Cross-Fabric Policy Reasoning for Hybrid InfiniBand / Ethernet HPC Systems

Policy-valid remediation across fabrics that are not interchangeable.

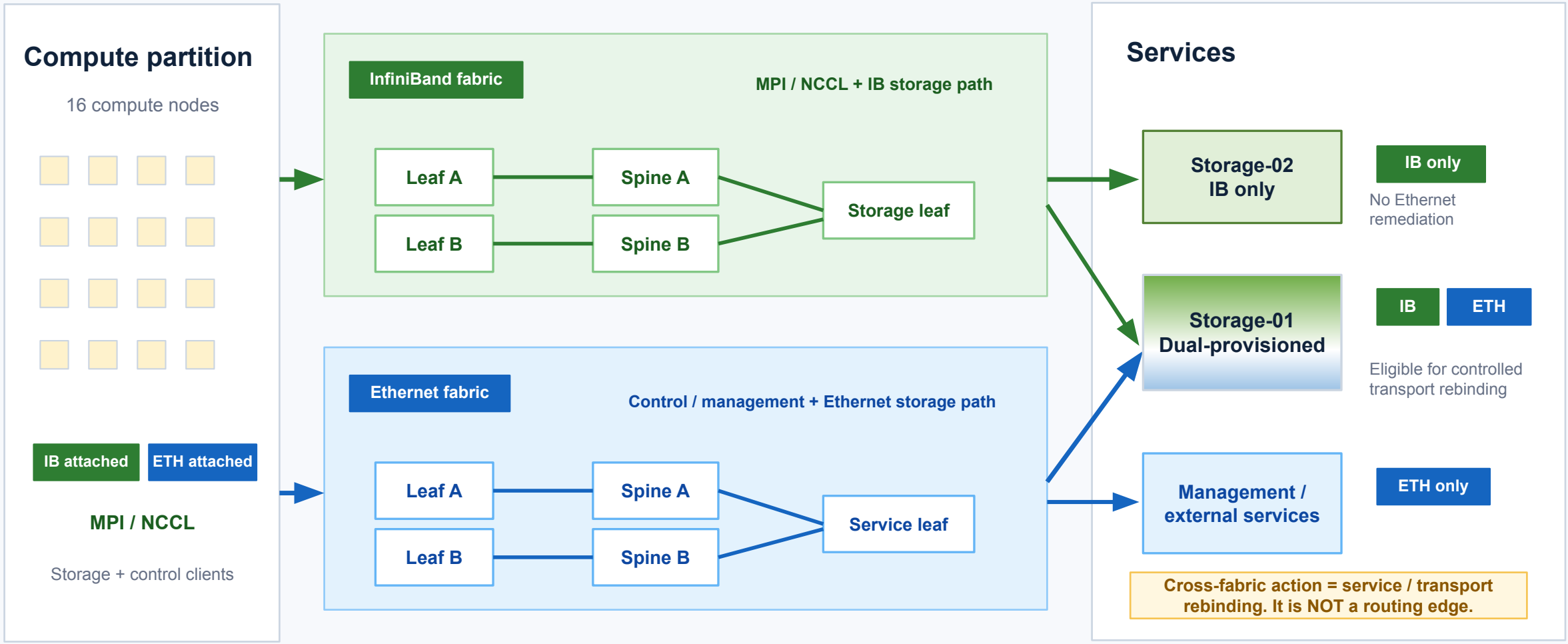


Vedant Sawal, Graduate Intern, NERSC, LBNL

Anna Giannakou, Research Scientist, LBNL

Architecture: Two fabrics, shared endpoints, explicit service provisioning

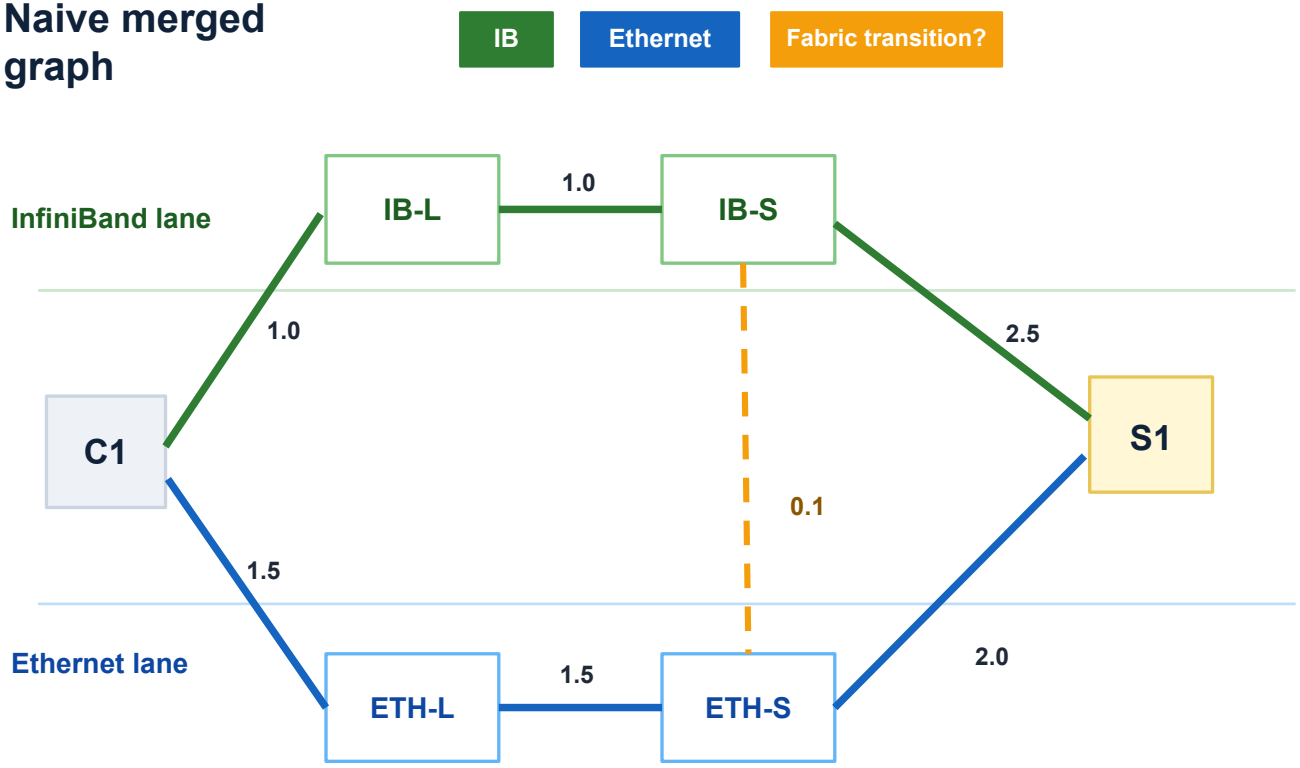
The prototype keeps InfiniBand and Ethernet as distinct transport domains.



A merged weighted graph can make Dijkstra choose an invalid path

Illustrative weights only: the issue is treating transport rebinding as an ordinary graph edge

Naive merged graph



Illustrative edge weights

If a fabric transition is encoded as an edge, Dijkstra can mix transport domains inside one "path."

Dijkstra from C1 to S1

Candidate	Cost
Valid IB path	4.5
Valid Ethernet path	5.0
Mixed IB -> ETH path	4.1

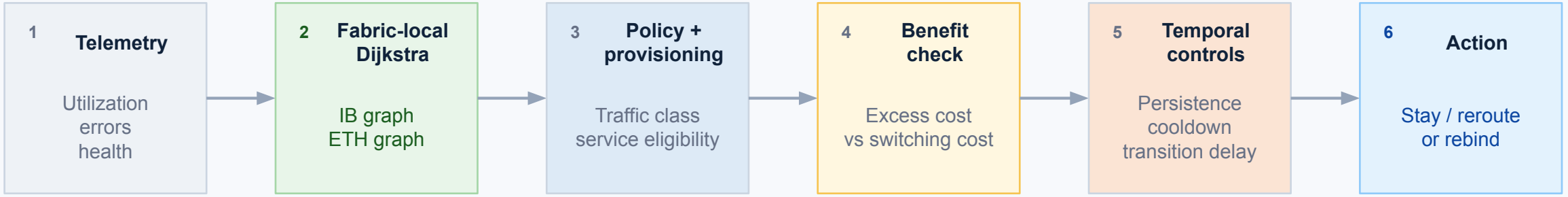
Naive shortest path

But a fabric transition is not a hop. It requires policy eligibility, transport support, and explicit reconfiguration.

CONTROLLER WORKFLOW

Local remediation first; cross-fabric transition only through policy

The controller stays within a fabric whenever possible, and escalates only when the active transport remains impaired.



FAST PATH

IB

ETH

Stay within the current fabric

If telemetry-aware routing finds a healthy alternate path, reroute locally and stop.

SLOW PATH

IB → ETH

Policy-constrained cross-fabric remediation

Only for explicitly dual-provisioned services, and only when benefit exceeds transition cost and temporal safeguards pass.

Key idea: InfiniBand and Ethernet remain separate graphs. Cross-fabric reasoning happens at the service-policy layer.

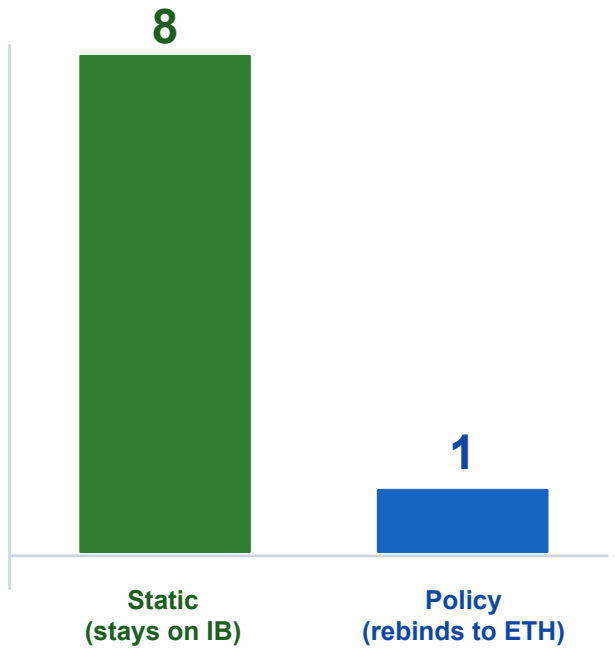
RESULTS

Persistent impairment triggers remediation, short-lived noise does not

Five deterministic scenarios compare static fabric preference with the policy controller.

Hard IB storage-path failure

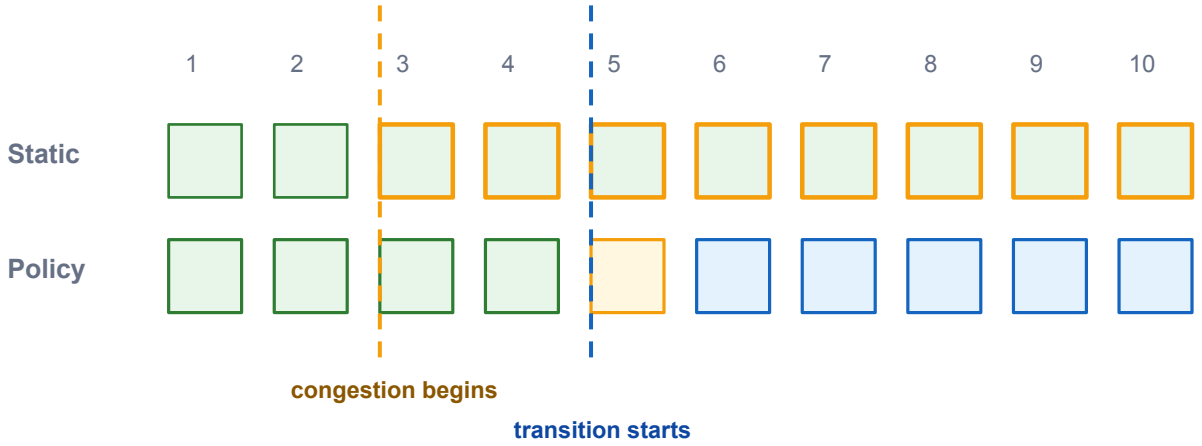
Unavailable simulation intervals



8 ⇨ 1

Sustained congestion timeline

Persistence = 3 • transition delay = 1



Transient congestion: 0 switches

Both fabrics congested: 0

Hard failure: failover

The controller reduces modeled unavailability under hard IB failure from 8 to 1 interval while avoiding unnecessary transport changes.

Sensitivity exposes the stability vs responsiveness tradeoff

50 configurations: persistence 1–5 x switching cost {0, 2.5, 5, 7.5, 10}.

Transient congestion: does the controller switch?

Switch to ETH | Stay on IB

Persistence \ Switching cost	0	2.5	5	7.5	10
1	Switch	Switch	Switch	Switch	Stay
2	Switch	Switch	Switch	Switch	Stay
3	Stay	Stay	Stay	Stay	Stay
4	Stay	Stay	Stay	Stay	Stay
5	Stay	Stay	Stay	Stay	Stay

Persistence >= 3 suppresses the two-step transient event.

What the sweep shows

Sustained congestion

Persistence	First transition
1	t = 3
2	t = 4
3	t = 5
4	t = 6
5	t = 7

Higher persistence --> later transition

Switching cost = 10 --> stay on IB
Benefit is too small to justify rebinding

TAKEAWAY

Keep the fabrics separate. Make Cross-fabric actions explicit

Cross-fabric remediation is a service-policy decision, not shortest-path routing over a merged topology.

1 Stay local first

Telemetry-aware routing handles intra-fabric degradation before considering a transport change.

2 Cross only when valid

Traffic class and per-service provisioning determine which transport transitions are admissible.

3 Model transition cost

Persistence, benefit threshold, cooldown, and explicit reconfiguration delay prevent flapping.

Prototype evidence

26	44	5	50	8 --> 1
nodes	links	core scenarios	sensitivity runs	failure unavailability

Main Claim: Cross-fabric remediation should be policy-constrained service/transport reasoning, not Dijkstra over one merged graph