

Navigating SRv6 uSID Use Cases with Open NOS & Whitebox Vendors

Lowest Common Denominator Baseline · Full Model Progression · BGP PIC/H-FIB/TI-LFA
C Struct vs. C++ OOP Containers · SAI vs. HAL/PAL · SONiC P4/PINS gRPC for 5G vRAN

LCD BASELINE	CONVERGENCE	NOS ARCHITECTURE
Oldest model = entry floor All newer = fully supported	BGP PIC Core H-FIB TI-LFA <50ms / sub-ms	Pure C vs C++ OOP SAI vs HAL/PAL direct

The Lowest Common Denominator (LCD) Methodology

Each whitebox vendor table starts from the **Lowest Common Denominator (LCD) model** — the oldest or most basic platform in that vendor’s lineup that still runs the Open NOS ecosystem (SONiC, Arrcus, OcNOS). The LCD model defines the **minimum hardware baseline**: the exact Broadcom silicon generation, minimum NOS software version, and which SRv6 uSID behaviors are supported at that floor. Every subsequent model in the vendor’s lineup meets or exceeds this baseline — newer generations progressively add native uSID endpoint capability, higher port density (100G → 400G → 800G → 1.6T), deeper DNX buffers, AI-fabric optimizations, and CPO optical integration. This progression means: **identify your oldest deployed hardware, find its row in the table, and every model above it in the same vendor family is fully supported.**

Whitebox Vendor Model Progression — LCD to Latest

Each table below starts with the oldest/lowest model (the LCD floor) and progresses upward through generations. The gold divider line marks the LCD baseline row. First row = minimum entry point. All subsequent rows = improvements in silicon capability, port density, and SRv6 uSID performance.

Edgecore Networks (Accton)

LCD Note: Lowest common denominator: AS7726-32X (Trident 3) — baseline for SONiC compatibility. All models from AS7726-32X onward inherit full open NOS ecosystem support. The AS7726-32X with Trident 3 marks the starting floor — it runs SONiC and OcNOS but does NOT natively support SRv6 uSID without software workarounds. AS7926-40XK (Trident 4) is the minimum for native uSID PE/endpoint capability.

MODEL	BROADCOM CHIPSET	PORT SPEED	uSID CAPABILITY & NOS NOTES
AS7726-32X	Trident 3 (TD3)	100G	Baseline — no native uSID endpoint; SONiC/OcNOS/Arrcus; transit-only for SRv6 uSID
AS7926-40XK	Trident 4 (TD4)	100G	uSID capable — minimum PE baseline; SONiC 202305+, OcNOS 6.3+, Arrcus 5.0.1+
AS9516-32D	Tomahawk 3 (TH3)	400G	No native uSID endpoint; transit-only; SONiC for DC fabric at 400G
AS9716-32D	Tomahawk 4 (TH4)	400G	uSID capable (DA-only 6-hop); SONiC 202411+; standard 400G DC spine
AS9726-32D	Tomahawk 5 (TH5)	800G	Hardware-optimized uSID; SONiC 202411+; AI backend fabric, lossless RoCEv2
AS9736-64D	Tomahawk 6 (TH6)	800G/1.6T	Next-gen AI DC; SONiC; highest density uSID + CPO ready

UfiSpace

LCD Note: Lowest common denominator: S9100 (Qumran 2c) — carrier-grade DNX baseline for OcNOS/Arrcus. DNX family native to SP/carrier SRv6 uSID deployments from J2 onward. S9100 with Qumran 2c represents the oldest DNX-family model supporting full SRv6 uSID PE behavior. All S9xxx-higher models provide enhanced scale, buffer depth, and DNX feature support.

MODEL	BROADCOM CHIPSET	PORT SPEED	uSID CAPABILITY & NOS NOTES
S9100 / S9110	Qumran 2c (Q2c)	100G	uSID PE baseline — DNX entry point; OcNOS 6.3+, Arrcus 5.0.1+; Cell Site Gateway
S9300-32D	Trident 4 (TD4)	400G	XGS variant; SONiC or OcNOS; SP aggregation/peering
S9321-64E	Tomahawk 5 (TH5)	800G	AI backend uSID; Arrcus or SONiC 202411+; 6-hop DA-only lossless
S9600-32C	Jericho 2c+ (J2C+)	400G	Carrier core; deep-buffer DNX; OcNOS or Arrcus; SP core/WAN SRv6 uSID
S9610-36D	Jericho 3 (J3)	400G/800G	AI-optimized DNX core; OcNOS; massive buffers + native uSID carrier perf

Celestica

LCD Note: Lowest common denominator: Seastone DX010 (Tomahawk 4) — hyperscale reference platform. All Celestica platforms from TH4 onward support native SRv6 uSID in DA-only mode. The DX010 with TH4 is the deployment baseline; newer TH5/TH6 platforms scale up performance and density while inheriting the same uSID capability.

MODEL	BROADCOM CHIPSET	PORT SPEED	uSID CAPABILITY & NOS NOTES
Seastone DX010	Tomahawk 4 (TH4)	400G	uSID baseline; SONiC/OcNOS; hyperscale DC fabric reference
Seastone DX030	Tomahawk 5 (TH5)	800G	AI backend uSID; SONiC 202411+; lossless RoCEv2 AI fabric
DS3000 R4039	Tomahawk 6 (TH6)	800G/1.6T	Next-gen AI DC CPO; SONiC; maximum density uSID
CS6456-54P	Trident 4 (TD4)	100G	Carrier aggregation; OcNOS or SONiC; peering/edge uSID PE

Quanta Cloud Technology (QCT)

LCD Note: Lowest common denominator: QuantaMesh T3048-LY9 (Tomahawk 1) — baseline transit-only, no uSID. The QuantaMesh T7032-IX7 (Tomahawk 3) is also transit-only. Minimum PE/endpoint capability starts at T9032-IX9 (Tomahawk 5). This 'jump' from TH3 to TH5 is a clear hardware generation breakpoint for uSID.

MODEL	BROADCOM CHIPSET	PORT SPEED	uSID CAPABILITY & NOS NOTES
T3048-LY9	Tomahawk 1 (TH1)	100G	Blind-transit P-node only; no SRv6 uSID; SONiC for basic L2/L3
T7032-IX7	Tomahawk 3 (TH3)	100G	Transit-only; no native uSID endpoint; SONiC DC fabric baseline
T7032-IX9D	Tomahawk 4 (TH4)	400G	uSID DA-only; SONiC; hyperscale cloud DC fabric
T9032-IX9	Tomahawk 5 (TH5)	800G	Hardware-optimized uSID; SONiC 202411+; AI backend gold standard

Delta Networks (Agema)

LCD Note: Lowest common denominator: Agema AG5648 (Trident 3) — DC fabric baseline, transit-only for uSID. Minimum PE capability starts at AG9064 (Trident 4 / Tomahawk 4). Delta covers the full XGS range from ToR through spine, plus DNX for SP carrier deployments.

MODEL	BROADCOM CHIPSET	PORT SPEED	uSID CAPABILITY & NOS NOTES
AG5648	Trident 3 (TD3)	100G	DC ToR baseline; transit-only for uSID; SONiC L2/L3 fabric
AG9064	Trident 4 (TD4)	400G	uSID PE minimum; SONiC, OcNOS, or Arrcus; full XGS/DNX range
AG9032V3	Tomahawk 4 (TH4)	400G	uSID capable; SONiC; hyperscale spine
AG9816	Tomahawk 5 (TH5)	800G	AI fabric; SONiC 202411+; hardware uSID optimized

Ragile Networks

LCD Note: Lowest common denominator: RA-B6920-64C (Tomahawk 5) — Ragile entered the market targeting AI/HPC fabrics from the outset, so even their baseline model natively supports SRv6 uSID. All Ragile platforms are uSID PE-capable out of the box.

MODEL	BROADCOM CHIPSET	PORT SPEED	uSID CAPABILITY & NOS NOTES
RA-B6920-64C	Tomahawk 5 (TH5)	800G	uSID baseline; SONiC custom SAI; AI/HPC GPU-to-GPU fabric
RA-B6920-128C	Tomahawk 6 (TH6)	800G/1.6T	Next-gen AI DC; SONiC; maximum density spine

Micas Networks

LCD Note: Lowest common denominator: Micas S9600 (Tomahawk 5) — AI-fabric focused from baseline. Micas contributed custom Broadcom SAI integrations for SONiC specifically targeting AI backend uSID workloads. All models support SRv6 uSID natively.

MODEL	BROADCOM CHIPSET	PORT SPEED	uSID CAPABILITY & NOS NOTES
S9600-64H	Tomahawk 5 (TH5)	800G	uSID baseline; SONiC; AI spine/leaf
S9700 (TH6-Davisson)	Tomahawk 6 (TH6) + CPO	1.6T	CPO-integrated; SONiC; next-gen AI DC ultrascale

Alpha Networks

LCD Note: Lowest common denominator: SNX-60D0-486F (Tomahawk 5) — 32x800G baseline. Alpha's 64x1.6T TH6 platform extends further. All models are AI-fabric native with uSID support.

MODEL	BROADCOM CHIPSET	PORT SPEED	uSID CAPABILITY & NOS NOTES
SNX-60D0-486F	Tomahawk 5 (TH5)	800G	uSID baseline 32x800G; SONiC; AI DC spine
SNX-60A0-xxx	Tomahawk 6 (TH6)	1.6T	64x1.6T ultra-dense AI spine; SONiC; max scale

Inventec / Wistron / DNI

LCD Note: Lowest common denominator: TH4-based 400G platforms. All TH5/TH6 variants are native uSID. These ODMs build for hyperscaler OEM programs; publicly disclosed model names vary by customer. Publicly known platforms shown below.

MODEL	BROADCOM CHIPSET	PORT SPEED	uSID CAPABILITY & NOS NOTES
D7332 (Inventec)	Tomahawk 4 (TH4)	400G	uSID DA-only; SONiC; DC fabric baseline
Wiwynn/Wistron TH5	Tomahawk 5 (TH5)	800G	AI fabric; SONiC; air/liquid-cooled DC
DNI TH6	Tomahawk 6 (TH6)	800G/1.6T	CPO AI DC; SONiC; OCP summit showcase

Lanner Networks

LCD Note: Lowest common denominator: Lanner NCA-4210 (Jericho 2) — SP aggregation entry point for uSID. Lanner focuses on carrier/SP platforms; DNX Jericho 2 is the baseline for SRv6 uSID PE capability in the cell site and metro aggregation role. All Jericho 3 platforms exceed this baseline.

MODEL	BROADCOM CHIPSET	PORT SPEED	uSID CAPABILITY & NOS NOTES
NCA-4210	Jericho 2 (J2)	100G	uSID PE baseline; OcNOS or Arrcus; SP aggregation/5G CSR
NCA-6210	Jericho 2c+ (J2C+)	400G	Enhanced scale; OcNOS or Arrcus; carrier core/metro
NCA-6540	Jericho 3 (J3)	400G	AI-optimized DNX; OcNOS; SP core massive buffers

Asterfusion

LCD Note: Lowest common denominator: CX532P-N (Intel Tofino 2) — P4-programmable baseline. Asterfusion targets programmable pipelines; Tofino-based platforms require custom P4 programs for SRv6 uSID. Marvell OCTEON-based variants provide routing-oriented NOS support.

MODEL	BROADCOM CHIPSET	PORT SPEED	uSID CAPABILITY & NOS NOTES
CX532P-N	Intel Tofino 2	100G/400G	P4 programmable; custom SONiC or P4Runtime; SRv6 uSID via P4 program
CX864E-N	Marvell OCTEON TX2	400G	Routing-focused; OcNOS or custom NOS; SP/carrier peering

Open NOS Minimum Versions for SRv6 uSID — Aligned to LCD Chipset

The table below cross-references the LCD chipset baseline to the exact NOS software version required to unlock SRv6 uSID (F3216) endpoint behavior on that silicon.

NOS	MIN VERSION FOR uSID	LCD CHIPSET TARGET	WHAT IT UNLOCKS AT THE LCD FLOOR
SONiC (Community/ Commercial)	202305 (basic) / 202411 PhoenixWing (AI-stable)	Trident 4 (TD4), TH4, Jericho 2+	SAI uSID VPN support (202305); full 6-hop DA-only AI fabric (202411); P4/PINS gRPC controller integration for host+fabric unified SRv6 domain
Arrcus ArcOS	v5.0.1	Jericho 2/2c+, Trident 4, Tomahawk 5	First full uSID production release; native SRv6-TE policy; 5G slicing; SAI direct (no Redis) → fastest SAI-based convergence
IP Infusion OcNOS	v6.3.0 SP Profile	Broadcom DNX J2/J2c+/J3, Trident 4	Production uSID + TI-LFA sub-50ms; full interop Cisco/Nokia/Juniper; HAL/PAL direct C → ~40ms node convergence; Class C/D PTP for 5G CSR
NVIDIA Cumulus Linux	4.3.x (Broadcom) / 5.x (Spectrum only)	FROZEN at Broadcom 4.3.x branch post-NVIDIA acquisition	■ No longer a general-purpose Broadcom whitebox NOS — 5.x targets NVIDIA Spectrum ASICs only. Use SONiC, Arrcus, or OcNOS for Broadcom uSID.
DriveNets Network Cloud	2022+ (SP profile)	DNX J2/J3, TH4+	Cloud-native microservices NOS; scales as virtual chassis; SRv6 TE emerging; AI cluster and WAN focus
Pica8 PICOS	3.7+ (basic SRv6)	Trident 3+, TH3+	Enterprise campus/access; limited SRv6 uSID at SP scale; strong traditional CLI model
DENT OS	2022+ (community)	Trident 3+	Lightweight Linux kernel Switchdev; inherits FRR SRv6; not SP-grade uSID at scale; retail/campus edge
Meta FBOSS	Meta internal release cycles	Trident 4, TH4/5	Hyperscale DC reference architecture; not publicly deployable outside Meta ecosystem

Convergence Architecture: BGP PIC/H-FIB + TI-LFA + SRv6 uSID uLoop

BGP PIC Core + H-FIB: Prefix-Independent Convergence

The Hierarchical FIB (H-FIB) is the architectural foundation that makes large-scale network convergence sub-second regardless of how many prefixes are in the routing table. Instead of storing every BGP prefix with its own explicit next-hop interface, the H-FIB decouples the prefix table from the next-hop table:

[L3 Prefix Table: 1,000,000 entries] → pointer → [Next-Hop/Encap Table] → pointer → [Physical Egress]

When a core link fails, the 1,000,000 prefix entries remain completely untouched. Only a single pointer inside the Next-Hop Table changes — all prefixes pointing to that next-hop are simultaneously redirected in a single hardware clock cycle. BGP PIC Core automates this: when the IGP detects a path change to a BGP next-hop, it swaps the next-hop group pointer without re-evaluating individual prefix entries. BGP PIC Edge pre-programs primary and backup egress PE objects — if PE-1 fails, the ingress router swings all customer traffic to the pre-programmed PE-2 object in one hardware operation.

TI-LFA + SRv6 uSID uLoop: Sub-50ms Protection

TI-LFA pre-computes a backup SID/uSID stack before any failure occurs. The backup path is programmed into the ASIC as a pre-installed next-hop group entry. When a link or node fails, the ASIC hardware detects the failure and switches to the backup SID stack immediately — bypassing the software control plane entirely for the initial switchover. With SRv6 uSID and the uLoop-avoidance mechanism, this switchover time is well under 50ms and approaches sub-millisecond in hardware-detected link failures (Loss of Signal). The three-tier nested protection: TI-LFA (local link/node) → BGP PIC Core (remote P-node) → BGP PIC Edge (egress PE failure) ensures every failure scenario has a pre-programmed hardware response that fires before the control plane even processes the event.

FAILURE TYPE	PROTECTION MECHANISM	OcNOS CONVERGENCE TIME	SONIC CONVERGENCE TIME	WHAT CHANGES IN HARDWARE
Link failure (LOS)	TI-LFA hardware switchover	Sub-50ms (hardw are-detected)	Sub-50ms (hardw are-detected)	ASIC de-activates primary next-hop index; pre-programmed backup SID stack fires at line rate
Node failure (BFD)	BFD → TI-LFA backup SID stack	~40ms (35ms BFD + 5ms C PAL)	60–180ms (35ms BFD + Redis lag)	BFD timeout → control plane updates next-hop pointer → ASIC swaps to backup uSID path
BGP next-hop loss (PIC Core)	H-FIB pointer swap (prefix-independent)	Single C struct pointer update	Redis ASIC_DB key update (faster than prefix-by-prefix)	H-FIB next-hop group pointer changed; 1M+ prefixes redirected in one hardware operation — no per-prefix reprogramming
Egress PE failure (PIC Edge)	Pre-programmed backup PE next-hop object	Near-instant (pre-programmed)	Near-instant via Redis ASIC_DB	Primary PE next-hop object invalidated; ASIC swings to pre-programmed backup PE object — customer traffic unaffected

NOS Architecture: C Structs vs. C++ OOP Containers

FRR: Pure C Stacked Structures — Common Routing Brain

FRRouting (FRR) is the common routing control plane shared by SONiC and Arrcus (and architecturally inherited by OcNOS through GNU Zebra lineage). FRR is written entirely in **pure C** using a multi-level hierarchy of C structs linked by pointer chains:

- Parent: struct route_node (Patricia/Radix tree node holding the prefix)
- Child: struct bgp_path_info (BGP) / struct isis_route_info (IS-IS) — protocol path attributes
- Next-hop chain: struct nexthop linked list (next* pointer per ECMP path)
- SRv6 uSID extension: parent IPv6 struct + stacked uSID carrier sub-structs per path

This entire RIB database lives in local RAM, optimized to byte boundaries for CPU L1/L2 cache locality. FRR handles all routing math — BGP, IS-IS, OSPF, SR-MPLS, SRv6 uSID — identically regardless of which NOS is running on top. The divergence is entirely in how each NOS takes FRR's output and programs the forwarding silicon.

SONiC: C++ OOP Containers — Redis Database Pipeline

SONiC wraps FRR in a containerized, database-driven architecture. Every daemon runs in its own Docker container. The hardware programming pipeline crosses multiple process boundaries:

FRR (C) → fpmSyncd (C++) → Redis APPL_DB (string key-value) → orchagent (C++ classes: RouteOrch, NextHopGroupOrch, NeighborOrch) → ASIC_DB → syncd → SAI API → Broadcom SDK → ASIC.

Each step serializes data into strings and crosses a Linux namespace. Under 1M+ BGP route churn, orchagent's C++ object instantiation and Redis pub/sub locking create millisecond-to-second CPU spikes. The benefit: container restart isolation — if FRR crashes, the ASIC keeps forwarding because state lives in Redis, not inside the crashed process.

OcNOS: Pure C Monolithic — Direct HAL/PAL Memory Execution

IP Infusion OcNOS inherits ZebOS's direct C execution model. No containers, no Redis, no C++ objects. Protocol daemons communicate via POSIX signals and direct memory passing. When the routing daemon selects the best path, the winning route struct is passed by C pointer to the PAL/HAL layer, which fires a single C-function call (e.g., bcm_l3_route_add()) directly into the Broadcom SDK. The SDK executes a memcpy or aligned register write into the ASIC TCAM. Near-zero serialization latency. OcNOS achieves ~40ms node-failure convergence (35ms BFD + 5ms software processing). SONiC typically achieves 60–180ms (35ms BFD + 25–150ms Redis pipeline).

Arrcus ArcOS: Ground-Up Modular — SAI Without Redis

Arrcus is not derived from GNU Zebra or FRR. Built from scratch as a modern modular routing OS, ArcOS uses a multi-threaded C/C++ routing engine that calls SAI directly — without the Redis mid-layer that creates SONiC's serialization bottleneck. This gives Arrcus SAI-standard silicon portability with closer-to-OcNOS convergence speed. ArcOS is the first NOS to achieve full SRv6 uSID production support with native SRv6-TE policy management.

SONiC P4/PINS + gRPC: Unified DC Fabric to 5G Host

P4 Integrated Network Stack (PINS) adds a P4Runtime gRPC server directly into SONiC. A single SDN controller can now program both whitebox switch ASICs (via P4Runtime/SAI) and FRR-based Linux compute hosts (via gRPC to FRR) using the same SRv6 uSID instruction set. In a 5G cloud-native vRAN deployment: the vDU host runs FRR, wraps radio packets in a uSID DA, and the SONiC ToR switch (programmed by the same gRPC controller) steers the packet through the fabric using the identical uSID carrier — host and switch become a single programmable SRv6 domain with no translation layer required. This is the architectural foundation for URLLC 5G slice enforcement from host NIC to DWDM ring.

Common Open NOS Architecture

DIMENSION	FRR (common to SONiC & Arrcus control plane)	SONiC (C++ OOP / Redis)	OcNOS (Pure C / HAL/PAL)	Arrcus ArcOS (Modular / SAI Direct)
Control plane language	Pure C — Patricia tree + nexthop linked list	FRR (C) inside Docker container	Pure C throughout — ZebOS heritage	C/C++ multi-threaded routing engine — ground-up
Hardware abstraction	N/A — FRR outputs to Netlink or FPM TCP	SAI (OCP standard) via syncd container	HAL/PAL (proprietary C direct call)	SAI (direct — no Redis mid-layer)
FIB programming path	zebra → Netlink (host) or FPM TCP (SONiC)	fpmsyncd→Redis APPL_DB→orchagent (C++ RouteOrch)→ASIC_DB→syncd→SAI→SDK	Protocol daemon → PAL/HAL (C pointer) → bcm_l3_route_add() → ASIC TCAM	Routing engine → SAI adapter → SDK → ASIC (3 hops vs SONiC's 7)
SRv6 uSID FIB write	uSID sub-struct pointer chain in nexthop struct	C++ EVPN/VRF object graph → multi-stage Redis → SAI uSID entry	Linear RAM: parent IPv6 + uSID sub-structs → PAL maps 128-bit DA shift mask → ALPM one C call	SRv6-TE engine → SAI path with minimal C++ object overhead
ISSU / HA model	N/A	Container restart; Redis state survives; hardware keeps forwarding — best failure isolation	Binary state freeze → NVRAM → reload C structs; zero HW touch if topology unchanged	In-process warm restart + RIB checkpointing
5G vRAN integration	FRR on vDU host: SRv6 uSID DA via Linux Netlink	SONiC+PINS: gRPC SDN controller programs both whitebox ASIC and FRR host — unified SRv6 domain	OcNOS + PTP/SyncE: Class C/D timing for vDU; lean monolithic footprint on CSR	Arrcus: SRv6-TE slice isolation + K8s API integration for cloud-native vRAN

NX-OS SRv6 uSID Clarification — Cloud Scale vs. Silicon One

■ **Frequently Asked: Does NX-OS support SRv6 uSID?** The answer depends on the ASIC. NX-OS 10.2(1) on Nexus 9300 (Cloud Scale ASIC) supports full 128-bit SRv6 SID ONLY — the Cloud Scale ASIC cannot perform Shift-and-Forward (NEXT-CSID) DA manipulation required for uSID at line rate. This is a hardware constraint. Nexus 9800 with Silicon One ASIC (NX-OS 10.6(2)F+) natively supports SRv6 uSID. Upgrading from Nexus 9300 to Nexus 9800 requires hardware replacement — not software-only.

h.Encaps.Red vs. h.Insert.Red — TI-LFA Encapsulation

H.Encaps.Red vs. H.Insert.Red — TI-LFA Head-End Encapsulation

H.Encaps.Red (Reduced Encapsulation): The head-end PE node adds a new outer IPv6 header containing the SRv6 uSID carrier in the Destination Address, with an optional SRH for paths requiring more than 6 hops. H.Encaps.Red is the standardized behavior defined in RFC 8986 and is the primary method used for PE-to-PE SRv6 uSID encapsulation. When overlay/underlay merge is disabled (vendor default), all 6 uSID slots in the DA carrier are available for transport path steering, and service SIDs are handled separately.

H.Insert.Red (Reduced Insertion): H.Insert.Red inserts SRv6 routing information into the existing IPv6 packet header rather than adding a new outer encapsulation. Important: H.Insert was NOT standardized in RFC 8986 (it was excluded). H.Insert.Red is used in specific scenarios such as TI-LFA Point of Local Repair (PLR) backup path insertion, where the PLR node needs to reroute the packet to the merge point without full re-encapsulation. At the PLR, the node inserts an additional IPv6 header with 1–3 uSIDs pointing to the merge point — this is the H.Insert.Red PLR encapsulation.

TI-LFA PLR Encapsulation Math (Broadcom, 2024):

With SRv6 uSID + TI-LFA (un-prefix SID scenario, ≤6 transport hops so no SRH at source):

Source node encap: IPv6 header (40 bytes) + uSID DA (6 entries in 128 bits)

PLR node adds: IPv6 header (40 bytes) + 1–3 uSIDs to merge point

Total overhead: 80 bytes (two IPv6 headers) + uSID carriers — far less than full SRv6+SRH

Compared to full 128-bit SRv6 TI-LFA: IPv6 (40B) + SRH + End SID + PLR IPv6 (40B) + SRH + 1–3 merge SIDs

uSID advantage: eliminates both SRH extension headers in the ≤6-hop case, reducing overhead and eliminating deep-parser recirculation at transit nodes.

Silicon Hop-Depth & Protocol Scale Tables

The following numbered tables use the carrier × 6 formula (derived from segment-routing.net MPLS WC 2022 demo methodology): uSID carriers × 6 uSIDs/carrier = total uSIDs without recirculation. Source for 26-uSID line-rate demo: Rakuten/Bell Canada/Cisco, MPLS WC Paris 2022, segment-routing.net.

Table 1: Broadcom XGS Family (Trident/Tomahawk) — SRv6 uSID Hop Depth per Generation

Sources: *segment-routing.net MPLS WC 2022 demo report*; *per-silicon carrier analysis*. Formula: $\text{uSID carriers} \times 6 \text{ uSIDs/carrier} = \text{total uSIDs without recirculation}$.

CHIPSET	STATUS	MPLS LABELS	FULL SIDs (1 pass)	uSID CARRIERS	TOTAL uSIDs	NOTES
Trident 2 / 2+	EOL	N/A	0	0	0	No SRv6 capability. VXLAN/L2 only.
Tomahawk 1/2 (TH1/2)	EOL	~8	0	0	0	Fixed pipeline. No SRv6 parser.
Trident 3 (TD3)	Legacy	~8	0	0	0	Basic SRv6 encap possible via software; no native uSID shift-and-forward at line rate.
Tomahawk 3 (TH3)	Legacy	~8–12	0	0	0	Bandwidth optimized fixed pipeline. No programmable SRv6 uSID. NX-OS Cloud Scale ASIC era; uSID not supported.
Trident 4 (TD4)	Modern — uSID baseline	~12	4	4×6=24	24	Per-silicon analysis based on <i>segment-routing.net</i> and MPLS WC 2022 demo data: 4 uSID carriers × 6 = 24 uSIDs in single pass. NPL/P4 programmable architecture.
Tomahawk 4 (TH4)	Modern	~12	4	4×6=24	24	4 full SID single-pass imposition. uSID DA-only (≤6) extremely efficient. SRH beyond triggers recirculation.
Tomahawk 5 (TH5)	AI Native	~12	4	4×6=24	24	Hardware-optimized AI backend. Same carrier math as TH4 but fully lossless at 51.2 Tbps. Gold standard for AI uSID fabrics.
Tomahawk 6 (TH6)	Next-Gen	~12	4+	4+×6=24+	24+	New gen; exact carrier count not yet publicly published. Exceeds TH5 performance. CPO-capable.

Table 2: Broadcom DNX Family (Jericho/Qumran) — SRv6 uSID Hop Depth per Generation

Sources: Silicon analysis per segment-routing.net MPLS WC 2022: Jericho 3–4 full SIDs = 18–24 uSIDs; Jericho 2c 6 full SIDs = 36 uSIDs. MPLS WC Paris 2022: 26 uSIDs at line rate on NCS5700/J2C+ (Rakuten/Bell Canada/Cisco, segment-routing.net).

CHIPSET	STATUS	MPLS LABELS	FULL SIDs (1 pass)	uSID CARRIERS	TOTAL uSIDs	NOTES
Qumran-MX / AX	EOL	~8	0	0	0	Jericho 1 generation. Basic MPLS only. No SRv6 uSID support.
Jericho 1 (J1)	EOL	~8	0	0	0	Cannot parse SRH at line rate. Deep parser buffer insufficient for SRv6.
Jericho 2 (J2)	Mature — uSID baseline	~12	4	4×6=24	24	Jericho family silicon analysis: 3–4 full SID single pass = 3×6=18 to 4×6=24 uSIDs. NCS 5700 (J2) demo: 26 uSIDs at line rate (4.33 carriers in single pass). Rakuten/Bell Canada/Cisco, MPLS WC Paris 2022, segment-routing.net .
Qumran 2c (Q2c)	Mature	~12	4	4×6=24	24	Cost-optimized J2 variant for cell site gateway. Same uSID carrier math as J2.
Jericho 2c+ (J2C+)	High Perf	~12	4–5	4×6=24 to ~26	24–26	Enhanced ingress pipeline vs J2. The 26-uSID demonstration (segment-routing.net) was specifically on J2/J2C+ NCS 5700 platforms.
Jericho 3 (J3)	AI/Core	~12–14	6	6×6=36	36	Jericho 2c silicon analysis: 6 full SID single pass = 6×6 = 36 uSIDs; massive deep buffers; AI cluster optimization. Segment-routing.net MPLS WC 2022 carrier-x-6 methodology.
Jericho 4 (J4)	New Gen 2025	TBD	6+	6+×6=36+	36+	J4 / BCM99450. HBM packet memory. 160× more traffic buffering vs on-chip. Exact uSID carrier count not yet publicly published. Expected to exceed J3.

Table 3: Protocol Comparison — MPLS / SR-MPLS / SRv6 Full SID / SRv6 uSID Hop Scale

Source: Daniel Voyer, Bell Canada, MPLS WC Paris 2022 (segment-routing.net): '26 uSIDs is 2–3× better than current MPLS deployed use-cases.' Carrier × 6 formula per segment-routing.net MPLS WC 2022 demo methodology. RFC 9800 NEXT-CSID.

PROTOCOL	CONTROL PLANE	DATA PLANE	MAX HOPS (line rate)	uSID CARRIERS	TOTAL uSIDs	SOURCE / NOTES
Legacy MPLS (LDP/RSVP-TE)	LDP + RSVP-TE — per-LSP soft state	4-byte label stack	~8–12 labels at line rate (platform dependent)	N/A	N/A	Industry average; no single published standard — varies by ASIC generation
SR-MPLS v4	IGP SR extensions (OSPF/ISIS SR)	4-byte MPLS label stack	~12 labels at line rate (J2/TH4 class silicon)	N/A	N/A	Same MPLS data plane as LDP; SR only changes label signaling, not hardware limits
SR-MPLS v6	IGP SR extensions + IPv6 transport	MPLS-in-IPv6 encap	~12 labels at line rate	N/A	N/A	Intermediate step; still label-stack constrained. Adds IPv6 outer header overhead.
SRv6 full 128-bit SID (with SRH)	IGP IPv6 SR extensions	IPv6 + SRH extension header	~4–6 full 128-bit SIDs at line rate without recirculation (J2/J3/TH4)	1 SID per carrier entry	N/A	Deep SRH parsing forces recirculation on older ASICs; 50–66% throughput penalty on pre-J2 silicon
SRv6 uSID ≤6 hops (DA-only, no SRH)	IGP IPv6 SR extensions	Single 128-bit IPv6 DA — NO SRH required	6 uSIDs per carrier in DA	1 carrier = 6 uSIDs	6 uSIDs (1 carrier) — zero recirculation, standard IPv6 transit on P-nodes	RFC 9800 NEXT-CSID. P-nodes see standard IPv6. Zero forklift for transit nodes.
SRv6 uSID with SRH (>6 hops)	IGP IPv6 SR extensions	IPv6 DA (6 uSIDs) + SRH (additional carriers)	Silicon analysis: Trident/TH = 4 carriers × 6 = 24 uSIDs; J2 = 4–5 × 6 = 24–26 uSIDs; J2C/J3 = 6 × 6 = 36 uSIDs (carriers × 6 methodology, segment-routing.net MPLS WC 2022)	Per silicon: 4–6 carriers	24–36 uSIDs (silicon dependent)	MPLS WC Paris 2022: NCS5700/J2C+ = 26 uSIDs at line rate (Rakuten/Bell Canada/Cisco demo, segment-routing.net). 2–3× better than MPLS.

Table 4: Incumbent Vendor uSID Hop Scale Comparison vs. Whitebox (Broadcom Silicon)

Sources: [segment-routing.net MPLS WC 2022 demo](#); [silicon carrier analysis](#); Cisco documentation; EANTC 2024/2025 interop reports. ■ Nexus 9300 (Cloud Scale ASIC): NX-OS 10.2(1) supports full 128-bit SRv6 SID ONLY — NOT uSID. Nexus 9800 (Silicon One): NX-OS 10.6(2)F+ supports uSID natively. Upgrade requires hardware replacement.

VENDOR	PLATFORM	SILICON / ASIC	MPLS LABELS	uSID CARRIERS	TOTAL uSIDs	SOURCE / NOTES
Cisco	NCS 5500 (NC55 Gen1)	Jericho 2	~12 labels	4 carriers	24 uSIDs	Broadcom J2; MPLS WC 2022 demo (segment-routing.net)
Cisco	NCS 5700 (NC57)	Jericho 2c+	~12 labels	4–5 carriers	24–26 uSIDs	26-uSID line-rate demo: Rakuten/Bell Canada/Cisco, MPLS WC Paris 2022, segment-routing.net
Cisco	8000 Series	Silicon One Q200	~12 labels	4–5 carriers	24–26 uSIDs	Built natively for NEXT-CSID from day one. IOS XR 7.3.2+
Cisco	ASR 9000 (LSP+)	Tomahawk Gen3	~8–12 labels	4 carriers	~24 uSIDs	Lab-validated: A9K-MOD400-TR via IOS XR 7.3.2+ microcode
Cisco	Nexus 9300	Cloud Scale ASIC	~8 labels	Full 128-bit SID only — uSID NOT supported	N/A	NX-OS 10.2(1): full SRv6 128-bit SID; uSID shift-and-forward not natively supported on Cloud Scale ASIC
Cisco	Nexus 9800	Silicon One G200/E100	~12 labels	4–5 carriers	24–26 uSIDs	NX-OS 10.6(2)F+. Silicon One architecture supports uSID natively. Upgrade path from Nexus 9300 Cloud Scale.
Juniper	MX (Trio 5, MPC10E)	Trio 5 Penta	~8–12 labels	4 carriers	~24 uSIDs	Junos 23.4R1+. uSID PE capable on MPC10E/11E/LC9600.
Juniper	PTX10000 (Express 4/5)	Express 4/5	~12 labels	4 carriers	~24 uSIDs	All PTX10000 flavors validated. Junos Evolved 23.4R1+.
Nokia	7750 SR FP5	FP5 uCode	~12 labels	4–5 carriers	~24 uSIDs	SR OS 22.5.R1+. FP5 fully programmable uCode upgrades.
Nokia	7250 IXR	Jericho 2 / FP4	~12 labels	4 carriers	~24 uSIDs	All models validated. SR OS 22.5.R1+ code upgrade only.
Arista	7280R3/7800R3	Jericho 2	~12 labels	4 carriers	~24 uSIDs	EOS 4.28.0F+. Same J2 silicon as NCS 5700.

Table 5: Complete Open NOS Platform Matrix — SRv6 uSID Support & Architecture

■ Cumulus Linux: post-NVIDIA acquisition, no longer a general-purpose Broadcom whitebox NOS — Broadcom support frozen at 4.3.x branch; 5.x targets NVIDIA Spectrum ASICs only. Cisco NX-OS platforms (Nexus 9300 and 9800) are covered in Table 4 — Incumbent Vendor Comparison.

NOS PLATFORM	LICENSE	ARCHITECTURE / HW ABSTRACTION	LCD SILICON / MIN VERSION FOR uSID	PRIMARY TARGET	KEY NOTES
SONiC (Community/Enterprise)	Open Source (Linux Foundation)	C++ SWSS/orchagent + Redis + FRR (C) <i>HW abstraction:</i> SAI (OCP standard)	Trident 4+, TH4+, J2+ Min ver: 202305 (basic) / 202411 (AI-stable)	Hyperscale DC, AI backend fabric	Best cloud-native 5G vRAN via P4/PINS gRPC unified host+fabric domain
Arrcus ArcOS	Commercial	Ground-up modular C/C++; multi-threaded routing engine <i>HW abstraction:</i> SAI (direct, no Redis)	J2/2c+, Trident 4, TH5 Min ver: v5.0.1	Core/WAN, AI DC, SRv6-TE	First to market full uSID; native SRv6-TE policy; fastest SAI-based convergence (~45ms node)
IP Infusion OcNOS	Commercial	Pure C monolithic (ZebOS/GNU Zebra heritage); HAL/PAL direct <i>HW abstraction:</i> HAL/PAL (proprietary C direct call)	DNX J2/2c+/J3, Trident 4 Min ver: v6.3.0 SP	5G CSR, SP Core, WAN, DWDM	Best Class C/D PTP stability; ~40ms node convergence; carrier-grade MPLS+SRv6 interop
DriveNets Network Cloud	Commercial	Cloud-native microservices NOS; distributed across whitebox cluster <i>HW abstraction:</i> Proprietary per-node SAI	DNX J2/J3, TH4+ Min ver: 2022+ (SP profile)	SP Core, hyperscale DCI, distributed spine	Scales as virtual chassis; SRv6 TE emerging; AI cluster and WAN focus
Pica8 PICOS	Commercial	Enterprise Linux-based NOS; OpenFlow/SDN support <i>HW abstraction:</i> SAI / OvS bridge	Trident 3+, TH3+ Min ver: 3.7+ (basic SRv6)	Enterprise campus, access edge, DC	Limited SRv6 uSID at SP scale; strong for enterprise campus/access; traditional CLI model
DENT OS	Open Source (Linux Foundation)	Linux kernel Switchdev; lightweight edge NOS <i>HW abstraction:</i> Linux Switchdev (kernel-native)	Trident 3+ Min ver: 2022+ (community)	Retail branch, campus edge	Lightweight footprint; inherits Linux kernel SRv6 via FRR; not SP-grade uSID at scale
NVIDIA Cumulus Linux	Commercial (NVIDIA, post-acq.)	Linux-based; formerly open whitebox NOS <i>HW abstraction:</i> Proprietary switchd (closed) / NVUE	NVIDIA Spectrum-2/3/4 ONLY — Broadcom FROZEN at 4.3.x Min ver: 4.3.x (Broadcom); 5.x (Spectrum only)	NVIDIA Spectrum DC fabrics only	■ NO longer a general-purpose Broadcom whitebox NOS. Broadcom support frozen at branch 4.3.x. 5.x = Spectrum ASICs only.
Meta FBOSS	Open Source (Meta/OCP)	C++ microservices; Facebook/Meta hyperscale DC NOS <i>HW abstraction:</i> SAI (Broadcom/Meta SAI fork)	Trident 4, TH4/5 Min ver: Meta internal release cycles	Meta/hyperscale DC fabric (not publicly deployable)	Not publicly deployable outside Meta ecosystem. Reference architecture for hyperscale open NOS design.