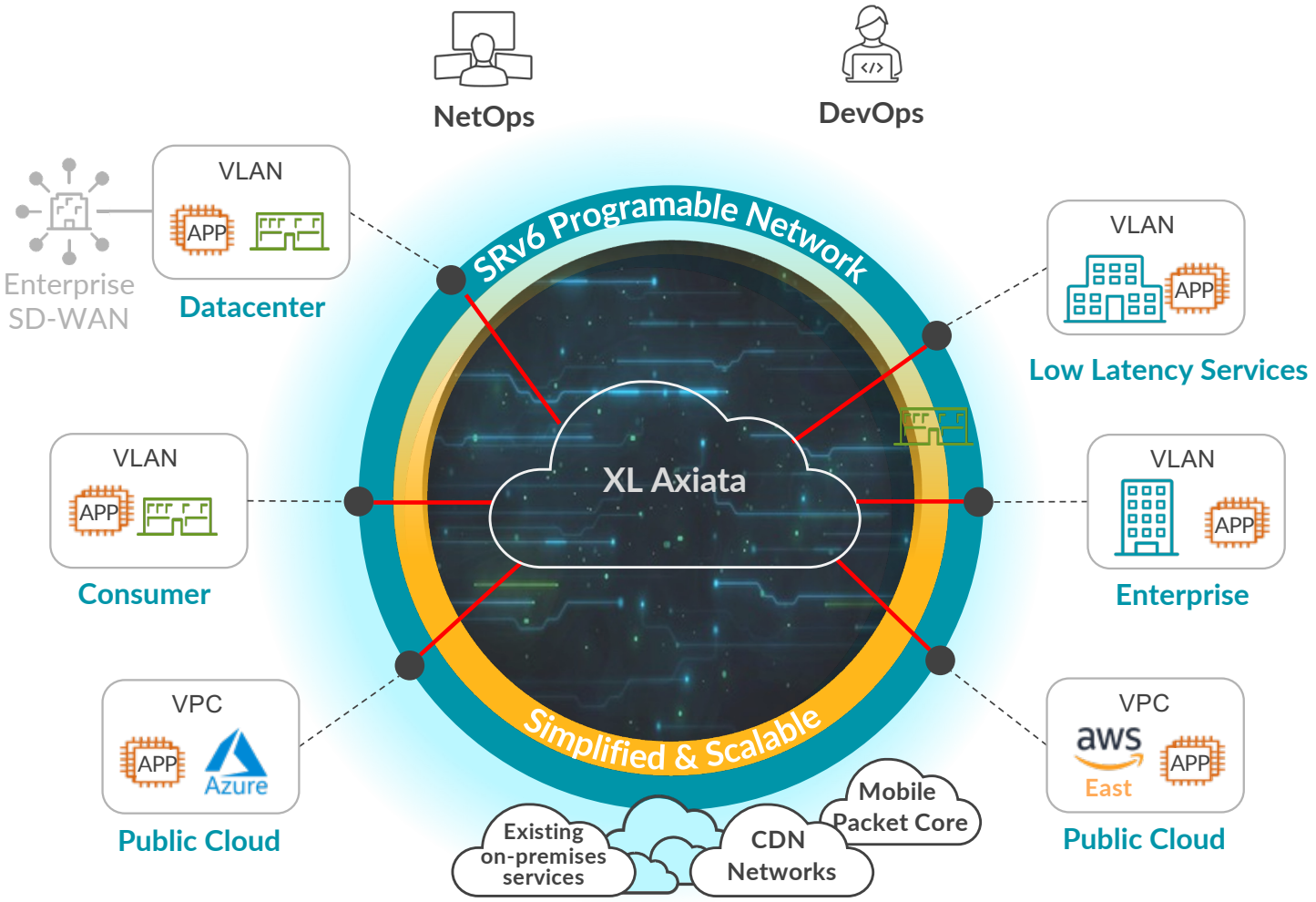
XL Axiata SRv6 Network....*the good* after the 3-year journey

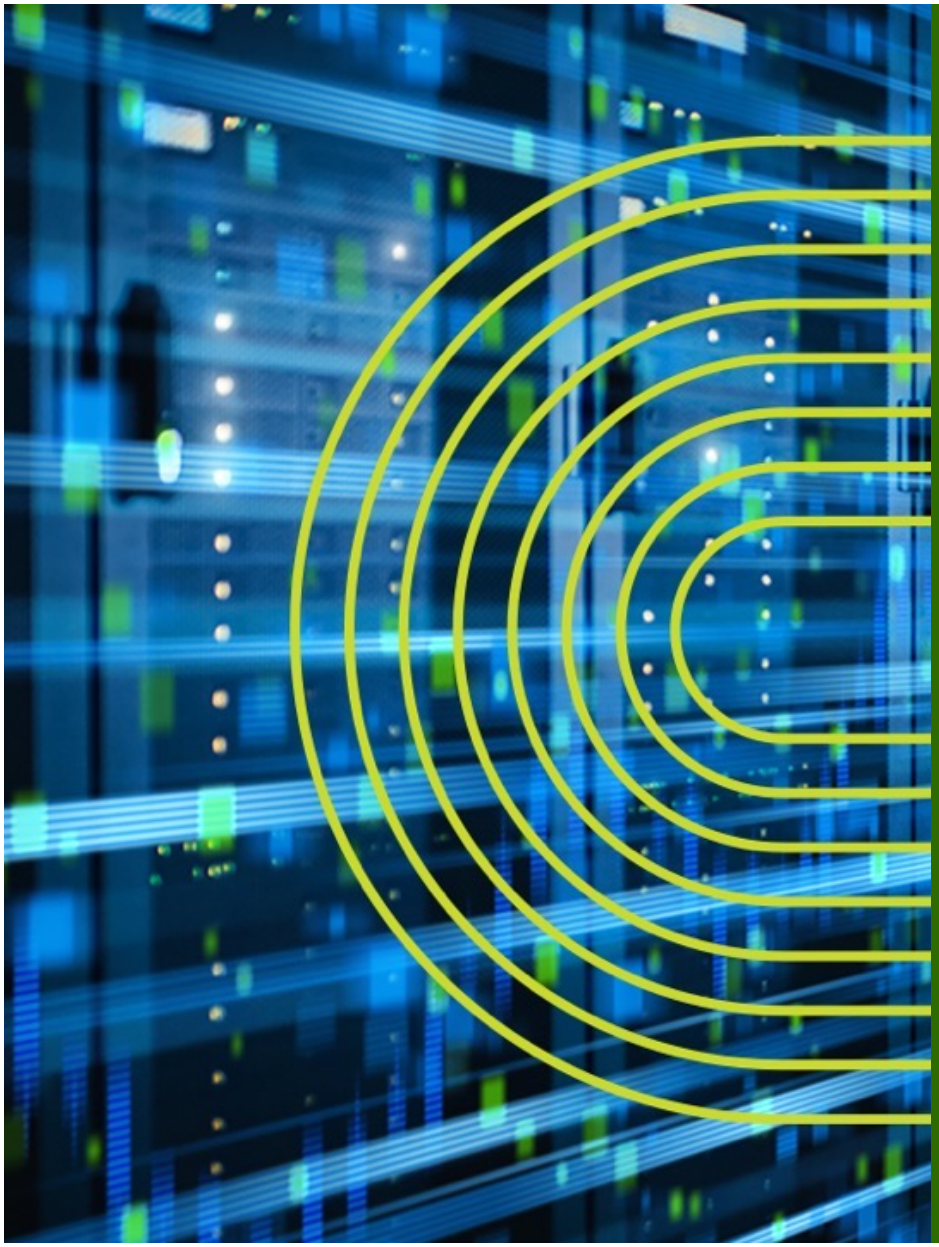
Massive Scale & Flexibility

Cost Optimization

Programmability Close Loop Automation

Any Service Anywhere





SRV6 DEPLOYMENT 2024

- SRv6 Compression
- SRv6 Flex-Algo
- SRv6 EVPN

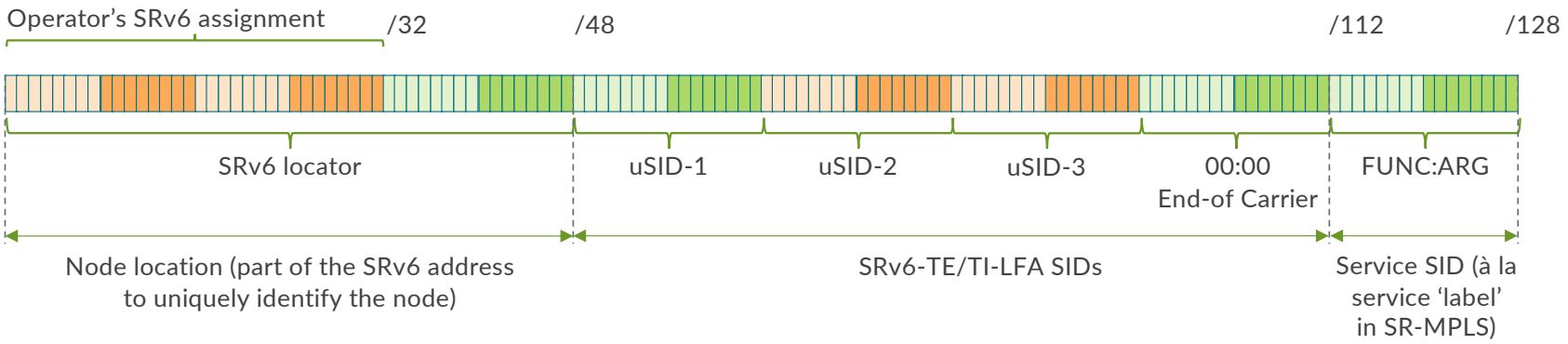
JUNIPER NETWORKS

SRv6 Deployment Goals 2024

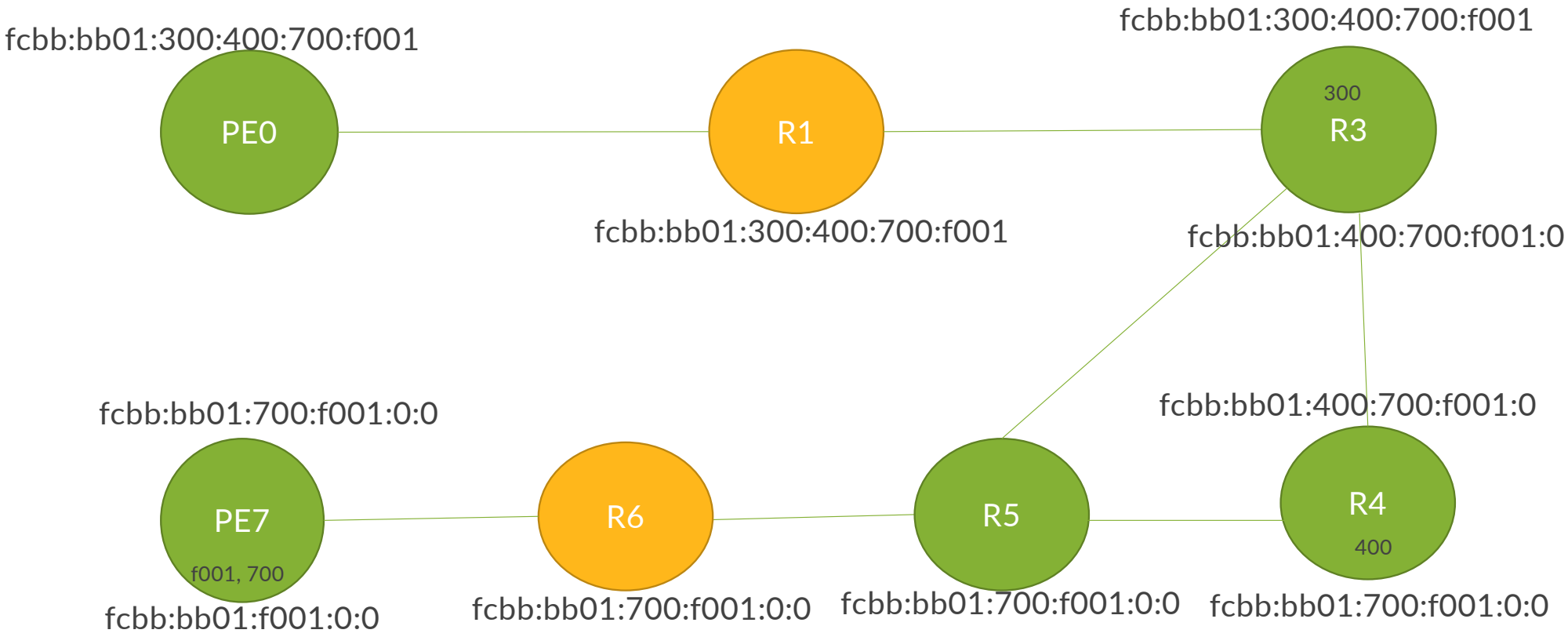
SRV6 Deployment	• SRv6 Compression – NEXT-CSID based Services (DT46/DX2/DT2) • SRv6 <> MPLS Interworking Option B • Flex-Algo Deployment with Delay "Metric" using Twamp-Lite ***(Multi-Vendor-Juniper, Cisco, ZTE & Huawei)***
Automated Service Prioritization	• Ensure customer SLAs are delivered • Link Delay Measurement and Delay Based Routing • Granular Service Prioritization.
Smarter Bandwidth Utilization	• Utilize inter-region links more efficiently • Allow all traffic to use optimal paths; re-grade based on Link utilization and congestion. • Intelligently congest inter-region links and maintain SLAs.
Services	• END.DT46 – L3VPN, Global • END.DX2 – EVPN-VPWS • ***END. DT2 – EVPN-ELAN (Multi-Vendor-Juniper, Cisco, ZTE & Huawei)***

NEXT-CSID - uSID

- SRH requirement for TI-LFA / TE will be minimized (potentially even removed for typical TI-LFA applications, or less advanced TE) with uSID (NEXT-CSID), where multiple SIDs can be embedded in single 128-bit address field. https://www.ietf.org/archive/id/draft-ietf-spring-srv6-srh-compression-09.txt
  - Example: /32 allocated to the operator, 100k nodes, 5 slices (flex-algo), 5 Regions
    - Per region SRv6 locators → 16 bits (2^{16} = 65,536) from address space enough for 100k locators → SRv6 locator is /48 (32+16)
    - 128 – 48 = 80 bits left for uSID and FUNC:ARG
    - 16-bit long uSID → 16 bits left for FUNC:ARG

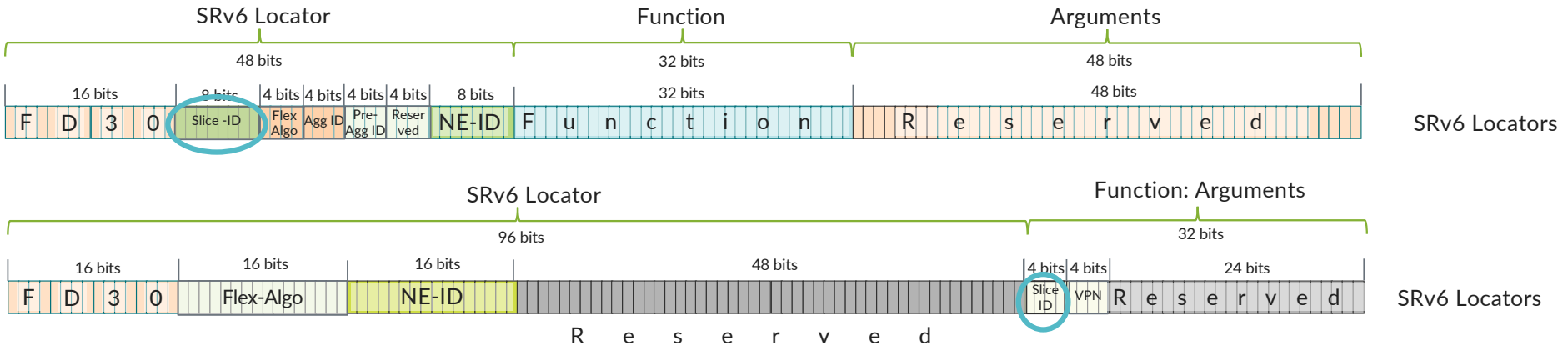


NEXT-CSID Forwarding Overview



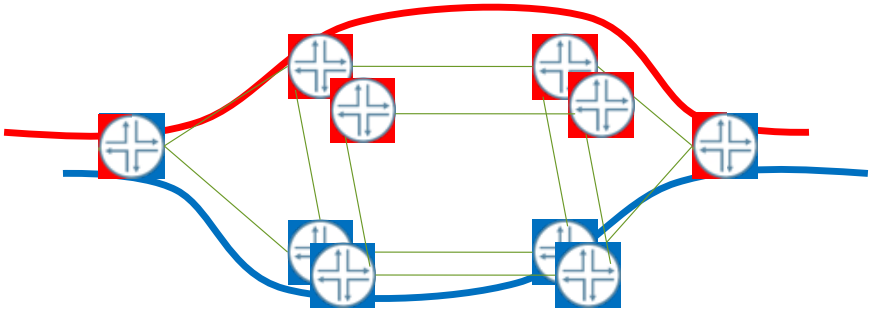
SRv6 Network Slicing
Slice ID in the SID

- SRv6 Locator, or SRv6 SID (END.DT4, END.DT6, END.DT46) can have dedicated space (few bits) to carry Slice ID (see below some examples)
- Transit router classifies packets to appropriate slice based on this slice ID and puts the packet to the slice on the link, where each slice on the link has capacity guarantees.

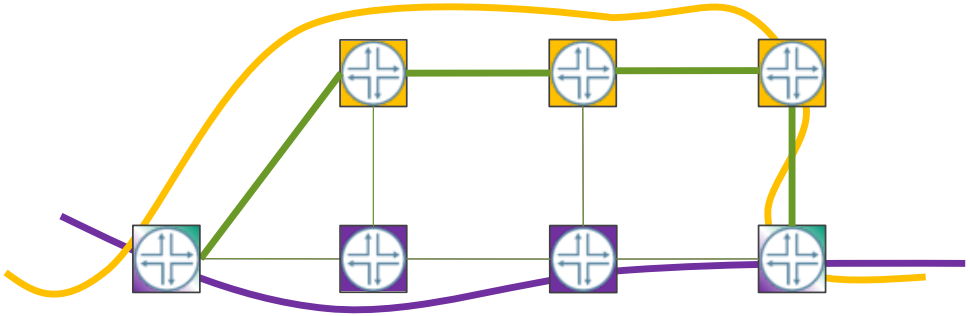


Flex-Algo: Low Latency / Service Diversity

- Separate Routing Planes with Geographical or Service Diversity Constrains

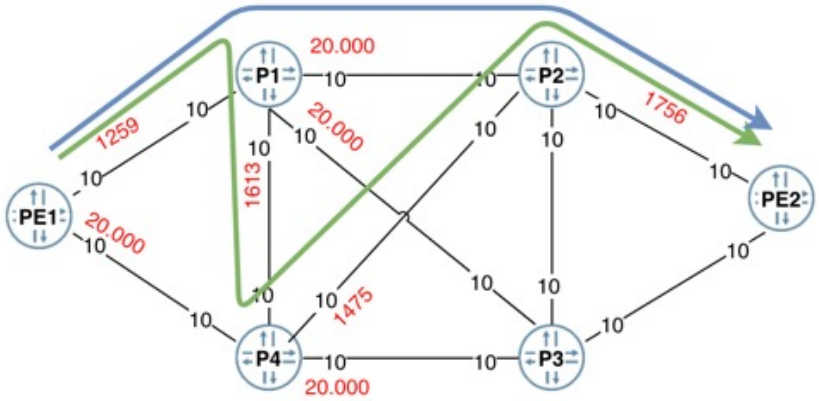


- Low Latency using Link Delay Metric (ISIS Twamp-Lite)



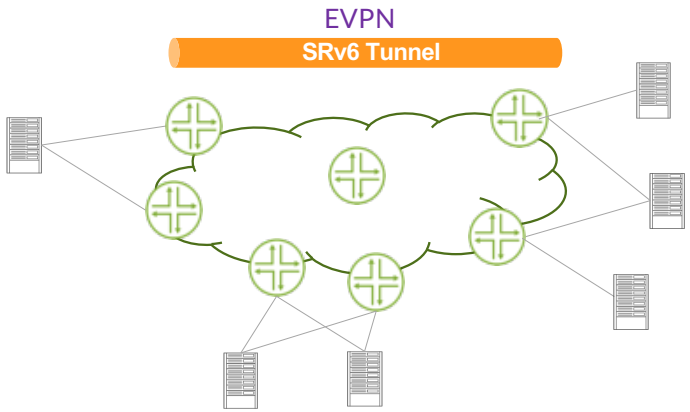
Using Flex-Algo with Delay-Metric in Action

- PE1 chooses least **metric** blue path to PE2 by default
- PE1 chooses least **delay-metric** green path to PE2 in Algo 128
- Default metric is 10
- Some links use static delay-metric of 20ms
- Other links use TWAMP light dynamic delay-metric calculation
- All devices participate in SRv6 Algo 128



EVPN over SRv6 with SRv6 SRH / uSID

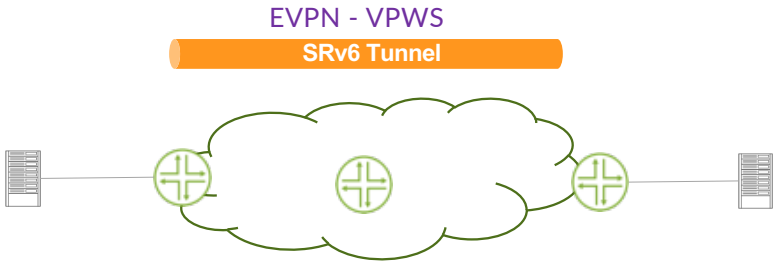
EVPN



EVPN service covers:

End.DT4/DT6/DT46 – Layer 3 table lookup
End.DX2U - Layer 2 unicast cast
End.DT2M – Layer 2 flooding

EVPN VPWS



EVPN VPWS service covers:

End.DX2 - Layer 2 cross connect

Recommended Reading for Operations Skill Enablement

- RFC 8754: **IPv6 Segment Routing Header (SRH)**
- RFC 8986: Segment Routing over IPv6 (SRv6) Network Programming
- Day One SRv6 Book : https://www.juniper.net/documentation/en_US/day-one-books/DayOne-Intro-SRv6.pdf
- SRv6 in Junos: https://www.juniper.net/documentation/us/en/software/junos/is-is/topics/topic-map/infocus-isis-srv6-network-programming.html
- TechPost 1: SRv6 Basics Locator and End-SIDs - https://community.juniper.net/blogs/krzysztof-szarkowicz/2022/06/29/srv6-basics-locator-and-end-sids
- TechPost 2: L3VPN on SRv6 - https://community.juniper.net/blogs/krzysztof-szarkowicz/2022/08/11/l3vpn-over-srv6
- TechPost 3: SRv6 Summarisation - https://community.juniper.net/blogs/krzysztof-szarkowicz/2022/09/28/srv6-summarization
- TechPost 4: SRv6 SID Encoding and Transposition - https://community.juniper.net/blogs/krzysztof-szarkowicz/2022/12/02/srv6-sid-encoding-and-transposition
- TechPost 5: SRv6 Inter Domain Services Option-C - https://community.juniper.net/blogs/krzysztof-szarkowicz/2023/02/06/srv6-l3vpn-inter-as-option-c

THANK YOU

JUNIPER NETWORKS | Driven by Experience™

SRv6 Standardization at IETF

- SRv6 Architecture – RFC 8402
- SRv6 Dataplane – RFC 8754
- SRv6 Network Programming – RFC 8986
- SRv6 ISIS Extensions - draft-ietf-lsr-isis-srv6-extensions→ RFC9352
- SRv6 BGP Extensions - draft-ietf-bess-srv6-services→ RFC9252
- SRv6 Policy - draft-ietf-spring-segment-routing-policy→ RFC9256
- SRv6 OAM - draft-ietf-6man-spring-srv6-oam→ RFC9259

"SHIPS IN THE NIGHT"

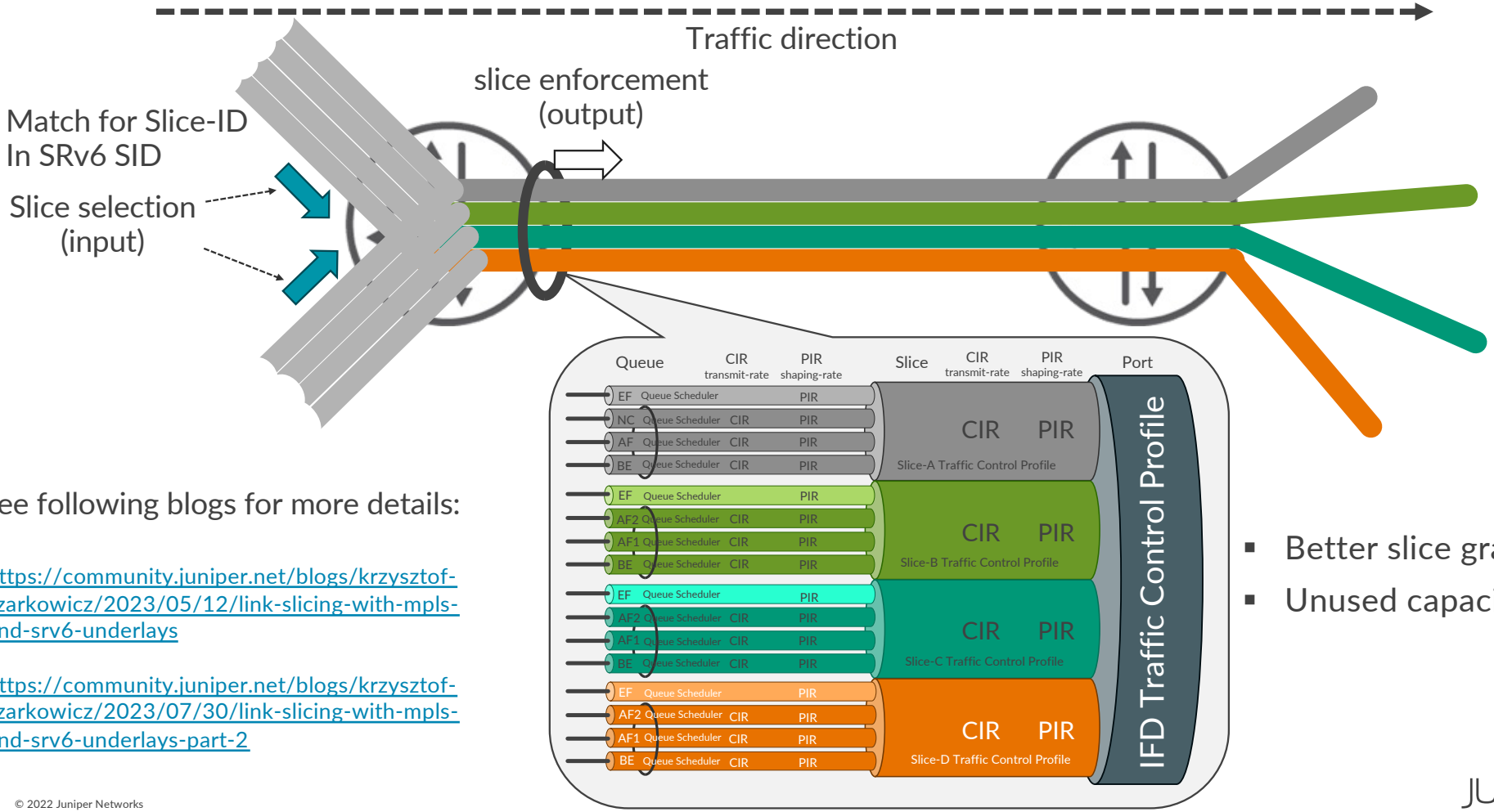
SRv6 can co-exist with IPv4 based RSVP, LDP and BGP-LU

- Protocol Preference RSVP → LDP →OSPFv2 → SRv6-ISIS
- MPLS and SRv6 co-existence for minimal disruption – IPv4/IPv6 Loopback and SRv6 Locator.
  - Continue using same IPv4 Loopback as concurrent active LDP/RSVP in classic MPLS without any disruption to services.
  - Flexibility in having a Service specific data plane- MPLS or SRv6 based on BGP NH (IPv4/IPv6/locator) at the ingress of the service.
  - SRv6 Locator as BGP END.DT4/6/DX would be required for new or reparenting existing MPLS services to run over SRv6 data plane.
  - Service – Service Mapping using Color for both SR/SRv6-TE or Flex-Algo constraints- Policy – color:0:129 (example Low Latency)
- Easy migration – Co-Existence with new SRv6 RAN rings and Core MPLSv4
  - Enable dual stack IPv6 in the Core network on new and existing Nodes – Node and Links
  - Multi-Instance ISIS – RAN<AGG>Core at AGG
  - Integrate New SRv6 RAN rings with AGG as ABR using SRv6 Migration Tools – SRv6 <> MPLS Interworking GW
  - SRv6 RAN<AG>PS-PS will ensure simplicity, scale and prevent leaking between instances, also
- End State
  - Enable SRv6 in IPBB (AG/CAG/PS) using L2-ISIS (Node/Adj/Anycast SID)
  - Deploy a combination of SRv6-TE and Flex-Algo to provide traffic engineering and Network Slicing Services like *"low latency"*
  - Assuming SRv6 deployment in the Core network is completed, remove MPLS (LDP/RSVP) from the Core.
  - Continue with MPLS<>SRv6 Interworking assuming some MPLS CSRs will continue to exist.
    - Integrate Old MPLS Nodes in the RAN using SRv6 Migration Tools- MPLS<>SRv6 Interworking GW

Guaranteed Link – Network Slicing

Link Channelization/Partitioning with SLA

- Match for Slice-ID In SRv6 SID



See following blogs for more details:

https://community.juniper.net/blogs/krzysztof-szarkowicz/2023/05/12/link-slicing-with-mpls-and-srv6-underlays

https://community.juniper.net/blogs/krzysztof-szarkowicz/2023/07/30/link-slicing-with-mpls-and-srv6-underlays-part-2

- Better slice granularity
- Unused capacity sharing