

AI Backend: Deploying SRv6 uSID and SONiC for Deterministic Load Balancing

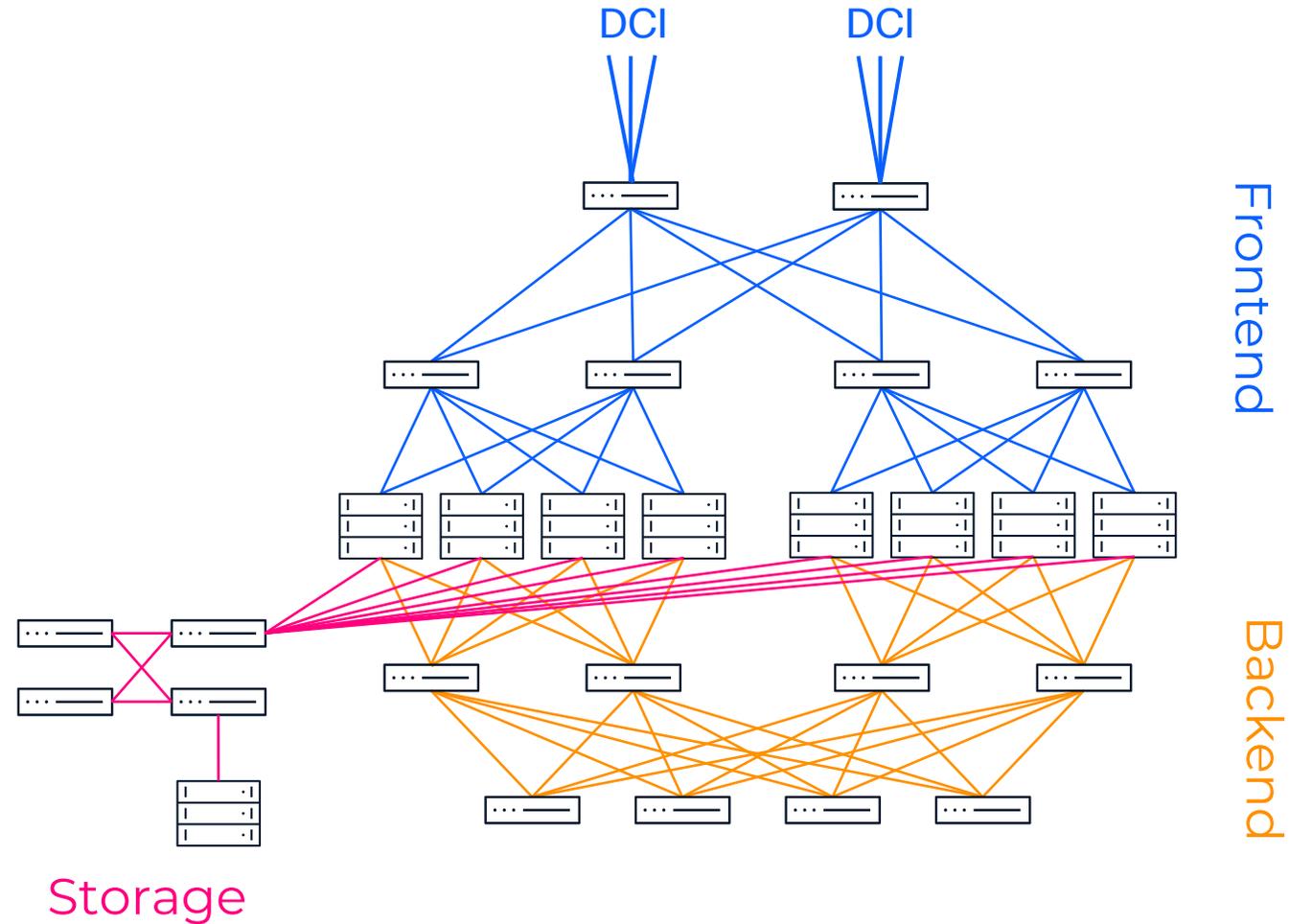
NANOG96

Pablo Camarillo – Cisco

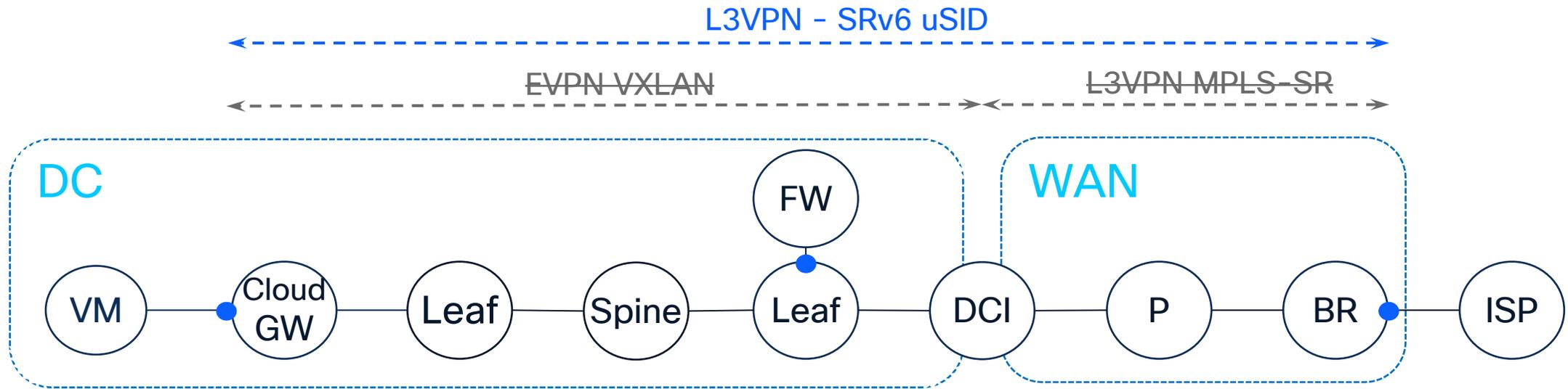
Rita Hui – Microsoft

AI DC

- Frontend:
 - Compute
 - Inference
- Backend:
 - Inter-XPU
 - High-throughput (1:1 oversubscription)
 - Low jitter
- Storage:
 - Moves large datasets between storage systems and GPU servers
 - Load initial dataset for training jobs.
 - Periodic Data Snapshots.



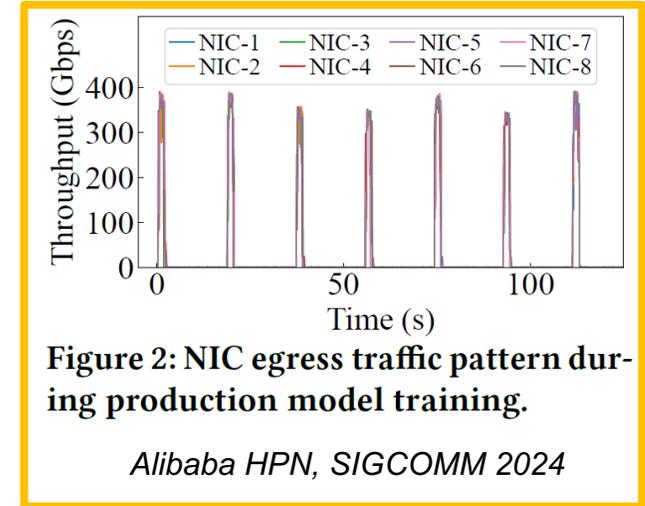
Frontend: SRv6 converged DC/WAN



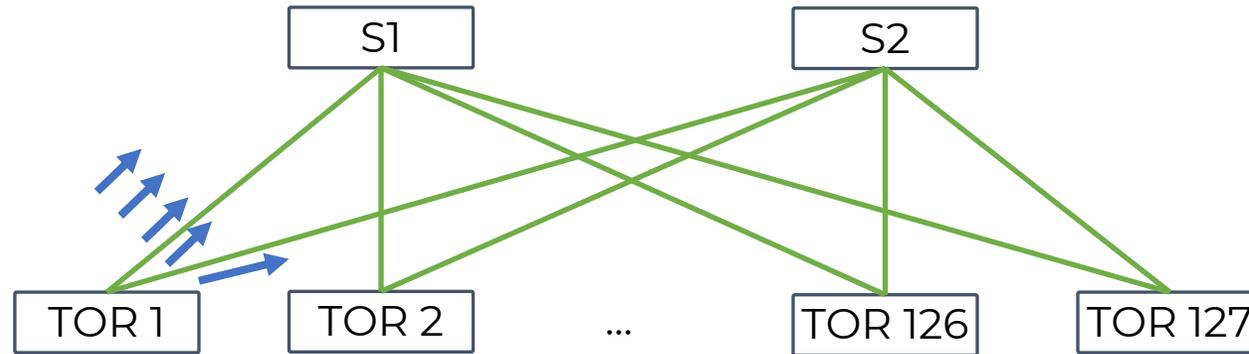
- SRv6 uSID converged end-to-End design from the DC Front End, through the metro, up to the Peering
 - Including FW service insertion
- Replaces the legacy design with VXLAN in the DC and SR-MPLS in the metro network.
- Alexey Gorovoy from Nebius ([recording](#))

Backend: AI Training Traffic

- Long lasting (weeks)
 - Highly synchronized
 - Periodic
 - Bursty, maximum link capacity
 - Low Entropy
 - Predictable!
-
- Failures of communications in LLM training are costly
 - An epoch of training is blocked until the synchronized collective communications of last epoch finishes
 - If an ongoing job crashes, all progress since the last checkpoint is lost (i.e., repeat last 15-30mins of computation)
 - META Llama3 (406B) training: 6k GPUs over 54 days; 419 unplanned interruptions *

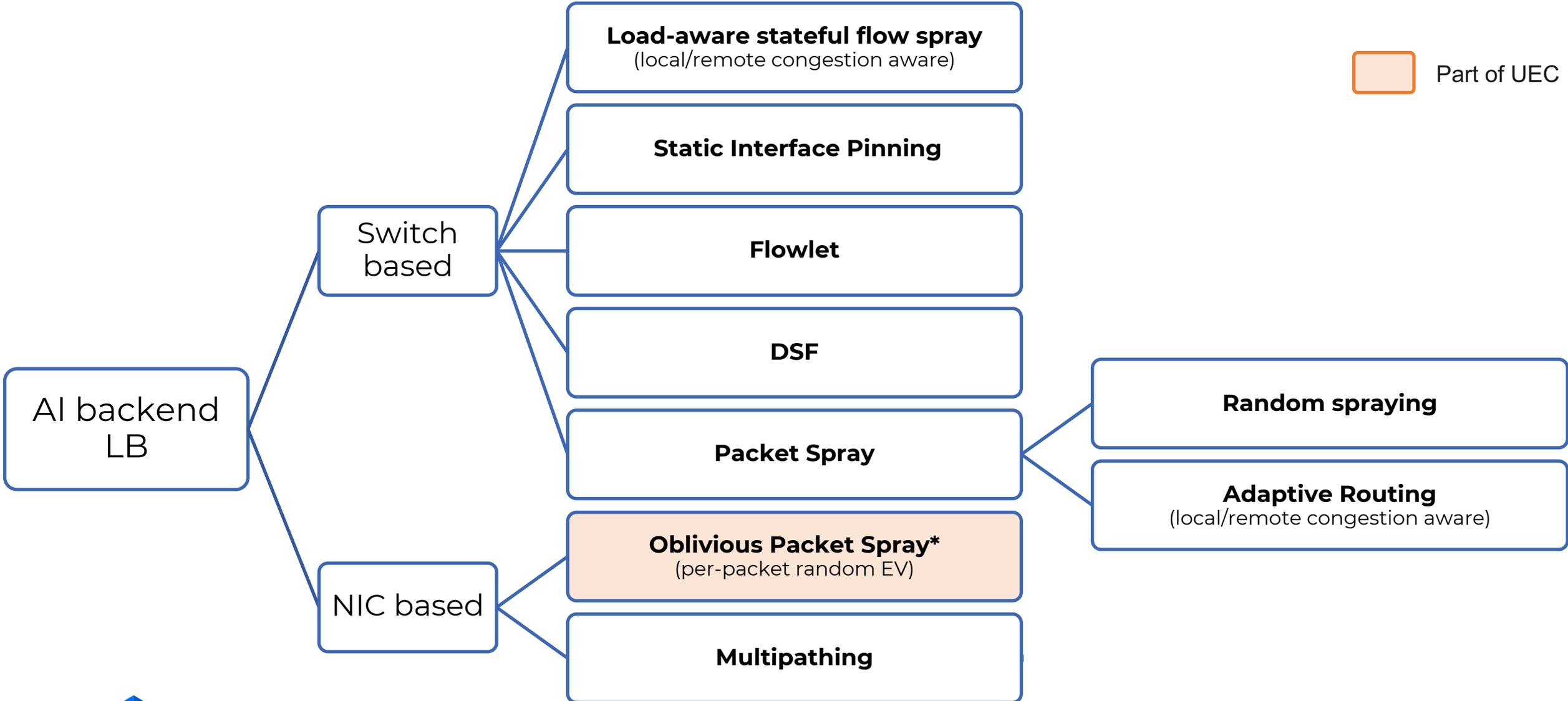


ECMP is fragile!



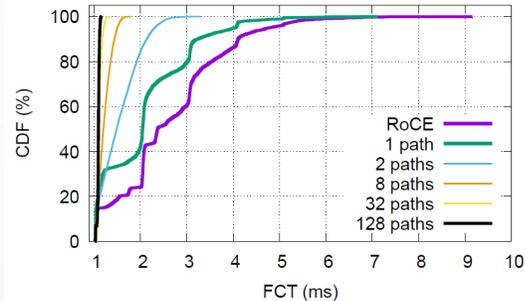
- As number of QPs is small, traditional ECMP load-balancing results in load-imbalance (hash collision).
- Leading to congestion, packet loss, and GPU halted waiting for retransmission.
- Many different efforts to overcome it.

Many attempts to solve the LB problem



The Need for Transport Multipathing

Ultra Ethernet Transport: a multipath replacement to RoCEv2.
Need multipath for pipeline parallelism



Takeaway: multipath is key to reduce tail FCT for ML training.
Flows start at line rate, but don't like PFC. Packet trimming being standardized.

Costin Raiciu, Load Balancing Approaches in AI/ML Networks, IETF123, July 2025

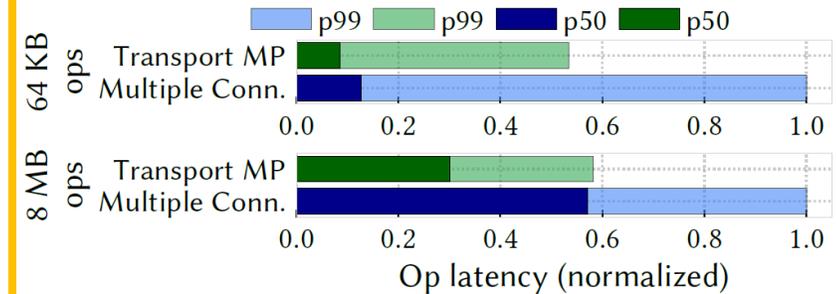


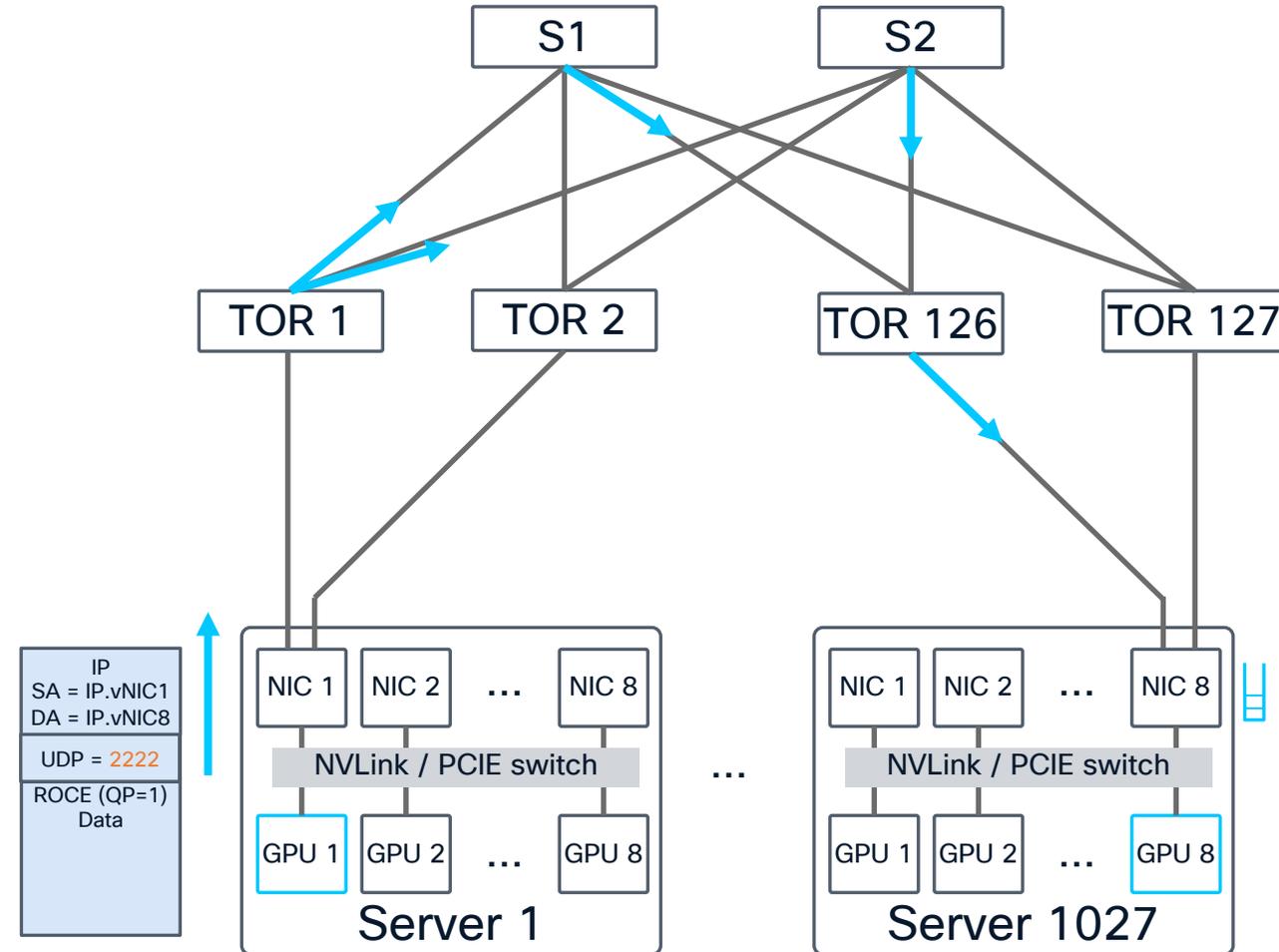
Figure 3: Multipathing benefits ML Workloads

Falcon: A Reliable Low Latency Transport, Google, SIGCOMM 2025

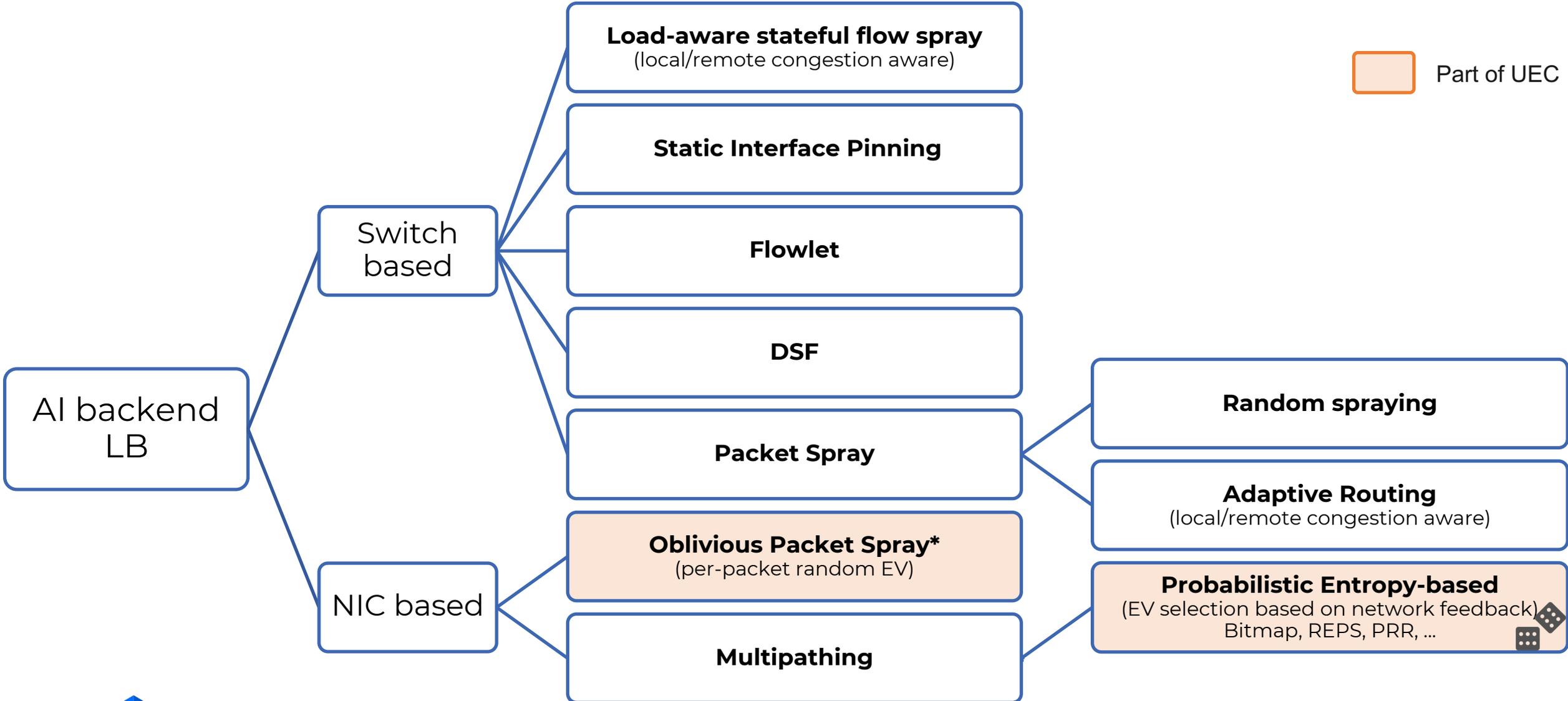
- Multipathing uses multiple paths for each QP
 - Host maintains state for each path.
 - Feedback loop and balancing done for each path.
- Typically, one single congestion window across all paths (to reduce RNIC overhead)
 - Some methods use a dedicated congestion window per path

A Probabilistic approach to Multipathing

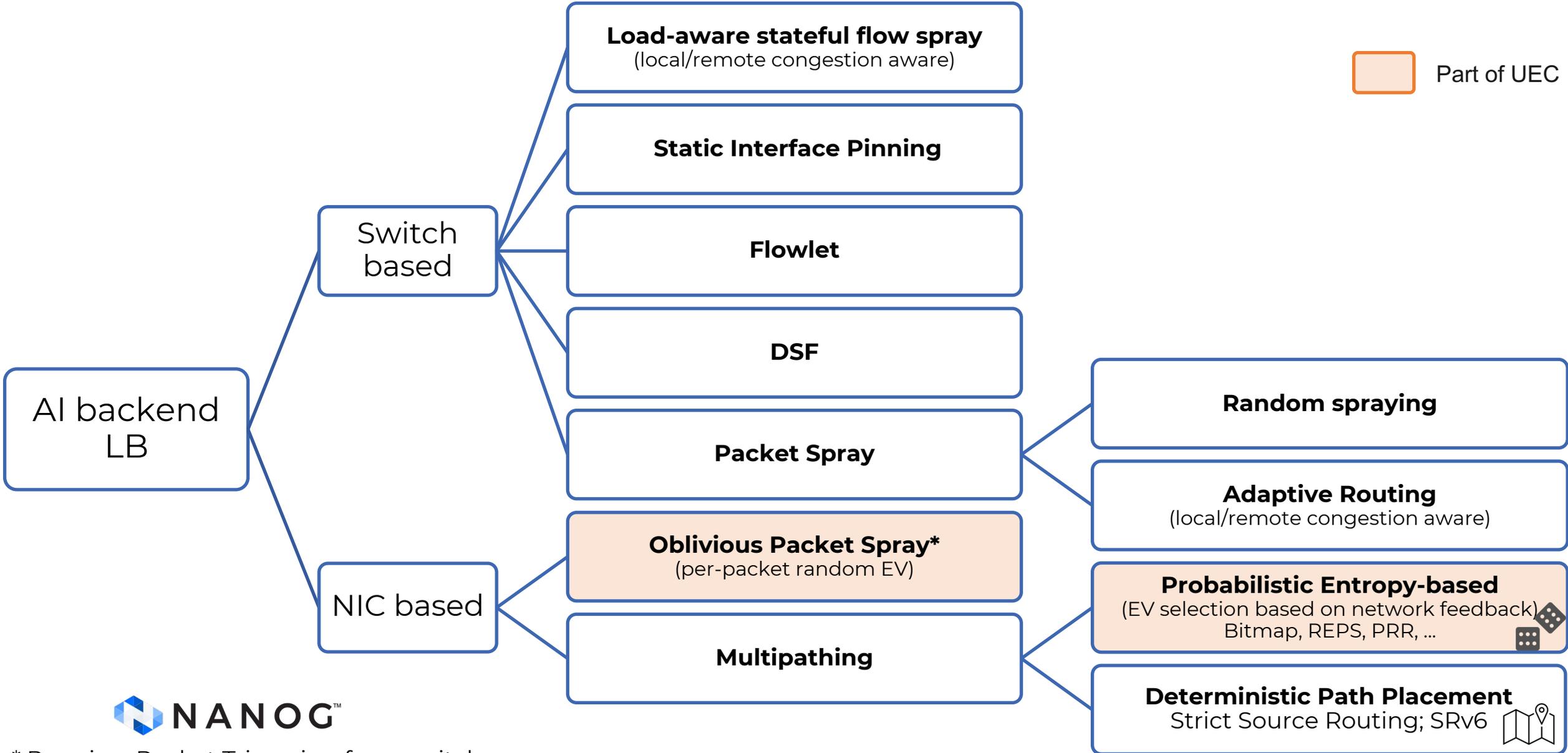
- Select a few **random EV** (Entropy Value)
 - Transmit data
 - Monitor the performance of a QP (e.g., RTT, ECN, Packet Trimming)
- **IF there's a performance degradation THEN change the EV** randomly
 - ...in hope to obtain a different path
- **Issue:** no guarantee that the new EV will be hashed through a different path in the fabric
- **A more refined version: predict** how EVs are hashed by intermediate switches
- **Issue:** Requires sticky load-balancing upon link failure (LB group change)



Many attempts to solve the LB problem



Many attempts to solve the LB problem



A deterministic approach: Source Routing

- GPU is in control of the end-to-end path
- Essence of SR: the application controls the end-to-end path as a network program in the packet header
- **Strict Source Routing**: full sequence of links the packet traverse
 - Traffic follows the explicit path step-by-step
 - Not exposed to ECMP
- Feedback loop is always faster at the NIC
 - Leverage of ECN and Packet Trimming (for congestion losses)
 - Change of path from is immediate (no state propagation through fabric)

SRv6 rich-ecosystem

Open-Source Networking Stacks

Network Equipment Manufacturers



Merchant Silicon



Smart NIC / DPU



Open-Source Applications



Partners



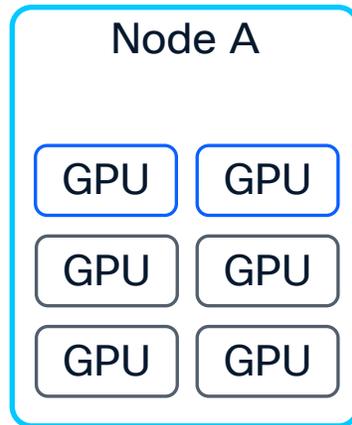
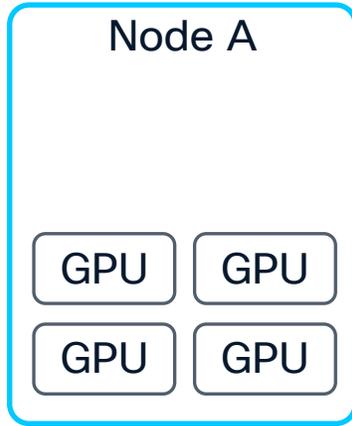
Failure resiliency

- An IP fabric with dynamic routing protocol is problematic for Backends
 - Assume 10ms to propagate a link flapping and ECMP group update in FIB
 - Assume 4KB MTU, 400Gbps link fully utilized
 - ...over 120,000 packets (~0.5GB) lost plus all related congestion control
- **Strict Source Routing is not ECMP exposed**
- **SRv6 can be deployed on a statically provisioned fabric**
 - **No dependency on dynamic routing protocols**

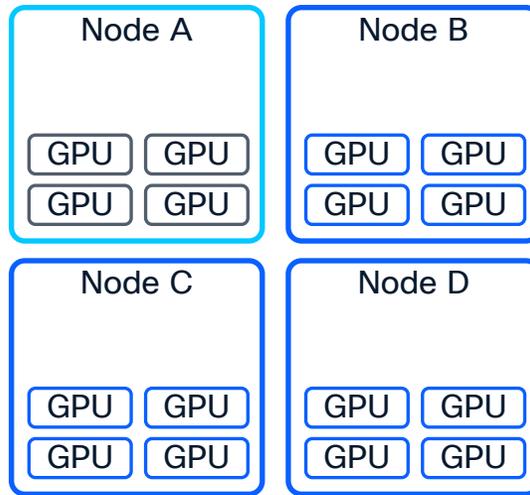
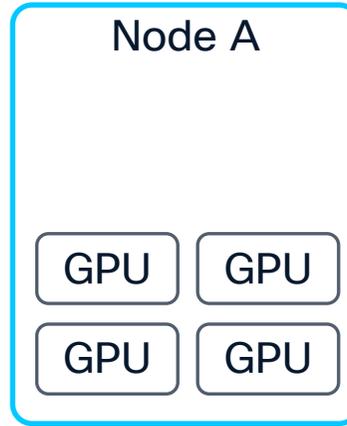
Virtualization

- SRv6 policy fulfills virtualization requirements in AI Backend:
- Efficient GPU Overlay/Tenancy separation
 - No MTU gain
 - No expensive DPU cycles
- Slicing of the backend
 - Multiple tenants in the same shared AI Backend
 - No impact by “noisy” neighbors

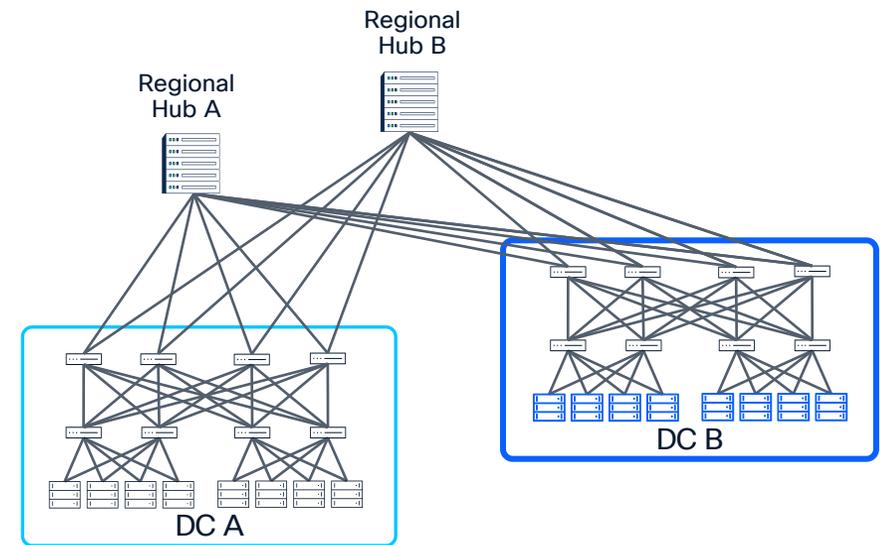
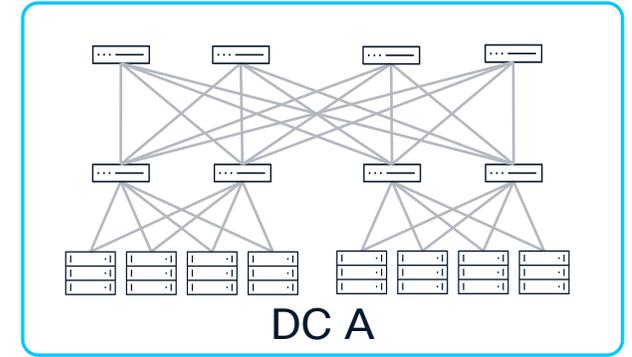
Scale Up



Scale Out

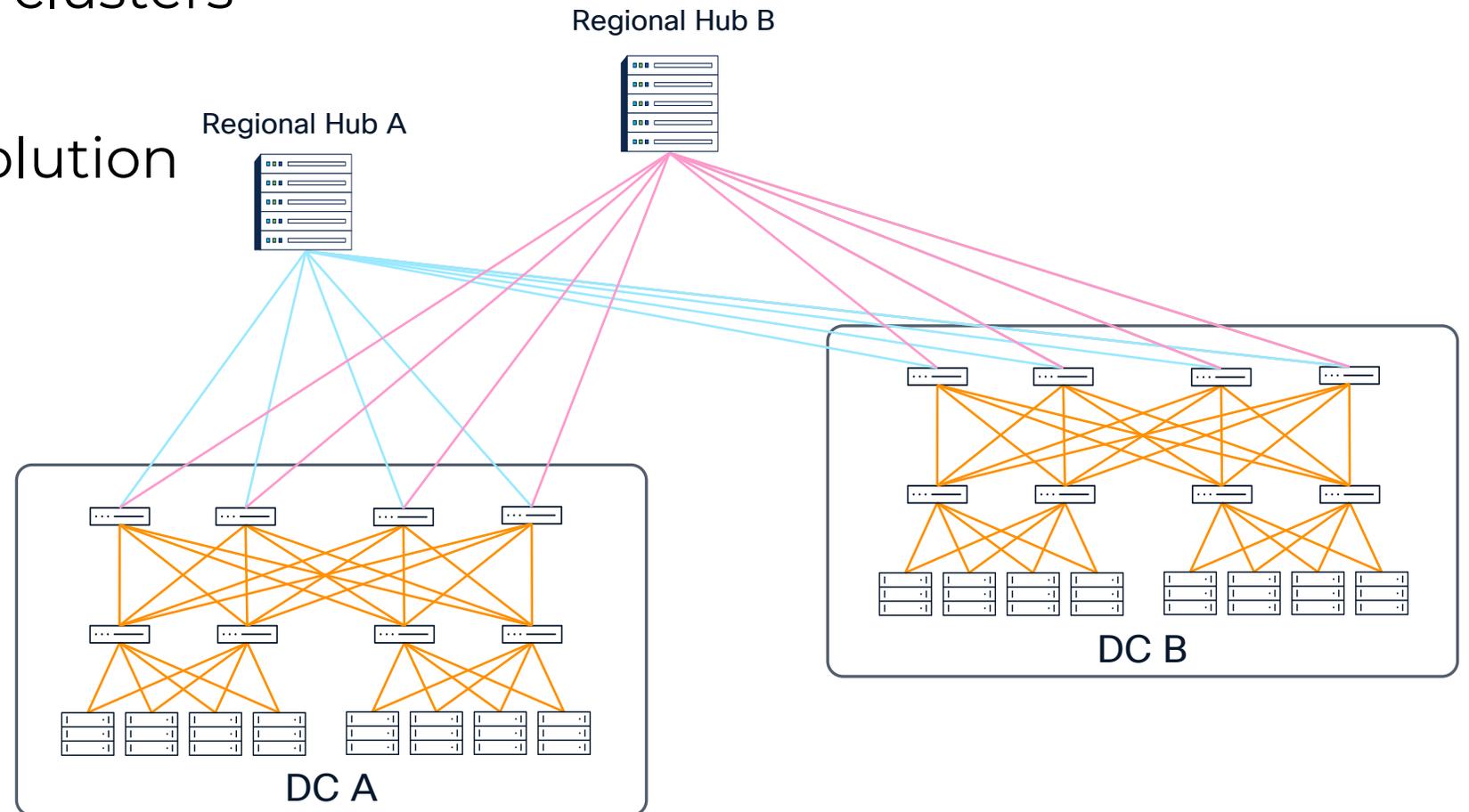


Scale Across



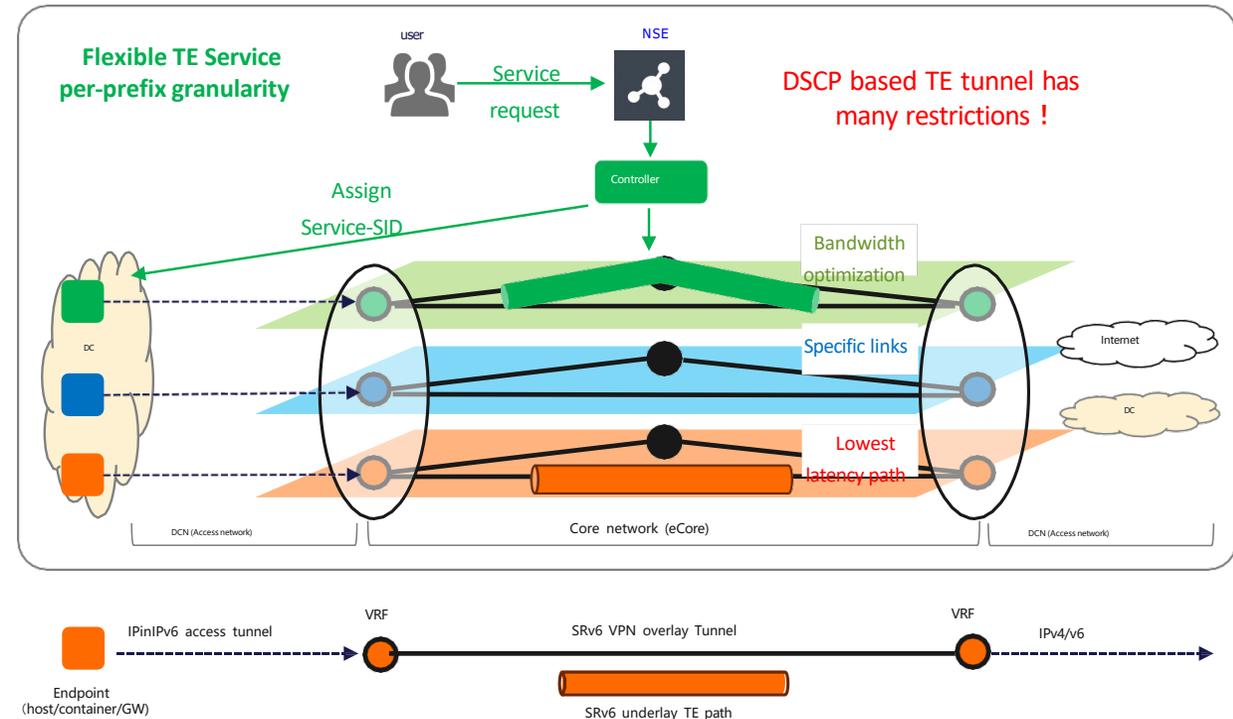
Scale Across: It's a TE Problem!

- The topology is less symmetric
- The capacity is more scarce
- Shortcuts between clusters
- SRLG's
- SR is the obvious solution



SRv6 DCI at Alibaba

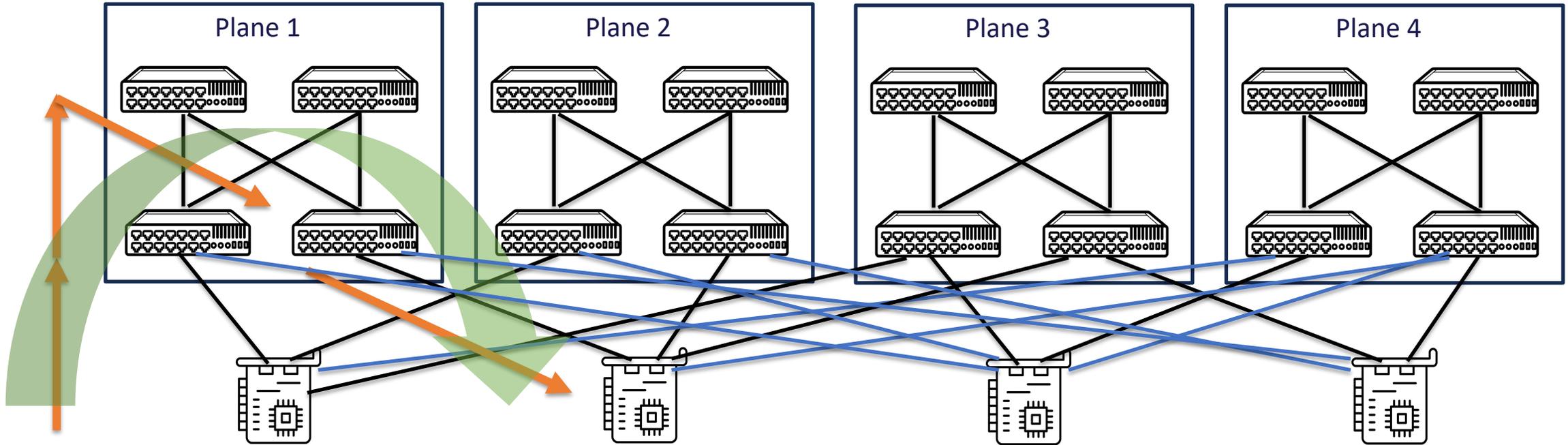
- eCore = Alibaba's Data Center Interconnect
 - Whitebox running SONiC
 - Multi-plane (redundancy)
 - Multi-Space (capacity scaling)
 - Multi—Routing domain (failure isolation; small blast radius)
- Service Aware Traffic forwarding with SRv6
- BGP Underlay Routing + ISIS (per domain)
- Presented at OCP 2025 ([link](#))



Microsoft: Backend Challenges

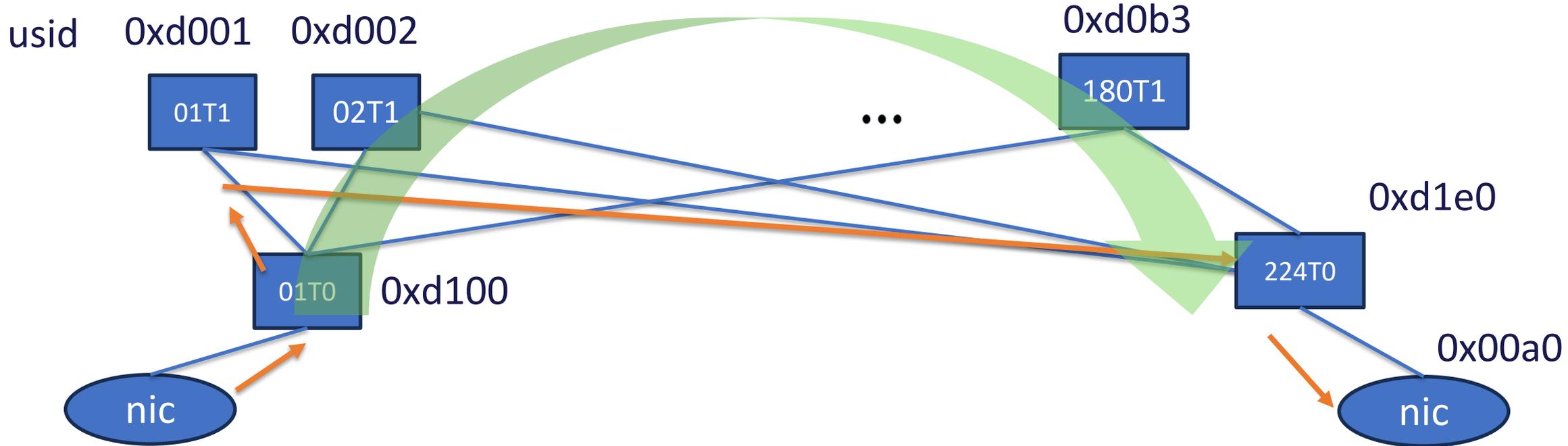
- The solution must be cost-effective and scalable
 - No Proprietary Technology
- Traditional passive hash-based load balancing mechanisms suffered from low entropy problem.
 - Need more active traffic engineering
- Failures is inevitable at this scale
 - Fast failover is necessary
- Multi-path transport is desired for efficient bandwidth utilization
 - Demand for fine-grained path control

SRv6 in AI Backend Network



- Provides fined-grained network control based on source routing
- Enables path enumeration for traffic management
- Integration with AI workloads flow scheduling provides optimal network performance
- Allow source to quickly reroute upon path failures or congestion

Failure handling with uSID

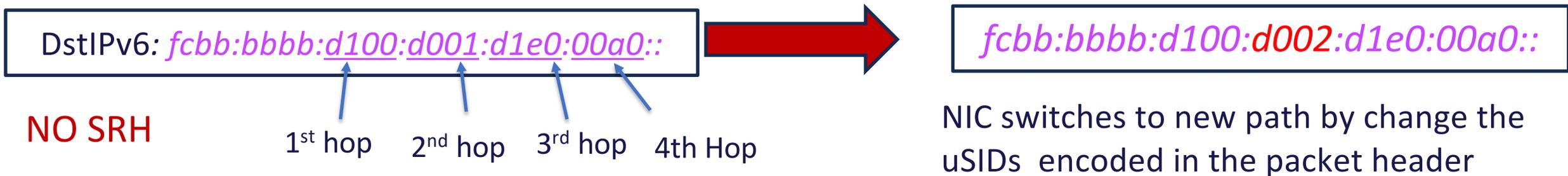
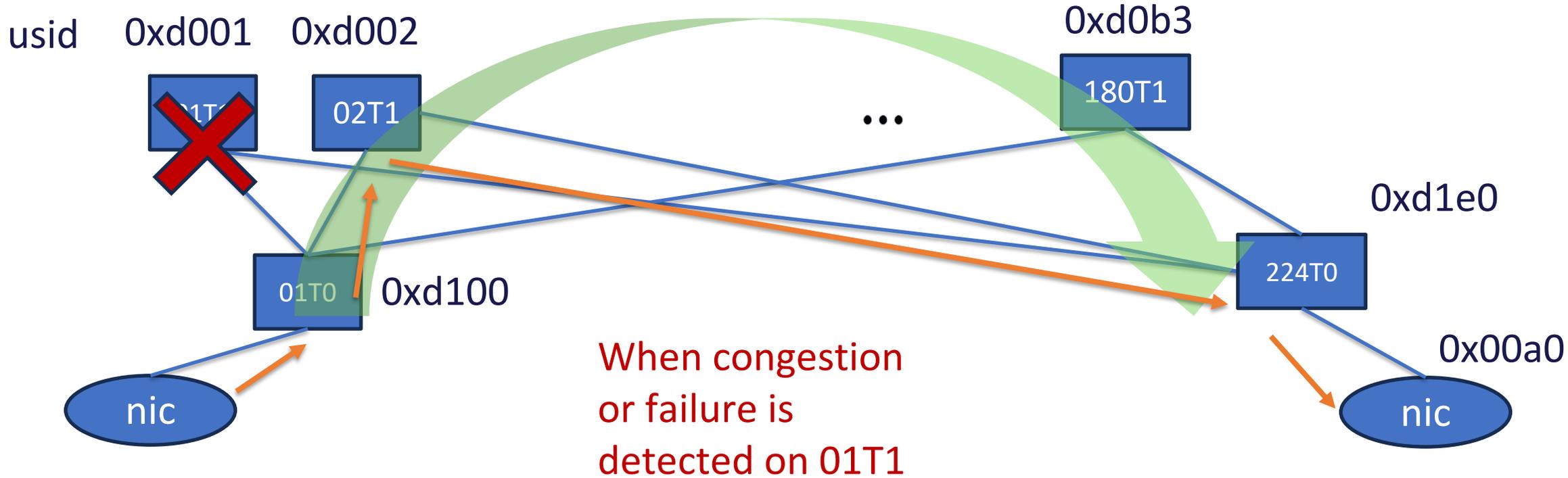


DstIPv6: *fcb:b:bbb:d100:d001:d1e0:00a0::*

NO SRH

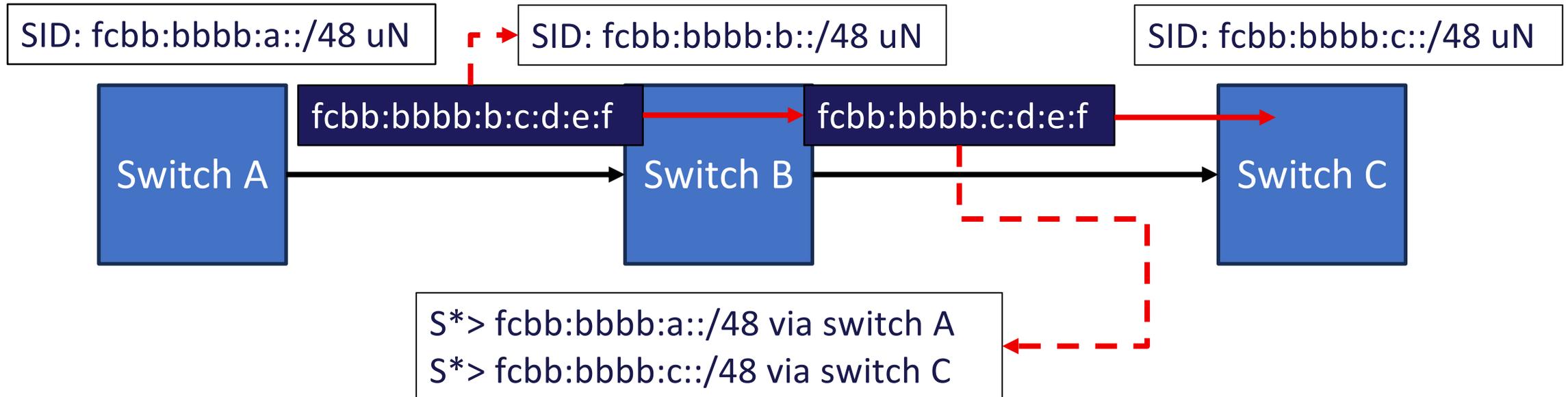
1st hop 2nd hop 3rd hop 4th Hop

Failure handling with uSID

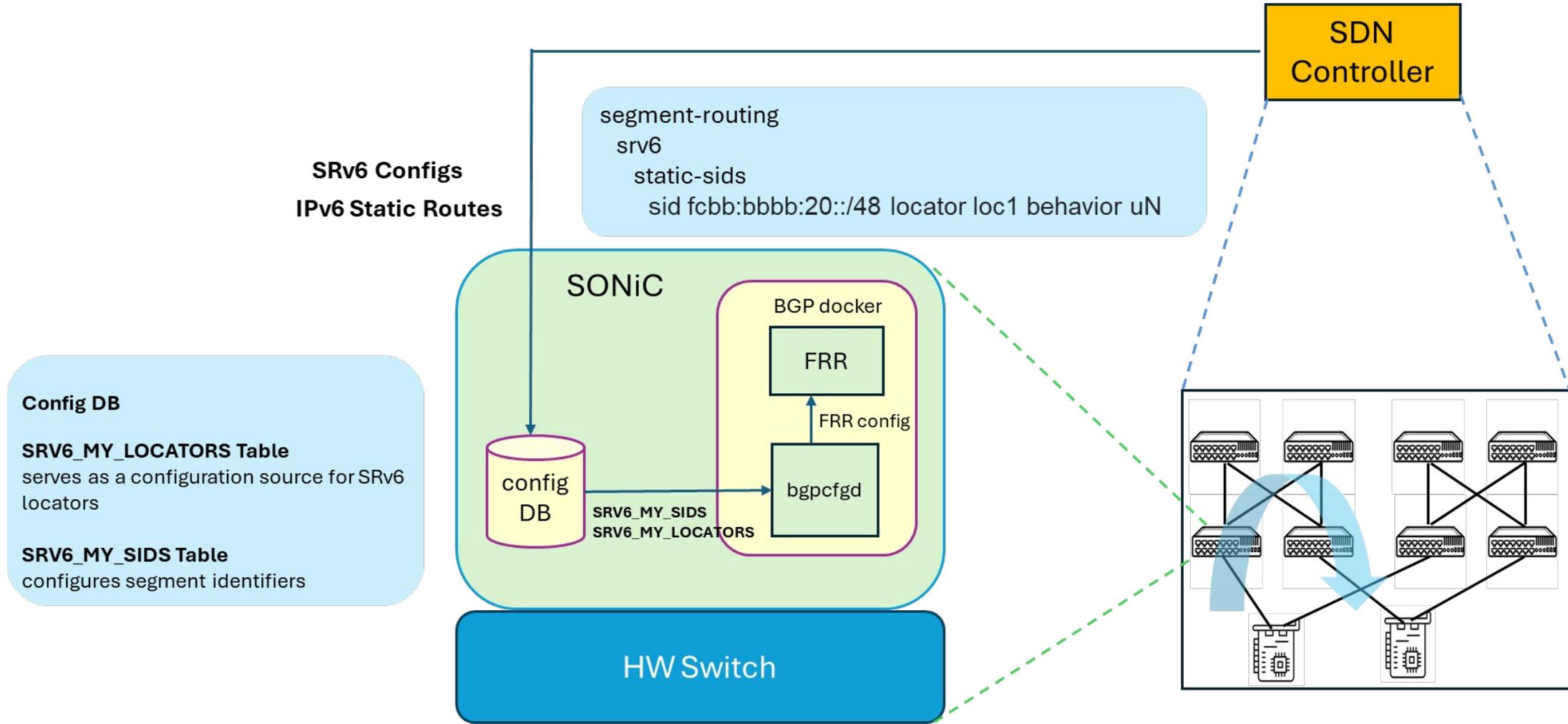


Simplify SRv6 with Static Config

- Segment Identifiers(SIDs) are configured on switches statically.
- The switch has a static route configured for each of its neighbor's SIDs.
- Hosts get the list of SRv6 paths (encoded as a list of SIDs) that it should use from the central controller.



SRv6 with static uSID in SONiC



SRv6 Ecosystem in SONiC

- Mature Support
 - uN and uDT4/6/46 functions
 - Static SID allocation and provisioning
 - Static steering of traffic with SRv6 SID list
 - BGP-EVPN based L3VPN Services.
 - Evolving with FRR routing stack
- Rich community
 - Contributors: Microsoft, Cisco, Alibaba, Broadcom, Nvidia
 - Use cases: Telecom, Enterprise, Cloud Network

Acknowledgments

Microsoft:

- Lihua Yuan
- Guohan Lu
- Riff Jiang
- Changrong Wu
- Abhishek Dosi
- Kumaresh Perumal

Cisco:

- Clarence Filsfils
- Ahmed Abdelsalam
- Carmine Scarpitta
- Jisu Bhattacharya
- Vijay Tapaskar
- Mani Veerachamy



Thank you

References

- SIGCOMM 2025:
 - A New Generation RDMA Network for Cloud AI, Alibaba ([link](#))
 - Falcon: A Reliable, Low Latency Hardware Transport, Google ([link](#))
- SMaRTT-REPS: Sender-based Marked Rapidly-adapting Trimmed & Timed Transport with Recycled Entropies, April 2024 ([link](#))
- REPS: Recycled Entropy Packet Spraying for Adaptive Load Balancing and Failure Mitigation, January 2026 ([link](#))
- Ultra Ethernet's Design Principles and Architectural Innovations, Aug 2025 ([link](#))
- @Scale 2025:
 - RDMA at Cloud Scale: The OCI Experience ([link](#))
 - (META) Scaling AI Network with DSF ([link](#))
- OCP 2025:
 - Microsoft: SRv6 for AI Backend Network ([link](#))
 - Alibaba: eCore Architecture; Alibaba's Service-Oriented DCI network ([link](#))

SRv6 - IETF

- SR Architecture – RFC 8402
- SR TE Policy Architecture - RFC 9256
- SRv6 Network Programming – RFC 8986
- IPv6 SR Header – RFC 8754
- Compressed SID Encoding – RFC 9800
- SRv6 BGP Services – RFC 9252
- SRv6 ISIS Extensions – RFC 9352
- SR Flex-Algo – RFC9350
- SRv6 OAM – RFC 9259
- Performance Management – RFC 5357