AI Clusters

Enabling AI Networking @ Scale with Multi-path Reliable Connections (MRC)

Eric Spada *Distinguished Engineer, Broadcom*

Guohan Lu *Member of Technical Stuff, OpenAI*

OCP PLATINUM Member

MRC Origins and Design Objectives

EARLY 2024



Proven foundations

Ultra Ethernet Consortium work and earlier vendor designs for multipath and packet spraying

SHARED AMBITION

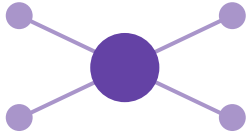
10^6 nodes

DESIGN DIRECTION

Diversity + resilience

OpenAI and Microsoft sought multipath and multiplane operation resilient to link and fabric failures

OPEN COLLABORATION



One open specification

AMD · Broadcom · Intel
Microsoft · NVIDIA · OpenAI

OCP MRC

AMD BROADCOM intel Microsoft NVIDIA OpenAI

Application Semantics and Transport Mechanisms

APPLICATION-VISIBLE CONTRACT

Reliable delivery
Connection-level reliability

Ordered completions
Disorder remains below the API

Familiar model
QP/CQ-style software abstractions

PRESERVED

MRC TRANSPORT MACHINERY



Spray
Per packet



Bound
Inflight state



Recover
Selectively



Control
Sender + paths

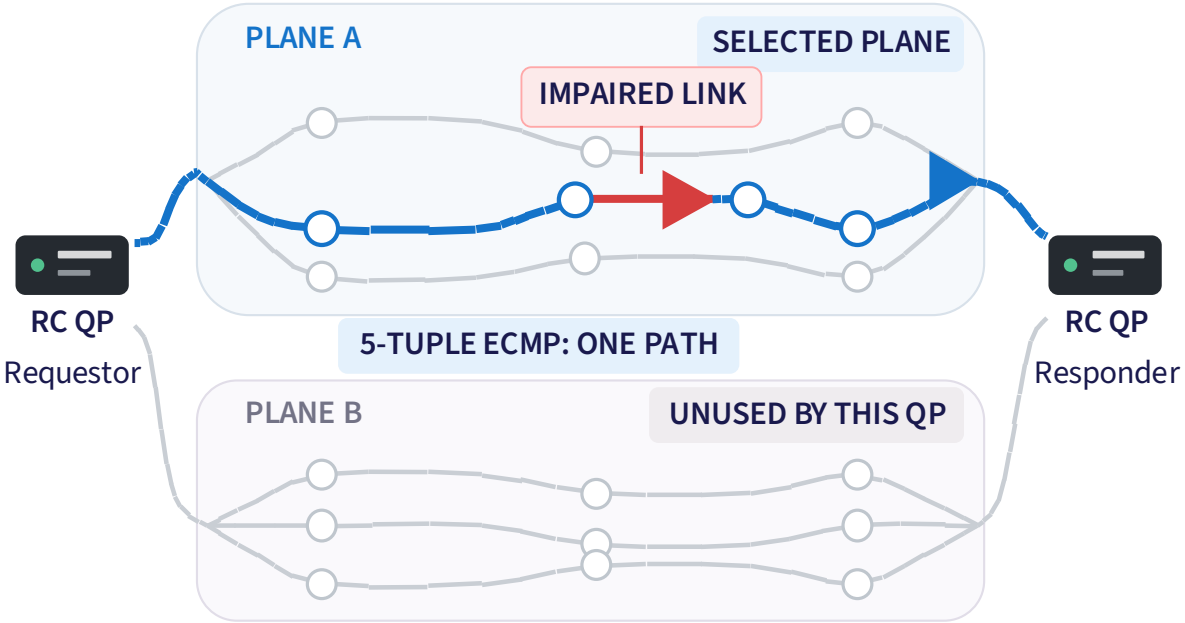


Avoid
Failed paths

Mandatory primitives ensure correctness • Optional primitives add efficiency and resilience

AMD BROADCOM intel Microsoft NVIDIA OpenAI

Limitations of RC at AI Scale



PINNED

One QP hashes to one ECMP path

REPLAY

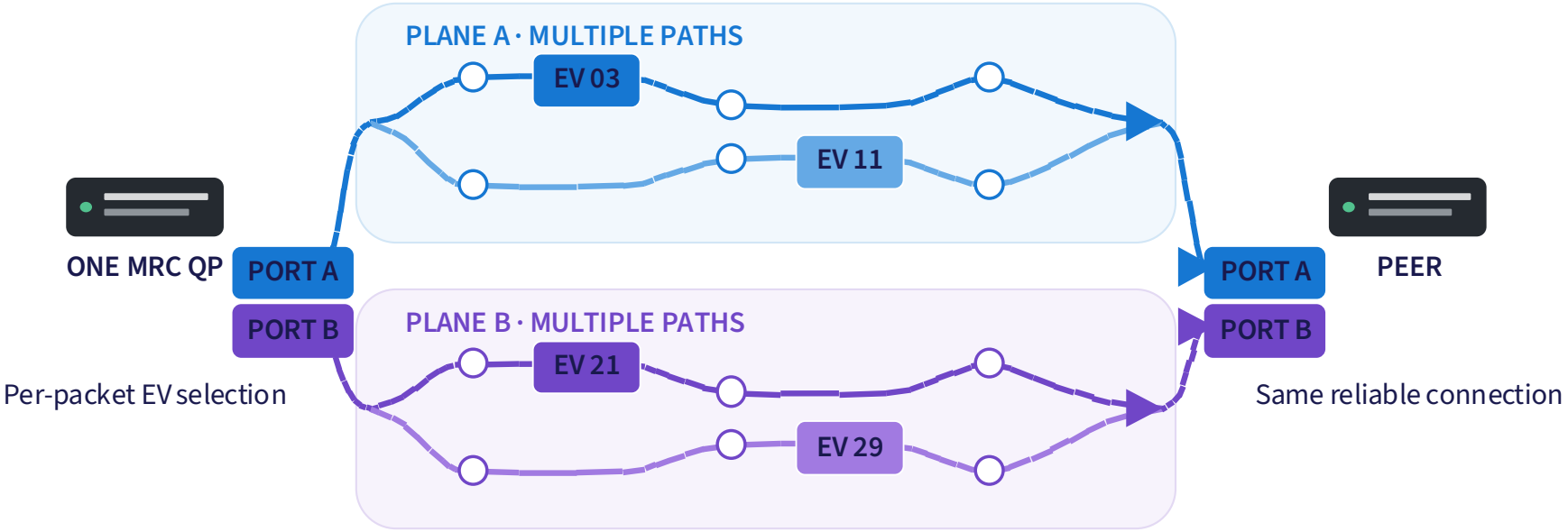
One loss triggers go-back-N

WAIT

Failure recovery depends on timeouts and routing convergence

AMD BROADCOM intel Microsoft NVIDIA OpenAI

Multipath and Multiplane



Multipath: routing diversity • **Multiplane:** port and failure-domain diversity

AMD BROADCOM intel Microsoft NVIDIA OpenAI

Source Routing With SRv6

ONE MRC ENTROPY VALUE ABSTRACTION
The transport can use implicit hashing or explicit source routes

CONVENTIONAL

ECMP entropy

Endpoint selects a hash input



PATH CHOSEN BY THE NETWORK

EXPLICIT

Structured EV

Hop-by-hop route encoded in packet fields



ROUTE IDENTIFIED BY EV

EXPLICIT

SRv6 uSID

Path encoded as IPv6 micro-segments



ROUTE IDENTIFIED BY EV

WHEN A ROUTE DEGRADES

Identify
Feedback maps to route

→

Exclude
Denylist unhealthy EV

→

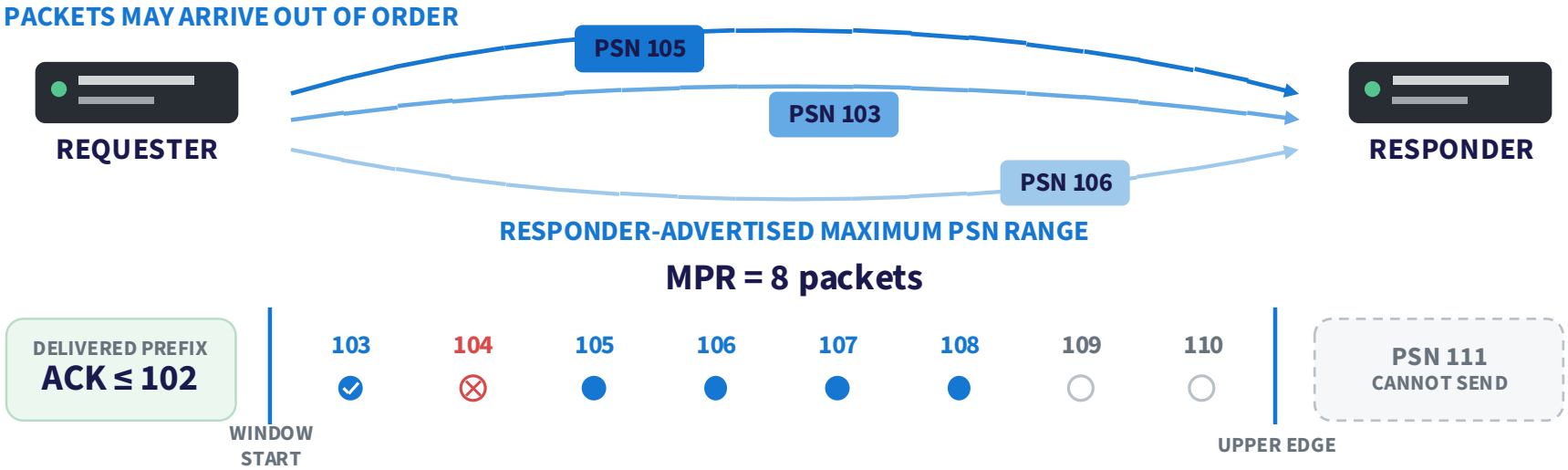
Reselect
Use another known route

NO NEED TO WAIT FOR ROUTING CONVERGENCE

Source routing does not accelerate the routing protocol—it removes the endpoint's dependency on it

AMD BROADCOM intel Microsoft NVIDIA OpenAI

Bounded Reordering



Optional Dynamic MPR resizes the window as demand and available NIC resources change

AMD BROADCOM intel Microsoft NVIDIA OpenAI

Selective Loss Detection and Recovery

CONVENTIONAL RC

Go-back-N replays the tail

One missing packet invalidates later progress



MRC

Selective recovery preserves progress

SACK identifies received packets and the actual gap



DETECT EARLY · RECOVER PRECISELY · GUARANTEE PROGRESS

SACK + NACK
Explicit delivery state
Mandatory reliability feedback

Packet trim
Header survives the drop
Optional network acceleration

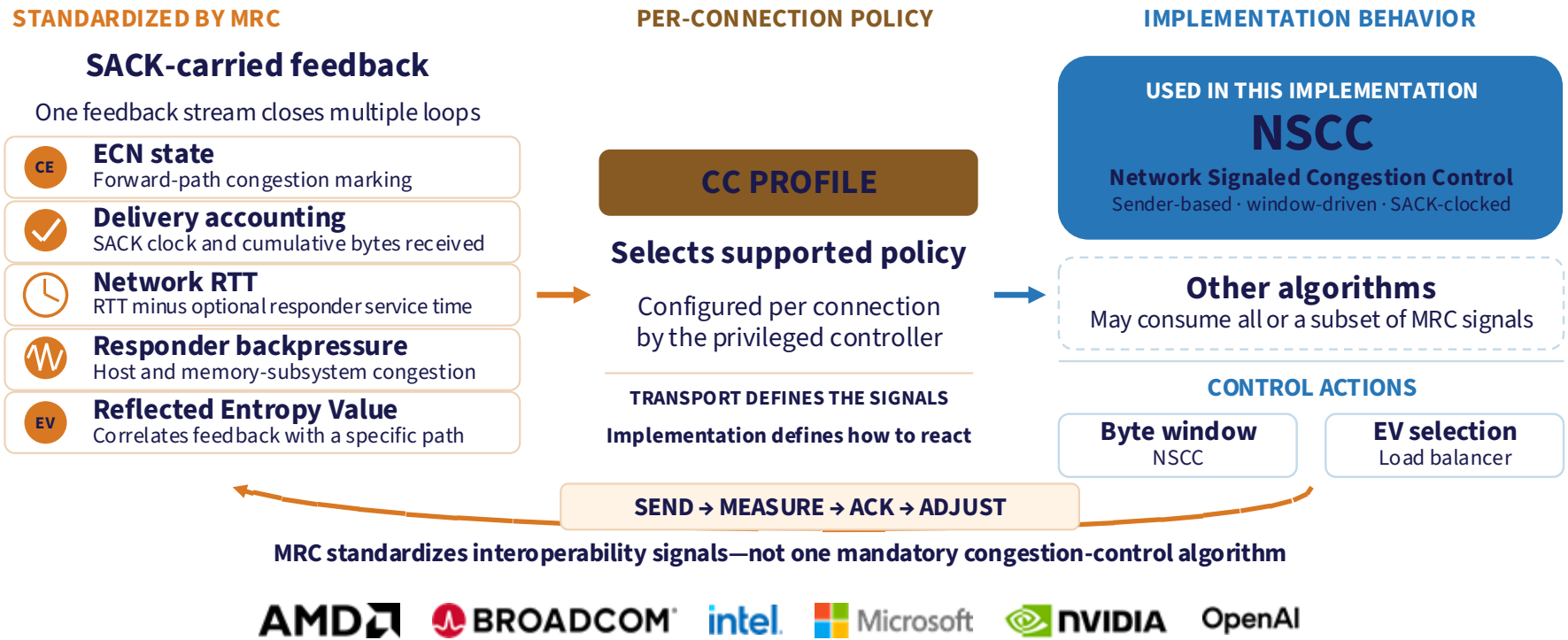
Probe
Elicit current SACK state
Optional proactive query

Timers + priority
Fallback and fast repair
Oldest gap retransmitted first

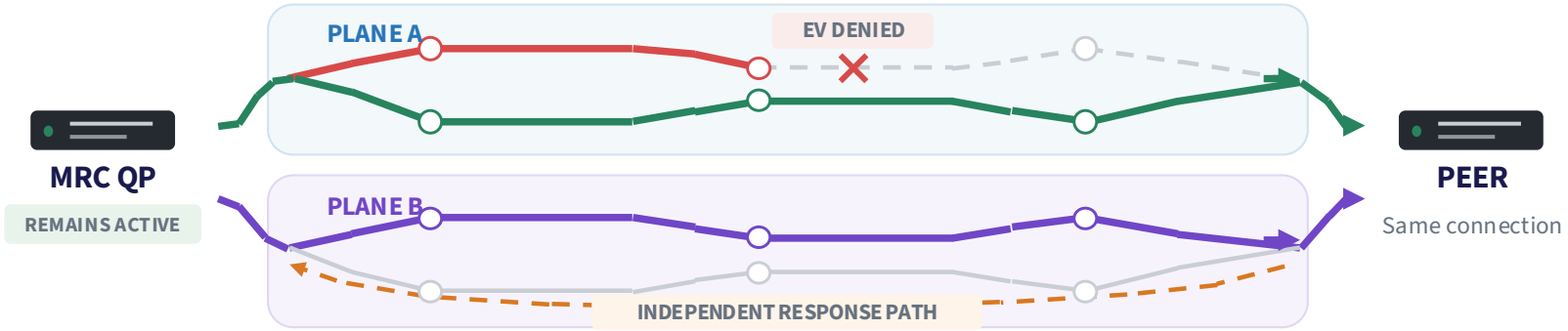
Reordering is not loss. One loss should not erase later progress.

AMD BROADCOM intel Microsoft NVIDIA OpenAI

Congestion Control Framework and NSCC



Fast Path and Plane Failure Avoidance



DATAPATH-TIMESCALE RESILIENCE

EV state
Exclude unhealthy routes
Path-selection state

EV probe
Test one explicit path
Path-scoped liveness

Port Status
Announce plane reachability
Asynchronous update

GID scoped
Update many QPs at once
Stateless endpoint operation

Failure becomes a path-selection event—not a connection failure

AMD BROADCOM intel Microsoft NVIDIA OpenAI

Application and Controller APIs

APPLICATION API — **mrc.h**

Familiar verbs model

Transport control and data-plane operations

- **Reuse libibverbs**
  Discovery and memory registration
- **Familiar resources**
  mrc_qp and mrc_cq lifecycle
- **Discover capabilities**
  Device and per-port feature queries
- **Configure and transmit**
  Write and Write-with-Immediate

query_device · query_port · modify_qp

DATA PLANE

OPERATE → **Reliable connection**

POLICY ←

BINDS TO

EV PROFILE | **CC PROFILE**

Connection setup remains out of band

Best-effort Ethernet
RoCEv2 / UDP / IP encapsulation

CONTROLLER API — **mrc_ctl.h**

Shared transport policy

Privileged process with CAP_NET_ADMIN

- **Query capabilities**
  Mandatory device and port discovery
- **Create EV profiles**
  Shared path universe and selection policy
- **Create CC profiles**
  Shared congestion-control policy
- **Optional path controls**
  EV-state events and EV probes

Policy and telemetry shared across many QPs

Familiar application semantics • centralized path and congestion policy

What happened when a T1 stopped forwarding

75K
GPUs in the production job

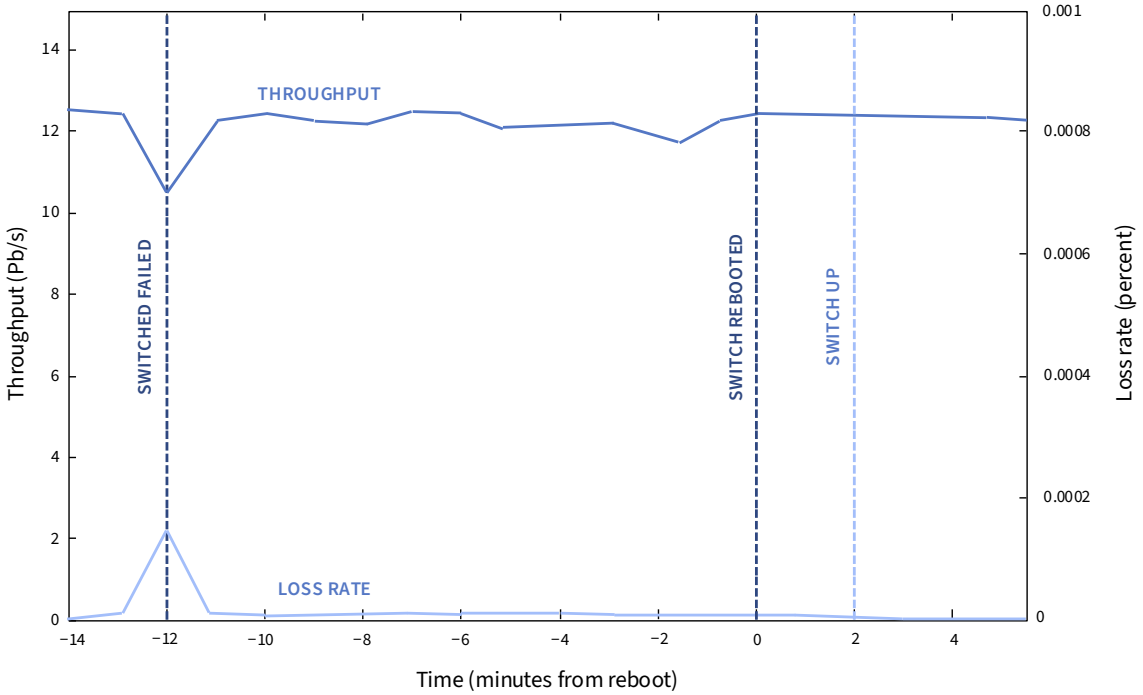
~25%
of QPs affected

~580K
packets dropped



Resilient AI Supercomputer Networking using MRC and SRv6

A T1 switch failure and reboot has minimal impact on training throughput



Real data captured during a training run showing MRC reacting to the complete loss of a T1 switch. The training job experienced a temporary slowdown, but recovered quickly.

[Call to Action]

- Where to find additional information (URL links)
  - MRC 1.0 Spec http://www.opencompute.org/documents/ocp-mrc-1-0-pdf
  - MRC API github: https://github.com/opencomputeproject/OCP-Multipath-Reliable-Connection


Thank
You!
OCP
GLOBAL
SUMMIT
October 12-15, 2026
San Jose, California