MPLS JAPAN

MPLS Japan 2025

IPv6 uSID for AI Backend NW

Teppei Kamata
Cisco Systems G.K.

October 2025

自己紹介

名前：鎌田徹平

所属：シスコシステムズ合同会社

役職：SP様向けのPre Sales SE

Interop Tokyo ShowNet NOC team member

MPLS Japanでの登壇は今回で6回目

まず初めに

- 本セッションの内容:
  AIのBackendネットワークにSRv6 uSIDを用いた事例が増えてきたので、その紹介と技術的な解説を行います

- SRv6自体の説明やAI BEネットワークの基礎についてはこれまでもMPLS Japanを始めとしたイベントで数多く発表されているので、本セッションでは解説しません

- 気になることがある方は個別にお知らせください

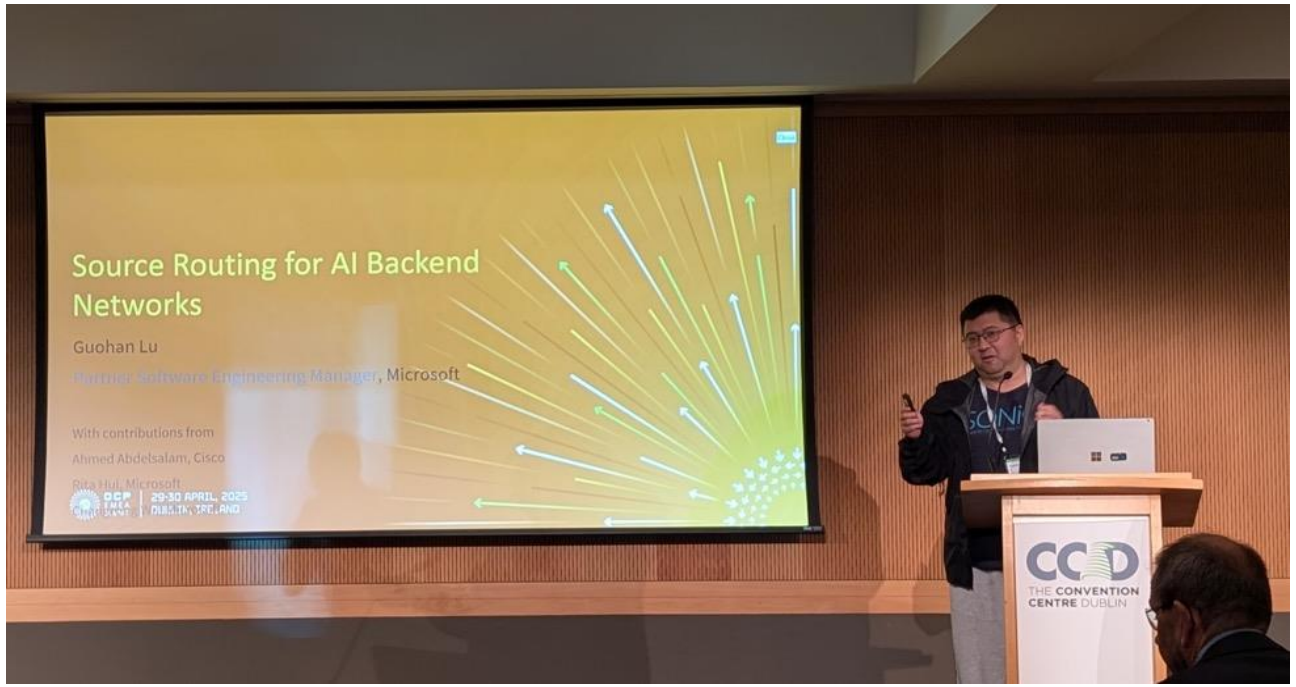
SRv6 uSID for AI @ Microsoft

"In the AI backend network, the training traffic is very bursty and each of the flow is elephant flow.

Traditional ECMP does not work for the AI backend network and it's particularly bad because the training job is highly synchronized. If flows collide on a link and it gets congested - it drags the whole job and delay the whole job significantly.

We use source routing for the backend network

SRv6 is a very perfect technology to do the source routing in the AI backend network."



Guohan Lu, AI Fabric, Microsoft. April 2025.

OCP Spring 2025 - Public Presentation Recording

Microsoft

SRv6 uSID for AI @ Microsoft

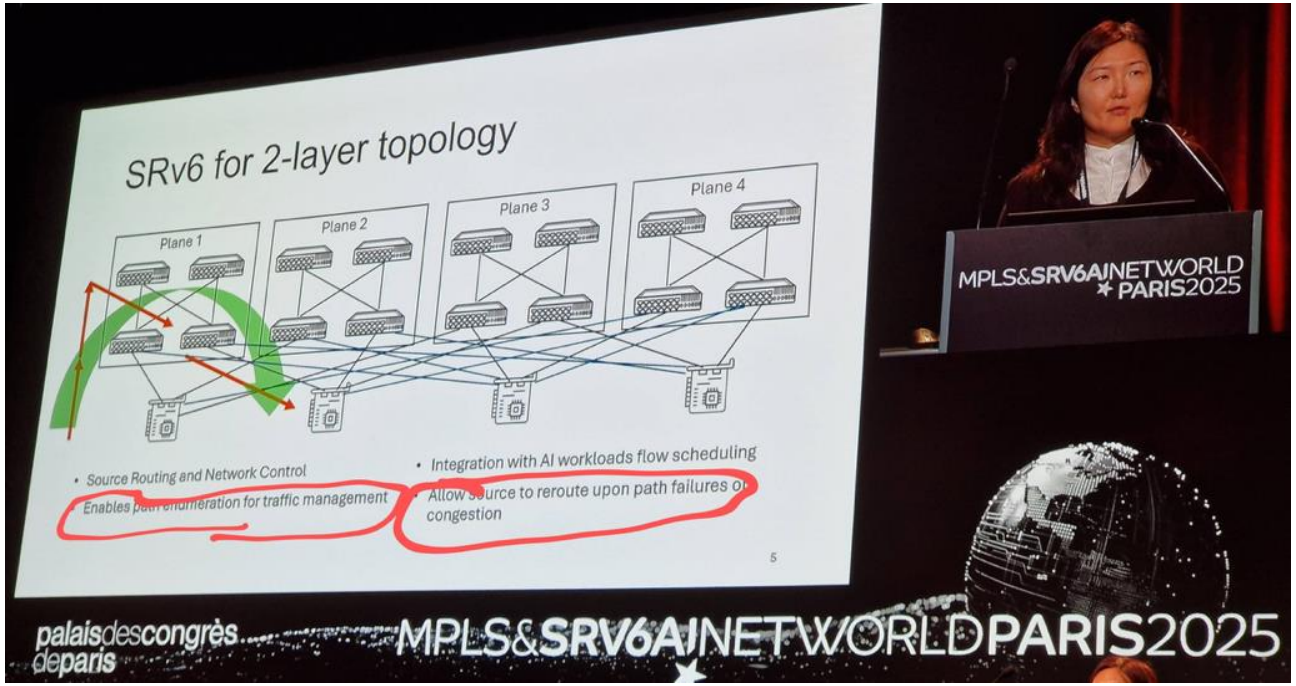
"We Like SRv6 for several reasons:

The first, the source routing capability, this is particularly useful for our users who want precise control over the traffic flows for optimal traffic performance.

The second, SRv6 enable detailed path enumeration allowing to manage the traffic for low latency requirements

The third, it enables the right coupling between AI workload flows and the scheduler.

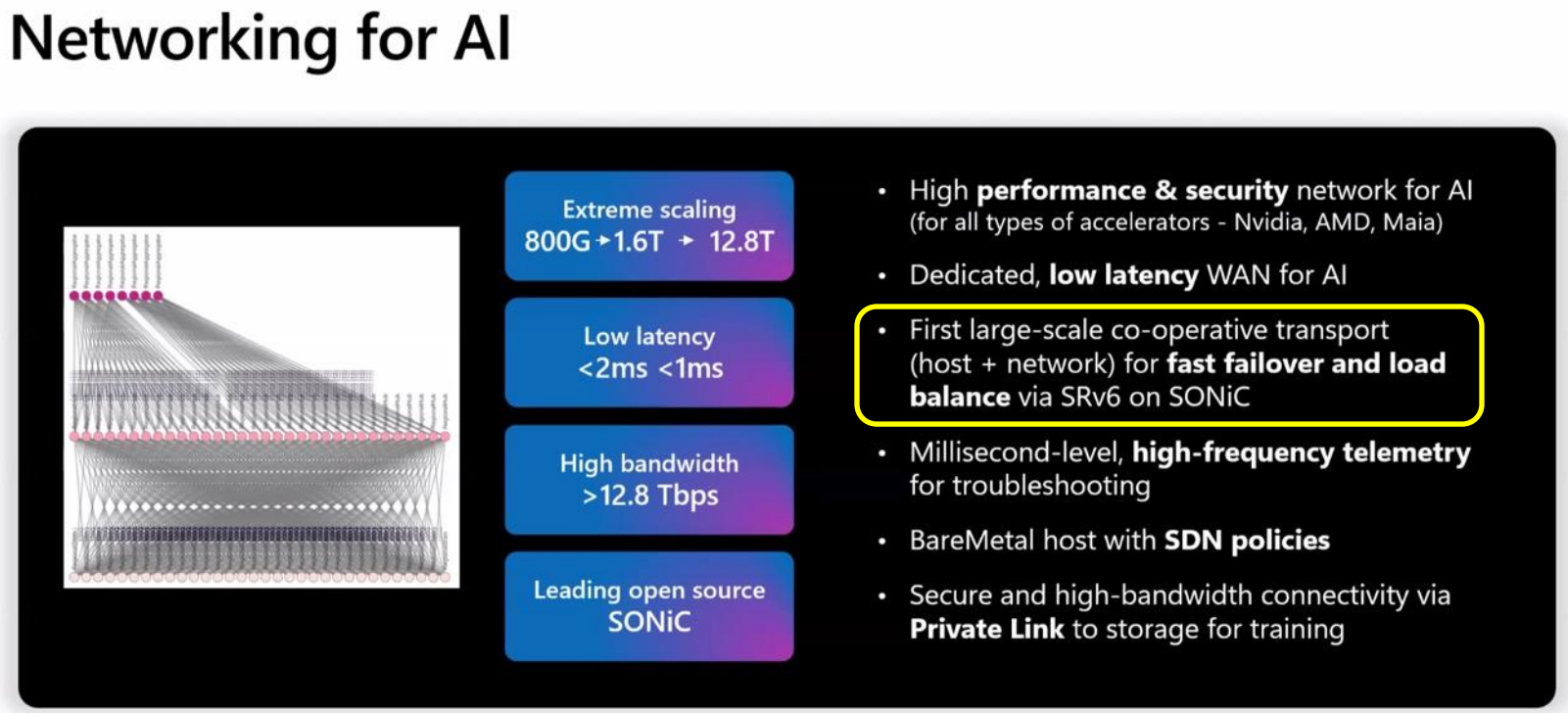
Finally, the network failure detection and respondence is at the source side. The source can recalculate the path and re-route the traffic without the dependency on the traditional routing protocols."



Microsoft

Rita Hui, MSFT AI Fabric, March 2025
SRv6 World - Public Presentation Recording

SRv6 uSID for AI @ Microsoft



- Deepak Bansal, Corporate VP, MSFT @ Hot Interconnect 2025 – Public Presentation

SRv6 uSID for AI Backed

Workgroup: SRv6 Operations
Published: 18 August 2025
Intended Status: Informational
Expires: 19 February 2026

C. Filsfils
Cisco Systems
C. Martin
Oracle Cloud
K. Pillai
IBM
P. Camarillo, Ed.
Cisco Systems
A. Abdelsalam
Cisco Systems
J. Tantsura
NVIDIA
K. Patel
Arrcus, Inc.

SRv6 for Deterministic Path Placement in AI Backends

Abstract

This document describes the use of SRv6 to enable deterministic path placement in AI backends, optimizing load balancing and congestion control for predictable GPU workloads.

Cisco, Oracle Cloud, IBM, NVIDIA, Arrcus

Benefits

* **Deterministic Path Placement**: SRv6 allows the NIC to control the path of each flow through the fabric.

* **Minimum-MTU**: A plain outer IPv6 encapsulation allows to encode 6 uSIDs in the outer DA. This implies that without the need of additional extension headers, only with 40Bytes of IPv6 encapsulation, we can encode up to 6 intermediate waypoints allowing to enforce a path in a 3-tier Clos network. This is sufficient to control a path hop-by-hop (link by link) through a leaf, spine, super-spine, spine, leaf.

* **Congestion Feedback Loop**: Instant rerouting at the source based on ECN, in-band measured One-Way and Two-Way latency, Packet Trimming feedback and in-band packet loss, without any dependency of routing protocols. There is neither any control-plane signaling involved between the GPU and the fabric, nor between the AI orchestrator and the fabric devices.

* **Standardization**: Open, vendor-agnostic implementation

* **Ease of operation**: as opposed to black-box proprietary solution which packs opaque layer-2 optimization, the SRv6 solution is minimalistic, IP based, fully standardized and a rich ecosystem (vendor, merchant and open source). The deterministic and open nature of the solution simplifies troubleshooting.

https://datatracker.ietf.org/doc/draft-filsfils-srv6ops-srv6-ai-backend/

AI Backend with IPv6 uSID

まずは背景について：AI Training Traffic

- Long lasting (weeks)
- Highly synchronized
- Periodic
- Bursty, maximum link capacity
- Low Entropy
- Predictable!

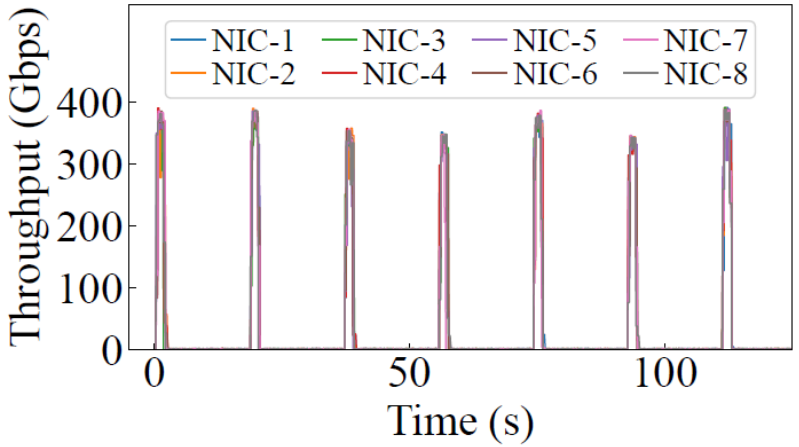


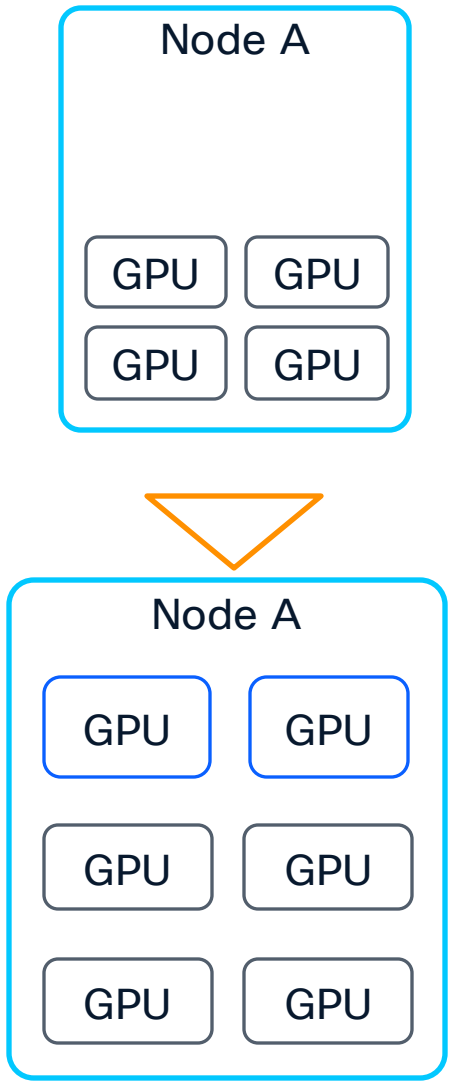
Figure 2: NIC egress traffic pattern during production model training.

- 従来のECMPベースのロードバランシングアルゴリズムはlow entropy問題を抱えていた
- AIのトレーニング中の障害はコストが高い
  - 直前のepochのCollective communicationの同期が終わるまでブロックされてしまう
  - 進行中のジョブが失敗した場合、チェックポイント以降の進捗が失われる

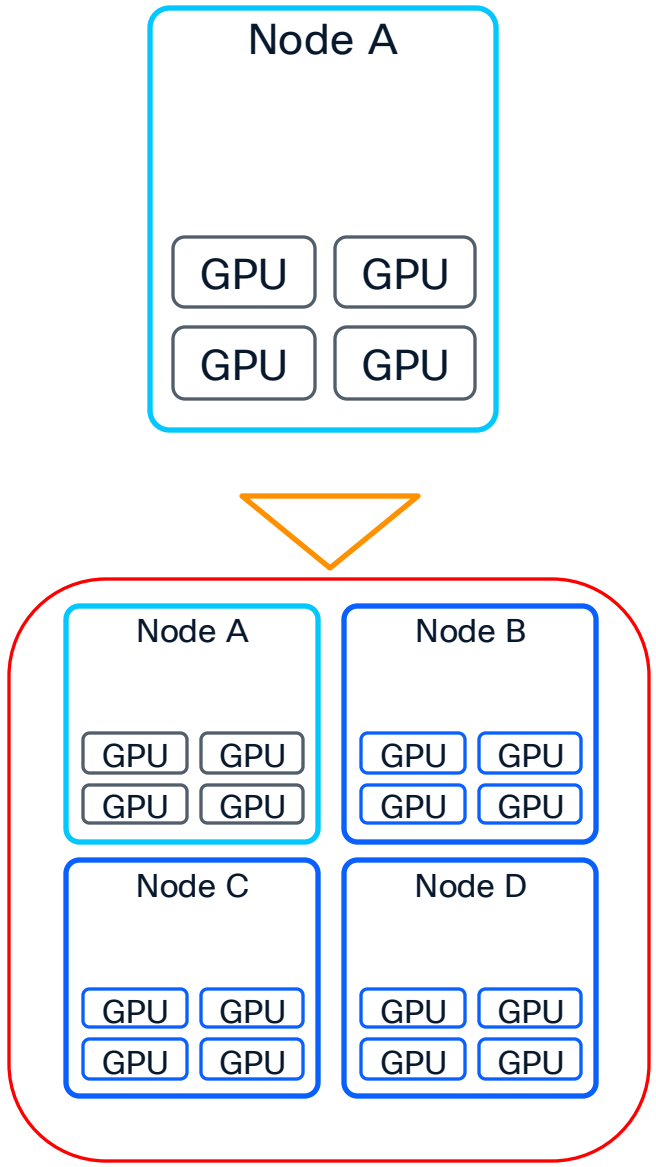
IPv6 uSIDを用いたアプローチ

- IPv6 uSIDを用いDeterministicにパスを選択
- GPUが end-to-endでパスをコントロール可能
  - Essence of SRv6: アプリケーションがnetwork programとしてend-to-endのパスを制御
  - NICレベルでFeedback loopを高速に回せる
- Ultra Scalable: ステートレスなSRv6ならでは
- Ultra Convergence: GPU（NIC）がSRv6のパスを切り替え可能
- オープンで標準化された技術
- 豊富なエコシステム
  - トラブルシュートが簡単: 標準が明確でwell-knownなSRのファンクションを利用
- さらに、uSIDはパスの選択を可能にするだけではなく、追加のヘッダー無しでマルチテナントを実現可能

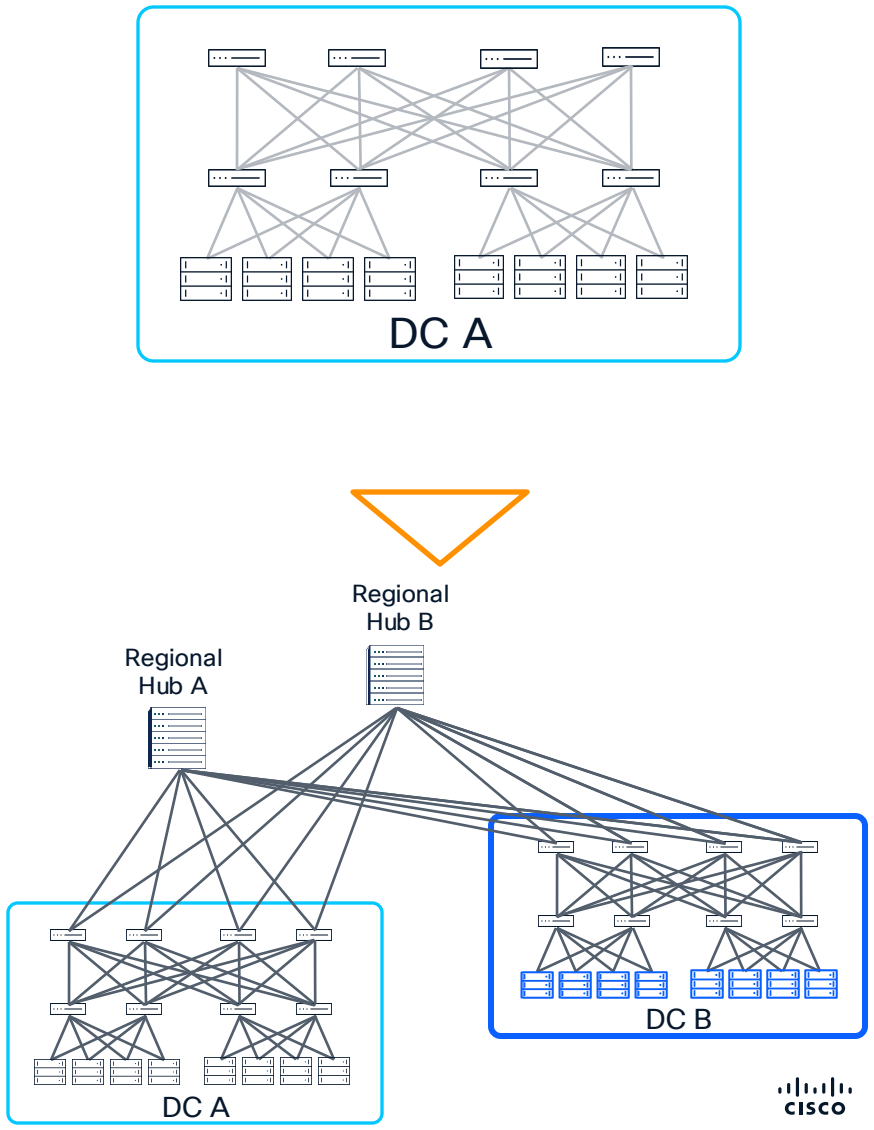
Scale Up



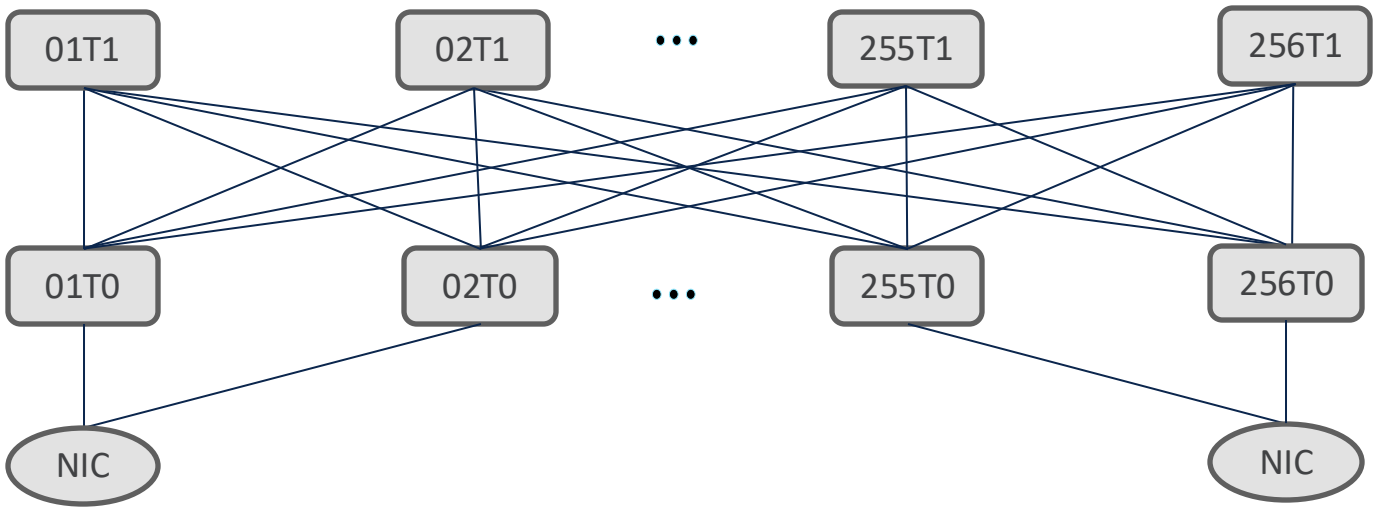
Scale Out



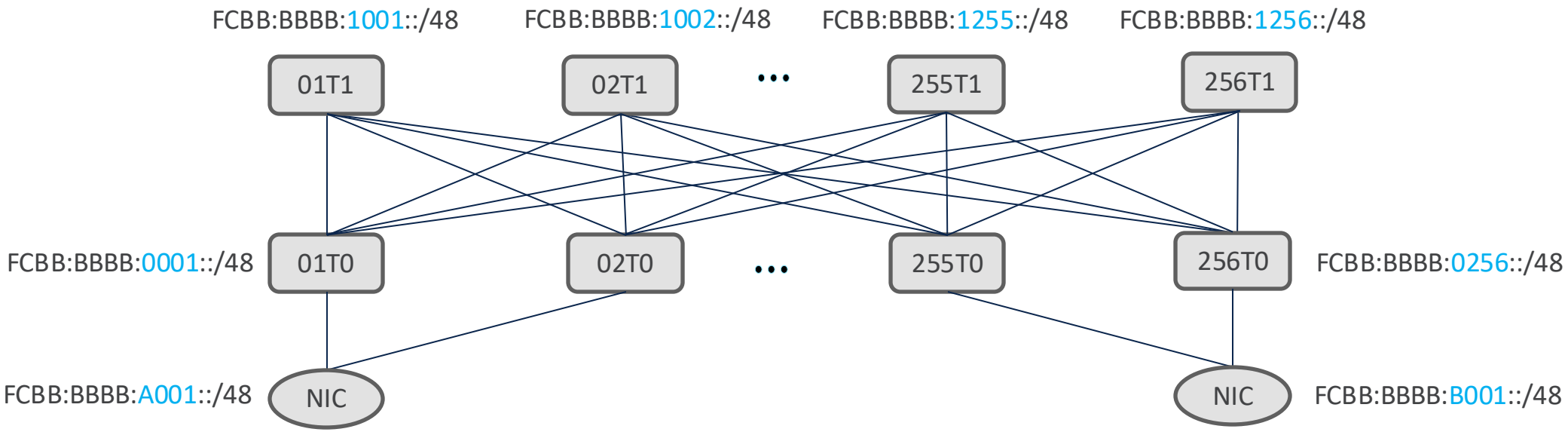
Scale Across



SRv6 uSID – Deterministic Path Placement

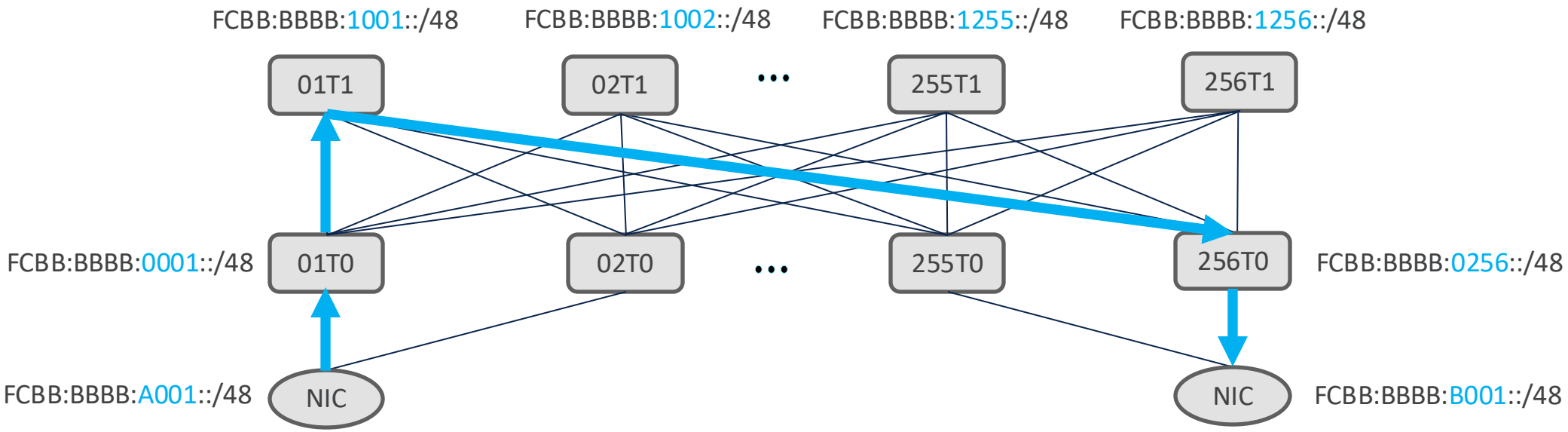


SRv6 uSID – Deterministic Path Placement



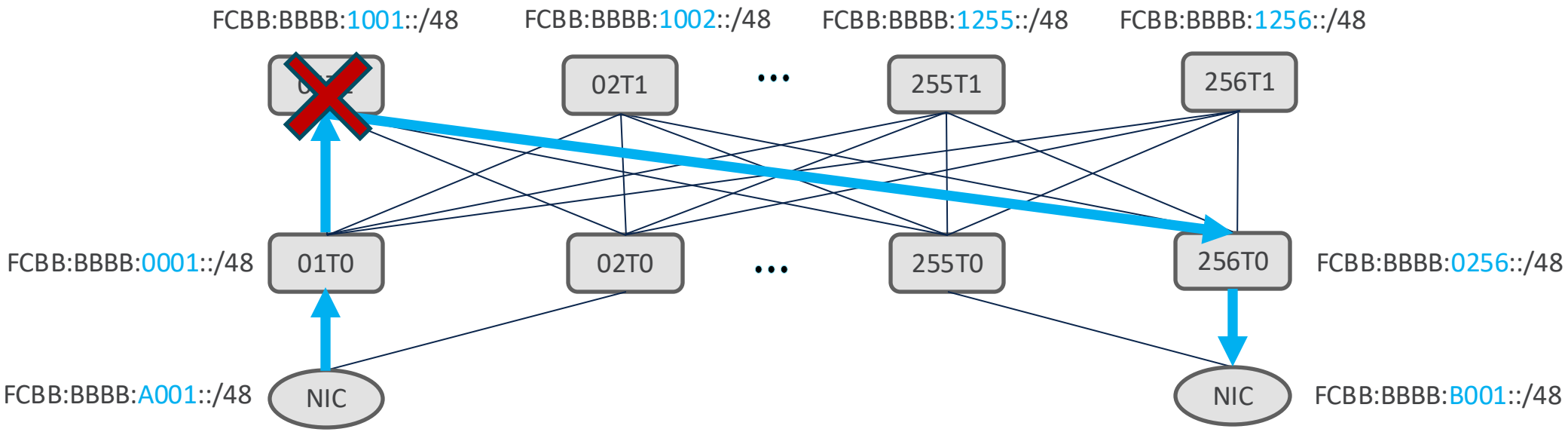
- uSID Block - FCBB:BBBB::/32
- uSID Locator - FCBB:BBBB:NNNN::/48
  - 01T0 => FCBB:BBBB:0001::/48

SRv6 uSID – Deterministic Path Placement



- NIC A to NIC B
  - Via FCBB:BBBB:0001:1001:0256:B001::
  - Go to 01T0 then 01T1 then 256T0 then NIC B

SRv6 uSID – Deterministic Path Placement



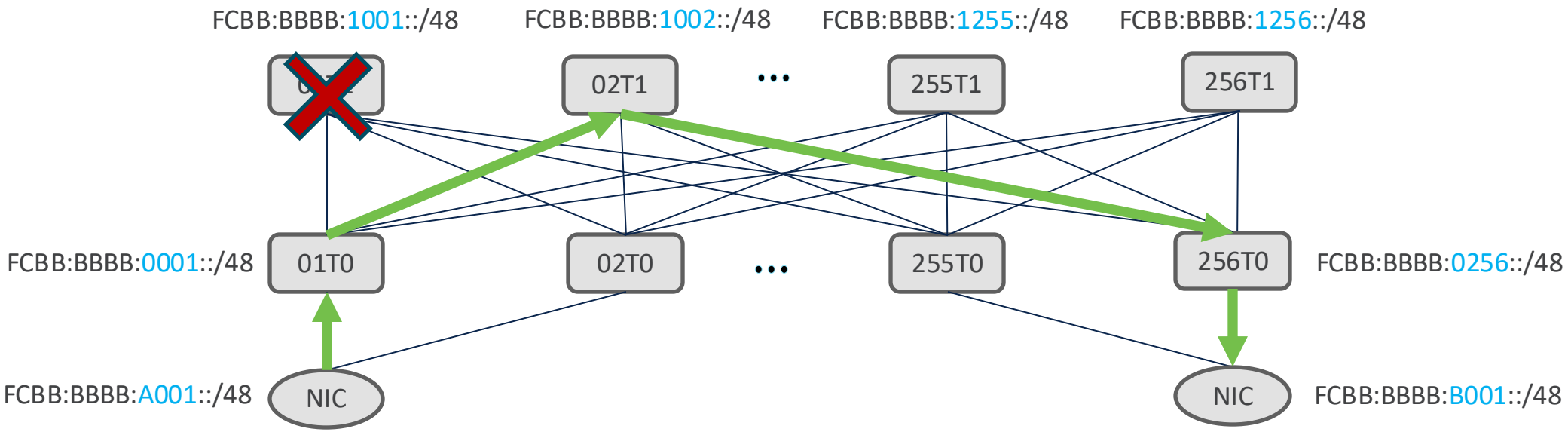
- NIC A to NIC B
  - Via FCBB:BBBB:0001:1001:0256:B001::
  - Go to 01T0 then 01T1 then 256T0 then NIC B

Congestion/Failure “ 01T1”

NICがパケットヘッダーにエンコードされたuSIDを変更することで新しいパスに切り替え

- NIC A to NIC B
  - via FCBB:BBBB:0001:1002:0256:B001::
  - Go to 01T0 then 01T1 then 256T0 then NIC B

SRv6 uSID – Deterministic Path Placement



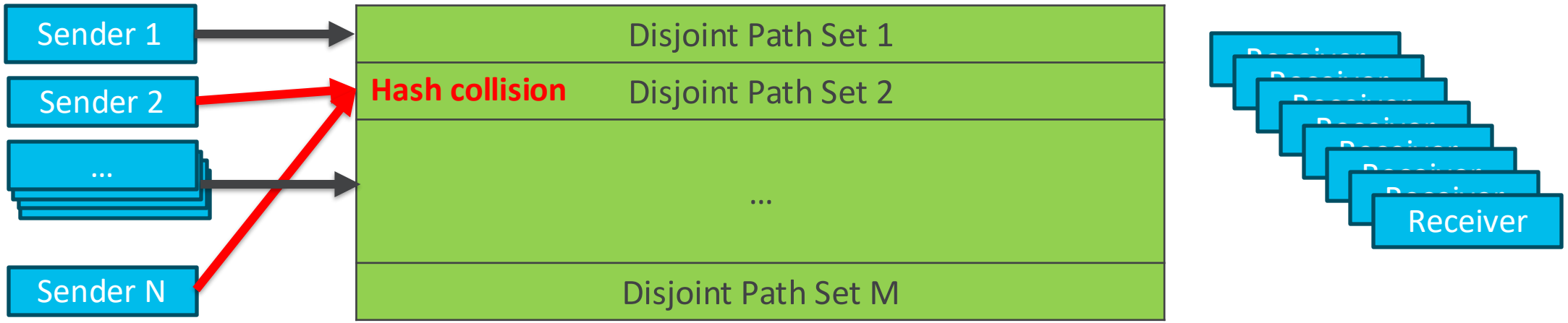
- NIC A to NIC B
  - via FCBB:BBBB:0001:1001:0256:B001::
  - Go to 01T0 then 01T1 then 256T0 then NIC B

Congestion/Failure “ 01T1”

NICがパケットヘッダーにエンコードされたuSIDを変更することで新しいパスに切り替え

- NIC A to NIC B
  - via FCBB:BBBB:0001:1002:0256:B001::
  - Go to 01T0 then 01T1 then 256T0 then NIC B

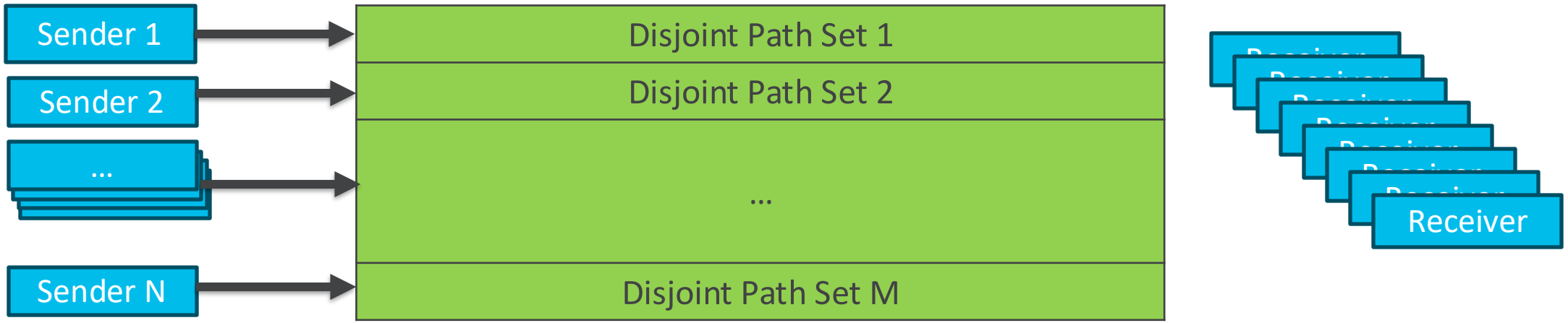
Path control via SRv6 maximize utilization
(Using incast as an example)



Case #1: using ECMP to distribute traffic

ハッシュベースのECMPでは、disjointパスが利用されることを保証できず、ほとんどのケースで経路間のロードバランスは不均一

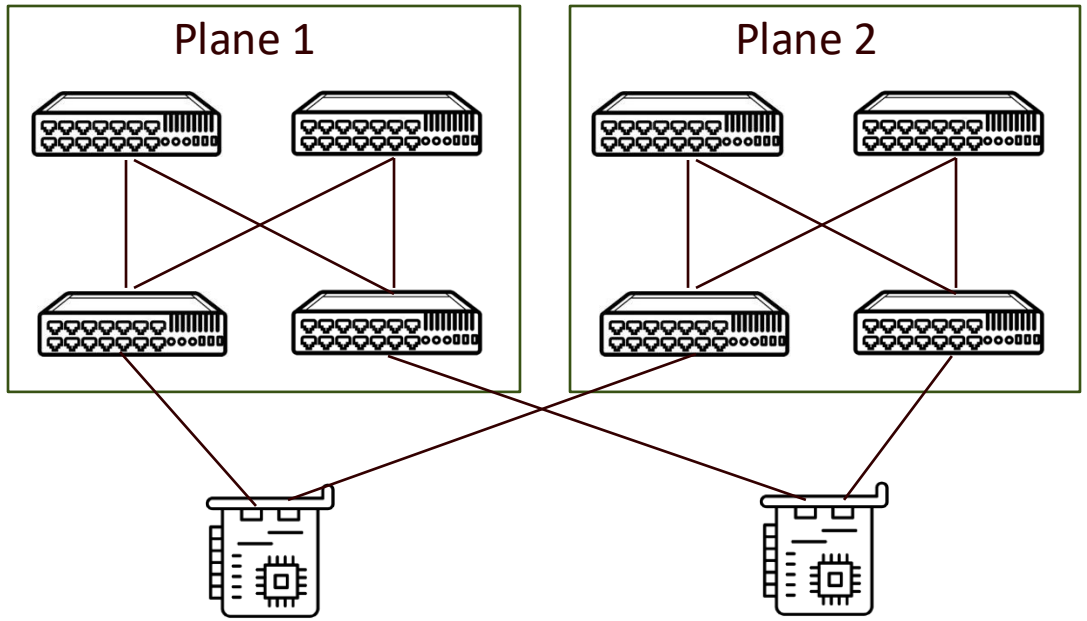
Path control via SRv6 maximize utilization
(Using incast as an example)

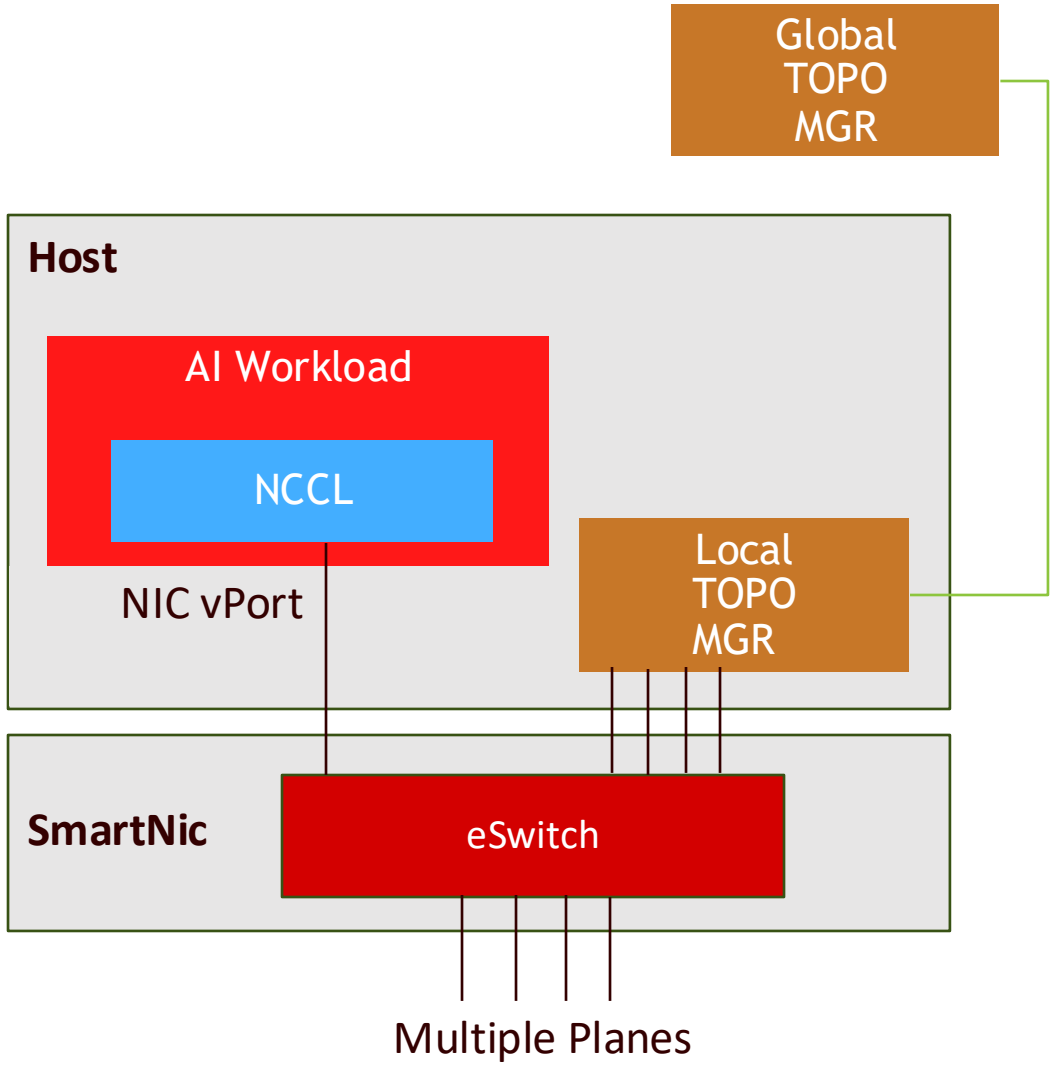


Case #2: using SRv6 to distribute traffic

SRv6ではdeterministicにパスを制御できるので、
全てのdisjointパスを利用でき、それぞれのパスがしきい値を超えない状態を実現可能

End-to-End Control





The NIC/DPU has many switch functionalities in multi-plane networks

より高度なCongenstion Controlにむけて

- 二年前に戻ります

Path Tracing の 特徴

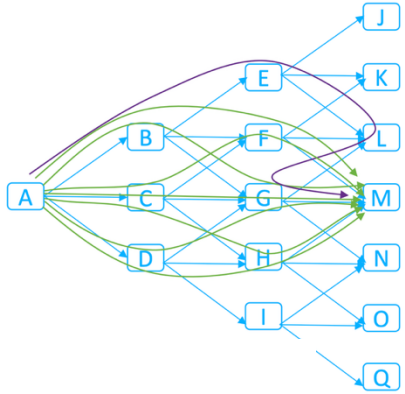
- 他の Inband Telemetry 方式 (INT, IFA, iOAM)に比較して、低オーバヘッド
- HW親和性とLine Rate処理
  - 現存のハードウェアでラインレートの処理が可能
  - コプロセッサやスローパスへのオフロード不要
- スケーラブルできめ細かいタイムスタンプ
  - 64 bit (Source Node, Sink Node)
  - 8 bit (Mid Node)
- スケーラブルな負荷測定

- INT (In-Network Telemetry) : https://github.com/p4lang/p4-applications/blob/master/docs/INT_v2_0.pdf
- IFA (Inband Flow Analyzer) : https://datatracker.ietf.org/doc/html/draft-kumar-ippm-ifa
- iOAM (In-Situ OAM) : https://datatracker.ietf.org/doc/html/rfc9197

16

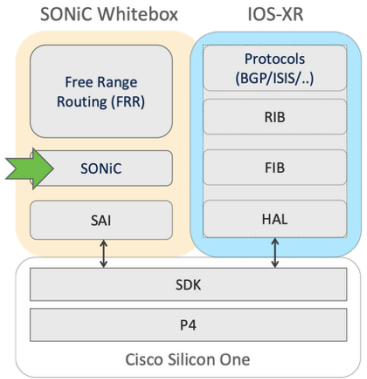
SRv6 Path Tracing ?!

- パケット単位のdeterministic（決定論的）なトレース
- HWパイプラインにLinerateで実装
  CPUへのpunt、Co-processor へのオフロードなし
- MTU効率の良さ：3 bytes / hop
  Interface (12 bits), Timestamp (8 bits), Load (4 bits)
- SRv6/IPv6をサポート
- シームレスなデプロイメント
  レガシーノードとのインターワーク



SRv6 uSID is fully supported in SONIC

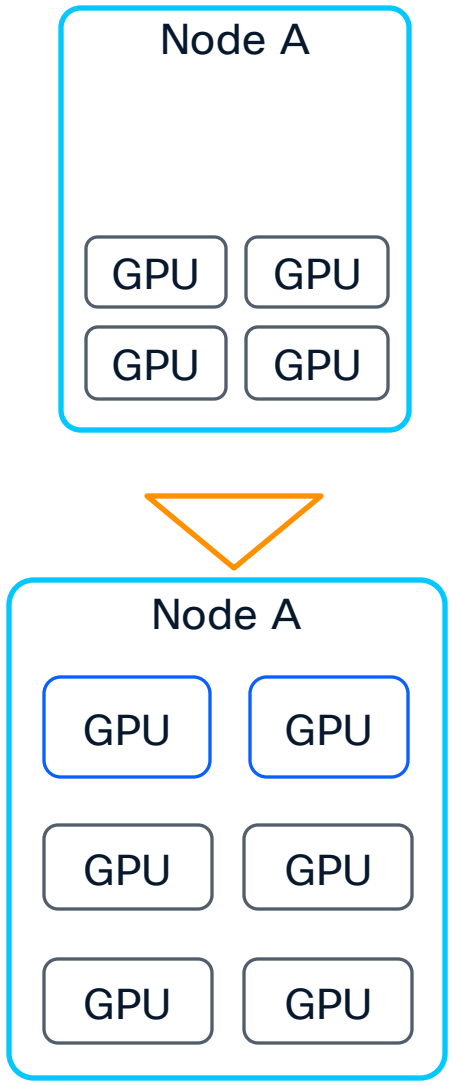
- SONIC 202211
  - uN/uA/uDT4/uDT6/uDT46
  - H.Encaps/H.Ecanps.Red
- Across all SONiC components
  - ConfigDB/App DB/Orchestration Agent
  - Fpmsyncd (for the integration with FRR)
- Path Tracing in SAI/SONiC
  - https://github.com/opencomputeproject/SAI/pull/1841
  - https://github.com/sonic-net/SONiC/pull/1456



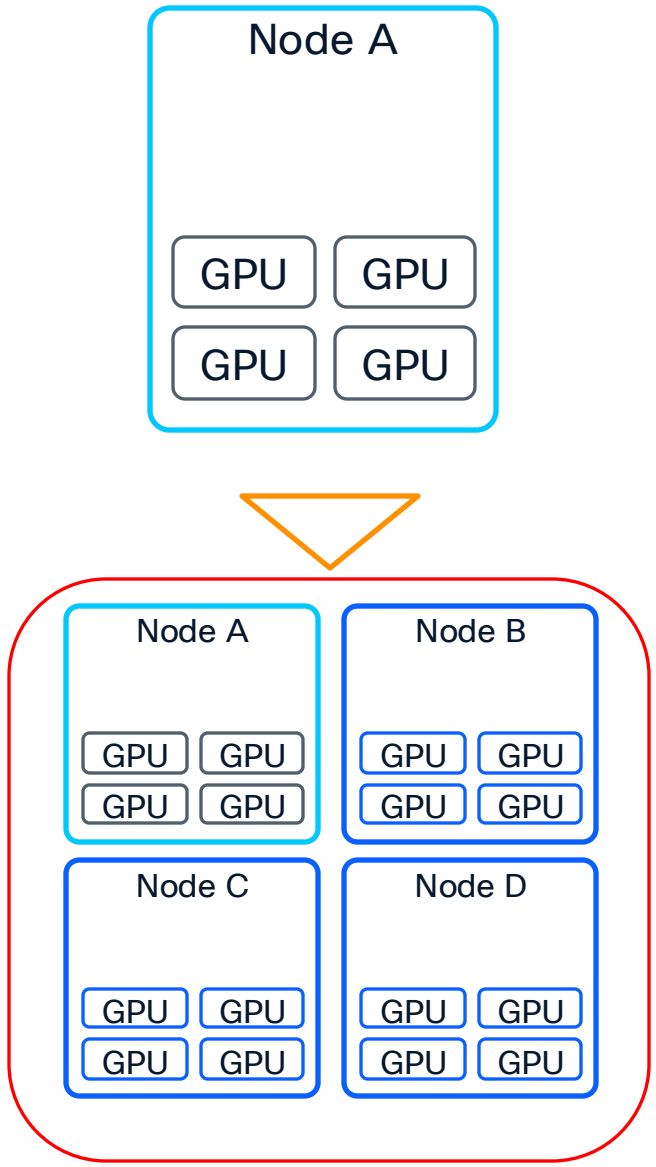
https://mpls.jp/2023/presentations/mpls2023-kohno.pdf
https://mpls.jp/2023/presentations/mpls2023-kamata.pdf

Scaling further...

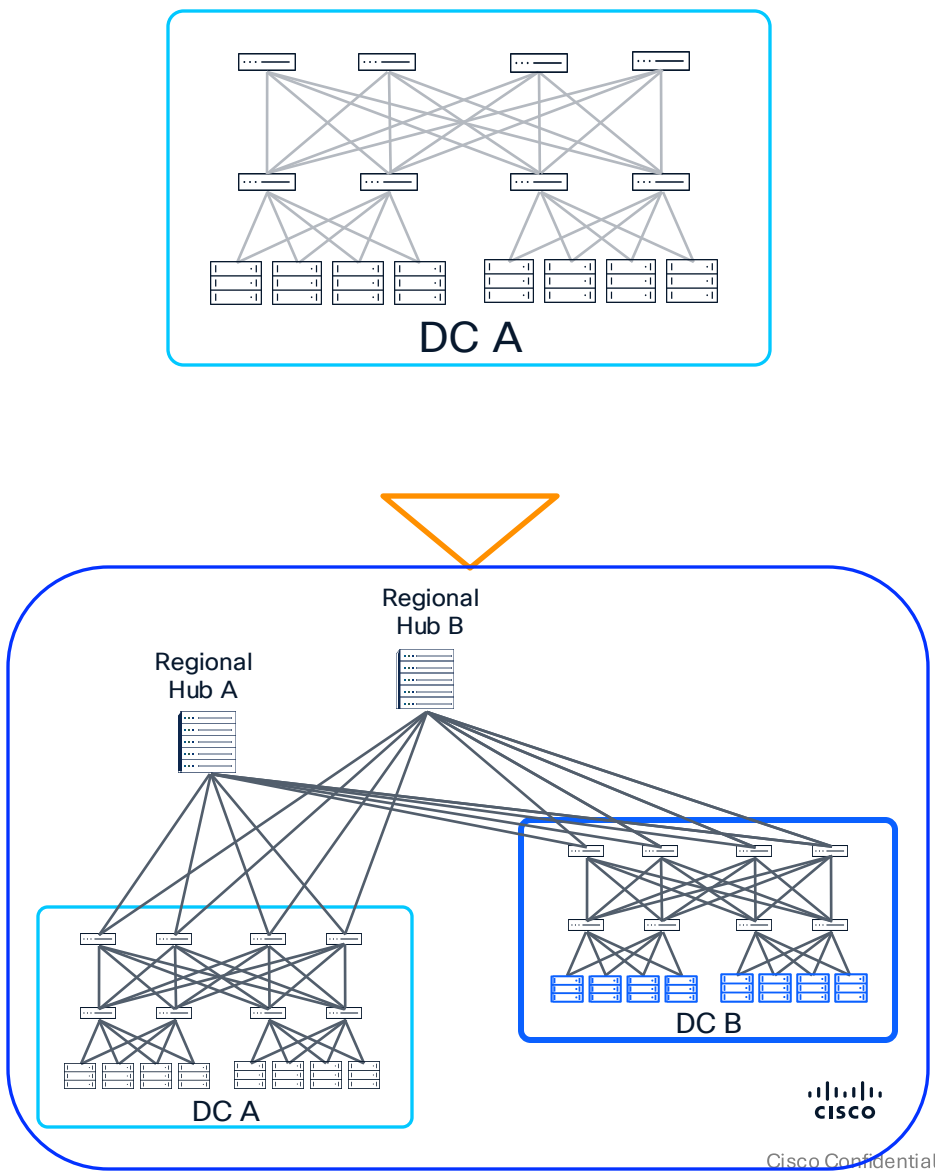
Scale Up



Scale Out

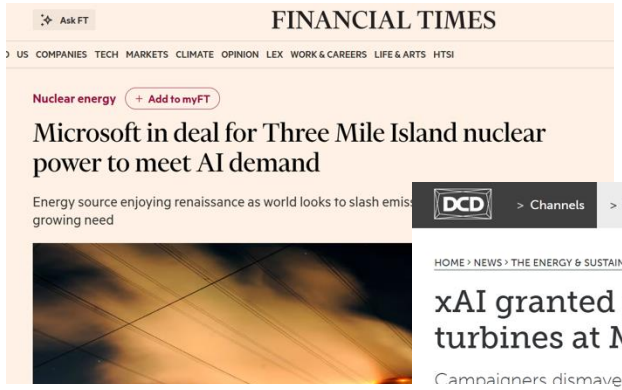


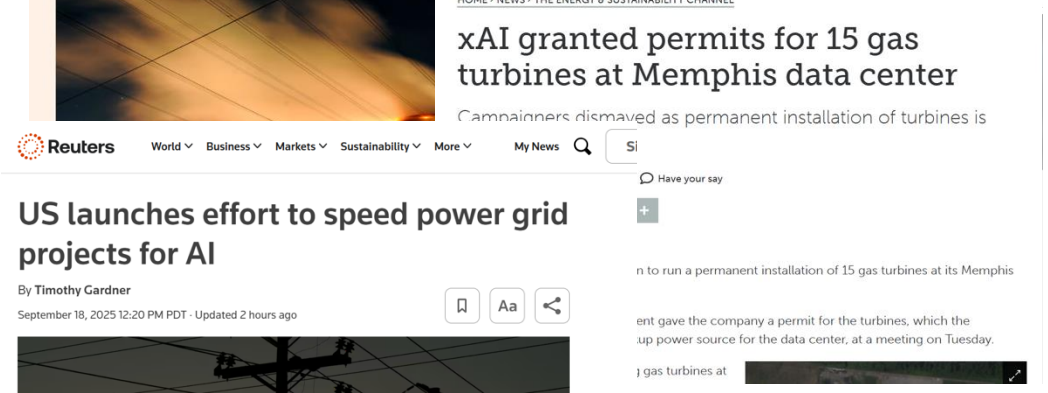
Scale Across

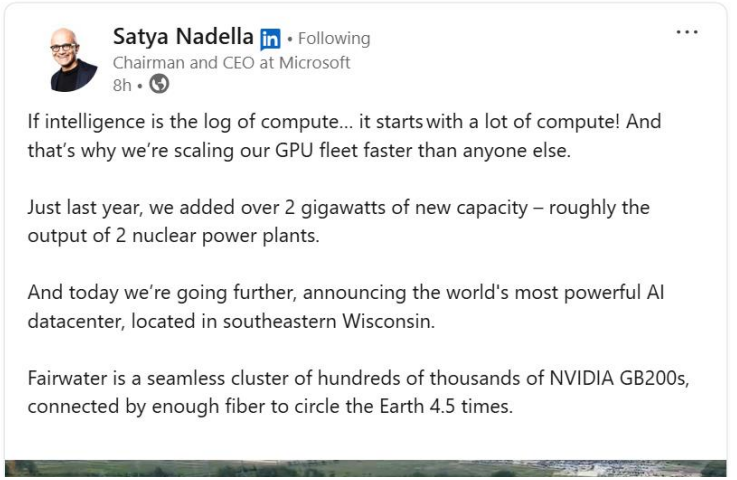


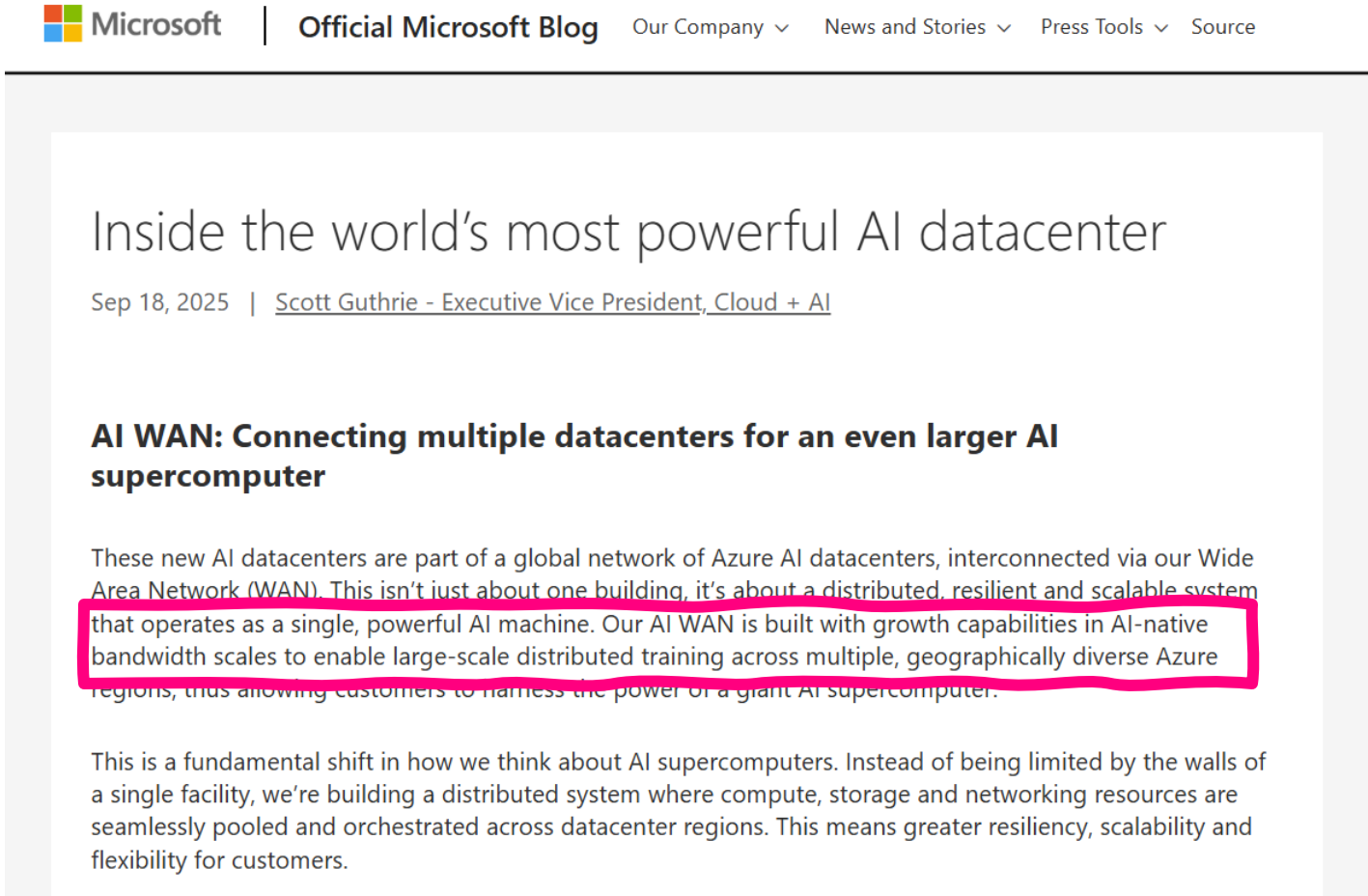
New challenges

- Why? → 電力不足について、もはや聞く必要あります?
  - 15kW/rackからMegaW/rackの世界へ
  - 空冷から水冷へ
- でも、DCI/Metroには異なる制約が
  - Transponderはちょっと高い
  - WANルータの容量に制限がある
  - Latencyに変動性がある(そもそも大きい)



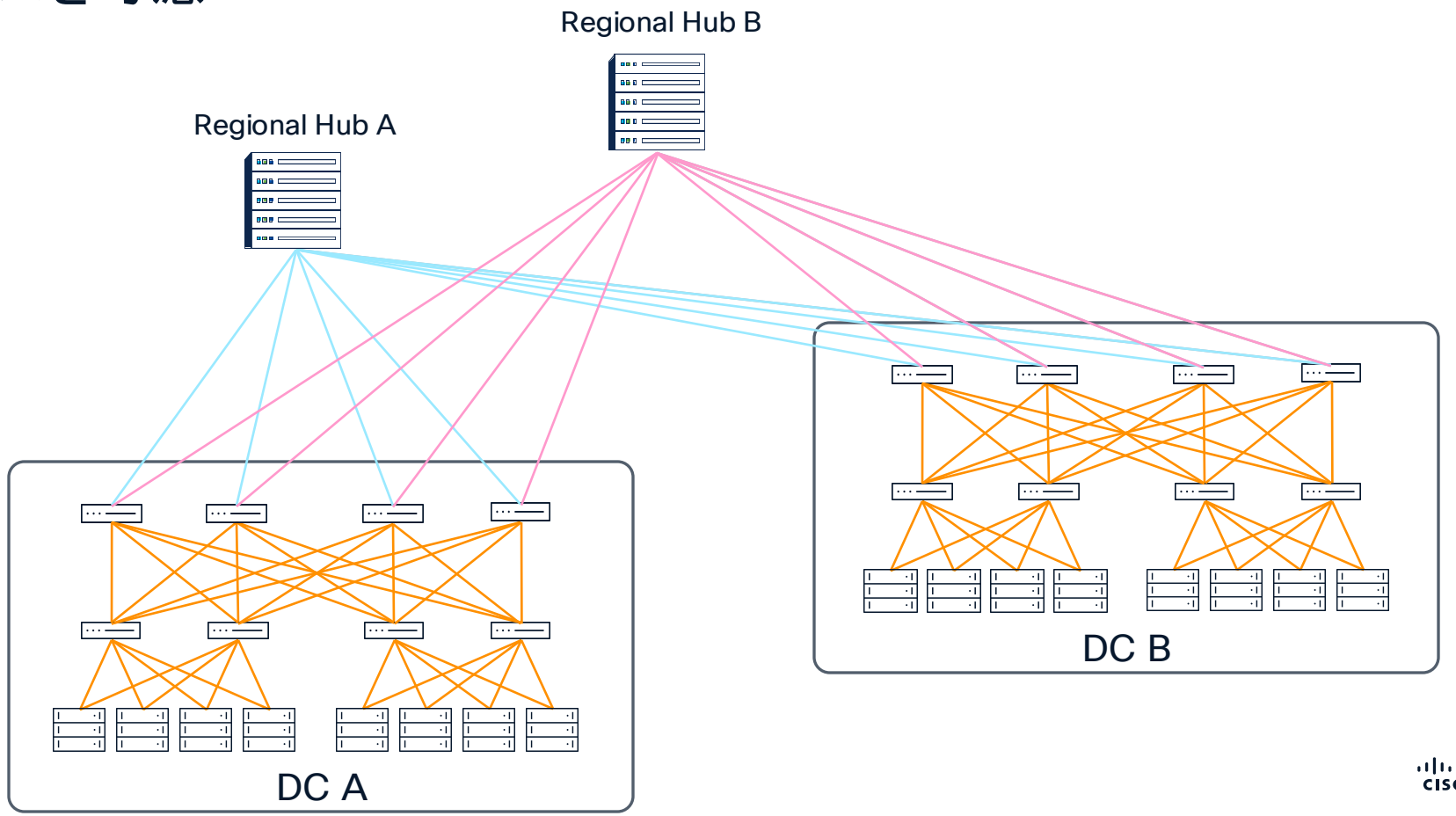






Scale Across: It’s a TE Problem!

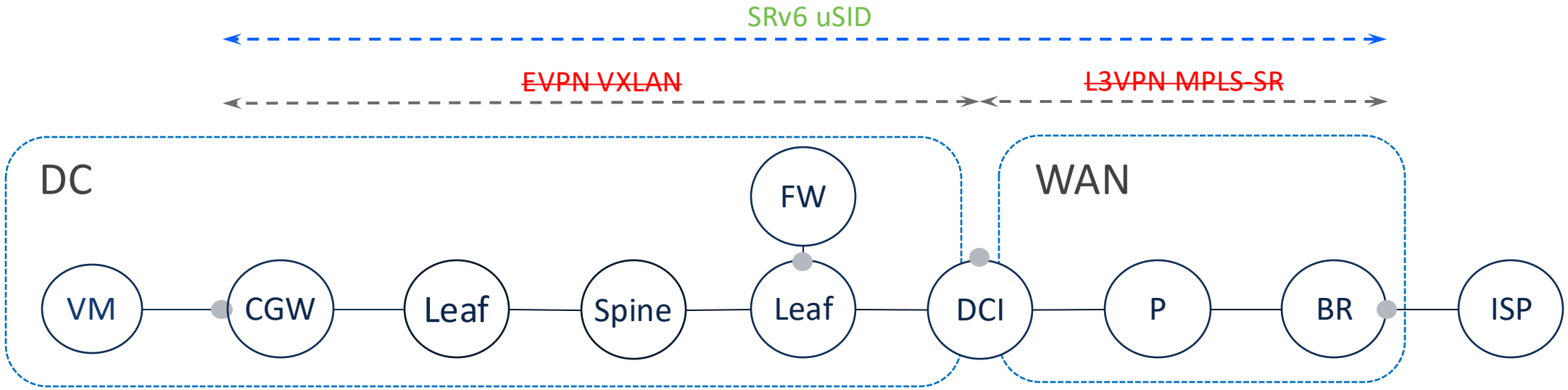
- トポロジが対照的ではない
- WANルータの容量不足
- 特定のワークロードのパスを考慮
- Latencyの考慮
- SRLG’s



IPv6 uSID for Scale Across

- IPv6 uSIDをAI Backendで採用する流れが続く可能性もある
- TEを用いるならSegment Routingを使うのは自然な流れ
- end-to-endでIPv6を利用可能:
  - TEといえばMPLSですが、DCにMPLSはないですよね。
  - GatewayでInter-AS OptionB? やりたくないですよね。
  - MPLSは複雑すぎる。

SRv6 in the Front-End DCの事例も



- SRv6 uSID
  - DCのFrontendからMetro、Peeringまで統一されたEnd-to-Endのデザインが可能
  - DCをVXLAN、Metro NetworkをSR-MPLSというデザインを置き換えることができる
  - Statelessなので、FirewallなどのNetwork functioを間にいれることができる
  - Traffic engineerinをネイティブサポート

Alexey Gorovoy, Nebius, March 2025
SRv6 World - Public Presentation Recording

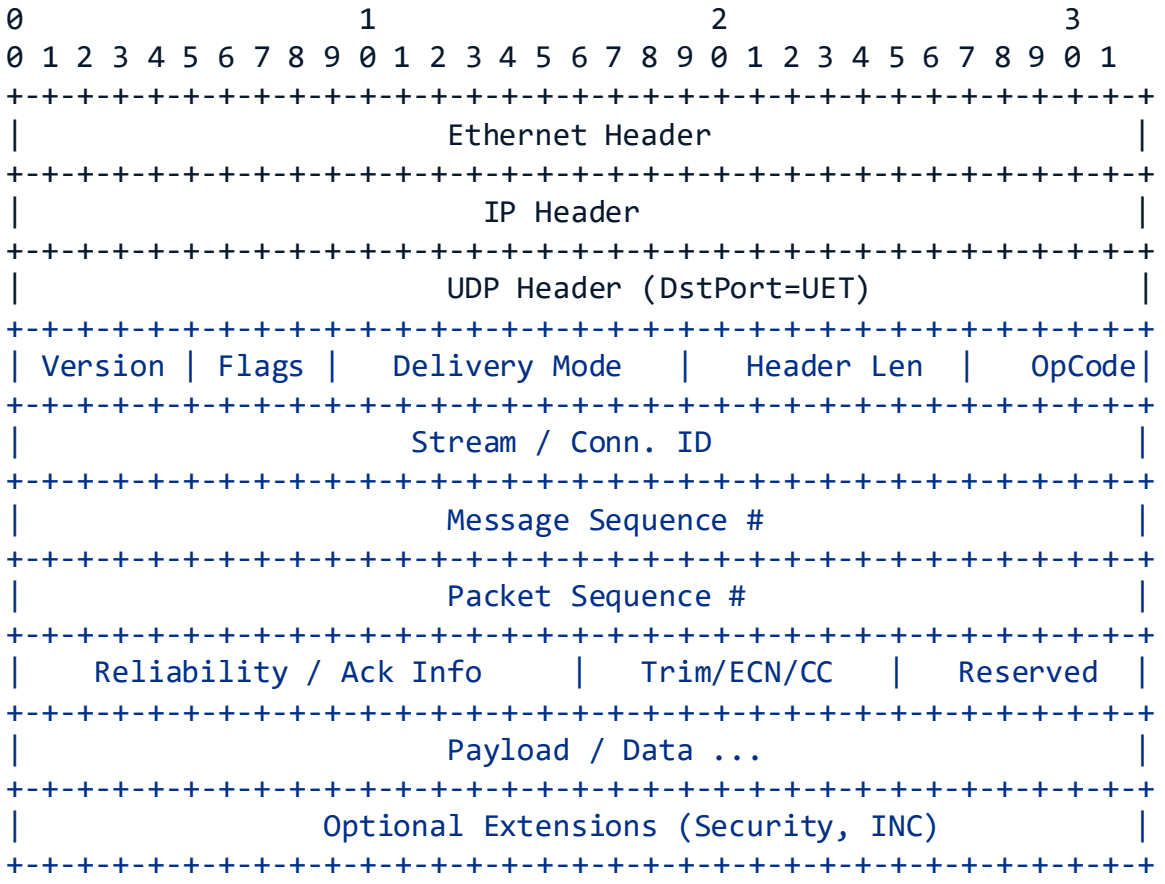
Conclusion

終わりに

- IPv6 uSIDのユニークなメリット:
  - Fabric内でNICがDeterministicなパスを選択することができる
  - パスの切り替えも即時実行可能 (Fabric内でStatelessなソリューション)
  - オープンで標準化された技術 (proprietaryなblack-boxではないですよ)
  - 業界内で広くサポートされている技術
- IPv6 uSIDによってRegion内でシームレスに拡張していくことが可能
  - Traffic Engineeringの要件は基本的に全て満たすことができる
- 今後も継続的に興味深いイノーベーションが続きます！

UET Header

- Key differences:
  - Explicit **Delivery Mode**
    - Reliable Ordered Delivery
    - Reliable Unordered Delivery
    - Unreliable Unordered Delivery
  - Split between **Message Seq#** and **Packet Seq#** → supports unordered spraying.
  - **Trim/CC metadata** → packet trimming, ECN congestion signaling.
  - **Optional in-network collective extensions** (unique to UET).
  - Headers are **lighter / more compressible**, especially for small messages.



Number of GPUs in a fabric

- No oversubscription
- 200Gbps interfaces at NIC (BF3 is 2x200G)
- 2-Tier
  - G200. 256 ports @ 200 G each; 128 GPUs each; 128 TORs; 64 spines.
    - Total: 8192 GPUs (1024 servers); N^2/2
  - G300. 512 ports @ 200G each; 256 GPUs each; 256 TORs; 128 spines.
    - Total: 131k GPUs (131.072); N^2/2
- 3-Tier
  - G200. 256 ports @ 200 G each; 128 GPUs each; 128 TORs; 128 spines; 128 super-spines
  - Total: 4M GPUs; N^3/4
- Three-tier challenges:
  - More optics. More switches. Cost and power consumption.
  - Higher latency (5 vs 3 hops).
  - Difficult load-balancing, congestion control, and more link flapping