LCD BASELINE · FULL MODEL PROGRESSION · BGP PIC/H-FIB · C vs C++ DEEP-DIVE · SONiC P4/PINS

Navigating SRv6 uSID Use Cases with Open NOS & Whitebox Vendors

LCD → Full Model Progression · BGP PIC/H-FIB/TI-LFA Convergence

C Structs vs. C++ OOP · SAI vs. HAL/PAL · SONiC P4/PINS 5G vRAN

11 Vendors	Sub-50ms	LCD→Latest
Full Model Tables	TI-LFA + H-FIB	Generation Guide

SRv6 TEK Innovations · Vendor-Neutral Network Engineering Consultancy

NAICS 541512 · 541519 · 541611 · MBE-Eligible Minority-Owned Small Business

GNU Zebra Lineage & The LCD Baseline Methodology

In 1996, Kunihiro Ishiguro created GNU Zebra and released it under the GPL. In 1999, he co-founded IP Infusion and exercised dual licensing: the same codebase became ZebOS (now OcNOS) commercially, while the open-source line continued as Quagga then FRRouting (FRR, Linux Foundation 2017). SONiC uses FRR inside a Docker container as its routing control plane. Arrcus ArcOS was built from scratch with no Zebra heritage — the third architectural path in the open NOS landscape.

"The LCD methodology: identify your oldest deployed hardware, find its row in the vendor table, and every model above it in the same family is fully supported — no additional analysis required."

Each vendor section below starts with the Lowest Common Denominator (LCD) model — the oldest or most entry-level platform that still runs the Open NOS ecosystem. The LCD row defines: the minimum Broadcom silicon generation, minimum NOS software version for SRv6 uSID support, and whether that platform acts as a blind-transit P-node only or as a full uSID PE/endpoint. Subsequent rows in each table show how capability scales upward through silicon generations.

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.**

Complete Whitebox Vendor Model Tables — LCD to Latest Generation

The tables below provide the complete model progression for each major whitebox vendor. The first row of each table (highlighted with a gold border above) is the LCD baseline. Silicon generations are listed oldest-to-newest, showing the exact point where native SRv6 uSID PE capability begins. Port speeds progress from 100G through 400G to 800G/1.6T.

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 PAIRING
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 PAIRING
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 PAIRING
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 PAIRING
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 PAIRING
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 PAIRING
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 PAIRING
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 PAIRING
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 PAIRING
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 PAIRING
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 PAIRING
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

Table A: Minimum Open NOS Software Versions — Aligned to LCD Chipset

The following table cross-references each NOS platform's minimum software version for SRv6 uSID (F3216) against the LCD chipset baseline it targets. Newer models in the same vendor family exceed these baselines exponentially in scale, feature depth, and port density.

NOS	MIN VERSION FOR uSID	LCD CHIPSET BASELINE	KEY CAPABILITY UNLOCKED AT LCD
SONiC (Community/ Commercial)	202305 basic / 202411 AI-stable	Trident 4 (TD4), TH4, Jericho 2+	P4/PINS gRPC controller; SAI uSID VPNs (202305); full 6-hop DA-only AI fabric (202411); FRR host+fabric unified SRv6 domain
Arrcus ArcOS	v5.0.1	Jericho 2/2c+, Trident 4, TH5	First production uSID; native SRv6-TE policy; 5G slicing; SAI direct (no Redis) → fastest SAI-based convergence ~45ms node failure
IP Infusion OcNOS	v6.3.0 SP Profile	Broadcom DNX J2/J2c+/J3, Trident 4	Production uSID + TI-LFA sub-50ms; HAL/PAL direct C → ~40ms node convergence; Class C/D PTP for 5G CSR; interop Cisco/Nokia/Juniper
NVIDIA Cumulus Linux	4.3.x (Broadcom frozen) / 5.x (Spectrum only)	■ NVIDIA Spectrum ASICs only post-acquisition	■ No longer general-purpose Broadcom whitebox NOS. Broadcom support frozen at 4.3.x. Use SONiC, Arrcus, or OcNOS for Broadcom XGS/DNX uSID deployments.
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

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

The combination of Hierarchical FIB (H-FIB), BGP Prefix-Independent Convergence (PIC), and Topology-Independent Loop-Free Alternate (TI-LFA) creates a three-tier nested protection system that fires pre-programmed hardware responses at every failure tier before the control plane has finished processing the event.

H-FIB: The Pointer Architecture

In a flat FIB, every prefix carries an explicit next-hop. A 1M-prefix BGP table requires 1M hardware updates when a path changes. The H-FIB decouples this: [L3 Prefix Table: 1M entries] → pointer → [Next-Hop/Encap Table: shared objects] → pointer → [Physical Egress]. When a core link fails, only the pointer in the Next-Hop Table changes. All 1M prefixes are simultaneously redirected in a single hardware clock cycle — prefix-independent convergence.

BGP PIC Core: Remote Transit Protection

BGP PIC Core pre-computes a backup BGP next-hop path. When an IGP change invalidates the primary core path, the BGP engine swaps the shared next-hop pointer in the H-FIB without re-evaluating individual prefix entries. Next-hop tracking (IGP→BGP callback) monitors IGP reachability to BGP next-hops continuously — any IGP change immediately triggers the pre-computed H-FIB pointer swap.

BGP PIC Edge: Egress PE Protection

For dual-homed customer sites, BGP PIC Edge pre-programs a primary PE object and backup PE object. If PE-1 fails, the ingress router invalidates the PE-1 Next-Hop Object in the H-FIB and swings to PE-2 in one hardware operation — all customer prefixes redirected simultaneously with zero per-prefix reprogramming.

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

TI-LFA pre-computes backup SID/uSID stacks before any failure occurs and installs them in the ASIC as pre-programmed next-hop group entries. Link failure (hardware-detected Loss of Signal): ASIC switches to backup uSID stack at line rate in under 1ms — entirely in hardware, bypassing the software control plane. Node failure (BFD-detected): 35ms BFD timeout + control-plane processing → total well under 50ms with OcNOS HAL/PAL direct C execution, or 60–180ms with SONiC's Redis pipeline under heavy load. The SRv6 uSID uLoop-avoidance mechanism prevents transient forwarding loops during the reconvergence window.

FAILURE TYPE	MECHANISM	OcNOS TIME	SONiC TIME	ARRCUS TIME	HARDWARE ACTION
Link failure (LOS)	TI-LFA hardware ASIC	**Sub-1ms**	**Sub-1ms**	**Sub-1ms**	ASIC fires backup uSID stack at line rate — control plane not involved
Node failure (BFD)	BFD → TI-LFA backup SID	**~40ms**	**60–180ms**	**~45ms**	BFD timeout → next-hop pointer update → ASIC swaps to pre-computed uSID backup
BGP next-hop loss (PIC Core)	H-FIB pointer swap	**Single C ptr update**	**Redis ASIC_DB key update**	**SAI direct update**	1M+ prefixes simultaneously redirected — zero per-prefix reprogramming via H-FIB
Egress PE failure (PIC Edge)	Pre-programmed backup PE object	**Near-instant**	**Near-instant**	**Near-instant**	Primary PE NH object invalidated; all customer traffic swings to backup PE in one HW op

IMPACT: **BGP PIC Core + H-FIB + TI-LFA: three tiers of pre-programmed hardware responses that fire before the control plane finishes processing. The only variable between NOS platforms is how fast each crosses the software/hardware boundary for the post-switchover cleanup.**

NOS Architecture Deep-Dive: C Structs vs. C++ OOP Containers (see Table C)

The routing control plane is common to all three NOS platforms at the mathematical level: FRR (SONiC, Arrcus) and OcNOS (ZebOS heritage) both use the same GNU Zebra-derived Patricia tree structure with linked-list nexthop chains. The divergence is entirely in how each platform crosses the software/hardware boundary.

FRR: Pure C Stacked Structures — The Common Brain

FRR represents its entire routing database as a hierarchy of C structs linked by raw pointers: • struct route_node: parent node in the Patricia/Radix tree; holds the prefix (IPv4, IPv6, MPLS label) • struct bgp_path_info / struct isis_route_info: protocol-specific child structs with path attributes • struct nexthop: linked list of next-hops (next* pointer per ECMP path; backup TI-LFA paths chained) • SRv6 uSID extension: parent IPv6 struct points to child uSID carrier sub-struct with 6×16-bit uSID fields This database lives entirely in local RAM. FRR optimizes every struct to byte boundaries for CPU L1/L2 cache locality. Walking 1M BGP prefixes or 500K IS-IS nodes costs microseconds, not milliseconds.

SONiC: C++ OOP Docker Containers — Redis Database Pipeline

SONiC runs each daemon in an isolated Docker container. The C++ orchestration framework (SWSS / orchagent) uses class hierarchies to model network objects: RouteOrch, NextHopGroupOrch, NeighborOrch, VxlanTunnelOrch, EvpnOrch — each inheriting from base orchestration engines with C++ virtual method tables, state machines, and dependency graphs. FIB programming pipeline: FRR (C) → fpmsyncd (C++) → Redis APPL_DB (string key-value) → orchagent (C++ object instantiation; dependency tracking) → ASIC_DB → syncd → SAI API → Broadcom SDK → ASIC. Every step serializes C data into strings and crosses a Linux namespace boundary. Under 1M+ BGP route churn, orchagent's C++ object instantiation and Redis pub/sub locking create CPU spikes that extend convergence to 60–180ms for node failures. The benefit: absolute container isolation — FRR crash never drops a forwarded packet because state lives in Redis.

OcNOS: Pure C Monolithic — HAL/PAL Direct Execution

OcNOS inherits ZebOS's direct C execution model. No containers, no Redis, no C++ objects. Protocol daemons communicate via POSIX signals and shared memory. The winning route struct is passed by raw C pointer to the PAL/HAL layer, which fires bcm_l3_route_add() directly into the Broadcom SDK. The SDK executes a memcpy or aligned register write into ASIC TCAM. OcNOS data structures use deeply nested C structs and arrays of function pointers — modular abstraction the UNIX way, without C++ virtual dispatch overhead. CPU execution traverses linear function pointer chains, maximizing L1/L2 cache hit rates. FIB programming pipeline: Protocol daemon (C struct) → PAL/HAL (C pointer) → bcm_l3_route_add() → ASIC TCAM. Three steps vs. SONiC's seven. ~40ms node convergence vs. SONiC's 60–180ms. Trade-off: a faulty C pointer causes a kernel panic vs. SONiC's isolated container restart.

Arrcus ArcOS: Ground-Up Modular — SAI Without Redis

Arrcus was built from scratch — not derived from GNU Zebra, FRR, or ZebOS. ArcOS uses a multi-threaded C/C++ routing engine that calls SAI directly from the routing module, without the Redis database mid-layer. This gives ArcOS OCP-standard silicon portability (same as SONiC) with convergence speeds approaching OcNOS (~45ms node failure). Arrcus was first to market with production SRv6 uSID and native SRv6-TE policy management.

© 2026 SRv6 TEK Innovations · Open NOS White Paper · Confidential

SONiC P4/PINS + gRPC: Unified DC Fabric to 5G Host SRv6 Domain

P4 Integrated Network Stack (PINS) adds a P4Runtime gRPC server into SONiC, enabling a single SDN controller to program both whitebox switch ASICs and FRR routing stacks on Linux compute hosts over the same gRPC channel. In a 5G cloud-native open vRAN deployment: the vDU host runs FRR, wraps radio packets in a SRv6 uSID DA, and the SONiC ToR switch (programmed by the same gRPC controller) steers the packet through the DC fabric using the identical uSID carrier. Host NIC and whitebox switch become a single unified SRv6 programming domain — no protocol translation layer, no separate controller plane. The same URLLC slice SID that the vDU host inserts is honored end-to-end from the Linux kernel forwarding table through the SONiC whitebox spine to the DWDM ring. OcNOS provides the 5G counter-answer: Class C/D PTP (G.8275.1) and SyncE integrated directly into the C execution model, with no Docker container jitter affecting clock synchronization — critical for the <1µs timing accuracy that vDU baseband processing requires. On a power-constrained Cell Site Router, OcNOS's monolithic footprint leaves far more CPU headroom for PTP/BFD than SONiC's 20+ container overhead.

DIMENSION	FRR (C — common RIB brain)	SONiC (C++ / Redis)	OcNOS (Pure C / HAL/PAL)	Arrcus (Modular / SAI Direct)
Control plane lang	Pure C — Patricia tree + nexthop linked list	FRR (C) inside Docker; SWSS orchagent in C++	Pure C — ZebOS heritage; struct + fn pointer arrays	C/C++ multi-threaded routing engine; ground-up
HW abstraction	Netlink (host) or FPM TCP (SONiC/Arrcus)	SAI (OCP) via syncd container	HAL/PAL proprietary C direct call	SAI direct (no Redis mid-layer)
FIB path hops	2 (zebra → Netlink/FPM)	**7 (FRR→fpmsyncd→Redis→orchagent→ASIC_DB→syncd→SAI)**	**3 (daemon→PAL→SDK)**	**3–4 (engine→SAI adapter→SDK)**
Node convergence	N/A (control plane only)	**60–180ms (Redis pipeline under load)**	**~40ms (HAL/PAL direct C)**	**~45ms (SAI direct, no Redis)**
SRv6 uSID FIB write	uSID sub-struct pointer chain in nexthop	C++ EVPN/VRF object graph → multi-stage Redis → SAI	Linear RAM: uSID ptr chains → PAL single C call → ALPM	SRv6-TE engine → SAI → SDK; min C++ object overhead
5G vRAN fit	FRR on vDU host: SRv6 uSID via Linux Netlink	**SONiC+PINS: gRPC unifies DC fabric + host — best cloud-native vRAN**	**OcNOS: Class C/D PTP + lean monolithic footprint — best CSR timing**	ArcOS: SRv6-TE slice isolation + K8s API — strong cloud-native mid-plane
ISSU model	N/A	**Container restart; Redis state persists; HW unaffected — best isolation**	Binary state freeze → NVRAM → reload C structs; zero HW touch	In-process warm restart + RIB checkpointing

SRv6 uSID Use-Case Pecking Order

USE CASE	#1	#2	#3	KEY DIFFERENTIATOR
Core/DWDM Transport	**Arrcus**	OcNOS	SONiC	Arrcus: fastest BGP PIC Core, best native SRv6-TE. OcNOS: MPLS legacy interop. SONiC: Redis lag under heavy churn
AI DC Backend Fabric	**Arrcus**	SONiC	OcNOS	Arrcus: adaptive SRv6-TE for GPU flows Dragonfly. SONiC: hyperscale cost/community. OcNOS: telco DNA mismatch
Standard DC Clos (EVPN)	**SONiC**	Arrcus	OcNOS	SONiC: community, cost, automation, cloud-native toolset. Arrcus: technically superior but premium cost. OcNOS: SP DNA

5G vRAN / vCU / vDU	OcNOS	Arrcus	SONiC	OcNOS: Class C/D PTP, ~40ms convergence, lean CSR footprint. SONiC: 20+ containers = timing noise
Cloud-Native Open vRAN	SONiC+PINS	Arrcus	OcNOS	SONiC+PINS+FRR: single gRPC controller programs DC fabric + vDU host as one SRv6 domain
Cell Site Router / ULH	OcNOS	Arrcus	SONiC	OcNOS: lean monolithic C, best PTP/SyncE stability, DNX match. SONiC: too heavy for power-constrained HW

"All three NOS platforms support SRv6 uSID. The pecking order is determined by how fast each crosses the software/hardware boundary under failure conditions."

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

■ **Frequently Asked: Does NX-OS support SRv6 uSID?** 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 for uSID at line rate — hardware constraint. Nexus 9800 with Silicon One (NX-OS 10.6(2)F+) natively supports SRv6 uSID. Upgrade from 9300 to 9800 requires hardware replacement.

■ **NX-OS Cloud Scale ASIC (Nexus 9300): Full 128-bit SRv6 SID ONLY — uSID NOT supported. Nexus 9800 Silicon One: uSID natively supported (NX-OS 10.6(2)F+). Hardware replacement required.**

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 Numbered Tables

Tables T1–T5 below provide the definitive silicon hop-depth reference. Formula: uSID carriers × 6 = total uSIDs without recirculation (segment-routing.net MPLS WC 2022 demo methodology). Key reference: Rakuten/Bell Canada/Cisco, 26-uSID line-rate demo, 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: uSID carriers × 6 uSIDs/carrier = 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 segment-routing.net 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-×-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) *HW abstraction:* 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 *HW abstraction:* 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 *HW abstraction:* 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 *HW abstraction:* 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 *HW abstraction:* 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 *HW abstraction:* 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 *HW abstraction:* 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 *HW abstraction:* 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.

Ready to Map Your Open NOS Architecture to Your Hardware?

SRv6 TEK Innovations provides vendor-neutral architecture assessments: LCD baseline audits, NOS selection, convergence benchmarking, and digital-twin validation against your exact hardware inventory.

NAICS: 541512 · 541519 · 541611 · MBE-Eligible Minority-Owned Small Business