

SRv6 uSID 6 Hops of Steering: The Sweet Spot & Migration Playbook

Skip SR-MPLS Entirely — The DA-Only, No-SRH
Path From Legacy MPLS Straight to SRv6 uSID

PARADIGM SHIFT

MPLS / SR-MPLS / SR-MPLSv6
skipped entirely

MIGRATION STRATEGY

Phase 1: PE-only uSID overlay
Phase 2: P-node hardware refresh

HARDWARE IMPACT

P-nodes stay vanilla IPv6 —
zero forklift for transit

The 6-Hop Sweet Spot: Why the uSID Carrier Changes Everything

SRv6 micro-segment (uSID) compression — standardized in RFC 9800 (Compressed SRv6 Segment List Encoding, NEXT-CSID flavor) — encodes up to six 16-bit micro-segments into a single 128-bit IPv6 Destination Address (DA), using the uSID-F3216 format: a 32-bit locator block followed by six 16-bit uSID containers. When the entire steering path fits within those six entries and the overlay/underlay merge is disabled (the default on most platforms), no Segment Routing Header (SRH) is required at all. The packet looks, to every transit node on the path, like a completely ordinary IPv6 unicast packet — routed purely on Longest Prefix Match (LPM) against the DA.

This single architectural fact is the foundation of the entire migration strategy: transit (P) nodes do not need to understand SRv6, do not need to understand uSID, and do not need any hardware or software upgrade. They forward the packet blindly based on a standard IPv6 FIB lookup. Only the nodes that originate or terminate uSID instructions — the SRv6 Endpoint nodes performing the Shift-and-Forward (NEXT-CSID) DA manipulation — require a programmable pipeline capable of masking, shifting, and rewriting the DA in a single pass at line rate.

“The 128-bit Forklift is a Myth. The uSID Blind-Spot is the Key.”

Transport SIDs vs. Service SIDs: H.Insert.Red Decoupling

With overlay/underlay merge disabled — the vendor default — all six uSID entries in the carrier are available for transport steering (the underlay path across the core). Service identification (the overlay — L2VPN/L3VPN endpoint behaviors such as End.DT4/DT6, End.DX2, VPN function/argument) is decoupled into a separate reduced encapsulation, typically H.Insert.Red or H.Encaps.Red. This separation means the full 6-hop transport budget is never consumed by service-layer SID bits — transport steering and service identification optimize independently, each using its full available space.

The Compression Gain: Classic SRv6 (SRH) vs. SRv6 uSID (DA-Only)

DIMENSION	CLASSIC SRv6 (128-bit SID + SRH)	SRv6 uSID (≤6 hops, DA only, no SRH)
Transit (P) node requirement	Must parse SRH extension header; deep parsing causes recirculation on older ASICs, cutting throughput 50–66%	Zero — packet is indistinguishable from standard IPv6 unicast; pure LPM forwarding at full line rate
Header overhead (6-hop path)	128-bit DA + SRH with 6 × 128-bit segments ≈96+ bytes	Single 128-bit DA carries all 6 uSIDs — 16 bytes total: 83% smaller / 6x vs. classic SRv6
Hardware upgrade boundary	Forklift — every node in the steered path needs deep-parser programmable silicon	Endpoint-only — P nodes stay on any silicon that can do standard IPv6 LPM (effectively all silicon, any generation)
MTU impact	SRH grows linearly with hop count — MTU planning required across the entire path	Fixed 40-byte IPv6 header regardless of hop count (≤6 hops) — no SRH growth
TI-LFA / SR-TE steering through P	Available if P nodes upgraded	Not available until P nodes are uSID-aware (Phase 2) — best-effort IGP shortest path until then

Compression by Hop Count: Why the Advantage Scales

uSID’s compression advantage is not flat — it scales directly with the number of hops being steered, and it is strongest at the 6-hop ceiling this playbook is built around. At 1 hop there is no advantage at all: a single classic 128-bit SID and a uSID DA both carry 16 bytes. The gain compounds as hops increase, since classic SRv6 pays 16 bytes per additional segment while uSID stays flat at 16 bytes total, regardless of how many of the 6 available micro-segments are in use.

Hops	Classic SRv6 (16B × N)	uSID (Flat 16B DA)	Bytes Saved	% Reduction
1	16 B	16 B	0 B	0%
2	32 B	16 B	16 B	50.0%
3	48 B	16 B	32 B	66.7%
4	64 B	16 B	48 B	75.0%
5	80 B	16 B	64 B	80.0%
6	96 B	16 B	80 B	83.3%

Classic SRv6 overhead reflects segment-list bytes only (16 bytes per 128-bit SID); adding the fixed 8-byte SRH base header brings the 6-hop classic total to 96+ bytes, matching the header-overhead figure above. uSID’s 16-byte DA is unaffected by hop count up to the 6-hop, no-SRH ceiling.

The takeaway: uSID’s compression argument should be made at realistic steering depths, not as a flat average. Multi-hop steering — traffic engineering, FRR, multi-domain paths — is where the 6-hop design delivers its full 83% / 6x benefit.

The Migration Playbook: Phase 1 (PE-Only) → Phase 2 (Core Refresh)

Phase 1 — “Blind Transit” Island (PE Upgrade Only)

Enable SRv6 uSID exclusively on PE (Provider Edge) nodes. The uSID-encoded packet's DA looks like a standard IPv6 address to every legacy P (core) node — they perform ordinary LPM and forward it without modification. ISIS/OSPF in the core simply advertises IPv6 reachability to PE loopback/locator prefixes; legacy P nodes forward uSID-encapsulated traffic blindly alongside all other IPv6 traffic.

- L2VPN (EVPN-VPWS/VPLS) and L3VPN services work 100% immediately — PEs handle service encaps/decap; the legacy core provides best-effort underlay reachability
- No explicit waypoint steering through P nodes — legacy P cannot perform Shift-and-Forward, so traffic follows IGP shortest path (best-effort) between PEs
- TI-LFA is unavailable until P nodes are uSID-capable — legacy P falls back to standard IP-FRR/LFA, which does not guarantee 100% link protection coverage
- RSVP-TE/SR-MPLS can remain active in parallel for traffic that requires strict steering (voice, critical services), while bulk/Internet/VPN traffic rides the new SRv6 best-effort plane

Phase 2 — Core (P-Node) Hardware Refresh

Once P nodes are refreshed to uSID-capable silicon and minimum software, enable SRv6 uSID on the core. TI-LFA and explicit SR-TE waypoint steering become available end-to-end, and the legacy MPLS/RSVP-TE plane can be fully decommissioned.

- P nodes become full uSID Endpoints — capable of Shift-and-Forward on the DA for TI-LFA backup path insertion and SR Policy explicit waypoints
- Decommission RSVP-TE / LDP / SR-MPLS entirely — single unified SRv6 uSID data plane end-to-end
- Result: 100% service enablement achieved in Phase 1 with a fraction of the hardware refreshed; Phase 2 hardware spend is targeted only at core nodes that require steering/FRR — not the entire fabric

***Phase 1 unlocks 100% of L2VPN/L3VPN service capability with zero core hardware changes.
Phase 2 is optional and can be scheduled on normal refresh cycles.***

Hardware Breakpoint Matrix — Incumbent Vendors

Lab-validated and vendor-documented breakpoints for SRv6 uSID (F3216, ≤ 6 hops, no SRH) Endpoint (PE) capability. Platforms below the breakpoint can still serve as Phase 1 “blind transit” P nodes routing plain IPv6.

VENDOR	PLATFORM / SERIES	PE / ENDPOINT — uSID CAPABLE	BLIND-TRANSIT (P) ONLY — NO UPGRADE NEEDED
Cisco	8000 Series (Silicon One Q100/Q200/P100/P200)	Native — built for NEXT-CSID from day one	—
Cisco	NCS 5500 / 5700 (Jericho 1 & 2 / NC57)	Validated — all flavors with IOS XR 7.3.2+, f3216 profile	Pre-7.3.2 / non-NC57 cards
Cisco	ASR 9000 / 9900	Tomahawk (Gen 3, e.g. A9K-MOD400) validated via IOS XR 7.3.2+; Lightspeed/Lightspeed-Plus native	Typhoon (Gen 2) and older — Gen 2 cannot do DA shift
Juniper	MX Series	Trio 5 (Penta, MPC10E) / Trio 6 (MPC11E, LC9600) — Junos 23.4R1+	Trio 1–4 (MPC2E/3E/7/8/9) — IPv6 transit only
Juniper	PTX Series	PTX10000 (Express 4/5) — all flavors validated	PTX3000/5000 (Express 1/2) — transit only
Juniper	ACX Series	ACX 7000 family (Jericho 2/2c/3 or Trident 4) — all flavors validated	Older Metro ACX
Nokia	7750 SR / 7950 XRS	FP4 / FP5 — SR OS 22.5.R1+, native End.uN/End.uA	FP3 — solid IPv6 transit, no uSID endpoint
Nokia	7250 IXR	All models validated — SR OS 22.5.R1+ code upgrade only, no hardware gap across the family	—
Arista	7280R3 / 7500R3 / 7800R3	Jericho 2 — EOS 4.28.0F+	7280R Gen-1 (Jericho 1) / 7050X (Trident 2) — transit only
Huawei	NetEngine 8000 (NE8000)	Solar-series chipsets — native, V800R021C00+ for compressed profile	Older NE40E / NE20E — standard IPv6 transit
Ciena	5164 / 5166 / 81xx	Modern Broadcom DNX/Qumran-class silicon — “SRv6 Ready”	Older 39xx/51xx (L2/MPLS-centric)

Whitebox Switch Vendors — Model, Chipset & Open NOS Pairing

Disaggregated whitebox platforms running Open NOS (SONiC, Arrcus ArcOS, IP Infusion OcNOS) via SAI inherit their uSID capability directly from the underlying Broadcom silicon generation. The table below maps the leading Edgecore-equivalent whitebox vendors and their flagship SRv6 uSID-capable models to the recommended Open NOS pairing.

VENDOR	MODEL / SERIES	BROADCOM CHIPSET	RECOMMENDED OPEN NOS FOR uSID	ROLE / TARGET USE CASE
UfiSpace	S9100 / S9110	Qumran 2c	IP Infusion OcNOS or Arrcus ArcOS	Carrier Edge / Cell Site Gateway
UfiSpace	S9300-32D	Trident 4	IP Infusion OcNOS	Aggregation / Peering
UfiSpace	S9321-64E	Tomahawk 5	Arrcus ArcOS or SONiC	AI Backend Fabric — 6-hop DA-only
Edgecore	AS7926-40XK	Trident 4	SONiC or IP Infusion OcNOS	Enterprise / DC Aggregation
Edgecore	AS9716-32D	Tomahawk 4	SONiC	Standard DC Fabric Spine/Leaf
Quanta Cloud Technology (QCT)	QuantaMesh T9032-IX9	Tomahawk 5	SONiC 202411+ (hyperscale-deployed) or Arrcus	AI Backend Fabric
Celestica	Seastone DX010	Tomahawk 4 / 5	SONiC or IP Infusion	Hyperscale Reference DC Fabric
Delta (Agema)	AG9064	Trident 4 / Tomahawk	SONiC, IP Infusion OcNOS, or Arrcus	Full XGS/DNX whitebox range
Ragile Networks	RA-B6920-64 C	Tomahawk 5	SONiC (custom SAI for AI)	AI/HPC GPU-to-GPU Fabric

Broadcom Silicon Timeline — XGS & DNX uSID Capability

XGS Family (Data Center Fabric / AI Backends — Trident & Tomahawk)

CHIPSET	STATUS	SRv6 uSID CAPABILITY	NOTES
Trident 2 / 2+	EOL	None	Legacy VXLAN era
Tomahawk 1 / 2	EOL	None	Fixed pipeline — no SRv6 capability
Trident 3 (TD3)	Legacy	Limited / Software	Programmable pipeline allows basic SRv6, but poor performance; no native uSID
Tomahawk 3 (TH3)	Legacy	None	Optimized for bandwidth; lacks programmable parser for SRv6 headers
Trident 4 (TD4) / Trident 5 (TD5)	Modern	Full Native uSID	Fully programmable (NPL) — complex uSID behaviors and deep SRH stacks
Tomahawk 4 (TH4)	Modern	Native (DA-Only)	Supports uSID in DA (6 hops); SRH extension headers may cause recirculation
Tomahawk 5 (TH5)	AI Native	Hardware Optimized	Gold standard for AI backends — line-rate 6-hop uSID, zero latency penalty

DNX Family (Core Router / WAN / Carrier — Jericho & Qumran)

CHIPSET	STATUS	SRv6 uSID CAPABILITY	NOTES
Qumran MX / AX	EOL	None	Jericho 1 generation — basic MPLS only
Jericho 1 (J1)	EOL	None	Does not support SRv6 parsing at scale
Jericho 2 (J2)	Mature	Native uSID	Industry workhorse — full SRv6/uSID, TE, and VPNs
Qumran 2c (Q2c)	Mature	Native uSID	Cost-optimized J2 variant — ideal for Cell Site Gateways (CSG)
Jericho 2c+ (J2C+)	High Performance	Advanced uSID	Enhanced logic for larger tables and deeper label stacks
Jericho 3 (J3)	AI / Core	AI Optimized	Massive buffers + native uSID — perfect load balancing for AI clusters

Minimum Open NOS / Software Versions for SRv6 uSID

Each Open NOS platform reached production-grade SRv6 uSID support at a specific release. The table below lists the minimum version per NOS, paired against the Broadcom silicon family it targets.

NETWORK OPERATING SYSTEM	MINIMUM VERSION FOR uSID	SILICON TARGET	KEY FEATURES & SUPPORT LEVEL
Enterprise SONiC	202305 (basic) / 202411 “PhoenixWing” (AI stable)	Trident 4/5, Tomahawk 4/5, Jericho 2/2c/3	202111 introduced experimental SRv6; 202305 added SAI support for uSID VPNs; 202411 is the standard for AI/backend fabrics with 6-hop DA-only support
Arrcus ArcOS	v5.0.1	Jericho 2/2c+, Trident 4, Tomahawk 5	Industry leader — first to support full uSID. Shift-and-Forward and deep SRH parsing. Native 5G slicing and TE policies
IP Infusion OcNOS	v6.3.0 (SP Profile)	Broadcom DNX (Qumran/Jericho), Trident 4	OcNOS-SP 6.3 introduced production uSID; optimized for DNX platforms; full interop with Cisco/Nokia SRv6 domains
NVIDIA Cumulus Linux	5.x (Spectrum-2/3/4 only)	NVIDIA Spectrum ASICs — Broadcom support frozen at 4.3.x	Note: Post-acquisition, Cumulus Linux 5.x dropped third-party Broadcom whitebox support — no longer a general-purpose whitebox NOS option for Broadcom-based uSID deployments

The Paradigm Shift: Skip SR-MPLS Entirely

Because the uSID 6-hop sweet spot eliminates the historical objection to SRv6 — the forklift hardware upgrade for transit nodes — the traditional intermediate step of migrating to SR-MPLS (or SR-MPLSv6) as a stepping stone toward SRv6 is no longer necessary for most networks. Operators can move directly from legacy LDP/RSVP-TE MPLS to SRv6 uSID, in a single coordinated PE-first rollout.

DIMENSION	LEGACY MPLS (LDP/RSVP-TE)	SR-MPLS / SR-MPLSv6	SRv6 uSID (≤6 hops, no SRH)
Control plane	LDP + RSVP-TE — separate label distribution and TE signaling protocols	IGP (ISIS/OSPF) extensions carry SR labels — eliminates LDP/RSVP-TE	IGP (ISIS/OSPF) advertises IPv6 locators — single unified IPv6 control plane
Data plane	MPLS label stack — 4 bytes/label, no native IPv6 integration	MPLS label stack (v4) or MPLS-in-IPv6 (v6) — still a label stack	Native IPv6 DA — no label stack, no separate data plane at all
Hardware requirement	Any MPLS-capable router — universally supported	Same MPLS-capable silicon — software-only upgrade from LDP/RSVP-TE	P nodes: any IPv6-capable silicon (zero upgrade). PE nodes: uSID-capable programmable pipeline (Jericho2+/Trio5+/FP4+/TH3+)
Network slicing / 5G	Not natively supported — requires separate VPN overlays	Improved via SR-TE policies, but still MPLS-label based	Native per-uSID network slicing — IPv6-native, cloud/Kubernetes integration via eBPF/Cilium
Migration as stepping stone	Starting point	Historically positioned as the “safe” intermediate step toward SRv6	With the 6-hop uSID sweet spot, this intermediate step can be skipped entirely — go straight from legacy MPLS to SRv6 uSID
End-state complexity	Two protocols (LDP + RSVP-TE) + MPLS data plane	One control-plane mechanism (SR) but still on MPLS data plane — an interim architecture, not an end-state	One data plane (IPv6), one control plane (IGP + SRv6), zero MPLS — true architectural end-state

MPLS → SR-MPLS → SRv6 was the textbook path. The 6-hop uSID sweet spot makes MPLS → SRv6 uSID the new direct path — fewer migrations, fewer protocols, one architectural end-state.

Important Caveats

- The 6-hop limit is a hard ceiling for the SRH-free path — any steering requirement beyond 6 segments (transport + service combined, when merge is enabled) requires SRH and re-introduces the deep-parsing hardware requirement
- TI-LFA and explicit SR-TE waypoint steering through the core require Phase 2 (P-node) uSID capability — Phase 1 alone provides best-effort underlay only
- Vendor “uSID support” claims should be validated against the specific line card / NPU generation, not just the chassis family — e.g. ASR 9000 spans Typhoon (gap), Tomahawk (validated via software), and Lightspeed (native)
- Overlay/underlay merge being disabled (the default) is what frees all 6 uSID entries for transport — if merge is enabled, service SIDs consume entries from the same 6-slot budget, reducing effective transport hop count

→ Cumulus Linux post-NVIDIA acquisition is no longer a general-purpose Broadcom whitebox NOS — 5.x targets Spectrum ASICs only; Broadcom support is frozen at the 4.3.x branch

Beyond F3216: uSID Carrier Format Flexibility

Every hardware breakpoint and hop-count figure in this playbook assumes the F3216 carrier format — a 32-bit uSID block plus six 16-bit uSID IDs — because F3216 is the default, and per Cisco's own IOS XR documentation it was the only format supported in the original 7.3.1 release. F3216 was a starting point, not a permanent ceiling.

Cisco's current ASR 9000 SRv6 configuration guide defines the carrier format using the notation "FbbUU" — bb is the uSID block size in bits, UU is the uSID ID size in bits — and confirms most current platforms now support far more than the original default. Block and ID sizes are expected to land on byte boundaries (16, 24, 32... for the block; 8, 16, 24, 32... for the ID) for hardware-friendly parsing on NPU and merchant-silicon (Broadcom) pipelines, but within that constraint, operators can choose the split that fits their network.

Juniper and Nokia take a different notational path to the same idea: instead of a single FbbUU code, they specify the split explicitly as Block:Node (B:N) — the underlying tradeoff is identical. A larger block leaves fewer bits for the ID field, shortening max steering depth; a smaller block (or a larger private space, e.g. ULA FC00::/8) leaves more ID bits and lets a single DA steer more hops before an SRH is needed. The carrier format is a design knob — size it to the node count the block must address and the SRH-free hop depth the use case requires, not the default.

Source: Cisco IOS XR SRv6 Configuration Guide, ASR 9000 (25.x) — "Configure SRv6 Micro-SID (uSID)", cisco.com.

Block:ID Permutation Matrix — Max SRH-Free Hops by Format

Format	Block (bits)	ID (bits)	Max Hops
F1608	16	8	14
F1616	16	16	7
F1624	16	24	4
F1632	16	32	3
F2408	24	8	13
F2416	24	16	6
F2424	24	24	4
F2432	24	32	3
F3208	32	8	12
F3216 *	32	16	6
F3224	32	24	4
F3232	32	32	3
F4008	40	8	11
F4016	40	16	5

Format	Block (bits)	ID (bits)	Max Hops
F4024	40	24	3
F4032	40	32	2
F4808	48	8	10
F4816	48	16	5
F4824	48	24	3
F4832	48	32	2
F5608	56	8	9
F5616	56	16	4
F5624	56	24	3
F5632	56	32	2
F6408	64	8	8
F6416	64	16	4
F6424	64	24	2
F6432	64	32	2

* F3216 = Cisco default. Max Hops = floor((128 – Block Size) / ID Size). Byte-aligned block sizes (16–64 bits) and ID sizes (8–32 bits) shown per Cisco's hardware-friendliness guidance for NPU/merchant-silicon register alignment; Juniper/Nokia B:N deployments follow the same formula. Smaller ID sizes trade per-node address space for deeper SRH-free steering — e.g., F1608 (8-bit IDs) reaches 14 hops but only 256 IDs per block, while F3216 balances a larger per-block node count against a practical 6-hop steering ceiling.