Google

Improving Network Availability with Protective ReRoute (PRR)

Yuchung Cheng

Google ML Systems and Cloud AI

July 2025. IETF 123@Madrid

The Bottom Line

- **PRR selects paths for reliability**
  - From hosts, with the IPv6 Flowlabel
- **Lots of production experience**
  - All Google internal and most Cloud for ~6 years
- **Clear upsides, without downsides**
  - Simple & efficient
  - Broadly applicable to transports
  - Easy to deploy
  - Quickly removes most outage time

We hope everyone will deploy these ideas for a more reliable Internet!

High Network Availability is Vital

$$\text{Availability} = \frac{\text{MTBF}}{\text{MTBF} + \text{MTTR}}$$

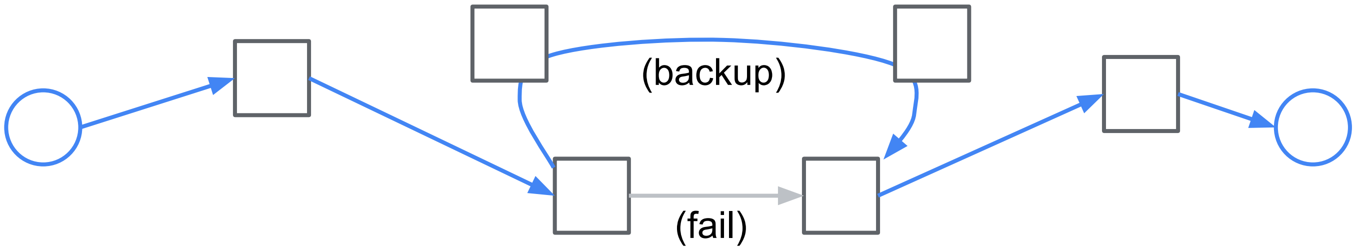
MTBF is Mean Time Between Failures
MTTR is Mean Time To Repair

Keep MTBF large

Focus on MTTR

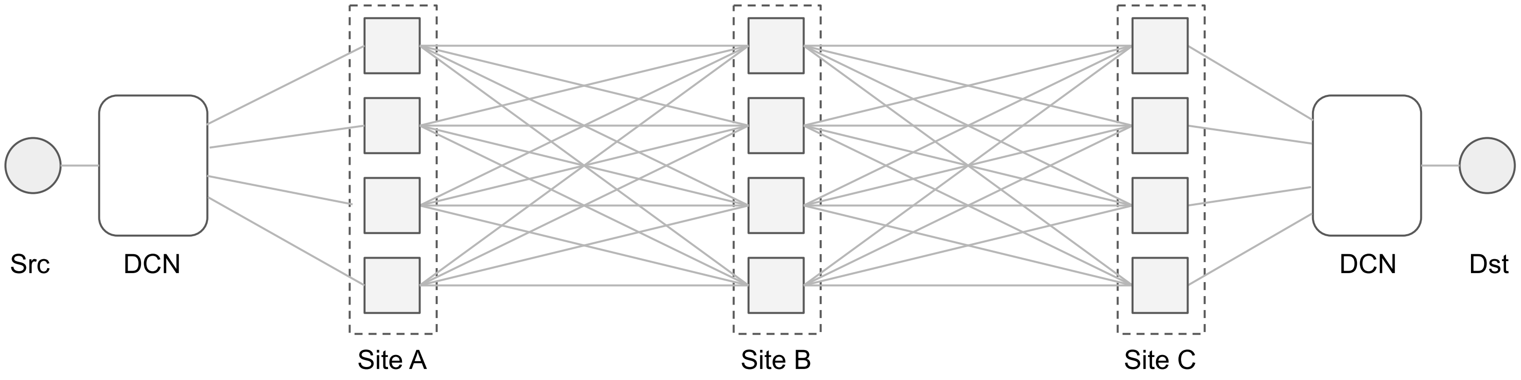
Google

Routing Isn't Sufficient for High Availability



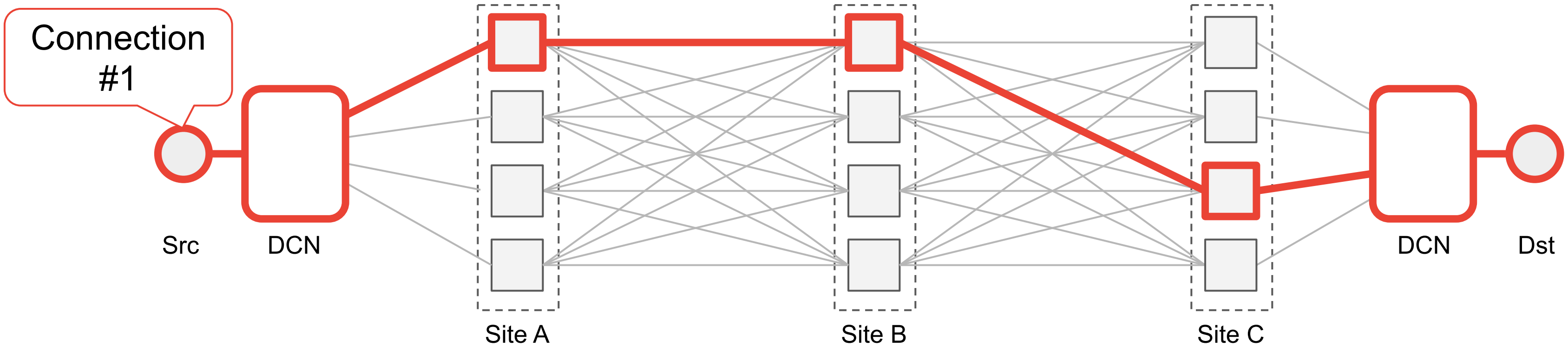
- Classic repair strategy is routing plus physical redundancy
- But it doesn't handle some complex, infrequent faults
  - Higher-level recovery results in long outages
- And it's not fast enough for common faults
  - Global repair when fast reroute fails (no backup, congestion)

Multipath is an Opportunity



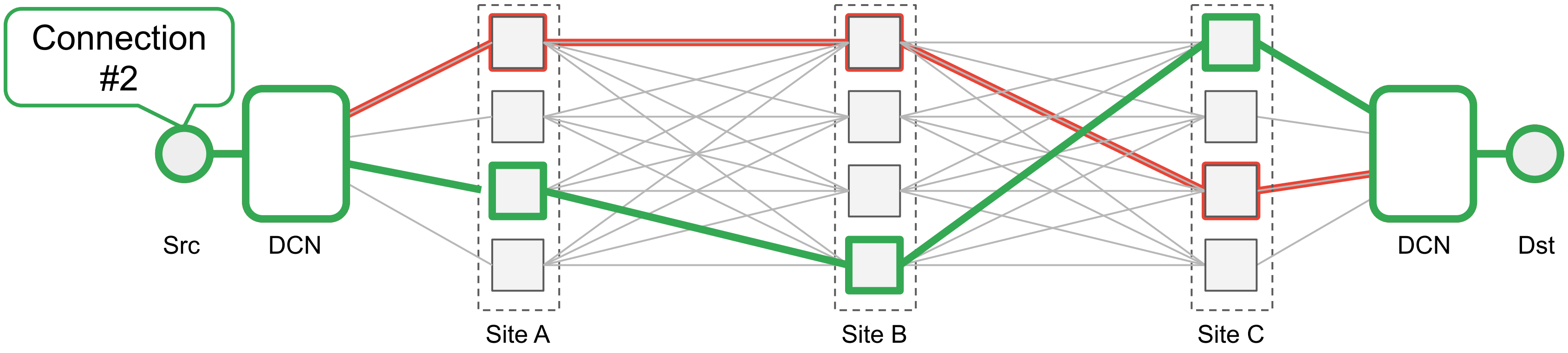
- Modern networks scale capacity by adding parallel links
  - ECMP distributes transport flows across paths
- **Hosts often have some working paths during an outage**
  - Just need to shift traffic to those paths to repair connectivity ...

Multipath is an Opportunity



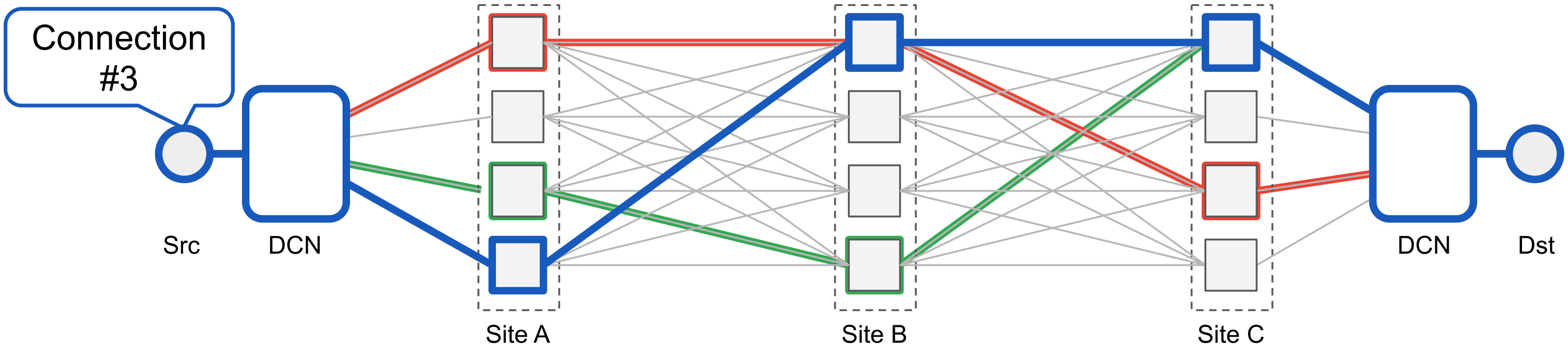
- Modern networks scale capacity by adding parallel links
  - ECMP distributes transport flows across paths
- **Hosts often have some working paths during an outage**
  - Just need to shift traffic to those paths to repair connectivity …

Multipath is an Opportunity



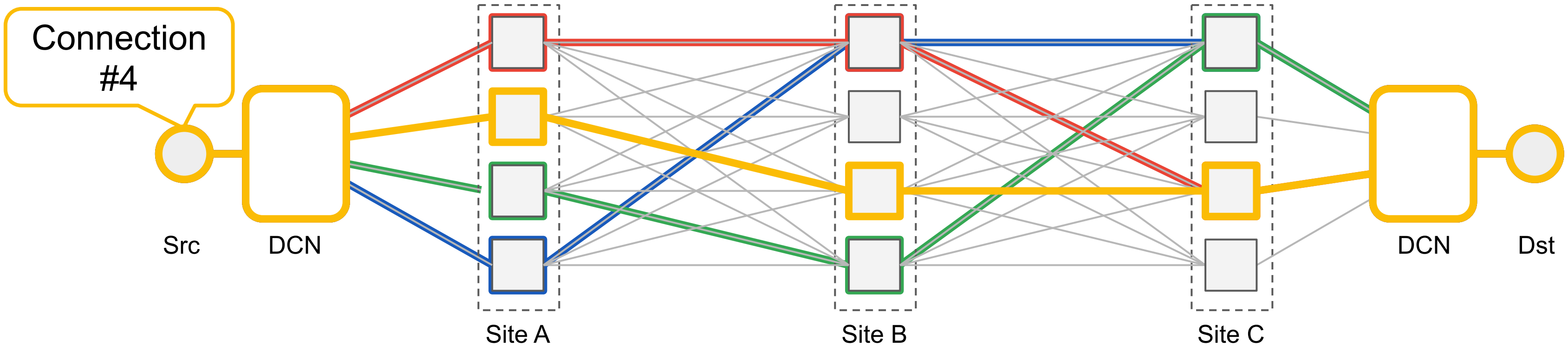
- ➔ Modern networks scale capacity by adding parallel links
  - ◆ ECMP distributes transport flows across paths
- ➔ **Hosts often have some working paths during an outage**
  - ◆ Just need to shift traffic to those paths to repair connectivity ...

Multipath is an Opportunity



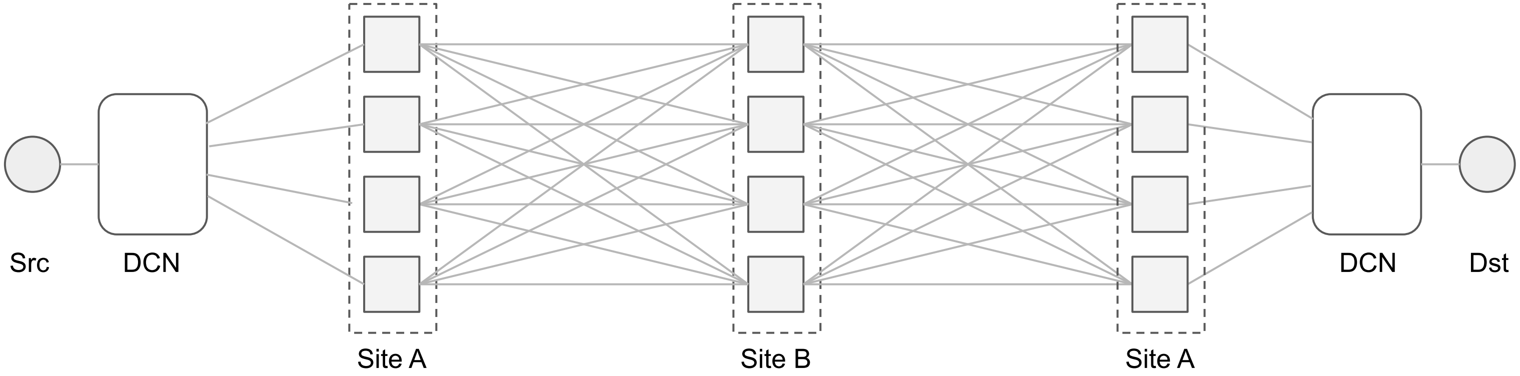
- ➔ Modern networks scale capacity by adding parallel links
  - ◆ ECMP distributes transport flows across paths
- ➔ **Hosts often have some working paths during an outage**
  - ◆ Just need to shift traffic to those paths to repair connectivity …

Multipath is an Opportunity



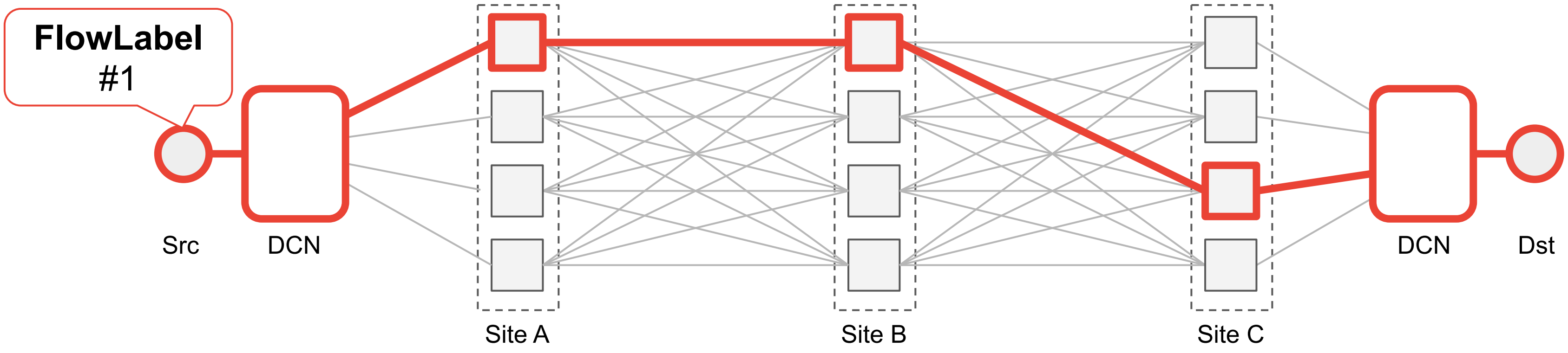
- → Modern networks scale capacity by adding parallel links
  - ◆ ECMP distributes transport flows across paths
- → **Hosts often have some working paths during an outage**
  - ◆ Just need to shift traffic to those paths to repair connectivity …

PRR uses the IPv6 FlowLabel



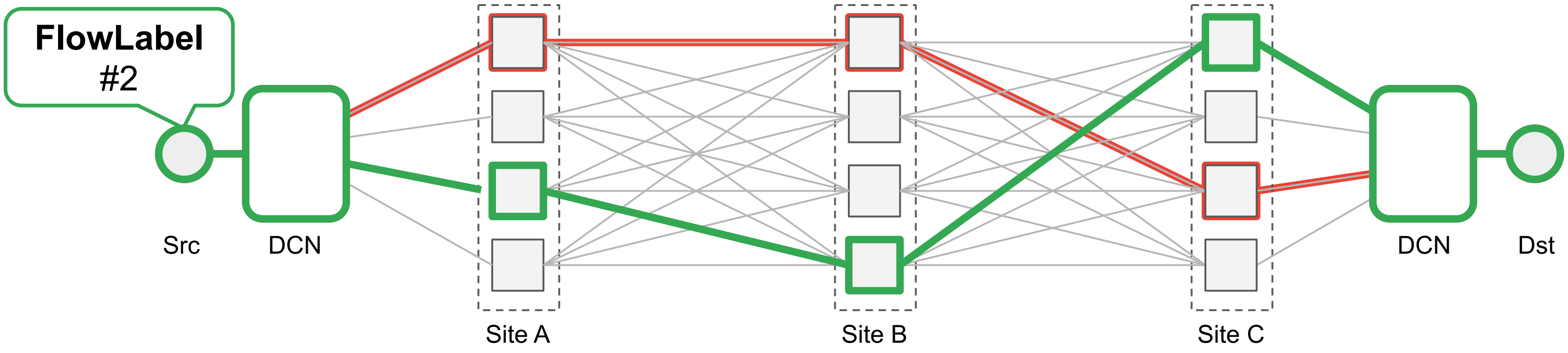
- → With traditional ECMP, each TCP connection is fixed to a path
  - ◆ Host can’t change the path without changing ports or IP addresses
- → With PRR, switches include IPv6 FlowLabel in ECMP
  - ◆ **Hosts repath a TCP connection by changing its FlowLabel**

PRR uses the IPv6 FlowLabel



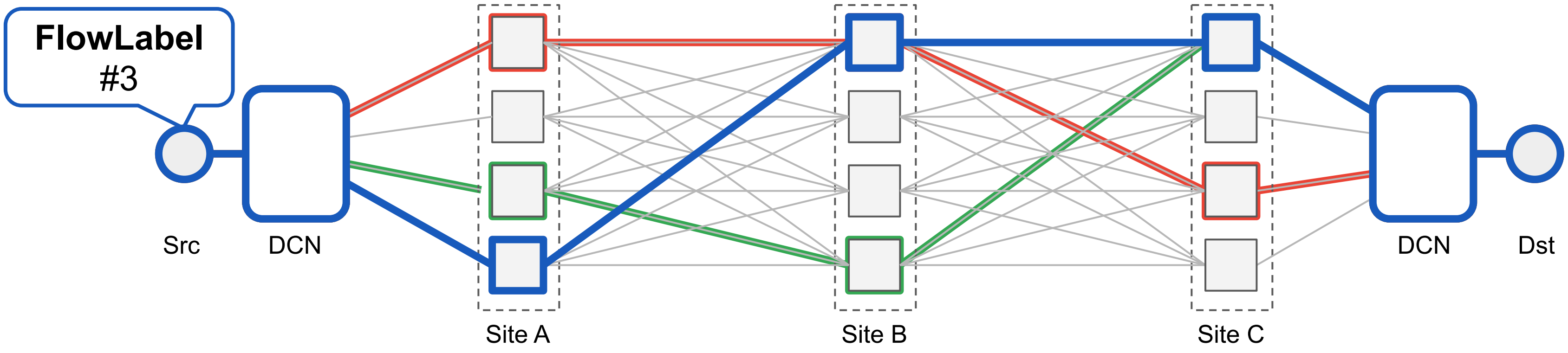
- With traditional ECMP, each TCP connection is fixed to a path
  - Host can’t change the path without changing ports or IP addresses
- With PRR, switches include IPv6 FlowLabel in ECMP
  - **Hosts repath a TCP connection by changing its FlowLabel**

PRR uses the IPv6 FlowLabel



- → With traditional ECMP, each TCP connection is fixed to a path
  - Host can't change the path without changing ports or IP addresses
- → With PRR, switches include IPv6 FlowLabel in ECMP
  - **Hosts repath a TCP connection by changing its FlowLabel**

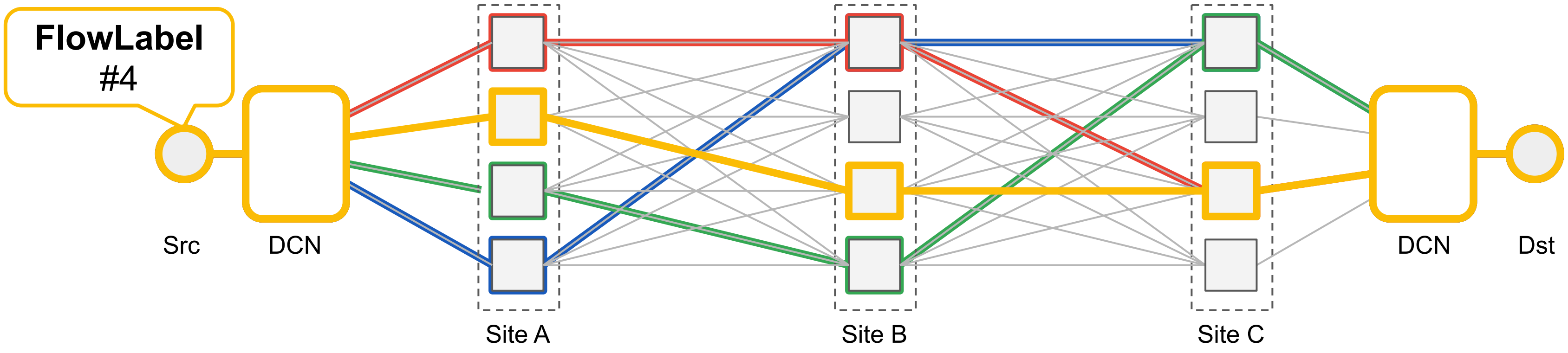
PRR uses the IPv6 FlowLabel



→ With traditional ECMP, each TCP connection is fixed to a path
- Host can't change the path without changing ports or IP addresses

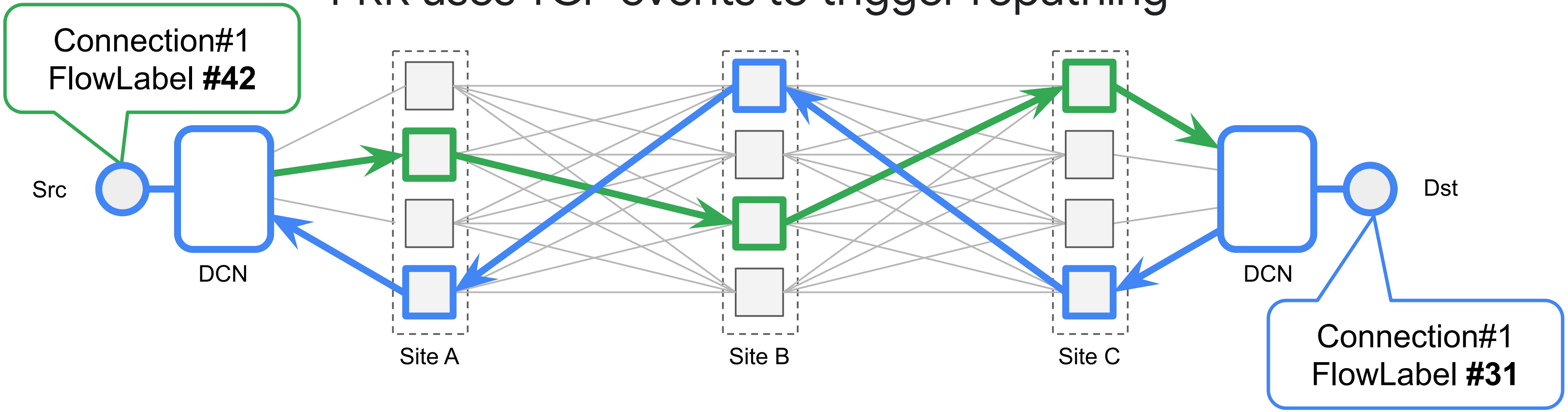
→ With PRR, switches include IPv6 FlowLabel in ECMP
- **Hosts repath a TCP connection by changing its FlowLabel**

PRR uses the IPv6 FlowLabel



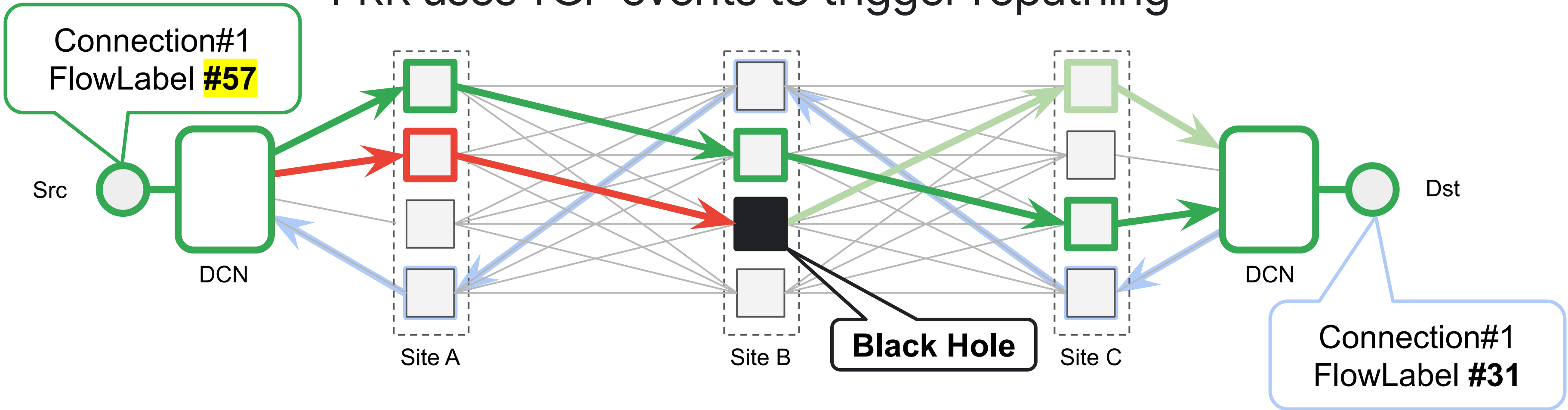
- With traditional ECMP, each TCP connection is fixed to a path
  - Host can't change the path without changing ports or IP addresses
- With PRR, switches include IPv6 FlowLabel in ECMP
  - **Hosts repath a TCP connection by changing its FlowLabel**

PRR uses TCP events to trigger repathing



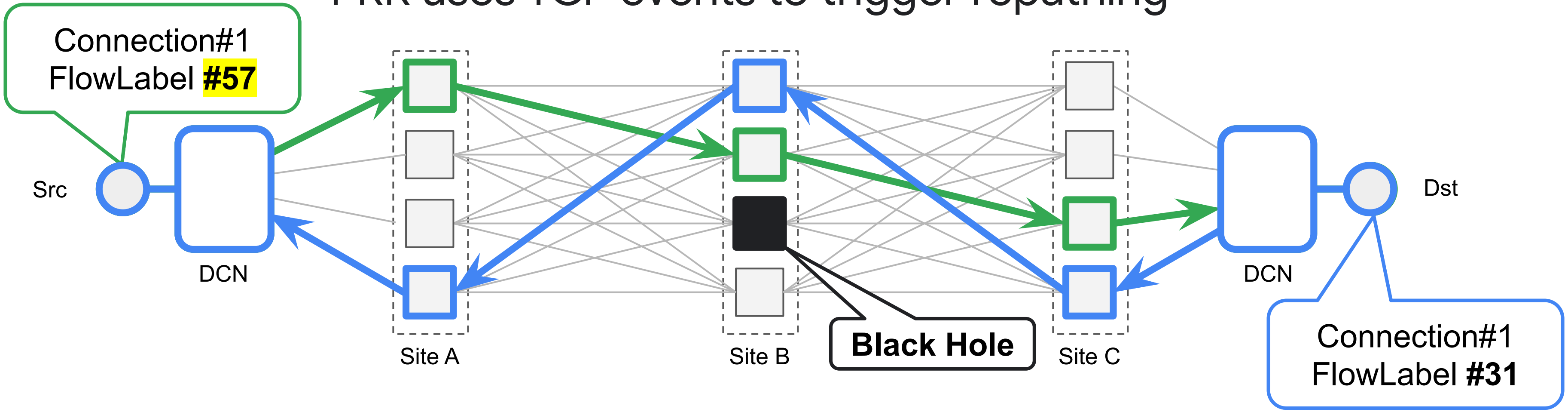
- → Hosts randomly repath (via FlowLabel) on connectivity problem
  - ◆ Sender on RTO, receiver on duplicate data reception; and for SYNs
- → Usually this fixes the fault; if not then try again at next RTO
  - ◆ Host local behavior, independent of routing, TE , SDN, etc.

PRR uses TCP events to trigger repathing



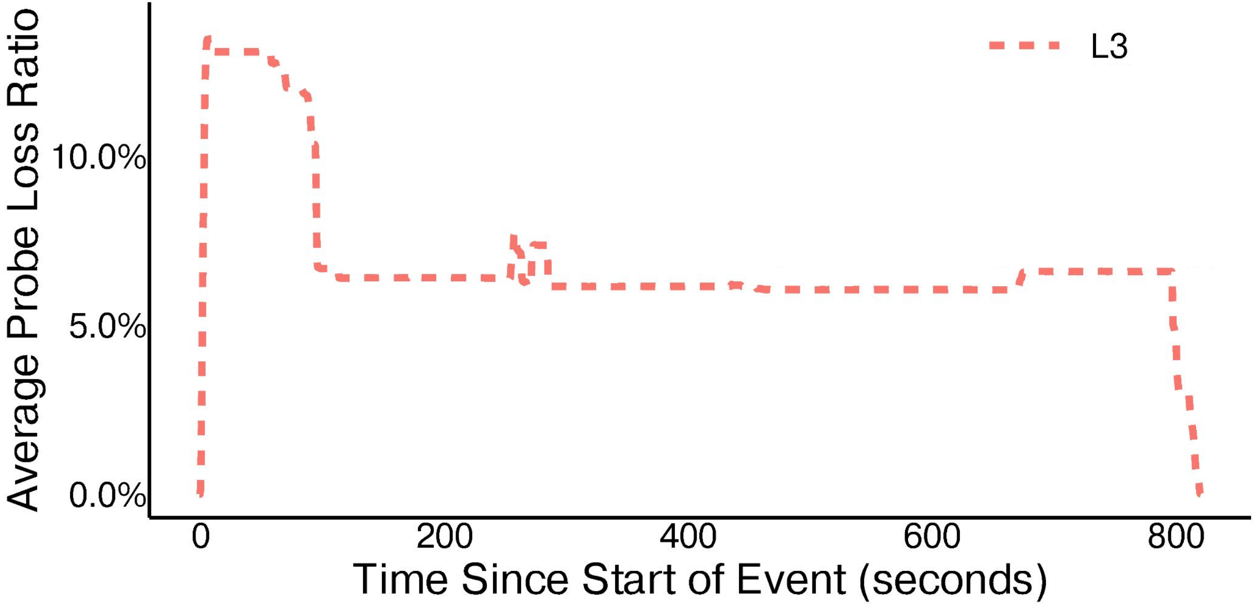
- Hosts randomly repath (via FlowLabel) on connectivity problem
  - Sender on RTO, receiver on duplicate data reception; and for SYNs
- Usually this fixes the fault; if not then try again at next RTO
  - Host local behavior, independent of routing, TE , SDN, etc.

PRR uses TCP events to trigger repathing



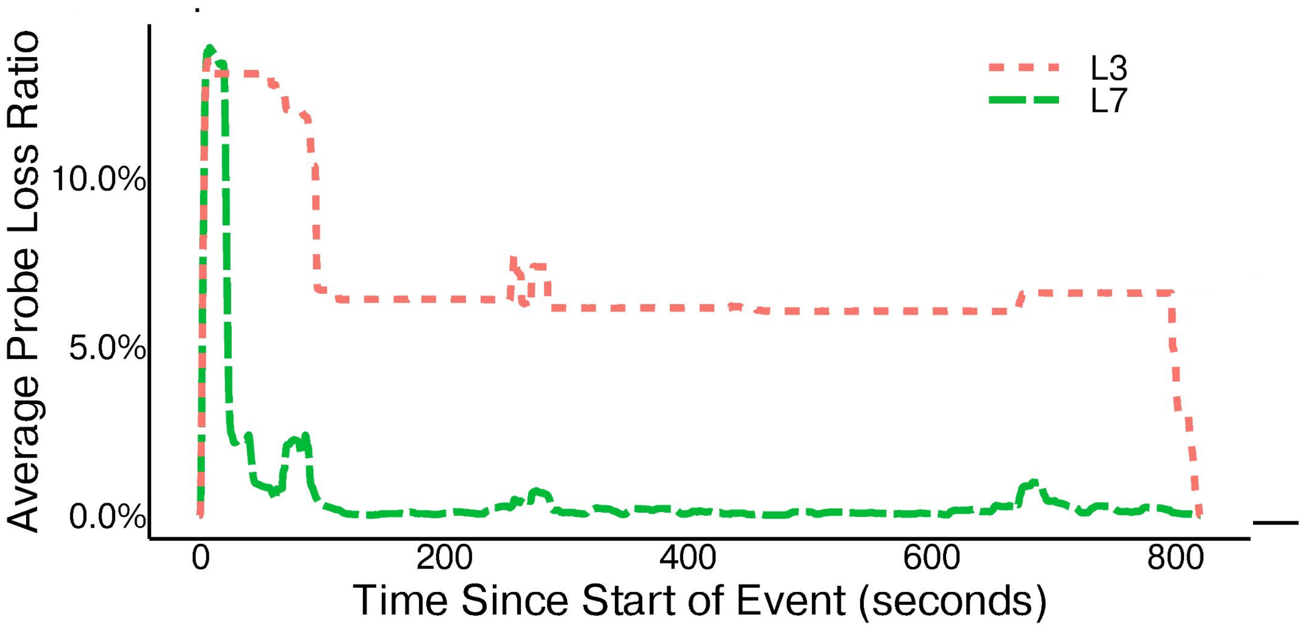
- Hosts randomly repath (via FlowLabel) on connectivity problem
  - Sender on RTO, receiver on duplicate data reception; and for SYNs
- Usually this fixes the fault; if not then try again at next RTO
  - Host local behavior, independent of routing, TE , SDN, etc.

Example B4 Outage



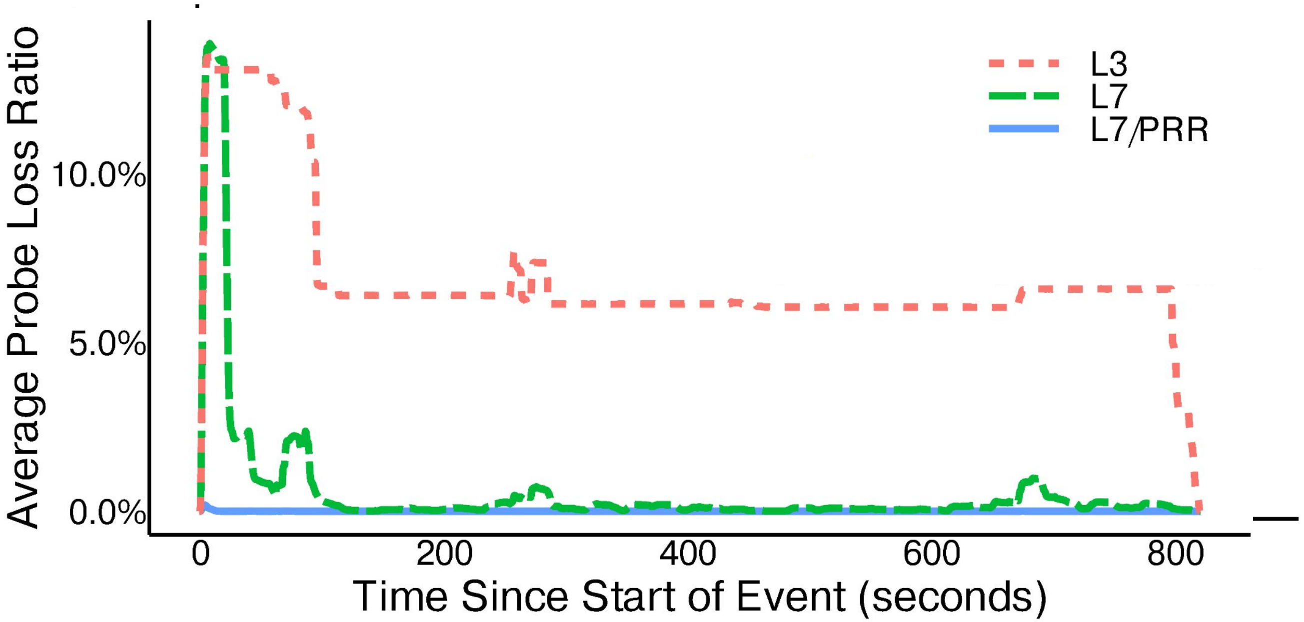
L3 (network) sees 14m outage until higher-level repair

Example B4 Outage



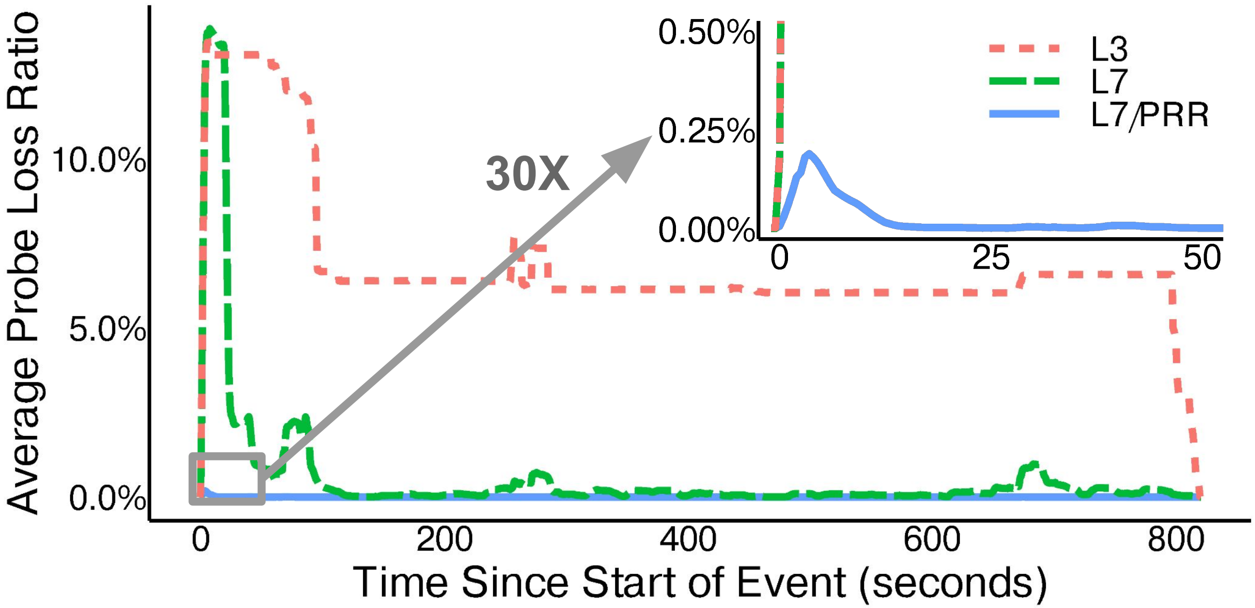
L7 (TCP user) benefits from gradual TCP re-connections

Example B4 Outage



L7 with PRR restores connectivity in seconds

Example B4 Outage



L7 with PRR restores connectivity in seconds

Deployment

- **Nearly zero capex or opex**
  - Transport stack change ~5 lines of code
  - No CPU or memory cost
  - Cloud VM coverage is more sophisticated but still straightforward
- **Incrementally deployable**
  - Upgrade transports, switches, in any order
  - Upstream Linux TCP supports by default
- **Partial rollout delivers good benefits**
  - Overall Google WAN observes 93% reduction in cumulative outage minutes

Great incentives for adoption when you use IPv6

Conclusion

→ **Use the IPv6 FlowLabel for ECMP**

- Intended architecture; lets hosts control paths

→ **Use routing to maintain diverse paths**

- Outages are partial connectivity issues

→ **Use host repathing for repair, not FRR**

- Quickly removes most outage time

Change how we design networks for availability