

CISCO *Live!*

GO BEYOND

#CiscoLiveAPJC



Network Best Practices for Artificial Intelligence Data Centre

Nemanja Kamenica
Technical Marketing Engineer
BRKDCN-2921

CISCO *Live!*

#CiscoLiveAPJC



Cisco Webex App

Questions?

Use Cisco Webex App to chat with the speaker after the session

How

- 1 Find this session in the Cisco Live Mobile App
- 2 Click “Join the Discussion”
- 3 Install the Webex App or go directly to the Webex space
- 4 Enter messages/questions in the Webex space

Webex spaces will be moderated by the speaker until November 15, 2024.

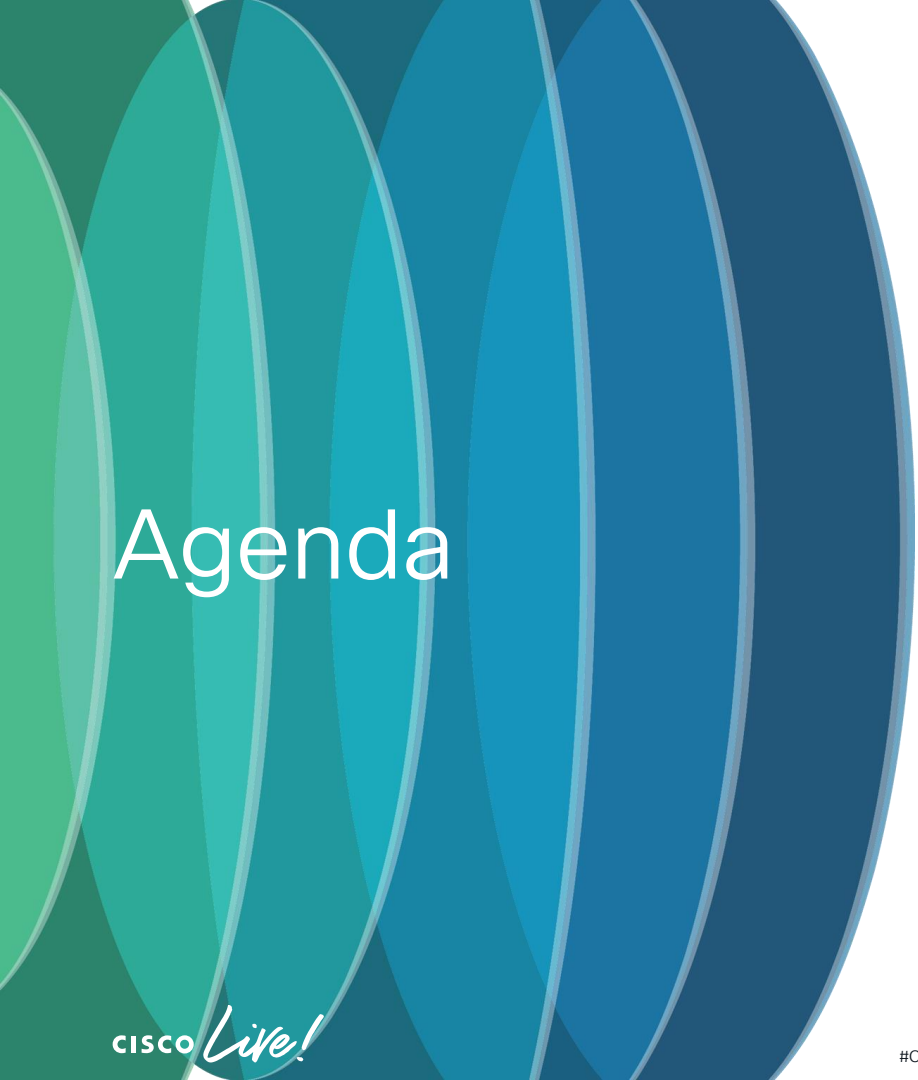


<https://ciscolive.ciscoevents.com/ciscolivebot/#BRKDCN-2921>



Abstract

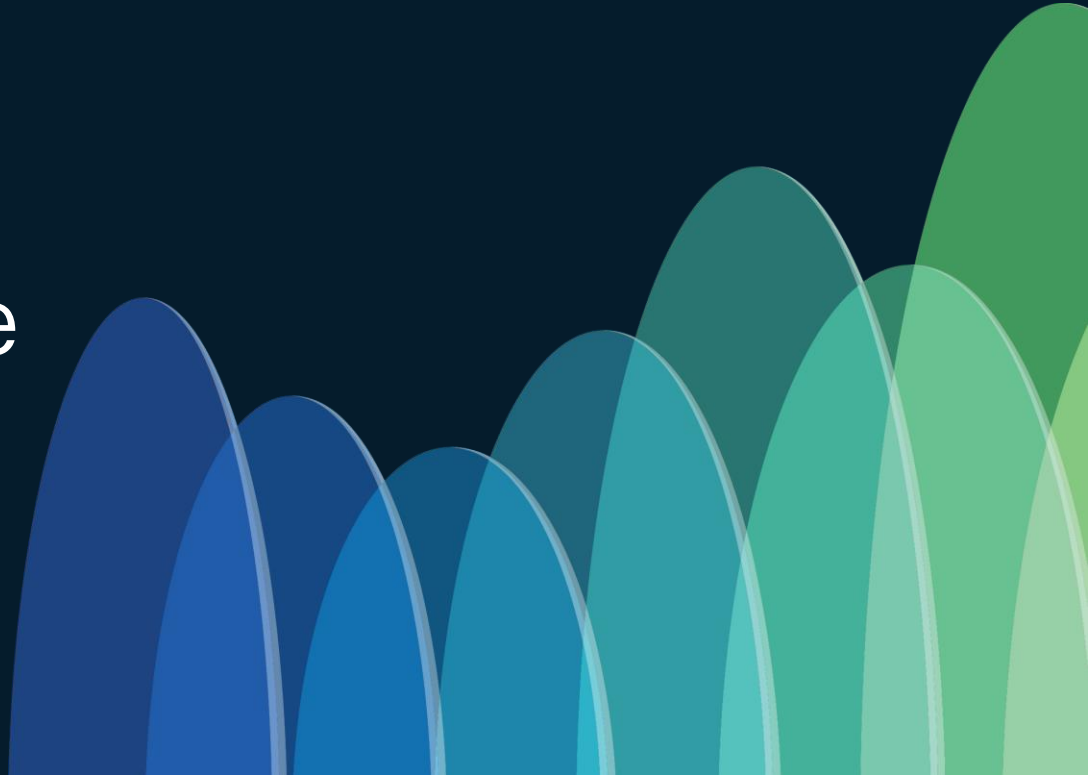
- AI/ML is dominating the news cycle. Using the right infrastructure is critical to run AI/ML workloads. Come to learn how to build a modern network suited to run your company's AI/ML workloads, manage congestion on Cisco Nexus 9000 switches, automate your network, and have visibility in AI workloads with Cisco Nexus Dashboard.



Agenda

- Why AI is important today and, in the future
- Network For AI Cluster
- Traffic Load-balancing
- Automation and Visibility
- The Blueprint For Today

Why AI is important today and, in the future



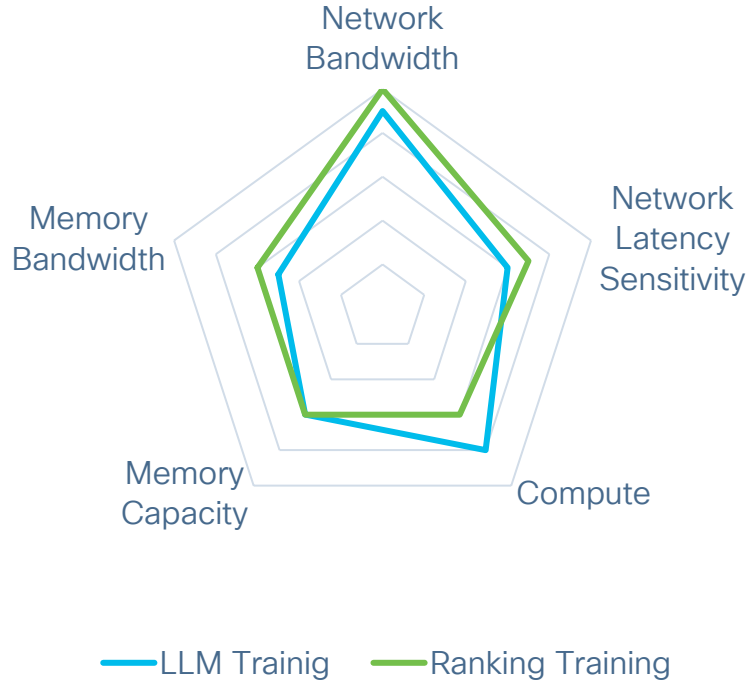
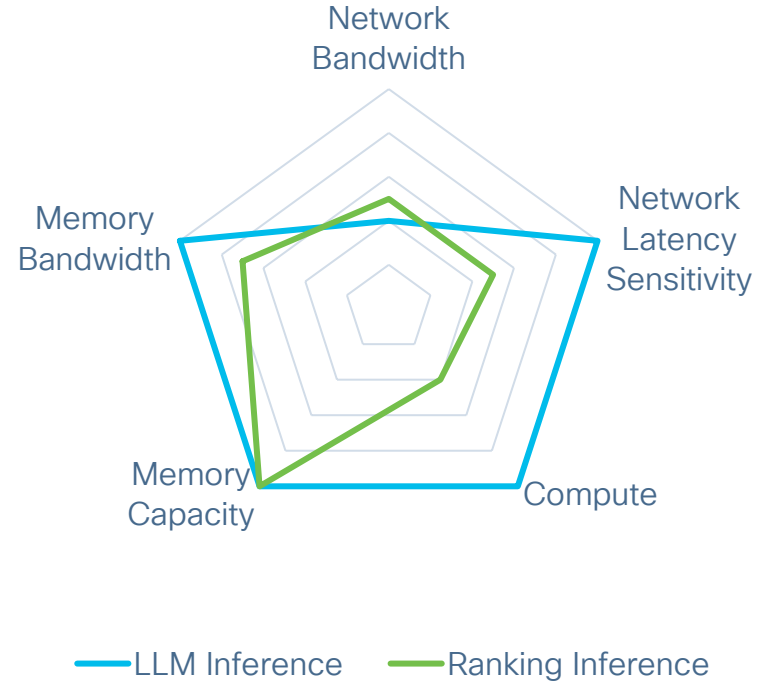
AI/ML can help many industries

Healthcare	Financial Services	Public Sector	Media and Entertainment	Manufacturing	Retail
Medical Risk Prediction	High Frequency Trading Analysis	Intelligent Public Transport	Speech Recognition	Visual Inspection	Personalized Recommendation
Early diagnostics	Quant Research	Security Log Analytics	Natural Language Processing	Anomaly Detection	Demand Forecasting
Medical Research	Fraud and Risk Analytics	Disaster Recovery Assistance	Content Classification	Asset Management	Visual Search

AI cluster types and interaction with network

	Distributed Training	Production Inference
Node to Node Bandwidth	High	Low
Key Metric	Training time of a model	High Availability and Latency
Operational Mode	Model training is offline	Usually online, requires real time response
Infrastructure requirement	Large network with many GPU/CPU hosts	Smaller network with mid size of CPU/GPU hosts

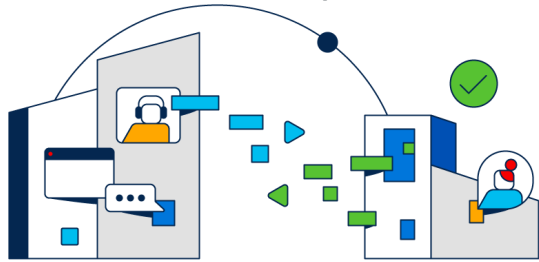
Training vs Inference



Where do you run your AI/ML training cluster

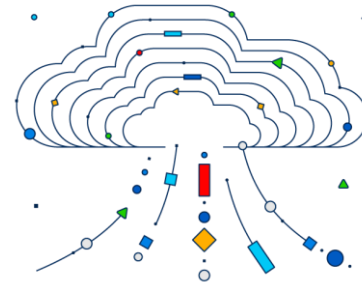
- On Prem

- Always available for enterprise to use
- Flexibility for large enterprise to leverage same cluster for different functions
- Data is stored on prem



- In Public Cloud

- Provides flexibility, pay for what you need
- Cost will grow with more data and training
- Challenge: Cost of egress data from the cloud



Large-scale distributed training

- Key Challenge of Training Cluster
 - Model Doubles every 2 months
 - Bigger model, higher accuracy
 - Most common single training runs on 512 GPUs



- Cluster Key Components
 - Compute Nodes
 - Network
 - Distributed File System/Storage
 - Job Scheduling and Orchestration
 - Software Framework for AI model



Large-scale distributed training

- Key Challenge of Training Cluster
 - Model Doubles every 2 months
 - Bigger model, higher accuracy
 - Most common single training runs on 512 GPUs



- Cluster Key Components
 - Compute Nodes
 - Network
 - Distributed File System/Storage
 - Job Scheduling and Orchestration
 - Software Framework for AI model



Network For AI Cluster



CISCO *Live!*



AI training network



For RoCEv2 transport the network must provide high throughput and low latency by passing CPU/GPU



The Cisco Nexus 9000 switches, provide the required low latency, and with up to 25.6Tbps of bandwidth per ASIC, to satisfy AI/ML clusters requirements

Shipping feature set with many customers in production

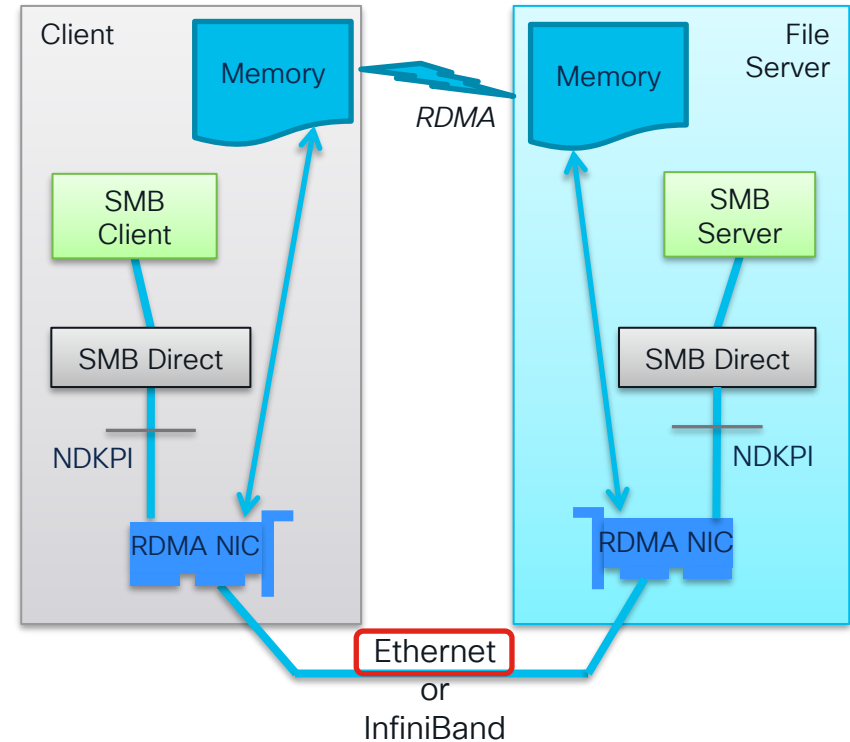
Nexus 9000 data center portfolio

- Nexus 9000 is Cisco's flagship data center networking portfolio
- Offer set of speed from 100Mbps to 400Gbps
- Runs in Standalone and ACI mode
- Based on Cisco Silicon



RDMA – remote data memory access

- Allows application software to communicate directly with the hardware (RDMA NIC)
- Bypasses OS stack
- RDMA delivers, low latency, high throughput, zero copy capabilities
- RDMA Hardware Technologies
 - RoCE: RDMA over Converged Ethernet
 - iWARP: RDMA over TCP/IP
 - Infiniband



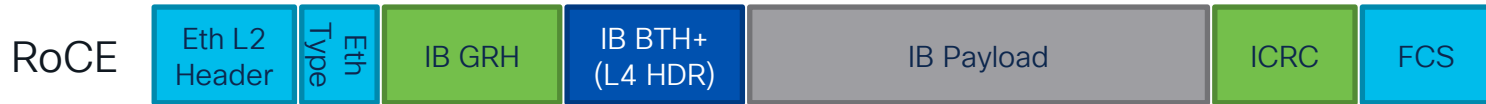
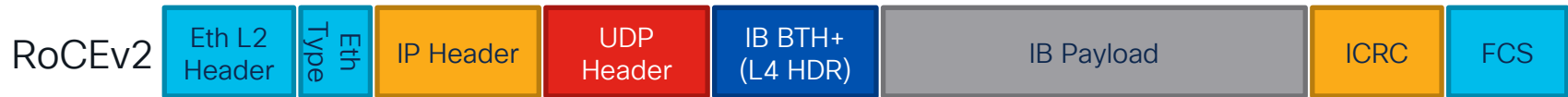


RDMA – remote data memory access

- Zero-Copy: applications can transfer data from one host to another by bypassing the software network stack. Data is being read & written, sent/received directly to memory buffers.
- Kernel Bypass: applications can perform data transfer directly from user space without the need to perform context switches over the kernel.
- CPU Bypass / Offload
 - Applications can access remote memory without any host CPU cycles in the remote machine.
 - The caches in the remote CPU(s) won't be filled with the accessed memory content.

RoCEv2 - basics

- Extension of RoCE protocol that involves a simple modification of the RoCE packet format
- Carry IP header and UDP header that serves as a stateless encapsulation layer for RDMA transport over IP



Source: https://en.wikipedia.org/wiki/RDMA_over_Converged_Ethernet

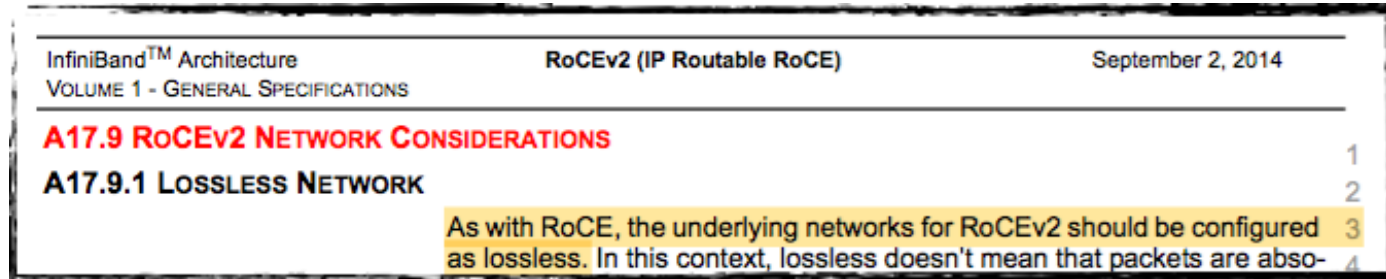


RoCEv2

- Uses well-known UDP Destination Port (dport) value 4791
- UDP Source Port (sport) serves as opaque flow identifier that can be used by networking infrastructure for packet forwarding optimizations (e.g., ECMP)
- Supports both IPv4 and IPv6
- Makes use of ECN field in IPv4/6 header for signaling of congestion

RoCEv2 End-To-End lossless behavior

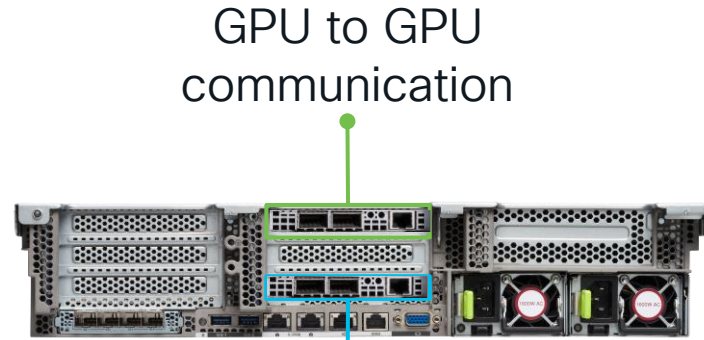
- Requires PFC to be enabled for RoCEv2 transport
- Traffic priority to be preserved between Layer 2 and Layer 3 network
 - Packet/Flow identification follows standard practices of IP/Ethernet networks (i.e., DSCP/802.1Q)
- ECN marking (WRED or consider AFD)
- Configure ETS
 - 802.1Qaz ETS



Networks – Why do I need them?



UCS C240 M7 Rack Server
Up to 2 GPUs



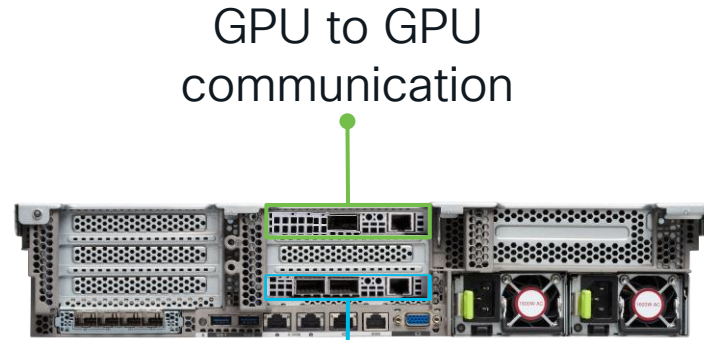
GPU to GPU
communication

Front-end and
Storage Network

Networks – Why do I need them?



UCS C240 M7 Rack Server
Up to 2 GPUs



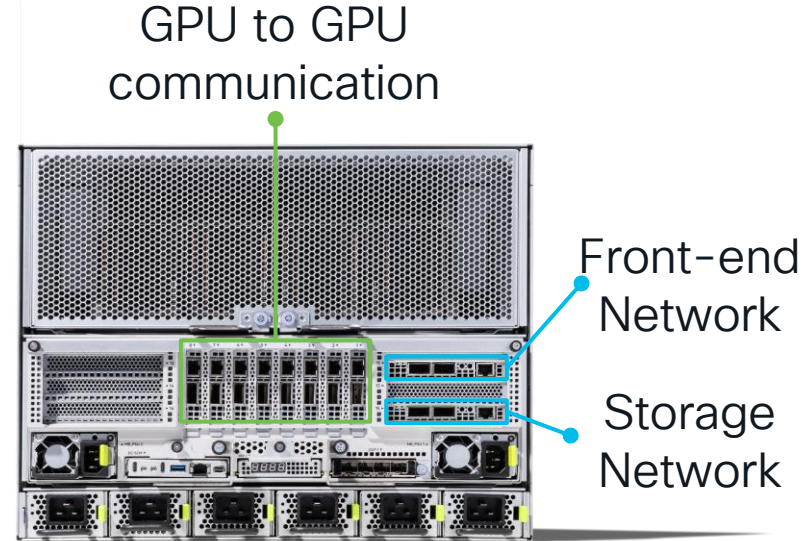
GPU to GPU
communication

Front-end and
Storage Network

Networks – Why do I need them?



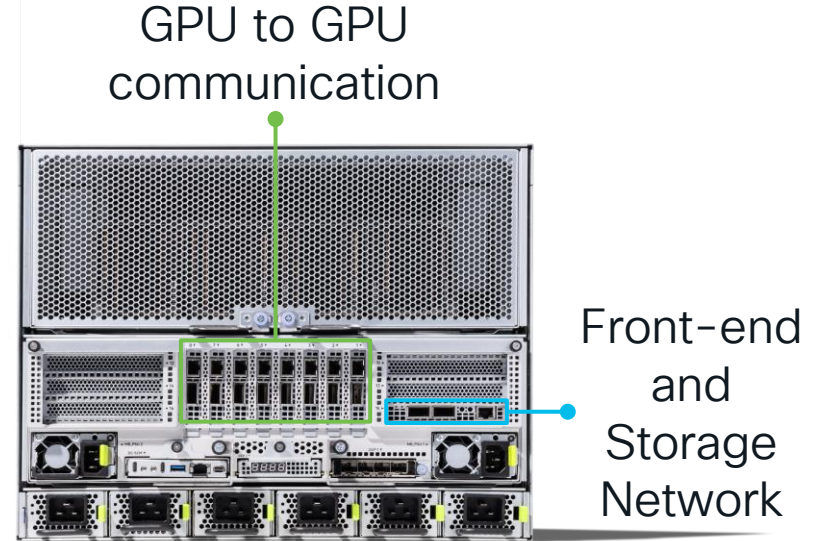
UCS C885A M8 Rack Server
8 GPUs



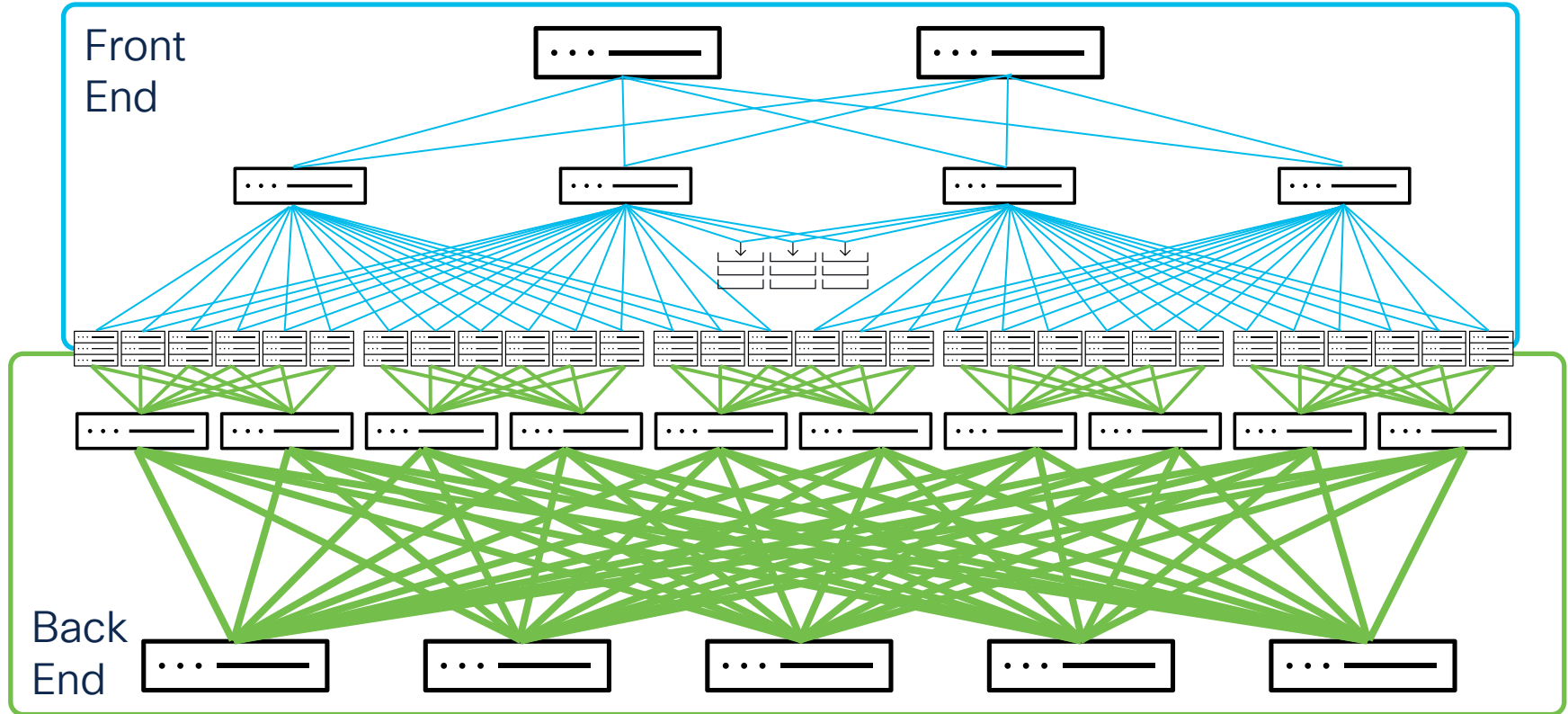
Networks – Why do I need them?



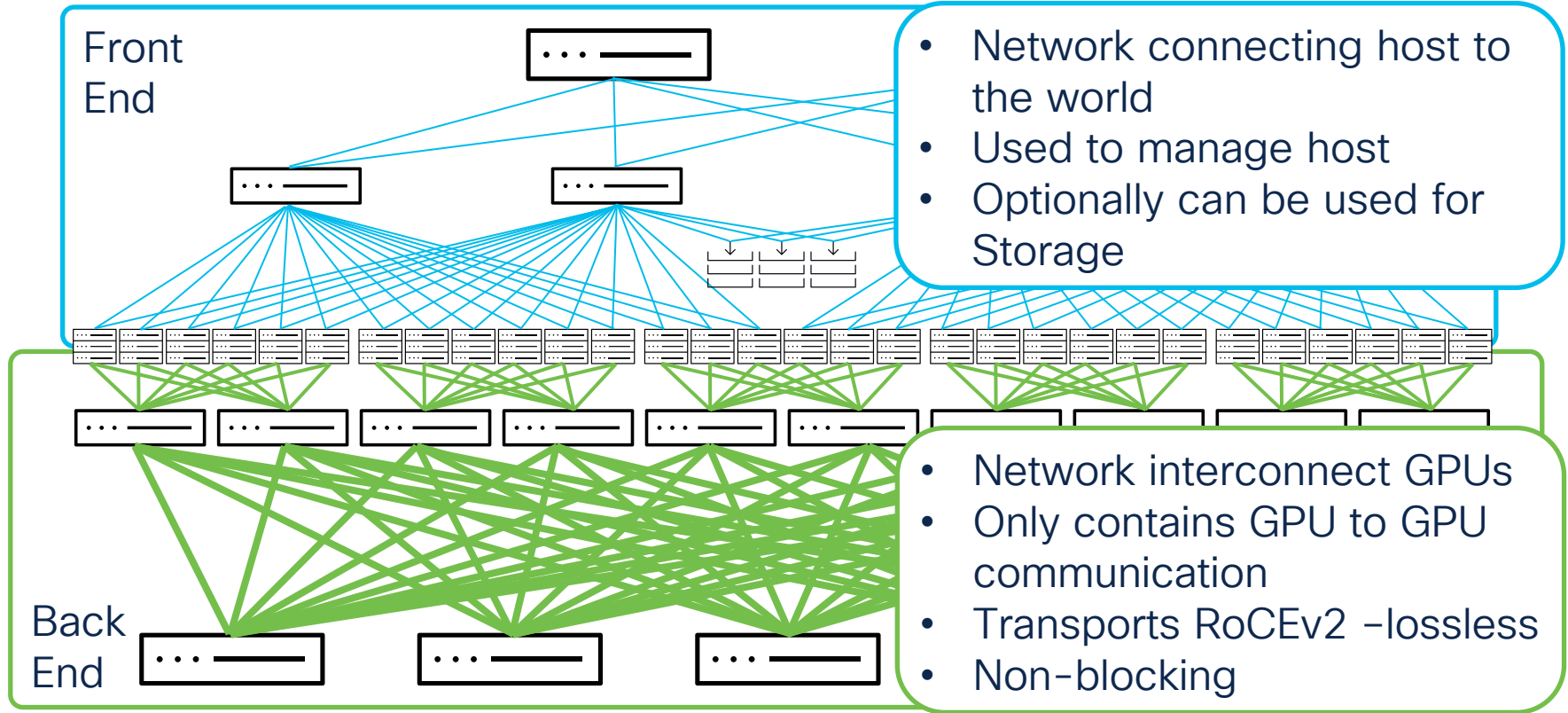
UCS C885A M8 Rack Server
8 GPUs



Non-blocking network

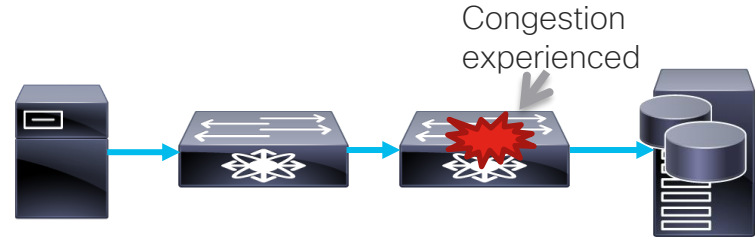


Non-blocking network



Explicit Congestion Notification (ECN)

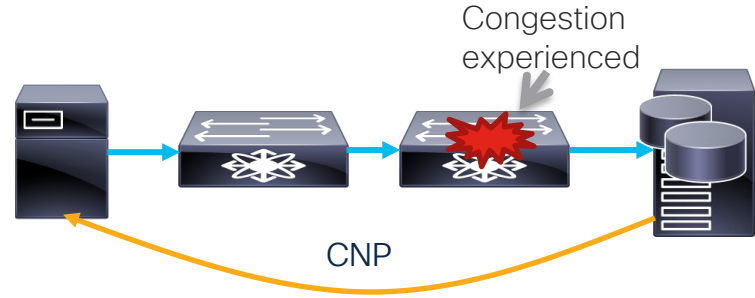
- IP Explicit Congestion Notification (ECN) is used for congestion notification.
- ECN enables end-to-end congestion notification between two endpoints on IP network
- ECN uses 2 LSB of Type of Service field in IP header



ECN	ECN Behavior
00	Non ECN Capable
10	ECN Capable Transport (0)
01	ECN Capable Transport (1)
11	Congestion Encountered

Explicit Congestion Notification (ECN)

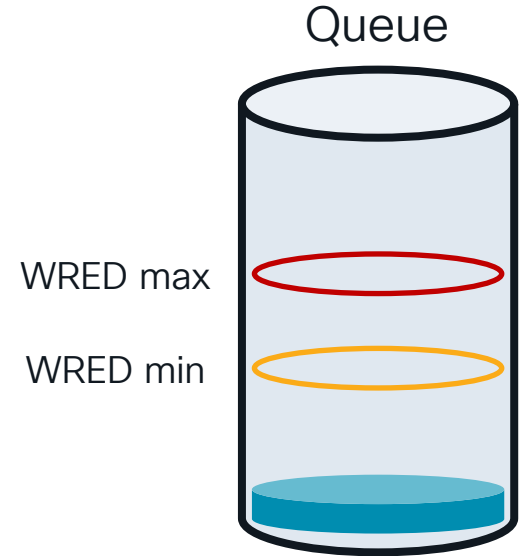
- IP Explicit Congestion Notification (ECN) is used for congestion notification.
- ECN enables end-to-end congestion notification between two endpoints on IP network
- ECN uses 2 LSB of Type of Service field in IP header
- In case of congestion, ECN gets transmitting device to reduce transmission rate using Congestion Notification Packet (CNP) without pausing traffic.



ECN	ECN Behavior
00	Non ECN Capable
10	ECN Capable Transport (0)
01	ECN Capable Transport (1)
11	Congestion Encountered

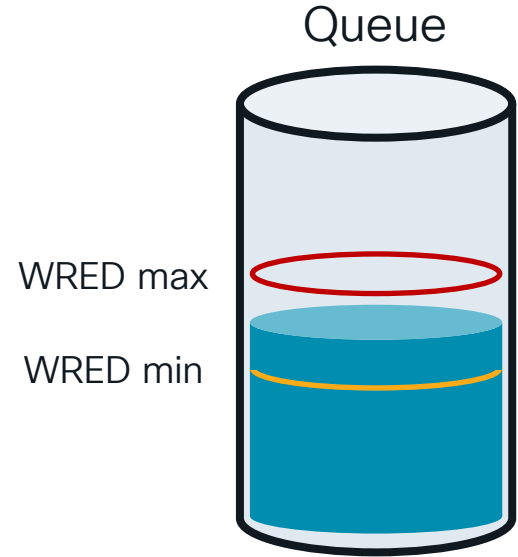
How does WRED ECN work?

- WRED (Weighted Random Early Detection) is used to signalize severity of congestion
- ECN is not marked when buffer usage is below WRED min threshold



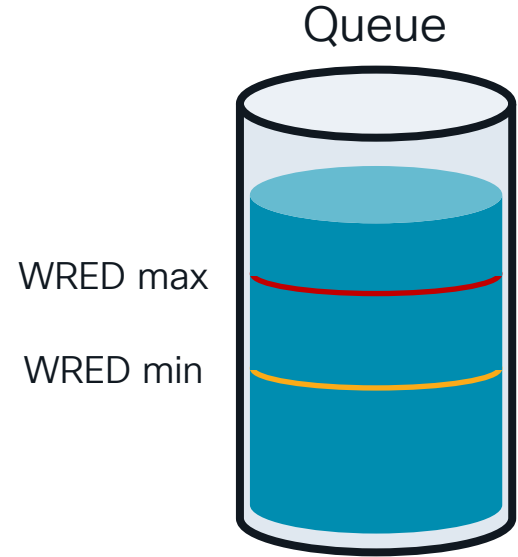
How does WRED ECN work?

- WRED (Weighted Random Early Detection) is used to signalize severity of congestion
- ECN is not marked when buffer usage is below WRED min threshold
- When buffer usage is minimal threshold, Congestion Encountered will be marked on N number of randomly selected packets (probability parameter)

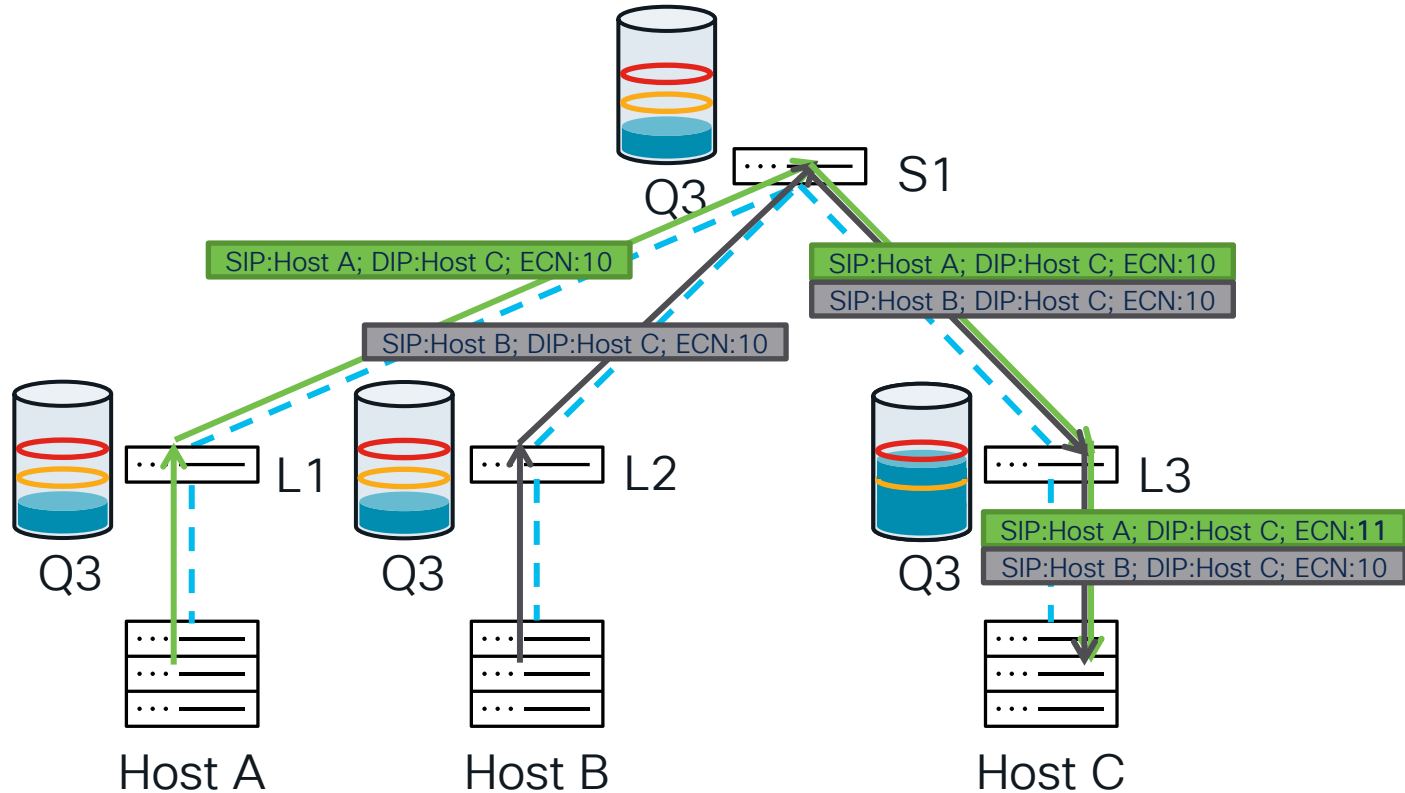


How does WRED ECN work?

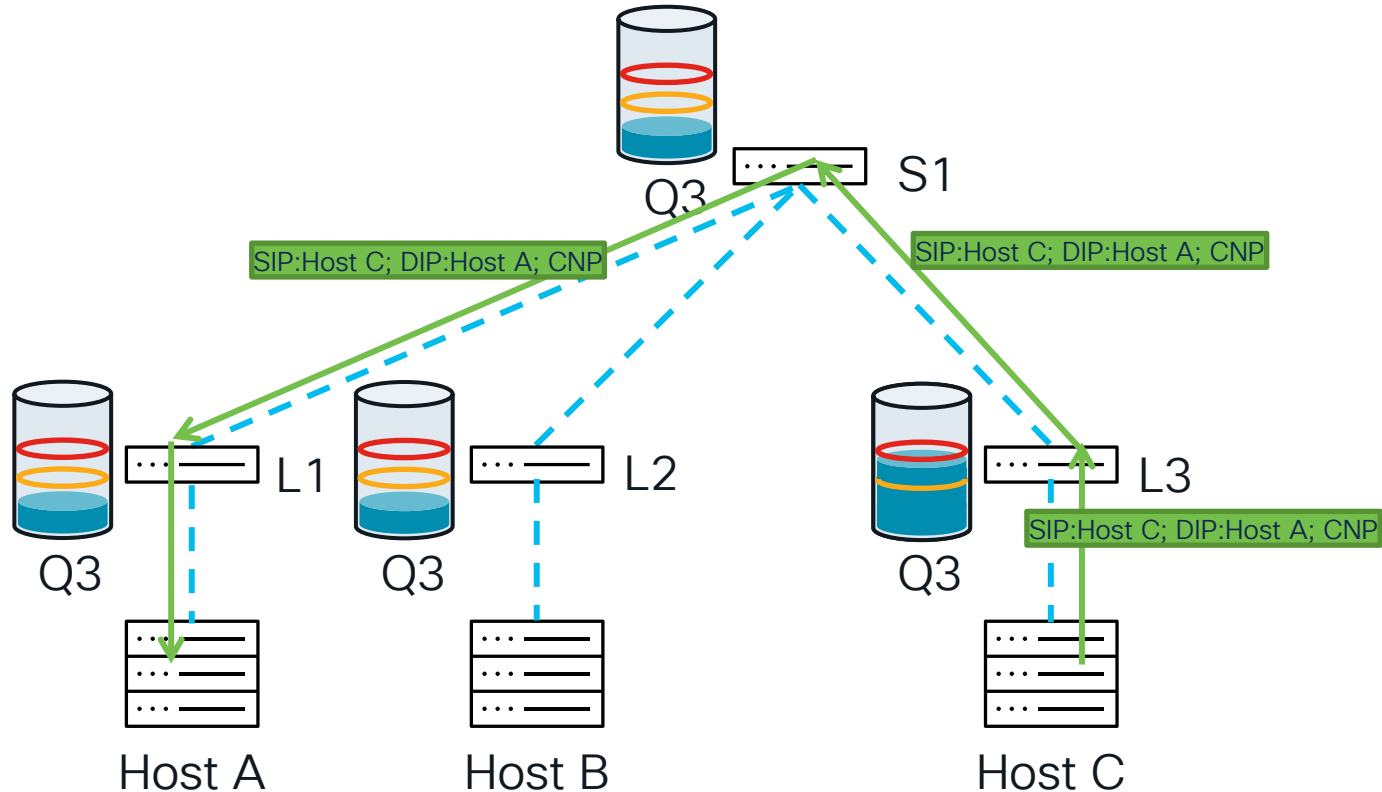
- WRED (Weighted Random Early Detection) is used to signalize severity of congestion
- ECN is not marked when buffer usage is below WRED min threshold
- When buffer usage is minimal threshold, Congestion Encountered will be marked on N number of randomly selected packets (probability parameter)
- After buffer usage crosses MAX threshold, every ECN capable packet will be marked with Congestion Encountered



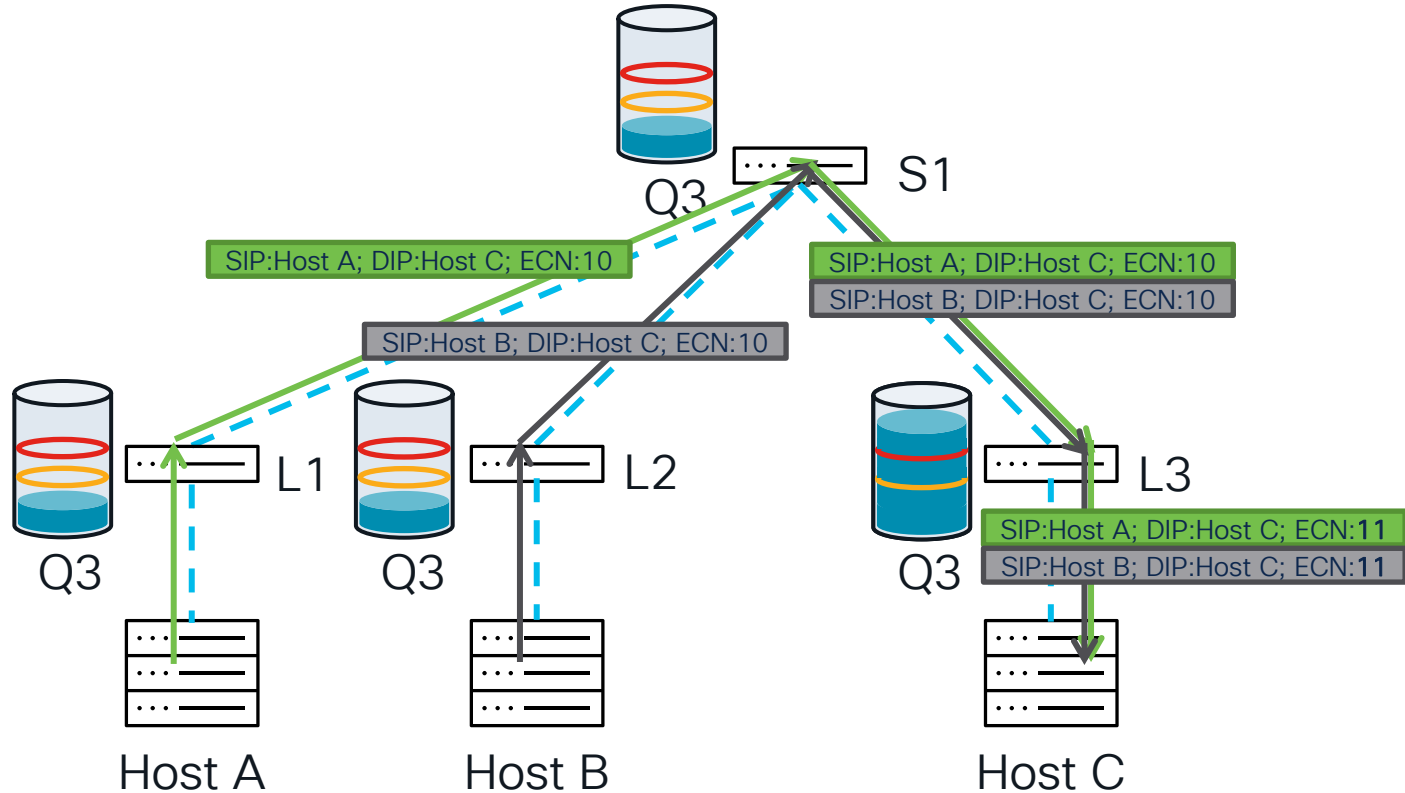
Explicit Congestion Notification – End to End



Explicit Congestion Notification – End to End



Explicit Congestion Notification – End to End

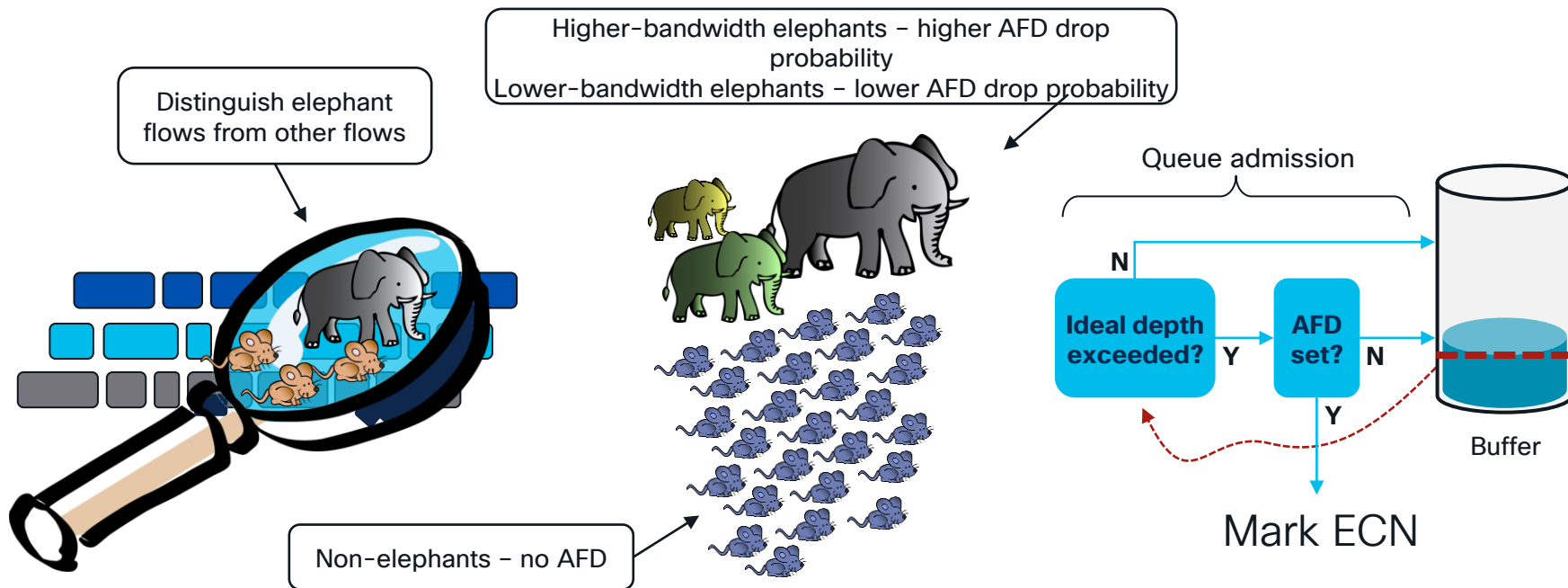


CISCO *Live!*



Approximate Fair Drop (AFD)

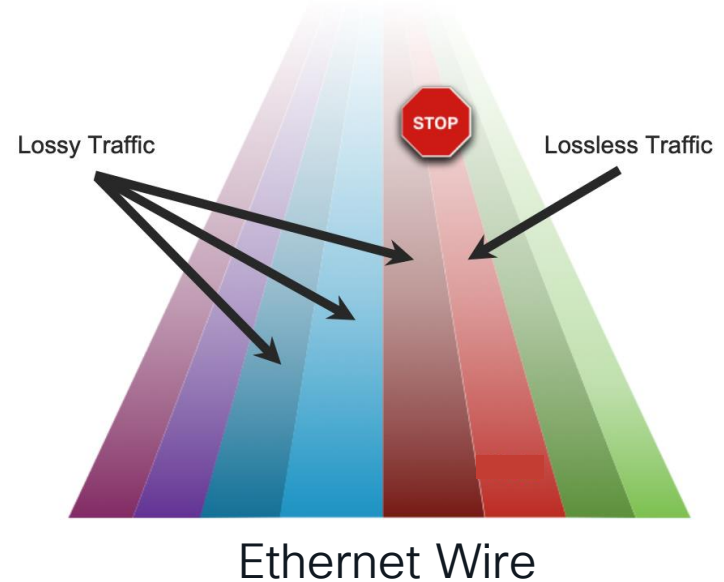
Maintain throughput while minimizing buffer consumption by elephant flows – **keep buffer state as close to the ideal as possible**



Priority Flow Control

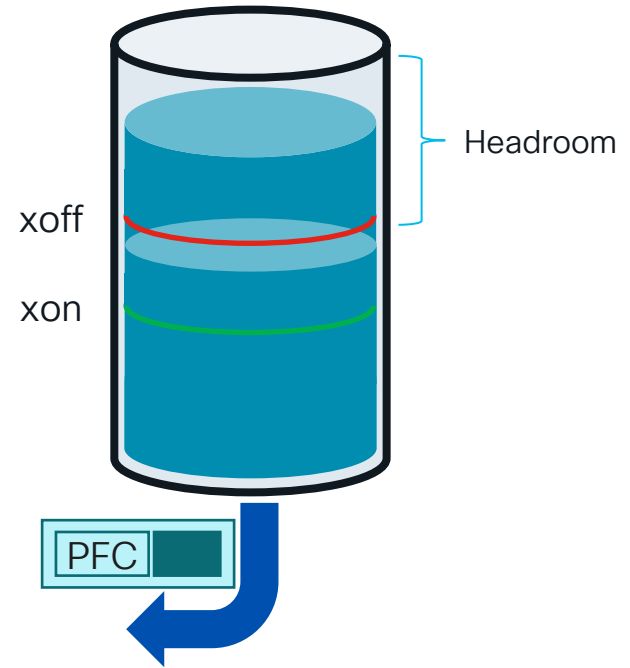
Flow Control Mechanism – 802.1Qbb

- A.k.a "Lossless Ethernet"
- PFC enables Flow Control on a Per-Priority basis
- PFC is also called Per-Priority-Pause
- Therefore, we have the ability to have lossless and lossy priorities at the same time on the same wire
- Allows traffic to operate over a lossless priority independent of other priorities
- Other traffic assigned to other priority will continue to transmit and rely on upper layer protocols for retransmission

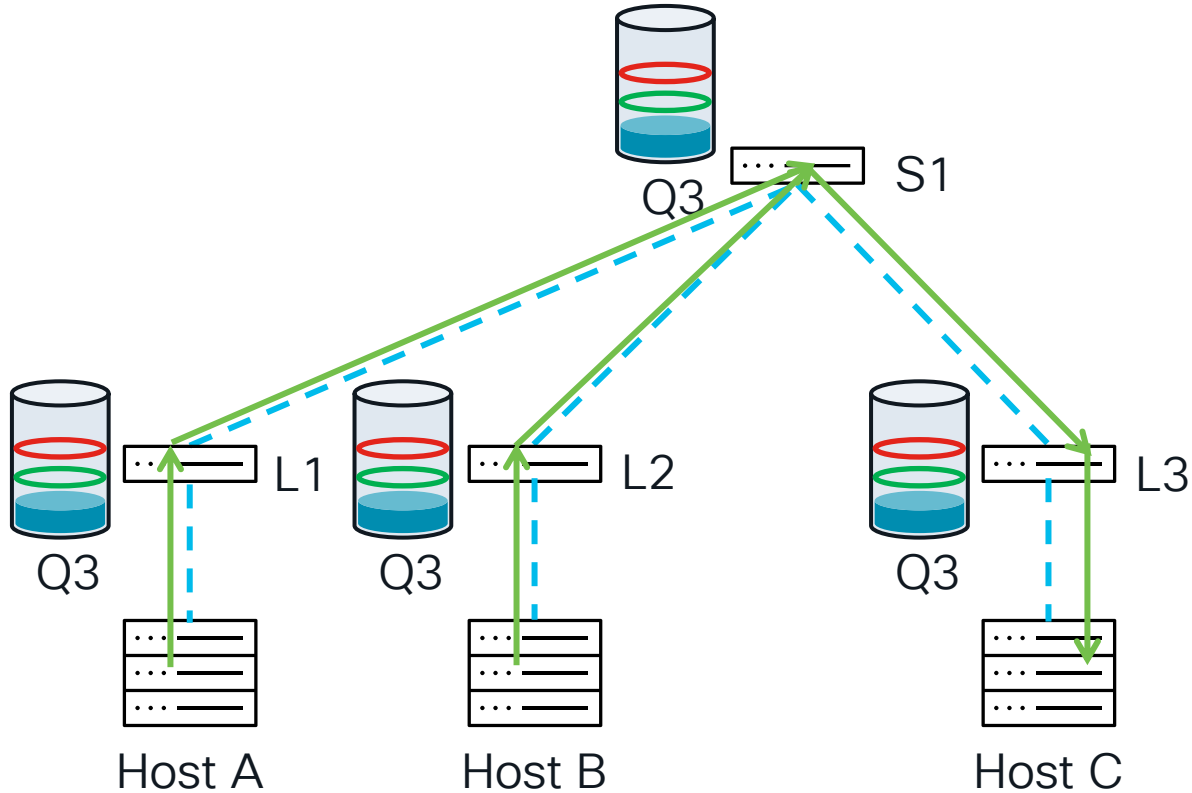


PFC – How pause frames are sent

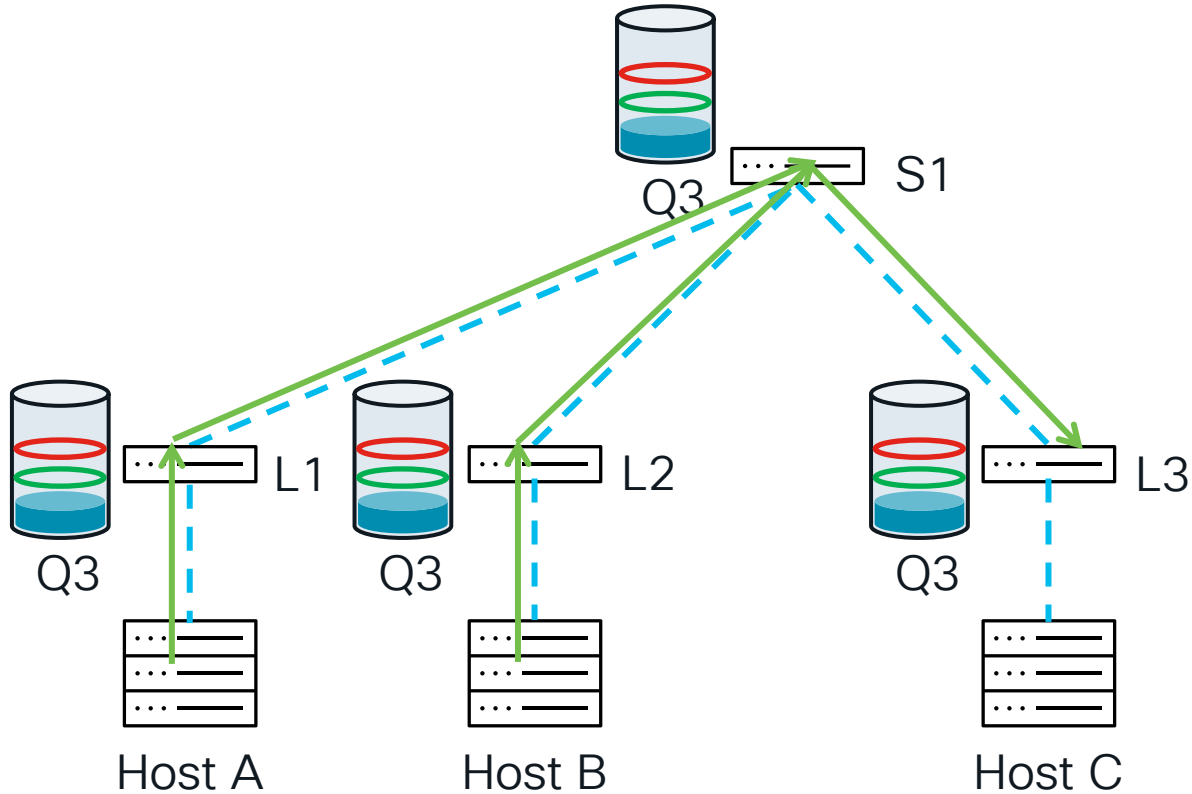
- PFC sets thresholds in no-drop queue
- Headroom is present to accommodate “in flight” packets
- Under congestion, traffic is buffered in non-drop queue
- PFC frames are sent toward sender after queue utilization exceeds *xoff* threshold
- While draining the queue, and utilization is below *xon* threshold system will stop sending PFC frames



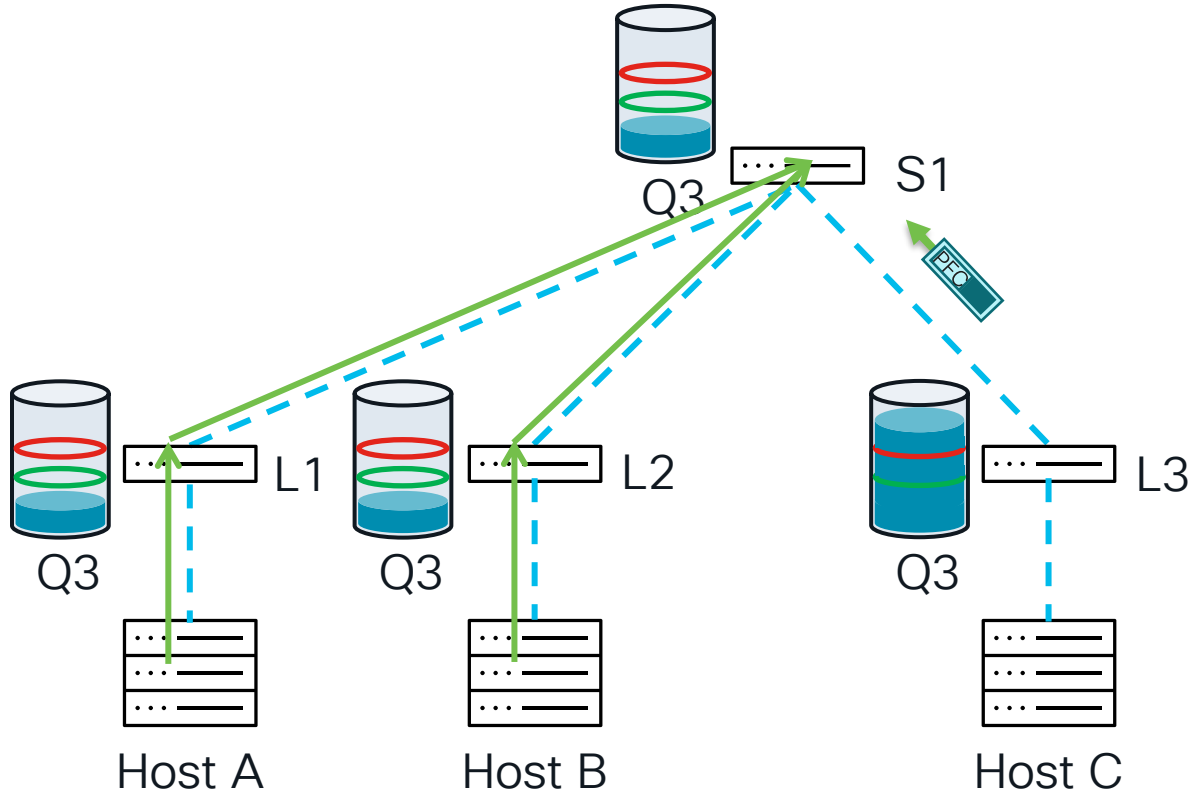
PFC hop by hop



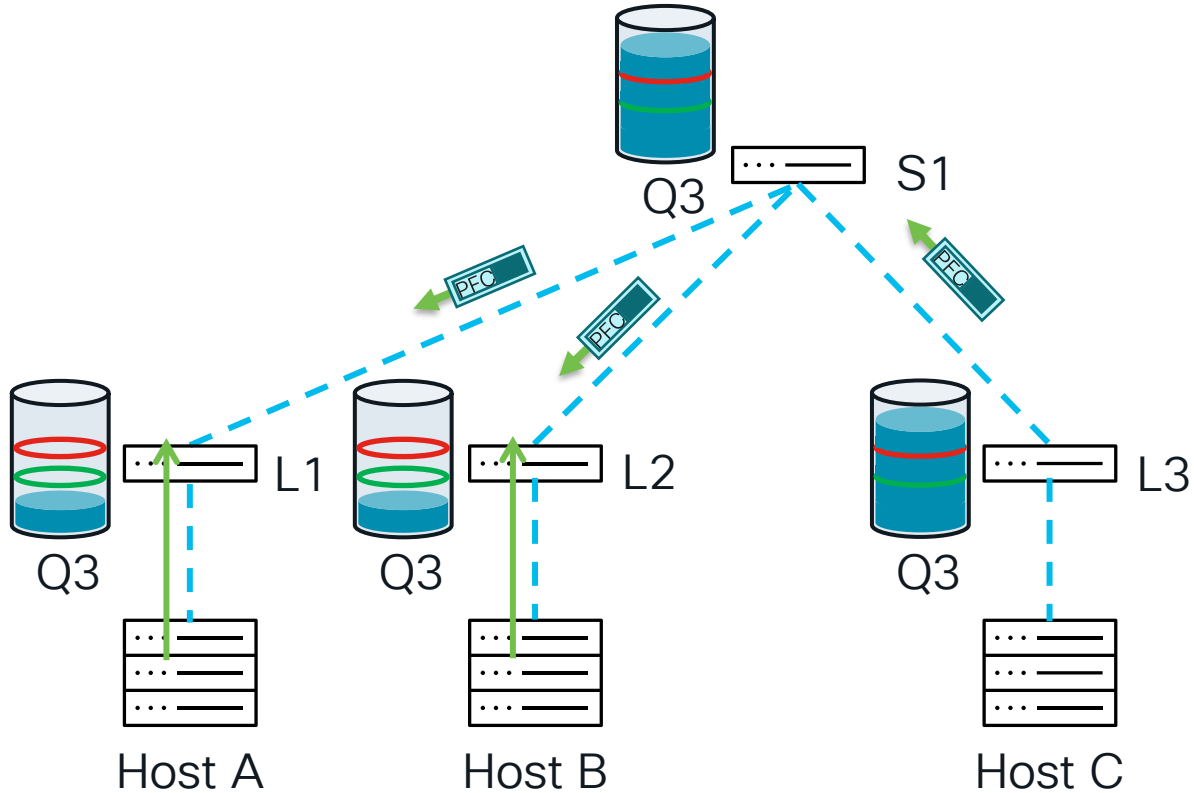
PFC hop by hop



PFC hop by hop



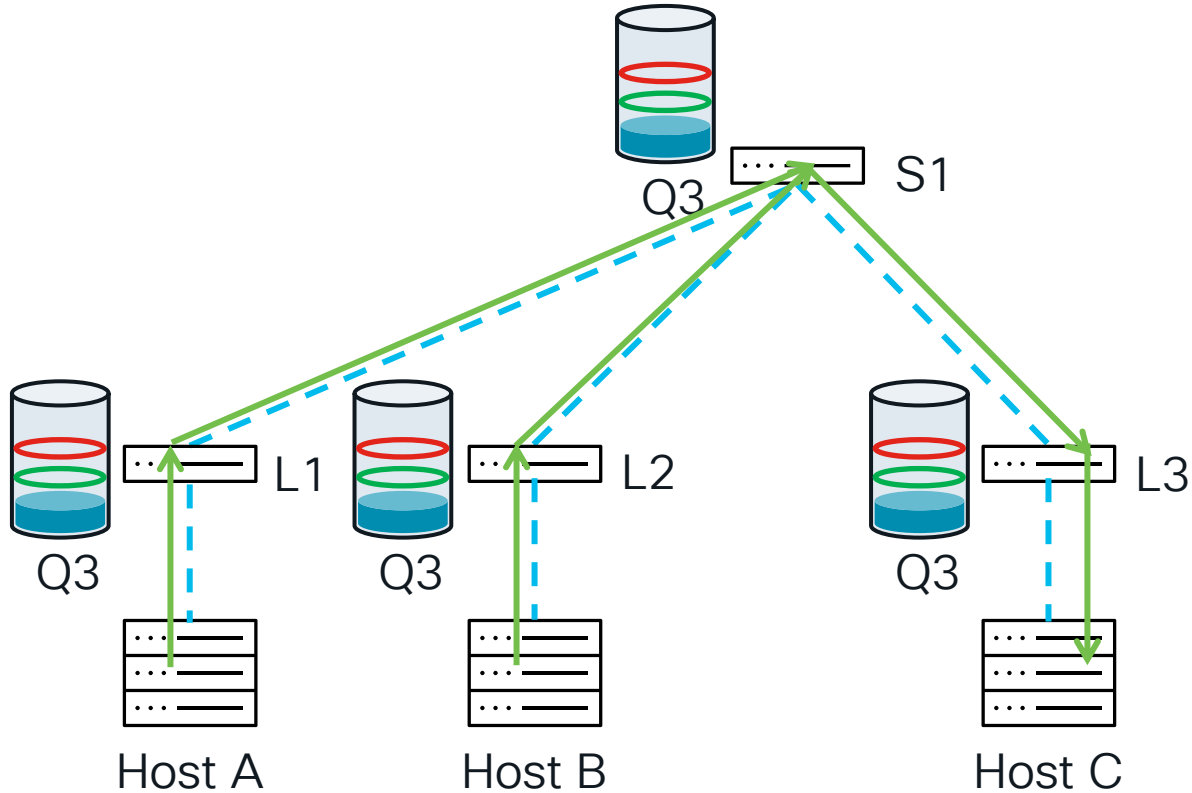
PFC hop by hop



CISCO *Live!*

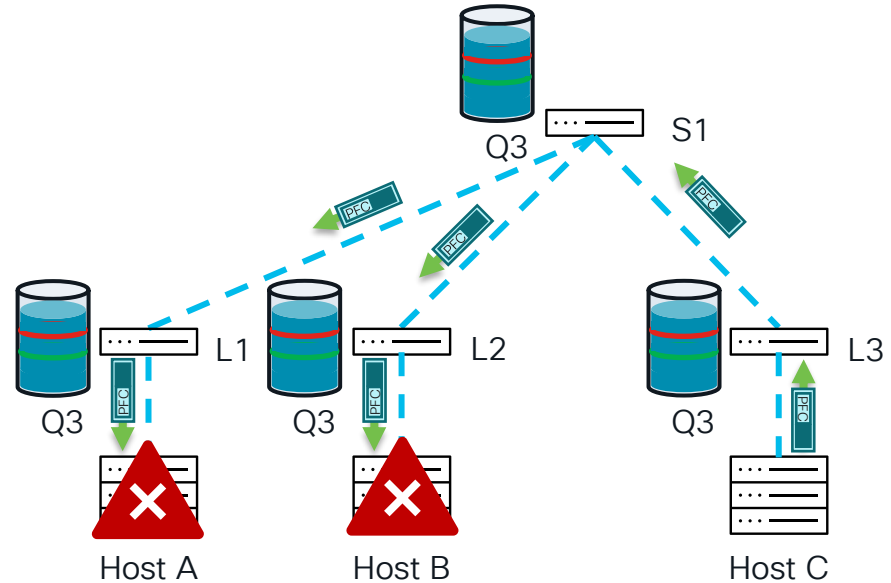


PFC hop by hop



PFC storm

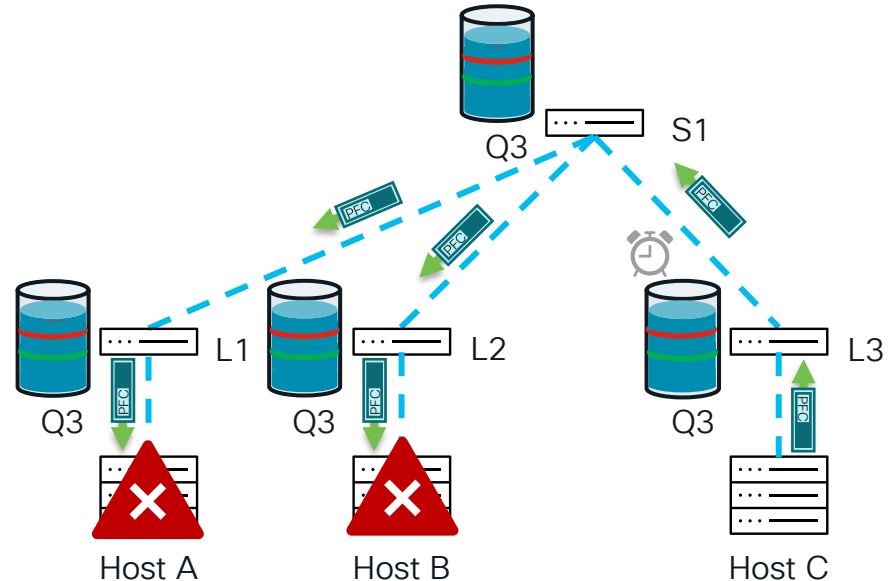
- In occasion of NIC or network malfunction, PFC storm can be triggered to send continues PFC frames in the network
- Network will propagate those frame to all senders
- PFC storm will stop traffic coming from sender
- PFC watchdog can drain the queue



https://www.microsoft.com/en-us/research/wp-content/uploads/2016/11/rdma_sigcomm2016.pdf

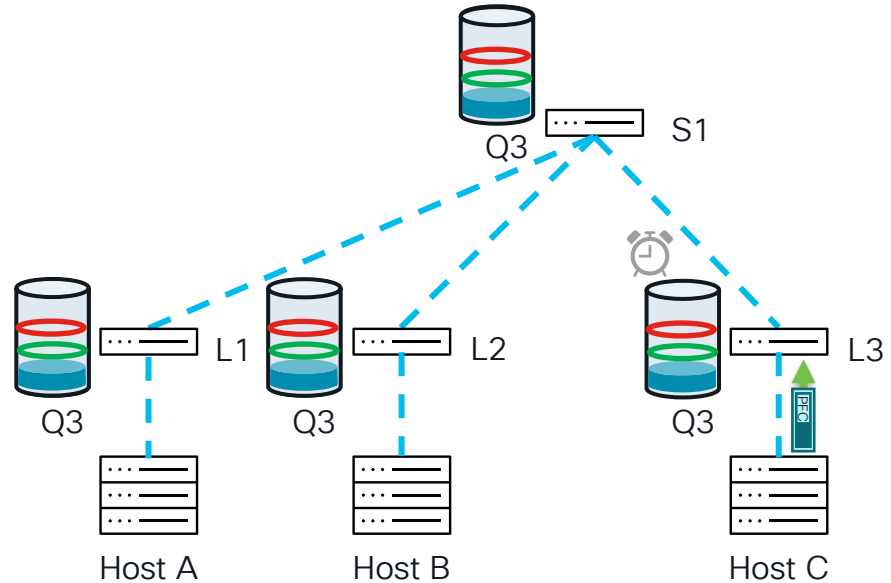
PFC watchdog

- PFC watchdog sets a timeout, if a packet exceeds time out, all packets from a queue will be cleared
- The watchdog prevents PFC frames propagating to sender and blocking it
- PFC Watchdog is supported on Cisco Nexus 9000 switches



PFC watchdog

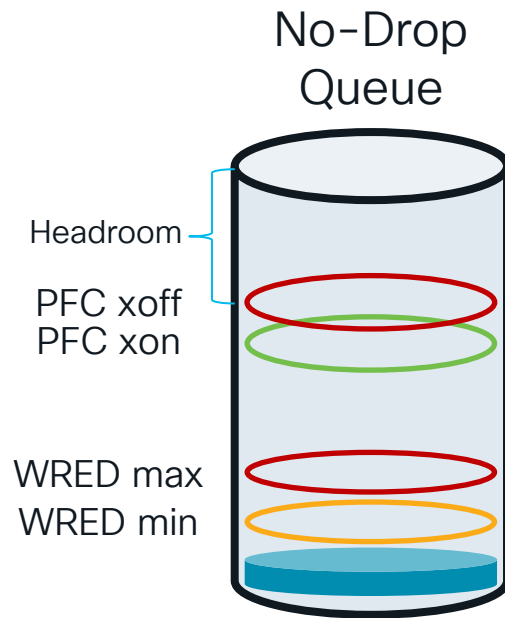
- PFC watchdog sets a timeout, if a packet exceeds time out, all packets from a queue will be cleared
- The watchdog prevents PFC frames propagating to sender and blocking it
- PFC Watchdog is supported on Cisco Nexus 9000 switches



RoCEv2: PFC and ECN together

How does it work?

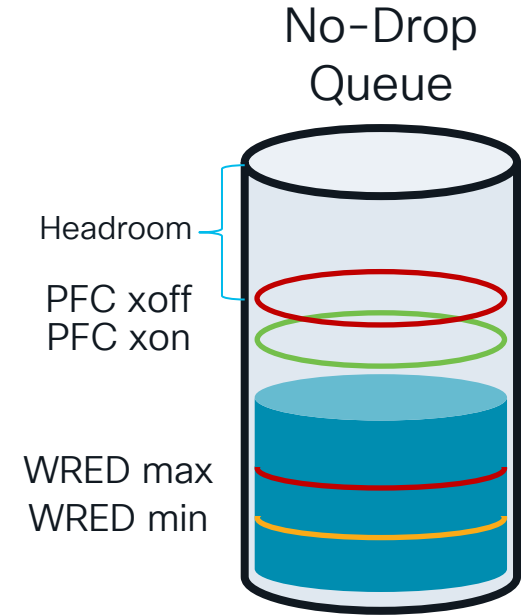
- WRED threshold are set low in no-drop queue
 - Signalize early for congestion, give enough time for end points to react
- PFC threshold are set higher than ECN
 - In case oversubscription buffers can be filled quickly without giving time to ECN to react
 - PFC will react and mitigate congestion



RoCEv2: PFC and ECN together

How does it work?

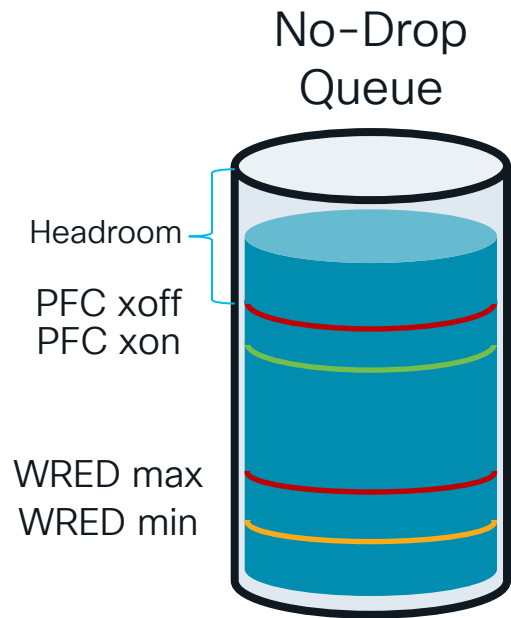
- WRED threshold are set low in no-drop queue
 - Signalize early for congestion, give enough time for end points to react
- PFC threshold are set higher than ECN
 - In case oversubscription buffers can be filled quickly without giving time to ECN to react
 - PFC will react and mitigate congestion



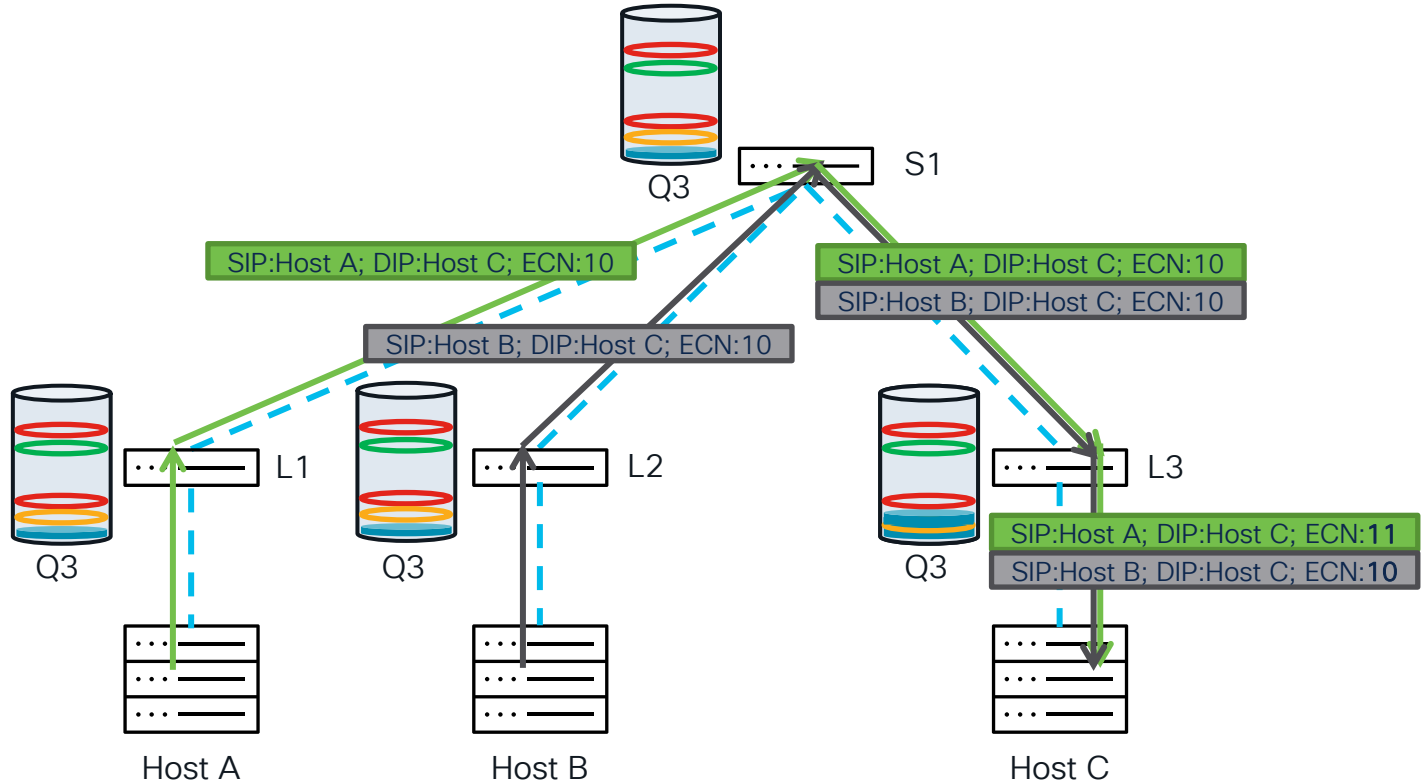
RoCEv2: PFC and ECN together

How does it work?

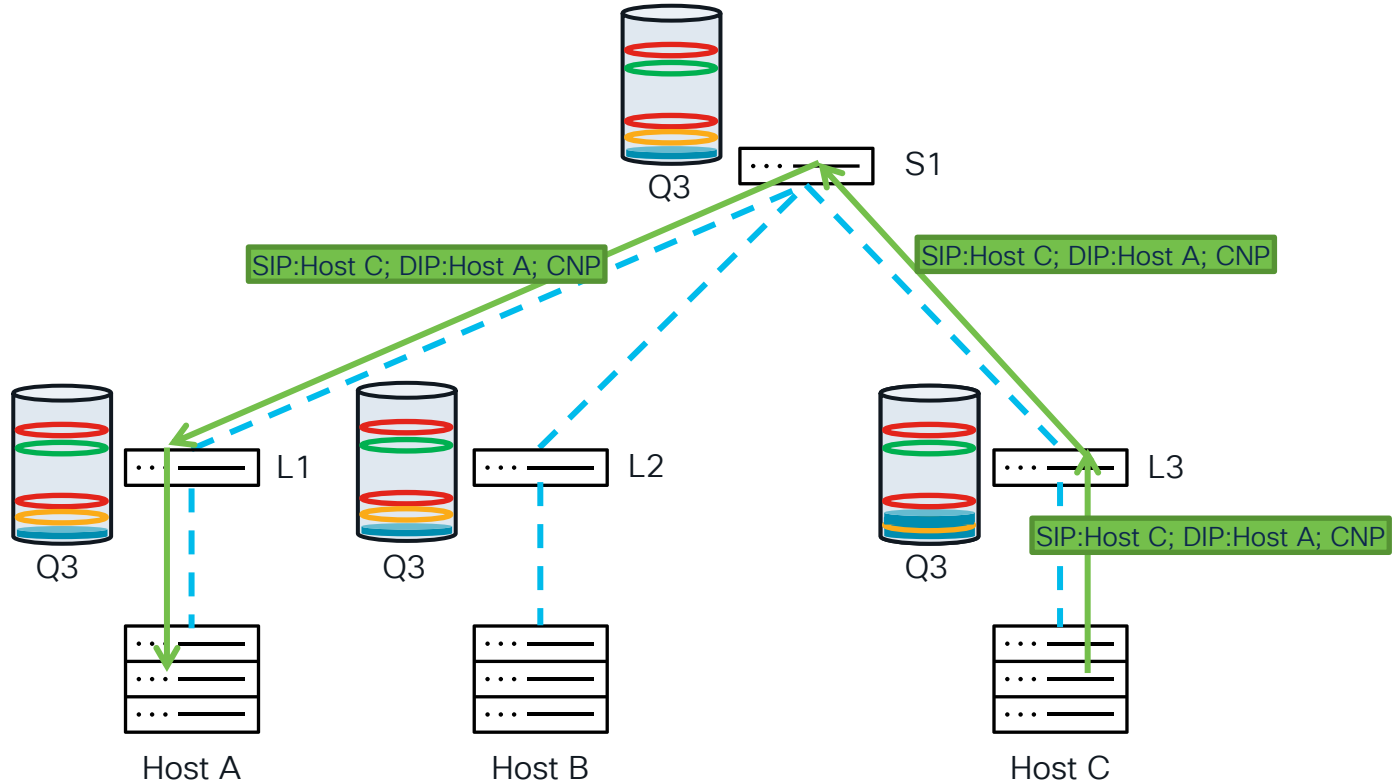
- WRED threshold are set low in no-drop queue
 - Signalize early for congestion, give enough time for end points to react
- PFC threshold are set higher than ECN
 - In case oversubscription buffers can be filled quickly without giving time to ECN to react
 - PFC will react and mitigate congestion



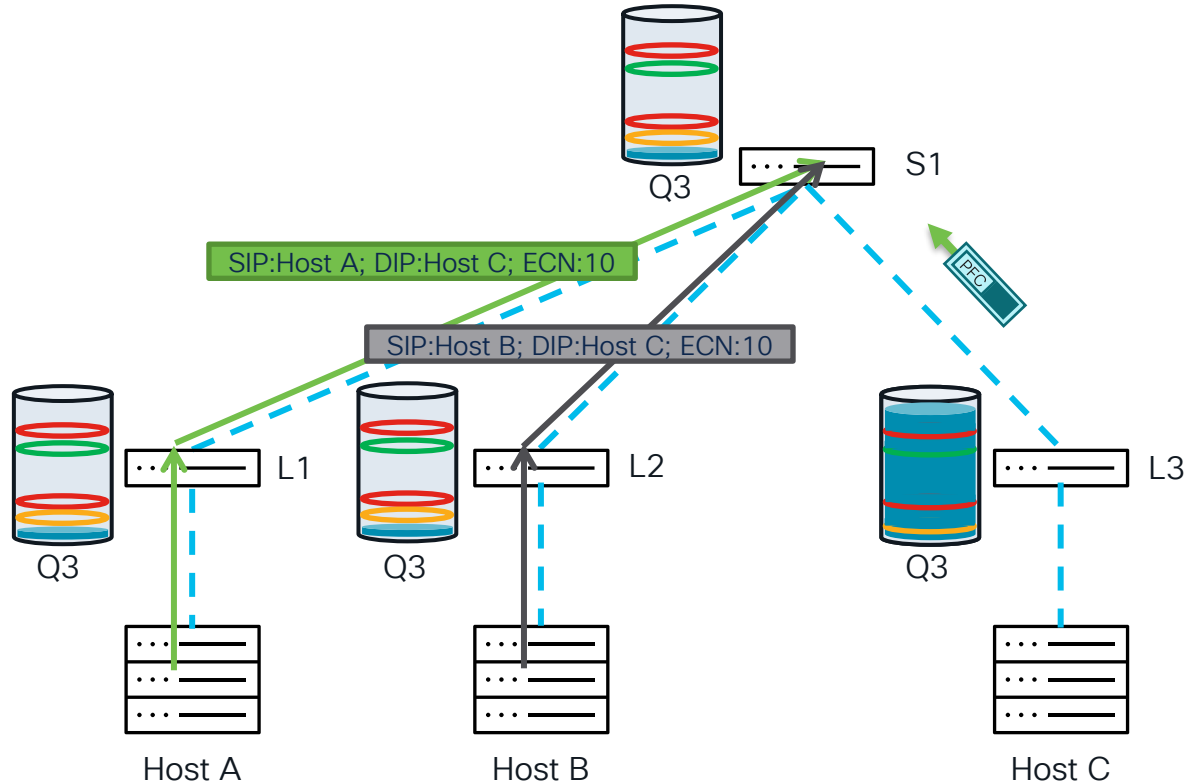
ECN and PFC make DCQCN



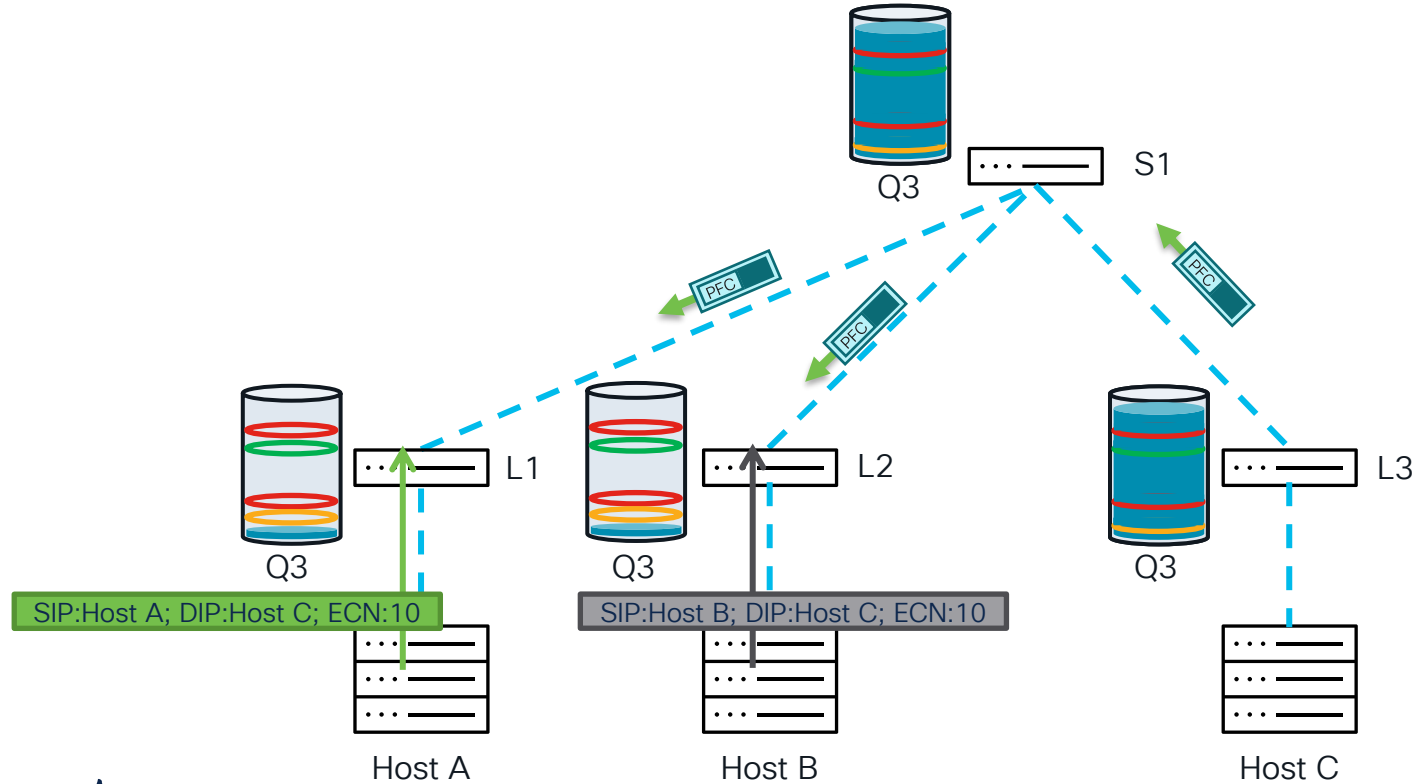
ECN and PFC make DCQCN



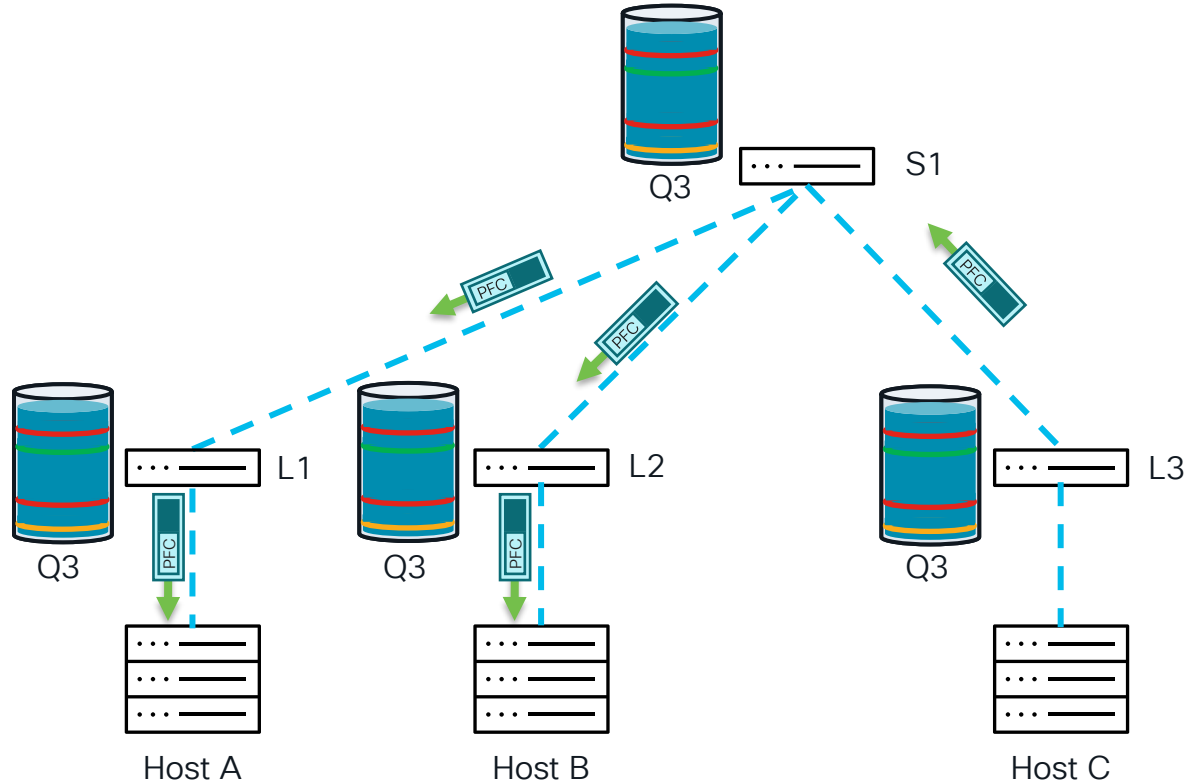
ECN and PFC make DCQCN



ECN and PFC make DCQCN

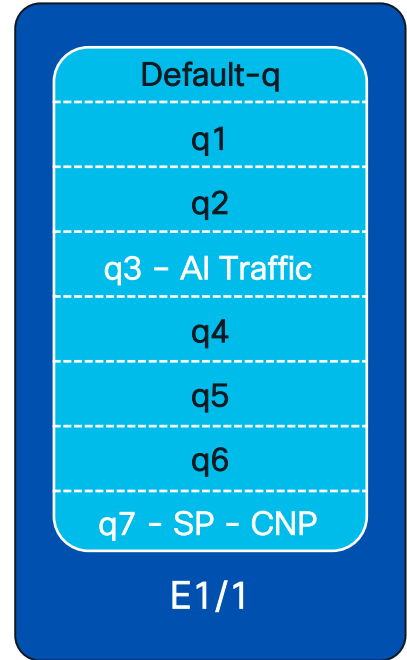


ECN and PFC make DCQCN



Quality of service

- Required separate queue for RoCEv2 traffic
 - Distinguished from other traffic in the port
 - Provide dedicated scheduling resources, to reduce latency
 - No contention for buffer resources with other traffic
 - RoCEv2 is not be part of strict priority queue, high volume of it might affect control plane
- RoCEv2 traffic requires ECN, PFC on the queue, while other traffic does not
- CNP traffic is part of strict priority queue, to deliver congestion signaling in time



Quality of service - Configuration

```
class-map type qos match-all class-roce  
  match dscp 24
```

Classification
for RoCE

```
class-map type qos match-all class-cnp  
  match dscp 48
```

Classification
for CNP

```
policy-map type qos QOS_classification_policy  
  class class-roce  
    set qos-group 3  
  class class-cnp  
    set qos-group 7  
  class class-default  
    set qos-group 0
```

Quality of service - Configuration

```
policy-map type queuing custom-8q-out-policy
  class type queuing c-out-8q-q7
    priority level 1
```

```
<snip>
```

```
  class type queuing c-out-8q-q3
    bandwidth remaining percent 99
```

```
    random-detect minimum-threshold 150 kbytes maximum-threshold 3000 kbytes drop-probability 7  
weight 0 ecn
```

```
<snip>
```

```
  class type queuing c-out-8q-q-default
    bandwidth remaining percent 1
```

```
policy-map type network-qos custom-8q-nq-policy
```

```
<snip>
```

```
  class type network-qos c-8q-nq3
    mtu 9216
```

```
    pause pfc-cos 3
```

```
<snip>
```

WRED min

WRED max

WRED random
marked packets

Enable PFC*

*PFC default thresholds

Quality of service- Configuration

```
system qos
```

```
service-policy type network-qos custom-8q-nq-policy
```

```
service-policy type queuing output custom-8q-out-policy
```

Enable WRED ECN

```
interface Ethernet1/1
```

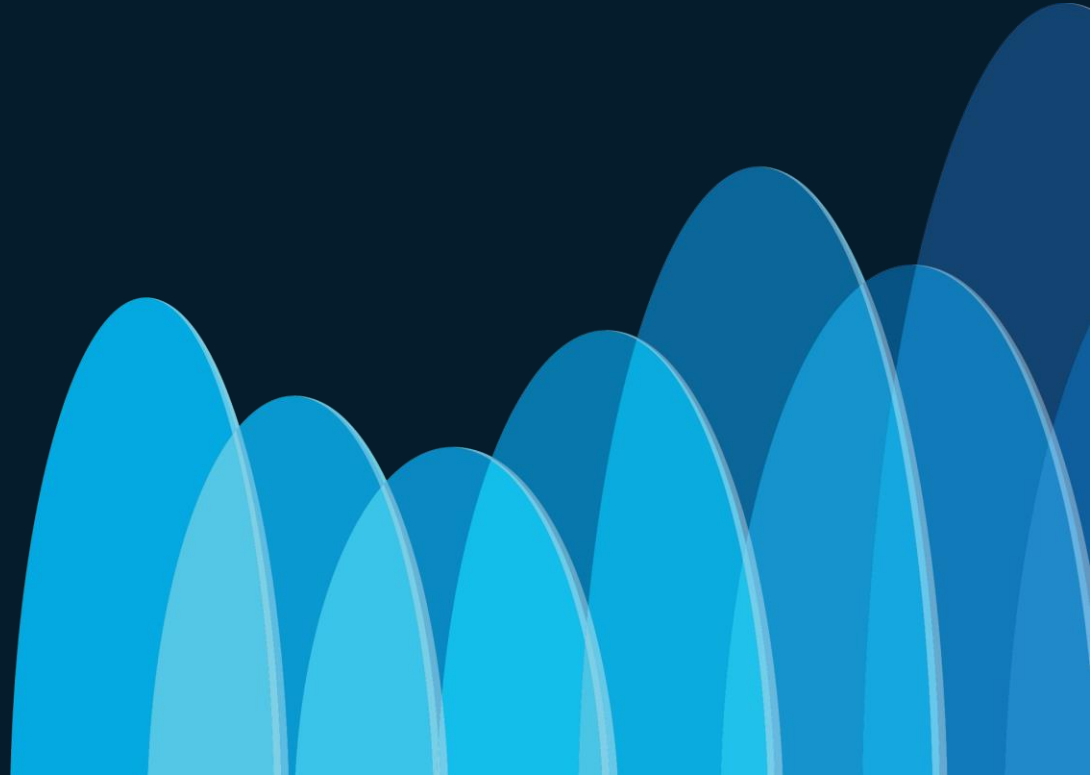
```
service-policy type qos input QOS_classification_policy
```

```
priority-flow-control mode on
```

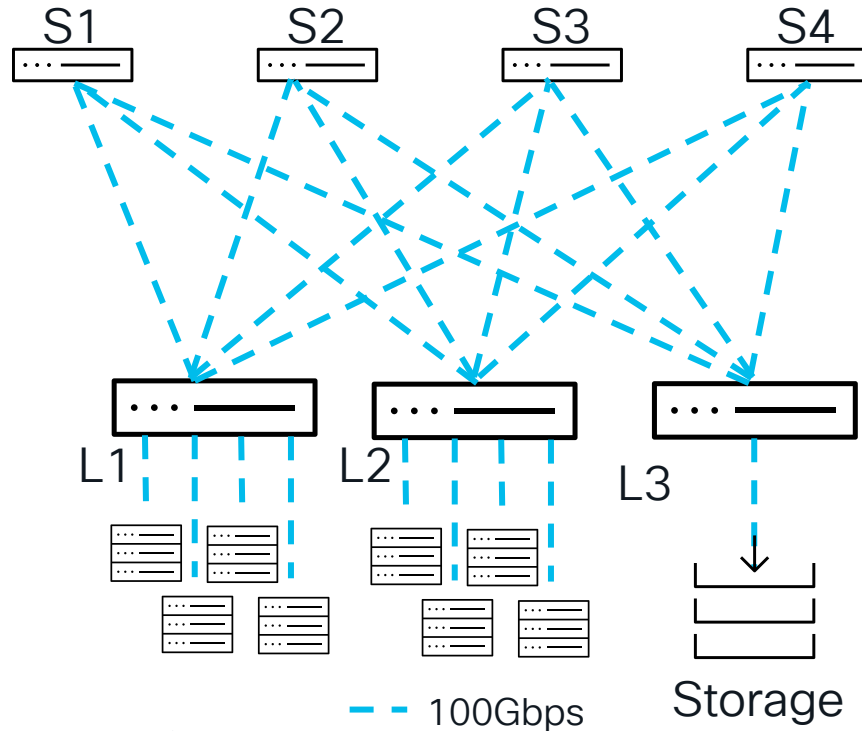
```
priority-flow-control watch-dog-interval on
```

Enable PFC
per interface

Traffic Load-balancing



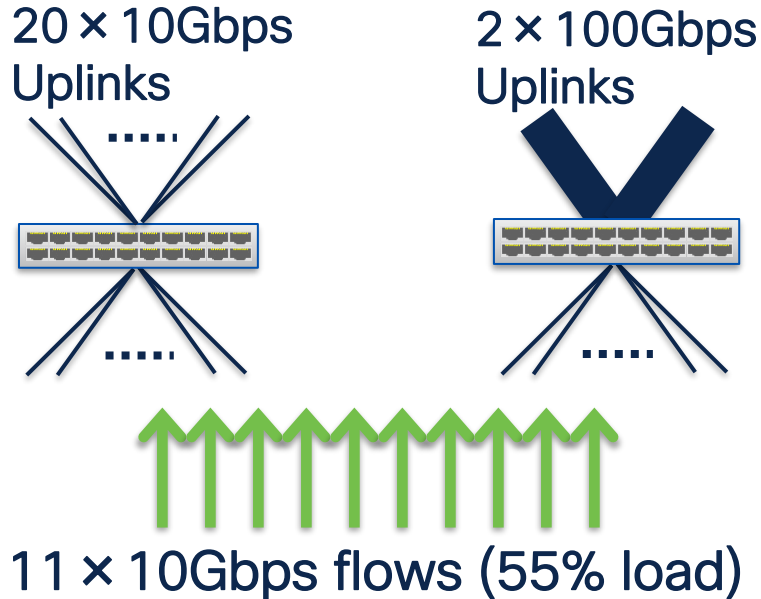
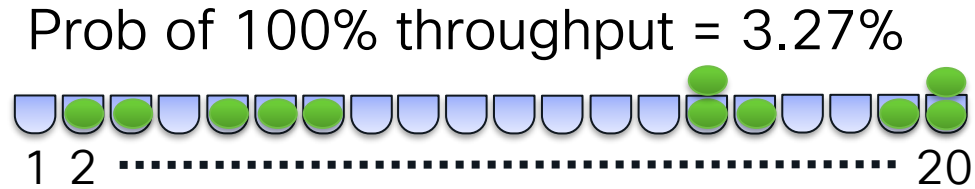
Non-blocking Network



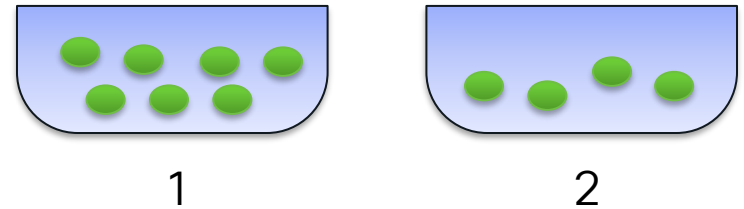
- Non-Blocking Network, allow host to talk to other hosts at full bandwidth
- Leaf: Same bandwidth to the host as to the spine
- Reduces need for congestion management to increase performance

Hashing – Where traffic goes?

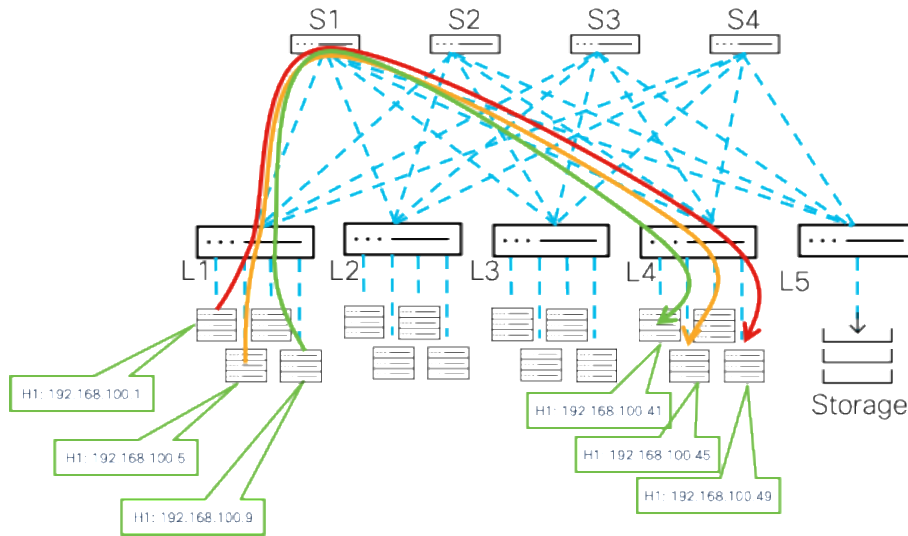
- Avoid oversubscribing link, allow multiple paths to ECMP



Prob of 100% throughput = 99.95%



Traffic polarization on the uplinks – Problem



- Traffic does not provide enough entropy to ECMP
- Traffic is polarised to a spine S1 only
- Traffic polarisation can affect application performance, triggering congestion management (PFC and ECN).

Lossless network enhancements – Problem

- All servers in back-end network, might be with in small subnets
- Two to three octets are the same among the all host ports
 - 192.168.100.x/24 or 192.168.x.x/16 could be used for all hosts
 - Destination port is the same for RoCEv2 is 4791
- This might lead for traffic polarisation on the uplinks
- Use higher bandwidth links
- Use enhanced ECMP algorithm, UDF, DLB...

Default ECMP algorithm

- By default, ECMP looks at source and destination IP and Layer 4 ports
- Hosts in AI fabric may belong to uniform subnets
- Layer 4 ports, destination port is 4791 for RoCEv2
- Entropy comes from Layer 4 Source Port

```
N9K-switch# show ip load-sharing
IPv4/IPv6 ECMP load sharing:
Universal-id (Random Seed): 2467474893
Load-share mode : address source-destination port source-destination
Rotate: 32
```

User Defined Field (UDF) ECMP algorithm

- UDF ECMP looks at source and destination IP and User Defined Field in a packet
- User can choose what field to look at to enhance entropy
- Every RoCE conversation is identified by Destination Queue Pair, in IB header
- Destination Queue Pair is 3-byte field

```
N9K-switch# show ip load-sharing
IPv4/IPv6 ECMP load sharing:
Universal-id (Random Seed): 908907021
Load-share mode : address source-destination udf offset 33 length 24
Rotate: 32
```

User Defined Field (UDF) – Destination Q Pair

- UDF offset of Nexus 9000 switches starts from first byte of IP header
- Destination Q Pair is 33 bytes from beginning of IP field (IP (20B) + UDP (8B) + IB (5B)) or 6th byte in InfiniBand header

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	172.16.103.11	172.16.101.11	RRoCE	1086	RC RDMA Read Response Middle QP=0x000a19
2	0.000020260	172.16.103.11	172.16.101.11	RRoCE	1086	RC RDMA Read Response Middle QP=0x000a19
5	0.116549167	172.16.103.11	172.16.104.11	RRoCE	78	RC RDMA Read Request QP=0x000a00
6	0.116561041	172.16.103.11	172.16.104.11	RRoCE	78	RC RDMA Read Request QP=0x000a00
7	0.122176963	172.16.103.11	172.16.112.11	RRoCE	1086	RC RDMA Read Response Middle QP=0x000903
8	0.122185013	172.16.103.11	172.16.112.11	RRoCE	1086	RC RDMA Read Response Middle QP=0x000903
9	0.236039751	172.16.103.11	172.16.101.11	RRoCE	78	RC RDMA Read Request QP=0x000a19
10	0.236050373	172.16.103.11	172.16.101.11	RRoCE	78	RC RDMA Read Request QP=0x000a19

> Frame 7: 1086 bytes on wire (8688 bits), 1086 bytes captured (8688 bits) on interface 0/17/0	0030	ff ff 00 00 09 03 00 14	72 81 f5 60
> Ethernet II, Src: MellanoxTech_c4:7c:ab (b8:ce:f6:c4:7c:ab), Dst: Cisco_23:d3:95	0040	64 3d 50 c3 60 3d db 77	94 3d 05 e0
> 802.1Q Virtual LAN, PRI: 0, DEI: 0, ID: 103	0050	1b 3d fd f2 da 3c c7 a1	87 3d 78 1e
> Internet Protocol Version 4, Src: 172.16.103.11, Dst: 172.16.112.11	0060	a2 3c 4f 94 47 3d 00 24	f4 3a bc 50
> User Datagram Protocol, Src Port: 49152, Dst Port: 4791	0070	8c 3d 89 88 16 3d 9d c4	a0 3c b5 c0
> InfiniBand	0080	57 3d a3 14 9d 3d 39 76	85 3d a5 20
> Base Transport Header	0090	a2 3d 43 f0 96 3d a9 a5	96 3d f2 e0
> Opcode: Reliable Connection (RC) – RDMA READ response Middle (14)	00a0	0c 3b 45 49 11 3d 61 df	82 3d 6b 70
> 0... = Solicited Event: False	00b0	1a 3d b5 8e 38 3d 25 33	ed 3c bd 1e
> 1.. = MigReq: True	00c0	57 3d 6d 44 55 3d 80 12	da 3a 90 30
> ..00 = Pad Count: 0	00d0	ad 3c 5d 92 34 3d 2b 2a	22 3c fd 30
> 0000 = Header Version: 0	00e0	31 3d 60 ca 70 3d 1f 68	6d 3d 6d 50
> Partition Key: 65535	00f0	10 3d 65 e8 aa 3c e4 4f	75 3d 4b e0
> Reserved: 00	0100	81 3d 5c 15 a5 3d b8 1e	82 3d 22 30
> Destination Queue Pair: 0x000903	0110	0f 3d 88 9e 1a 3d 65 97	7f 3c 29 c0
> 0... = Acknowledge Request: False	0120	8d 3d a7 8c 41 3d 58 48	ff 3c 07 50
> .000 0000 = Reserved (7 bits): 0	0130	17 3d 5b 72 64 3c d5 b7	d7 3a 71 50
> Packet Sequence Number: 1340033	0140	a1 3d d9 f9 45 3d eb 6a	1c 3c 03 30
> Invariant CRC: 0xb5baee4d	0150	06 3d 73 c4 60 3d b1 8d	a3 3d 50 40
> Data (1024 bytes)	0160	7a 3c a0 6e eb 3b a0 16	6d 3c b9 40
	0170	28 3d ec b9 5e 3d eb f6	28 3c a3 30
	0180	13 3d 4b 93 86 3d 85 28	c4 3c 6b e0
	0190	53 3a 53 c9 f4 3c 6c fa	51 3d 64 20
	01a0	91 3d f4 4a 69 3d f2 8e	90 3d 0b 10
	01b0	5f 3d 91 cb 97 3d 43 51	ba 3c e5 c0

User Defined Field (UDF) – Destination Q Pair

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	172.16.103.11	172.16.101.11	RRoCE	1086	RC RDMA Read Response Middle QP=0x000a19
2	0.000020260	172.16.103.11	172.16.101.11	RRoCE	1086	RC RDMA Read Response Middle QP=0x000a19
5	0.116549167	172.16.103.11	172.16.104.11	RRoCE	78	RC RDMA Read Request QP=0x000a00
6	0.116561041	172.16.103.11	172.16.104.11	RRoCE	78	RC RDMA Read Request QP=0x000a00
7	0.122176963	172.16.103.11	172.16.112.11	RRoCE	1086	RC RDMA Read Response Middle QP=0x000903
8	0.122185013	172.16.103.11	172.16.112.11	RRoCE	1086	RC RDMA Read Response Middle QP=0x000903
9	0.236039751	172.16.103.11	172.16.101.11	RRoCE	78	RC RDMA Read Request QP=0x000a19
10	0.236050373	172.16.103.11	172.16.101.11	RRoCE	78	RC RDMA Read Request QP=0x000a19

> Frame 9: 78 bytes on wire (624 bits), 78 bytes captured (624 bits) on interface ps-inb, id 0

> Ethernet II, Src: MellanoxTech_c4:7c:ab (b8:ce:f6:c4:7c:ab), Dst: Cisco_23:d3:95 (00:fc:ba:23:d3:95)

> 802.1Q Virtual LAN, PRI: 0, DEI: 0, ID: 103

> Internet Protocol Version 4, Src: 172.16.103.11, Dst: 172.16.101.11

> User Datagram Protocol, Src Port: 49152, Dst Port: 4791

> InfiniBand

Base Transport Header

Opcode: Reliable Connection (RC) – RDMA READ Request (12)

0... = Solicited Event: False

1... = MigReq: True

..00 = Pad Count: 0

.... 0000 = Header Version: 0

Partition Key: 65535

Reserved: 00

Destination Queue Pair: 0x000a19

1... = Acknowledge Request: True

..000 0000 = Reserved (7 bits): 0

Packet Sequence Number: 1365501

RETH – RDMA Extended Transport Header

Invariant CRC: 0x4f0ee094

0000 00 fc ba 23 d3 95 b8 ce f6 c4 7c ab 81 00 00 67

0010 08 00 45 6a 00 3c 90 3b 40 00 ff 11 c6 d3 ac 10

0020 67 0b ac 10 65 0b c0 00 12 b7 00 28 00 00 0c 40

0030 ff ff 00 00 0a 19 80 14 d5 fd 00 00 00 00 0e 83

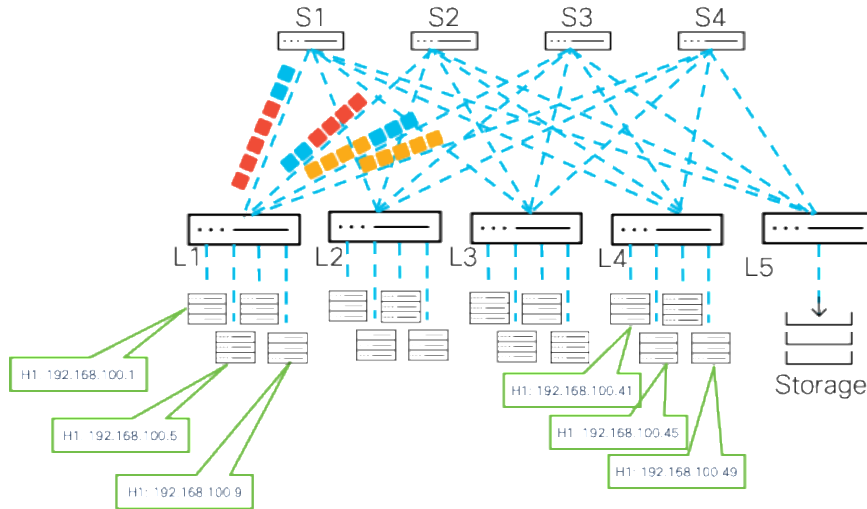
0040 38 c0 00 12 5c 52 00 00 03 40 4f 0e e0 94



Dynamic Load Balancing – Introduction

- DLB is dynamic way of load-balancing traffic over equal cost paths. It allows better traffic distribution, and prevents polarization of traffic over one link
- DLB on Nexus 9000 devices allows different modes of traffic load balancing
- **Flowlet mode:** Flow of 5-tuple is split in flowlets (group of packets) and distributed across equal cost paths
- **Static Pinning:** Allows for pairs of ingress and egress ports to be pinned together, as such manually distributed load over equal cost paths
- **Per Packet Load balancing:** Allows individual packets to be sent over different links, to allow better traffic distribution
- Detection of link failure is done in hardware, reducing reaction time for fast failover, with all of the modes

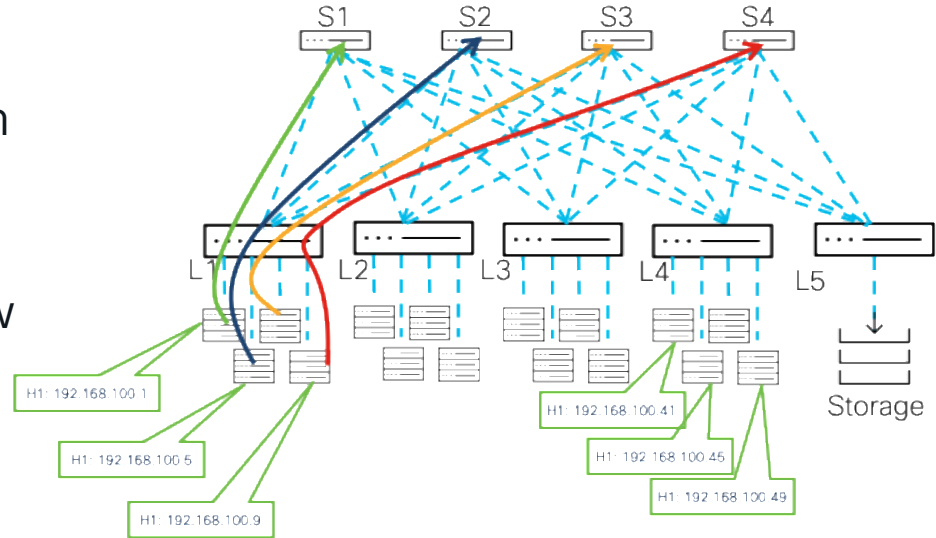
Dynamic Load Balancing – Flowlet Mode



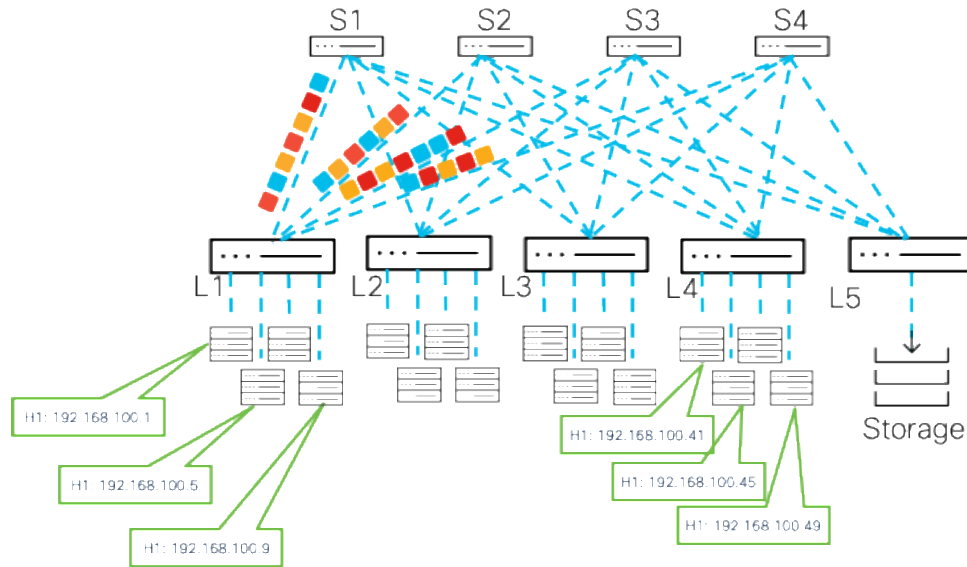
- Flowlets maintain the link, or dynamically change the link
- Least utilized link is chosen, for new flow, or at time flows are moved to different link
- Ageing time is used avoid out of order packets, it should be $2 \times \text{RTT}$
- All workloads are balanced fairly and if flows are bursty they can dynamically change path

Dynamic Load Balancing – Static Pinning

- Allows static pinning between ingress and egress ports
- Valid only for multipath destination traffic
- Steady bit rate flows, or undersubscribed network, to allow fair distribution of the traffic
- Manual configuration, or API

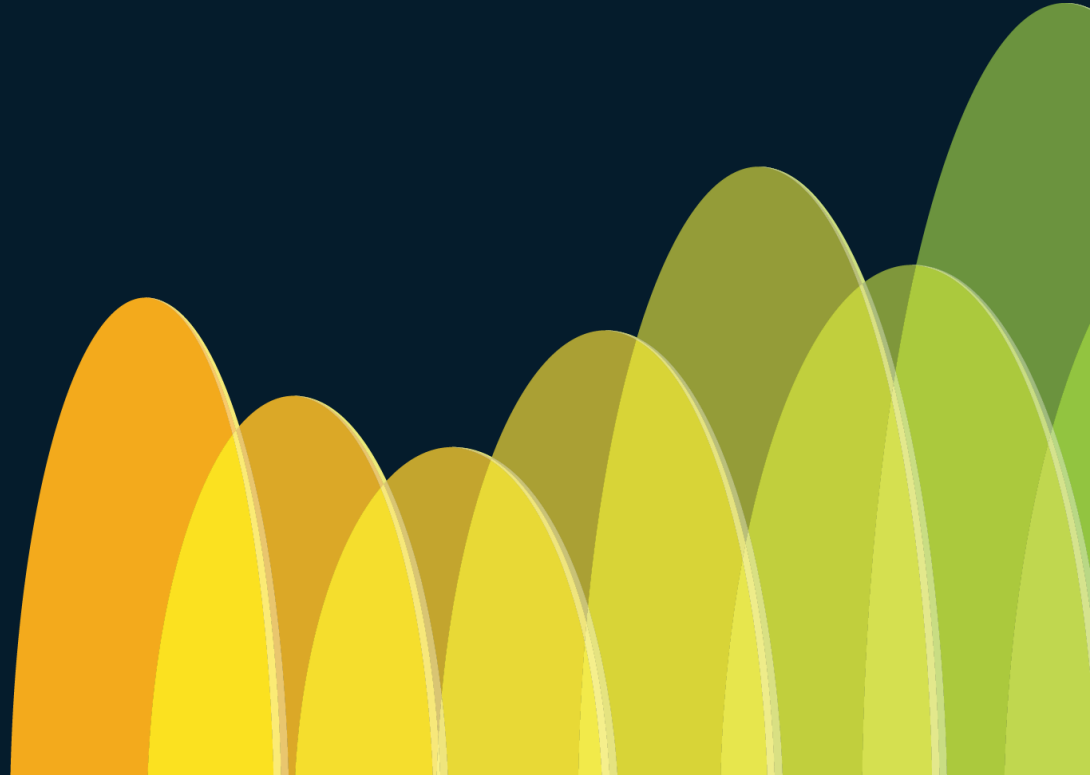


Dynamic Load Balancing – Per-packet Mode



- Per packet mode load balancing, allows fair distribution of traffic
- Least utilized link is chosen per packet, delivering uniform usage of the links
- As packet of same flow, will take different path, may happen that they arrive at destination endpoints out of order
- Expected in deployments with per-packet load balancing for receiver to put packet in order (host needs to do packet reordering)

Automation and Visibility



Nexus Dashboard Fabric Controller

Create Fabric?—×

N9K Cloud Scale Platform Queuing Policy	
Select an Option	Queuing Policy for all 92xx, -EX, -FX, -FX2, -FX3, -GX series switches in the fabric
N9K R-Series Platform Queuing Policy	
Select an Option	Queuing Policy for all R-Series switches in the fabric
Other N9K Platform Queuing Policy	
Select an Option	Queuing Policy for all other switches in the fabric
Enable AI / ML QoS and Queuing Policies	
☒	Configures QoS and Queuing Policies specific to N9K Cloud Scale switch fabric for AI / ML network loads
AI / ML QoS & Queuing Policy*	
AI_Fabric_QOS_100G	Queuing Policy based on predominant fabric link speed: 400G / 100G / 25G
AI_Fabric_QOS_400G	
AI_Fabric_QOS_100G	Enable MACsec in the fabric
AI_Fabric_QOS_25G	
	Cisco Type 7 Encrypted Octet String

Nexus Dashboard Fabric Controller

Edit interface(s) ? — ×

Additional CLI for the interface

Enable Interface*

☒

Uncheck to disable the interface

Enable Netflow

☐

Netflow is supported only if it is enabled on fabric

Netflow Monitor

Provide the Layer 3 Monitor Name

Netflow Sampler

Netflow sampler name, applicable to N7K only

Enable priority flow control

☒

Enable priority flow control

Enable QoS Configuration

☒

Enable to configure a QoS Policy for this interface. If AI/ML Queuing is enabled on the fabric, will use the QOS_CLASSIFICATION policy. Enter a custom policy below to override

Custom QoS Policy

Custom QoS Policy must be defined previously

Custom Queuing Policy

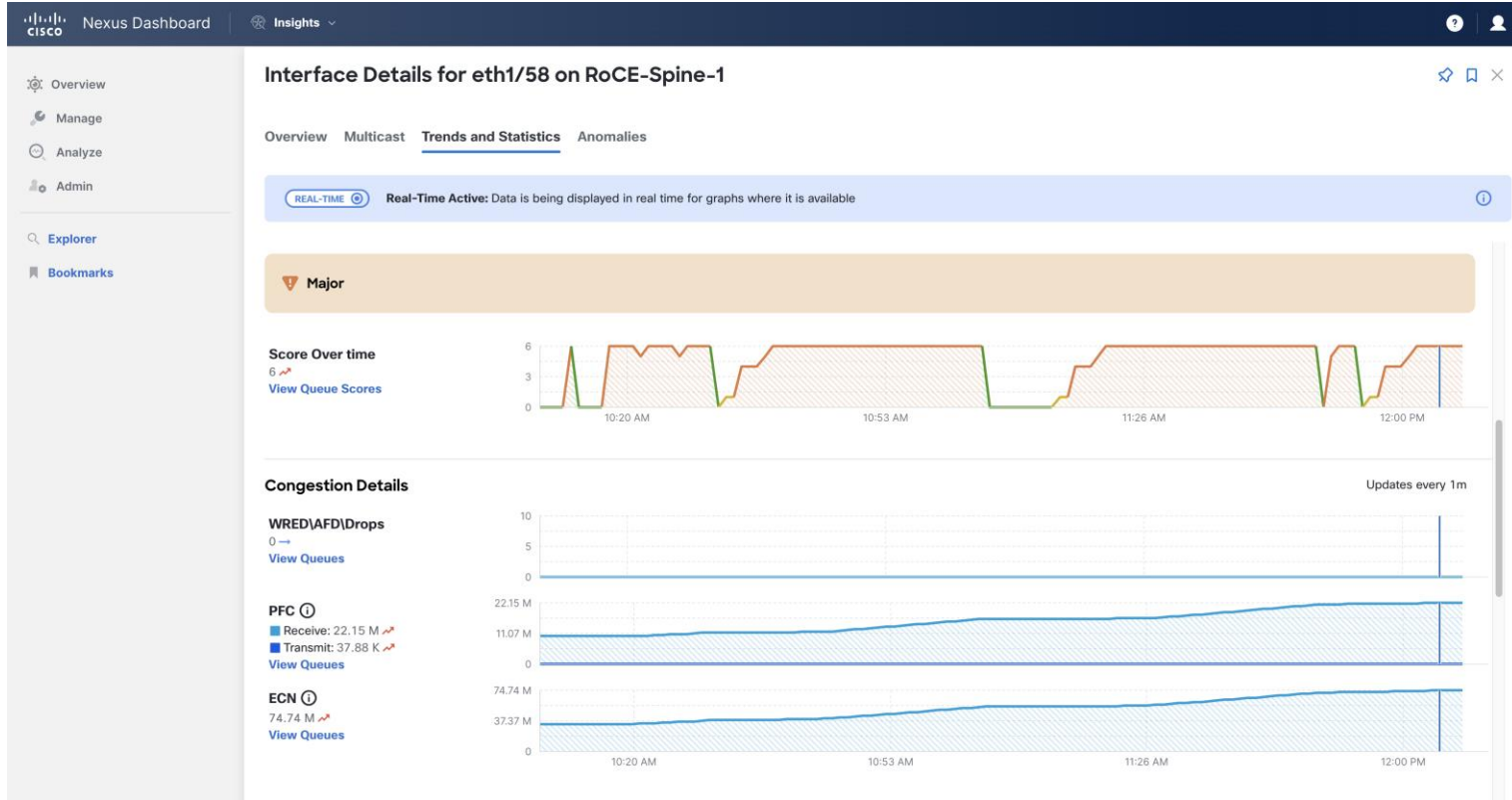
Queuing Policy must be defined previously

Visibility – Flow table

- Collects full flow information plus metadata
 - 5-tuple flow info
 - Interface/queue info
 - Flow start/stop time
 - Packet disposition (drop indicators)
 - Burst measurement
- Export data to collector
- Leveraged by Nexus Dashboard Insights



Nexus Dashboard Insights – Congestion visibility



Nexus Dashboard Insights – Congestion visibility

 Nexus Dashboard

Insights

Overview

Manage

Analyze

Admin

Explorer

Bookmarks

Interface Details for eth1/58 on RoCE-Spine-1

Overview Multicast Trends and Statistics Anomalies

REAL-TIME Real-Time Active: Data is being displayed in real time for graphs where it is available

Microbursts

Microbursts by

Queue	Start Time	Number of Bursts	Max Duration (ns)	Avg Duration (ns)	Max Peak	Avg Peak
queue-3	May 31 2024 12:05:00.000000 PM	172	2.09 ms	551.82 ns	4,554,368	2,738,949
queue-3	May 31 2024 12:00:00.000000 PM	538	1.94 ms	493.66 ns	5,137,184	2,709,682
queue-3	May 31 2024 11:55:00.000000 AM	76	550.57 ns	243.96 ns	3,174,080	2,203,202
queue-3	May 31 2024 11:50:00.000000 AM	331	3.11 ms	590.47 ns	7,363,616	2,874,888
queue-3	May 31 2024 11:45:00.000000 AM	279	8.18 ms	737.11 ns	6,466,720	2,965,094
queue-3	May 31 2024 11:40:00.000000 AM	386	2.70 ms	648.39 ns	6,902,688	2,952,173
queue-3	May 31 2024 11:35:00.000000 AM	354	3.64 ms	688.12 ns	6,847,776	2,963,263
queue-3	May 31 2024 11:30:00.000000 AM	266	3.22 ms	641.42 ns	6,009,536	2,913,820

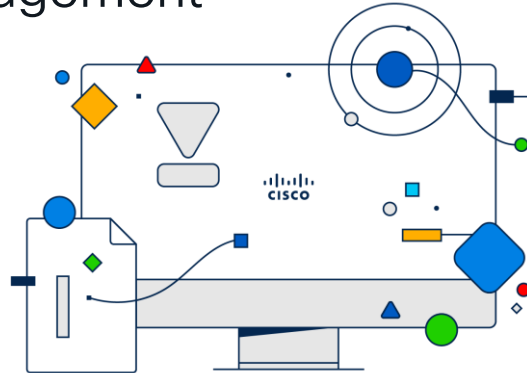
Blueprint of Today

CISCO *Live!*



Choosing the right infrastructure

- Build a Clos Fabric / Spine-Leaf
- Fixed Switches
 - Lower Latency, single AISC
 - Power Efficient
- Right Congestion management
- Routed Fabric
 - Use BGP for control plane
- Scalable design
 - Two tier design
 - Three tier design



Customer request #1

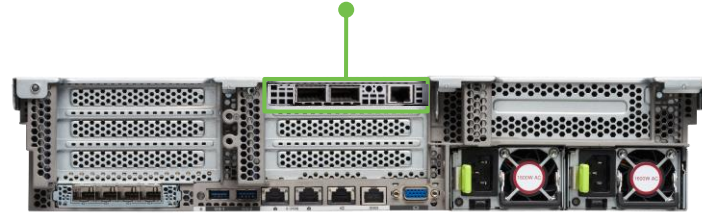
- Build Cluster of 260 GPUs
 - Use case: Inferencing for 100s of users, RAG, and fine tuning of a model
 - Cluster is built with stand alone server (e.g. Cisco UCS-C240-M7)
- Build non-blocking network, for GPU communication
- Predictable and low latency for efficient training
- Host connectivity at dual 100Gbps
- Fabric at 400Gbps for efficient load-balancing

Networks – Why do I need them?



UCS C240 M7 Rack Server
Up to 2 GPUs

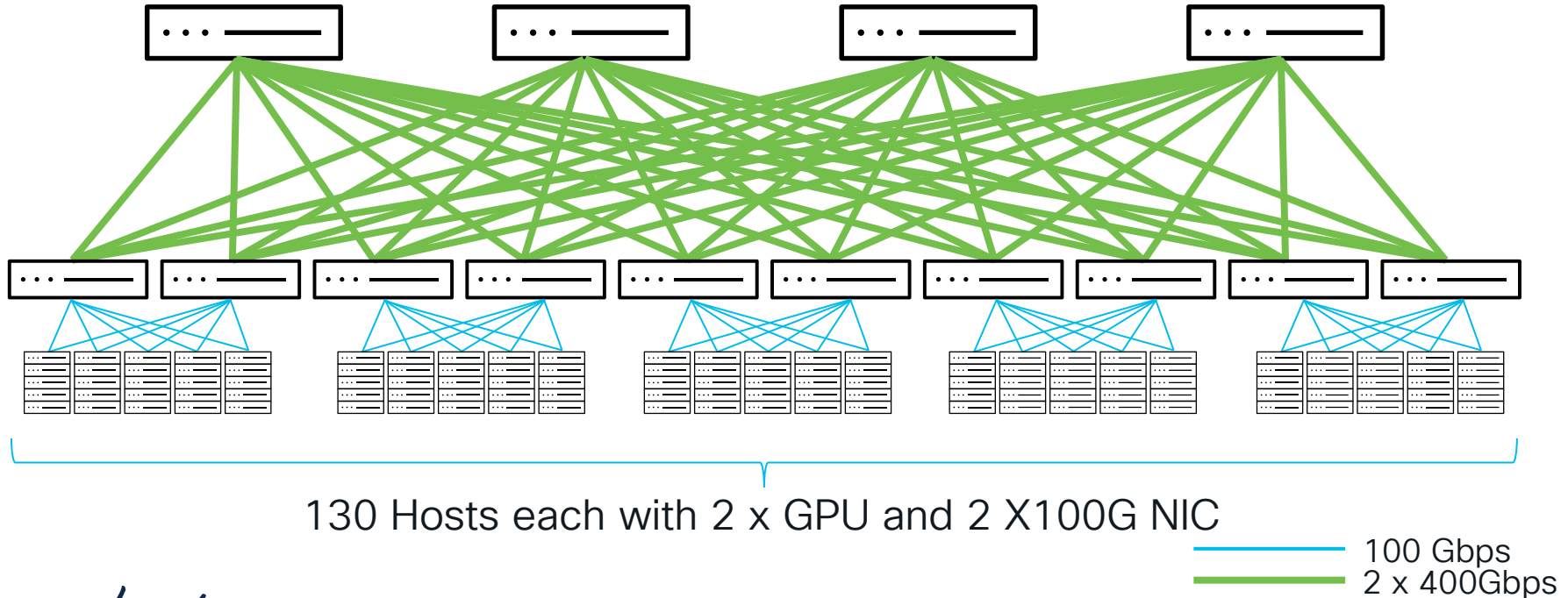
GPU to GPU, Front-end and
Storage Network



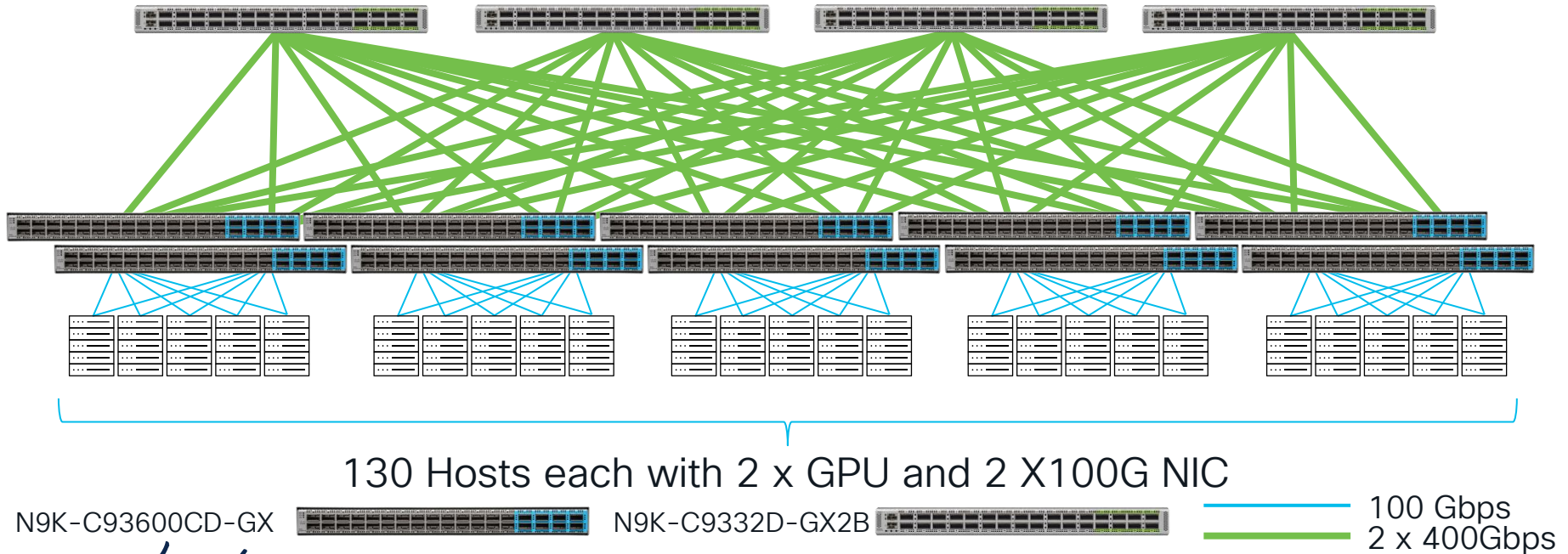
Customer request #1 – Proposal

- Standalone server can have up to 2 GPUs
 - As required is 260 GPUs, 130 Stand alone servers are needed
 - Each standalone server is equipped with dual 100G port NIC
- 260 x 100G ports required for host connectivity in leaf layer
 - 26 x 100G host interfaces per leaf switch, for 10 leaf switches
 - For seeped up have 8 x 400G uplinks per switch for Spine connectivity
- Total of 4 spines 20 x 400G ports used per Spine
- Leave room for future expansion

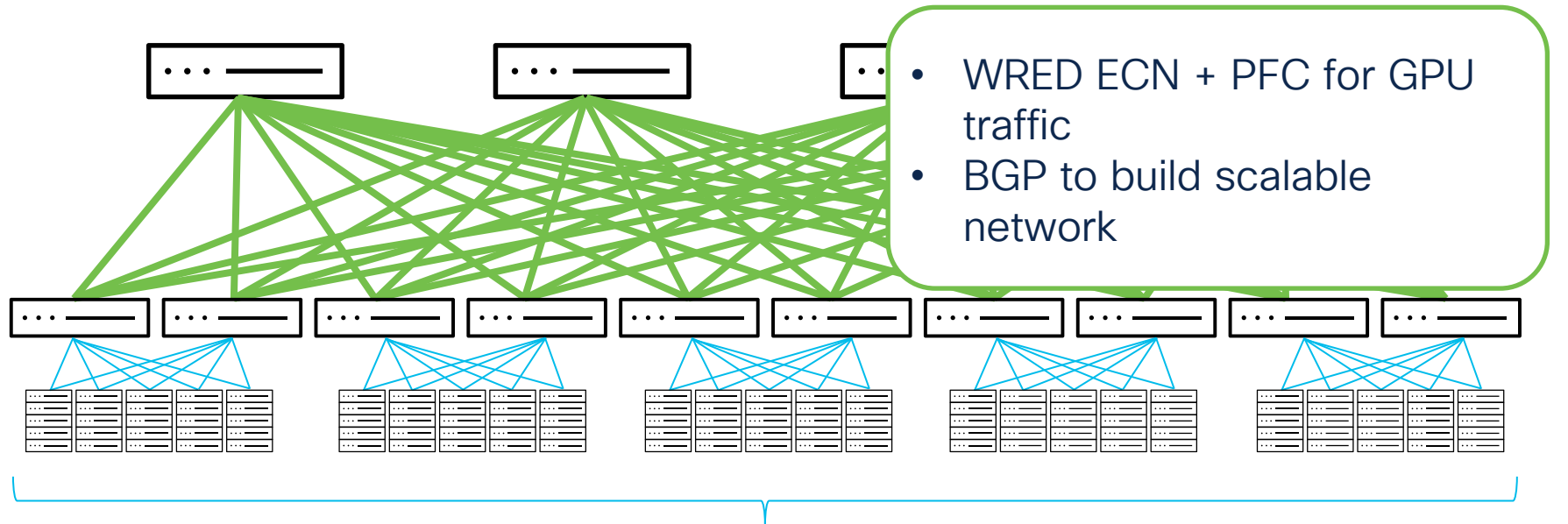
Customer request #1 - Design



Customer request #1 - Design



Customer request #1 - Design



130 Hosts each with 2 x GPU and 2 X100G NIC

— 100 Gbps
— 2 x 400Gbps

Customer request #1 – Design

- Fabric is running BGP in the fabric
- Non-blocking network design
- Congestion management through PFC and ECN for optimal performance
- Efficient traffic load-balancing in the fabric
- Automated with NDFC

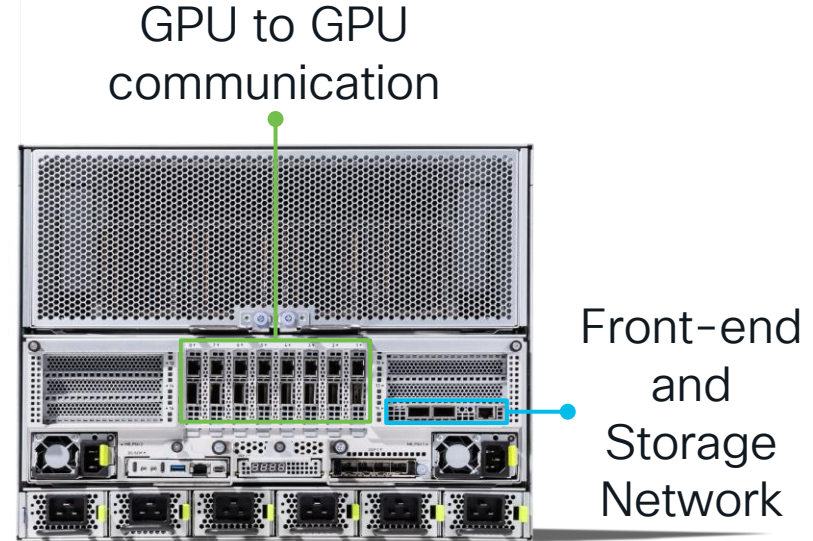
Customer request #2

- Build Powerful Cluster of 320 GPUs, for large training model use case
 - Cluster is built with powerful accelerators (e.g. 8-way GPU server)
- Build non-blocking back-end network, for GPU communication
 - Predictable and low latency for efficient training
 - Host connectivity at 8 ports of 400Gbps
 - Fabric at 400Gbps for efficient load-balancing
- Build front-end network, for server-to-server interaction, and storage connectivity
 - 2 x 100Gbps ports for connectivity, non-blocking and future expansion

Networks – Why do I need them?



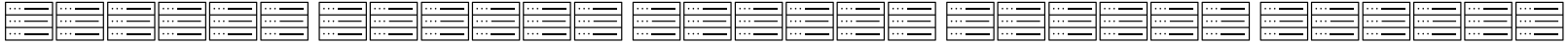
UCS C885A M8 Rack Server
8 GPUs



Customer request #2 – Proposal for Back-End

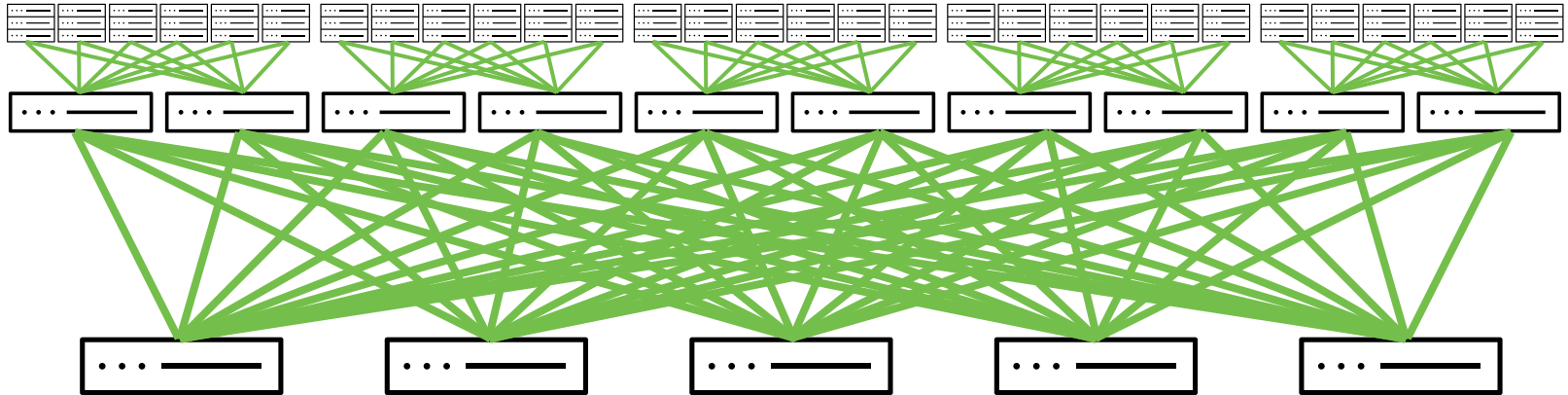
- Powerful AI training accelerator, has 8 GPUs per server
 - As required is 320 GPUs, 40 host are sufficient
 - Each host is equipped with 4 NICs each with dual QSFP-DD ports
- 320 x 400G ports required for host connectivity in leaf layer
 - 32 x 400G host interfaces per leaf switch, for 10 leaf switches
 - Non-blocking network 32 x 400G uplinks per leaf for Spine connectivity
- Total of 5 spines 64 x 400G ports used per Spine

Front-End and Back-End Cluster Network

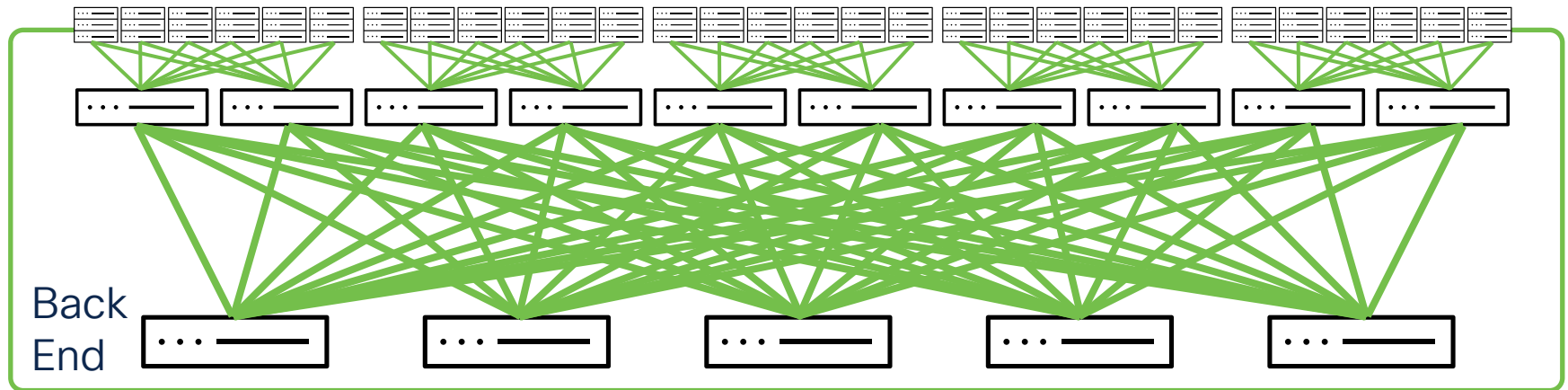


40 Hosts each with 8 x GPU

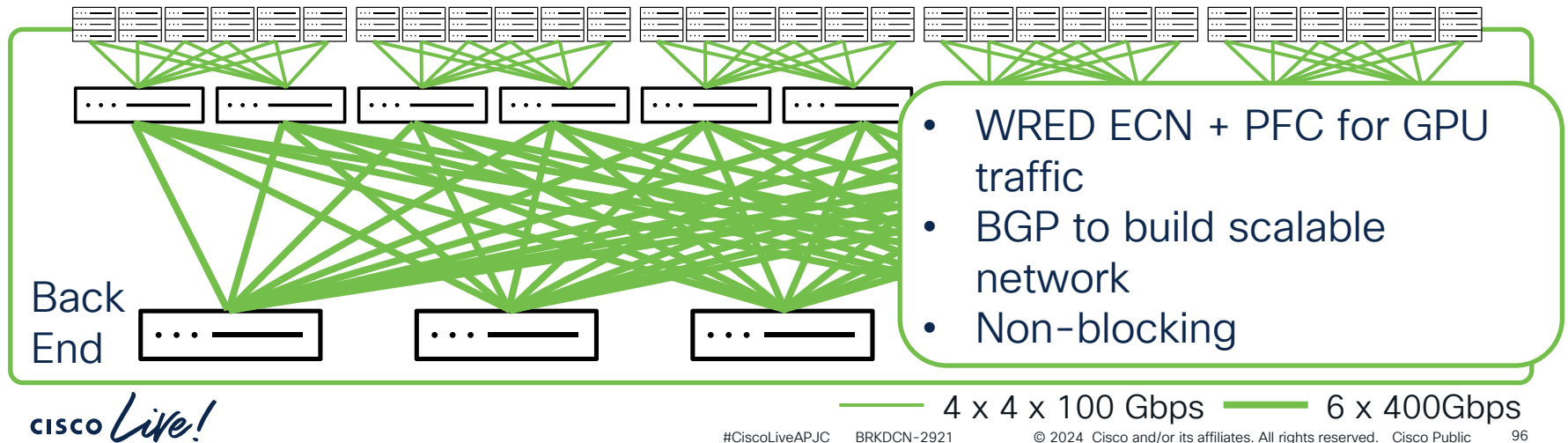
Back-End Cluster Network



Back-End Cluster Network



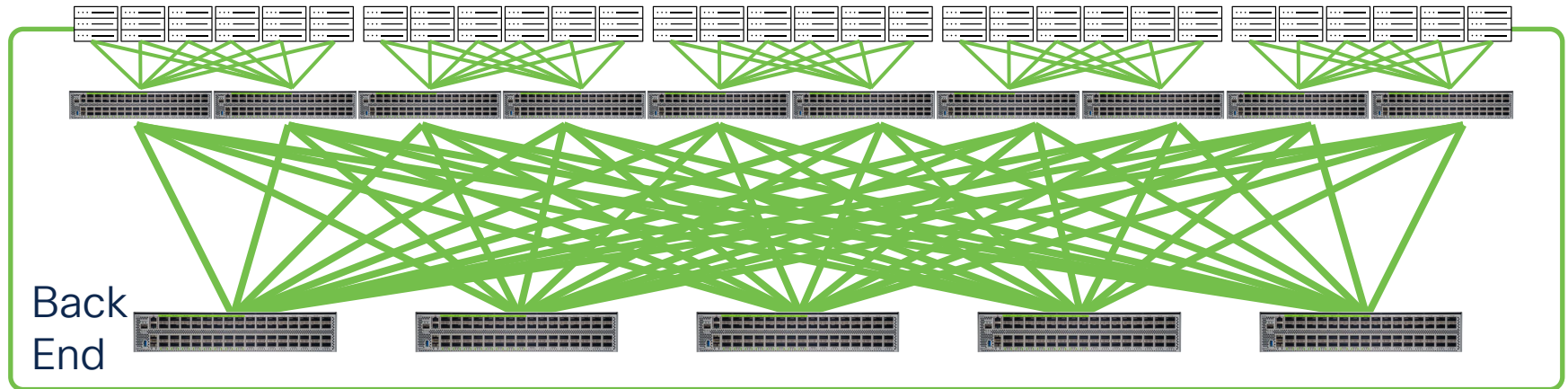
Back-End Cluster Network



Back-End Cluster Network



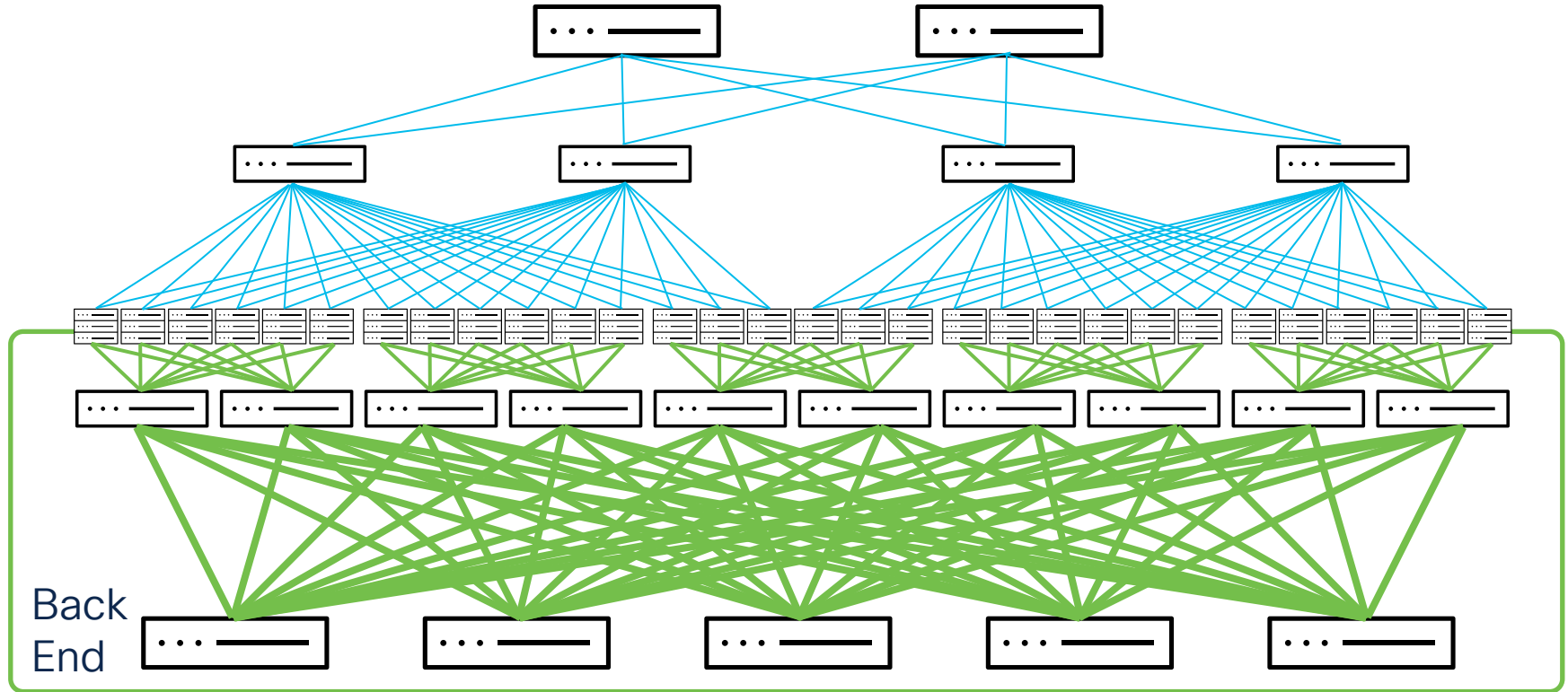
N9K-C9364D-GX2A



Customer request #2 – Proposal for Front-End

- Front End network for host communication, and storage
 - Each server has 2 x 100G ports
- 80 x 100G ports required for host connectivity in leaf layer
 - 20 x 100G host interfaces per leaf switch, for 4 leaf switches
 - Non-blocking network 20 x 100G uplinks per leaf for Spine connectivity
- Total of 2 spines 64 x 100G ports used per Spine
- Storage NFS network in Front End
 - 3 Storage array connected to leaf
 - RoCEv2 for storage networks

Front-End and Back-End Cluster Network



Back
End

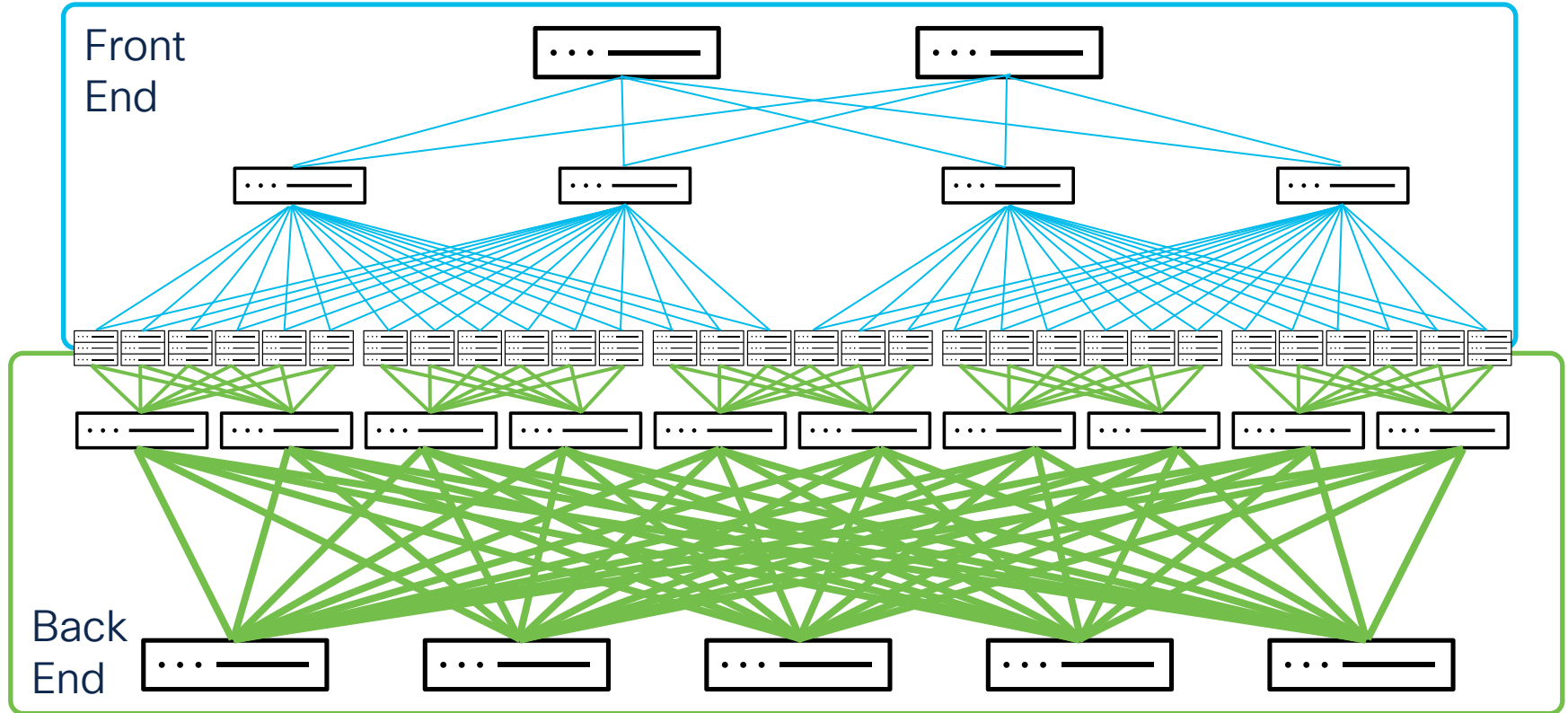
cisco *Live!*

— 100 Gbps

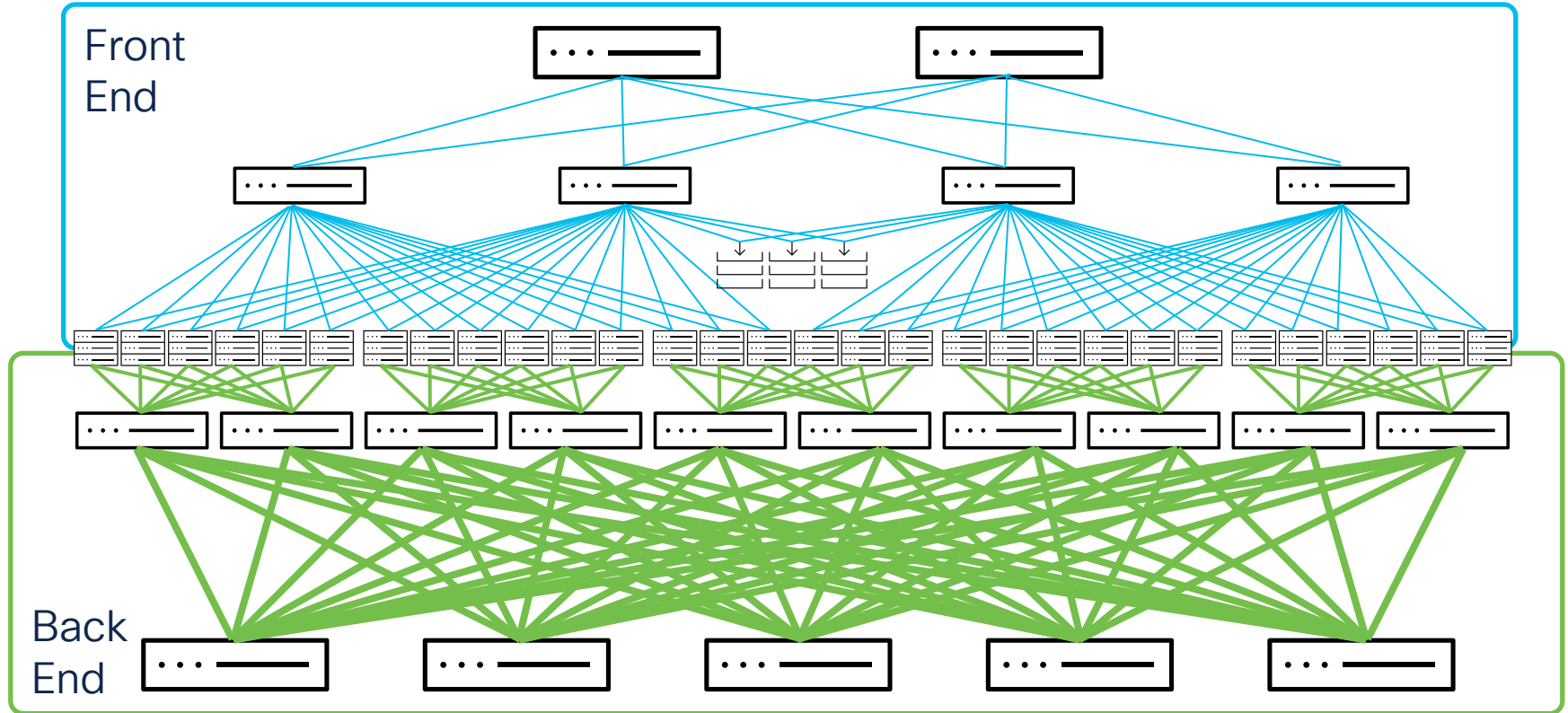
— 4 x 4 x 100 Gbps

— 6 x 400Gbps

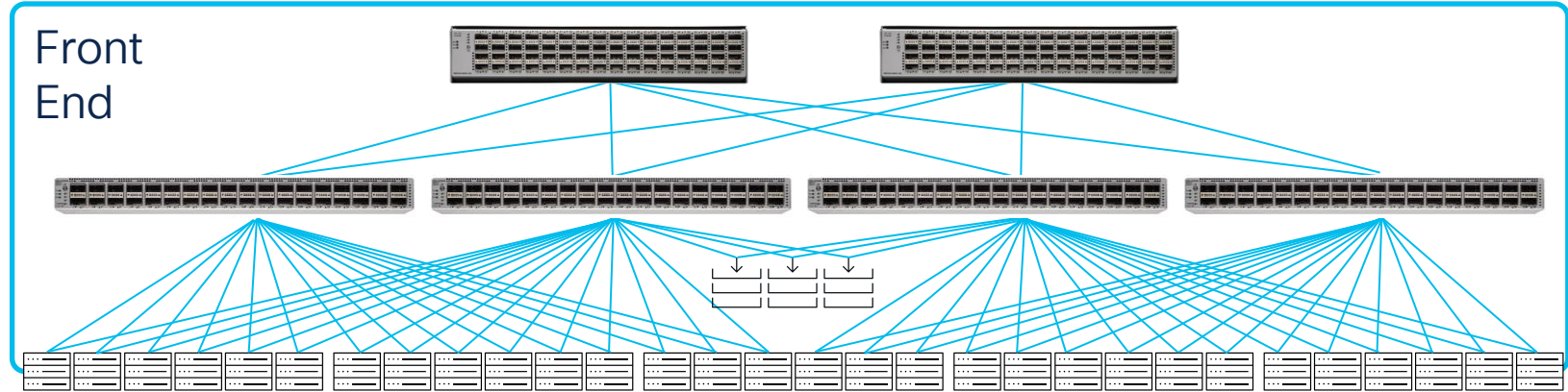
Front-End and Back-End Cluster Network



Front-End and Back-End Cluster Network



Front-End Cluster Network

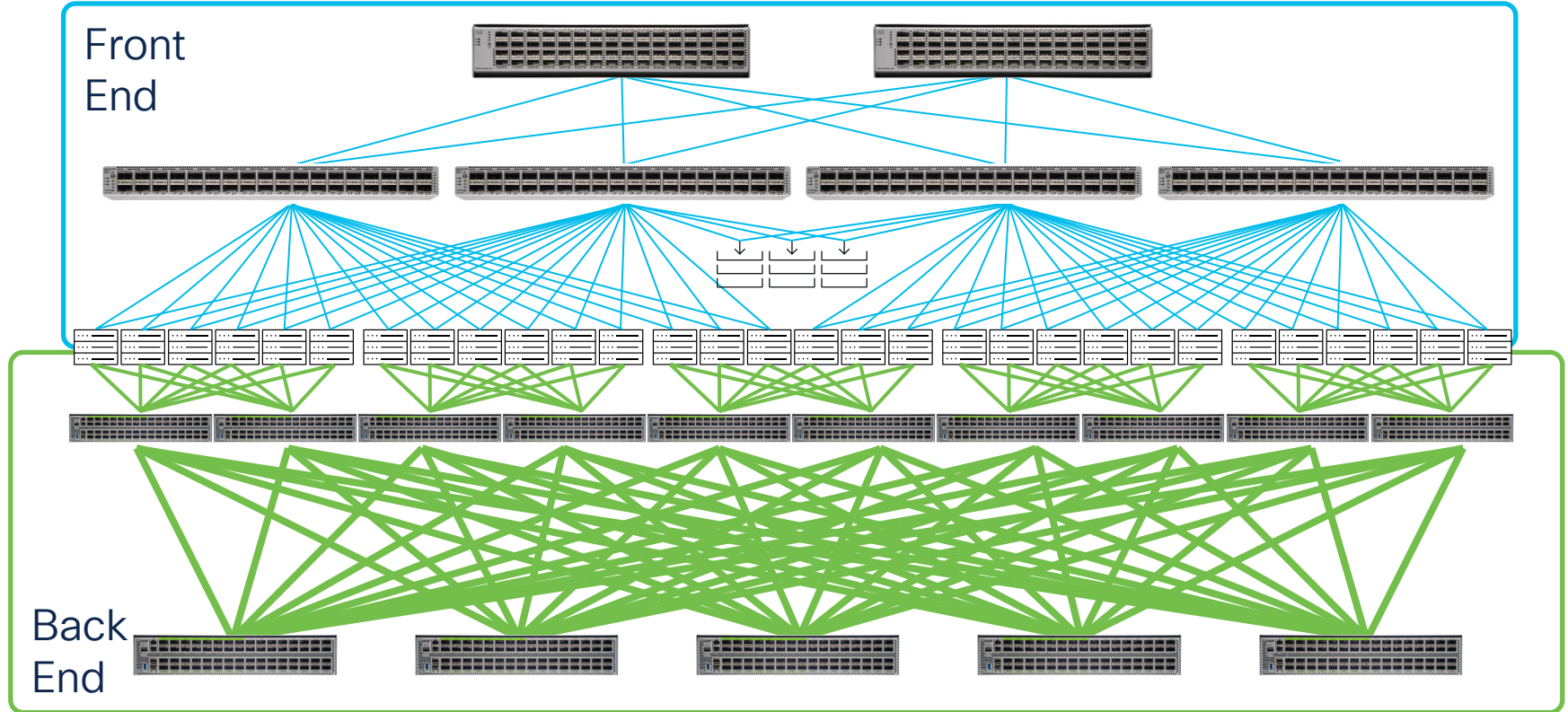


N9K-C9364C-GX

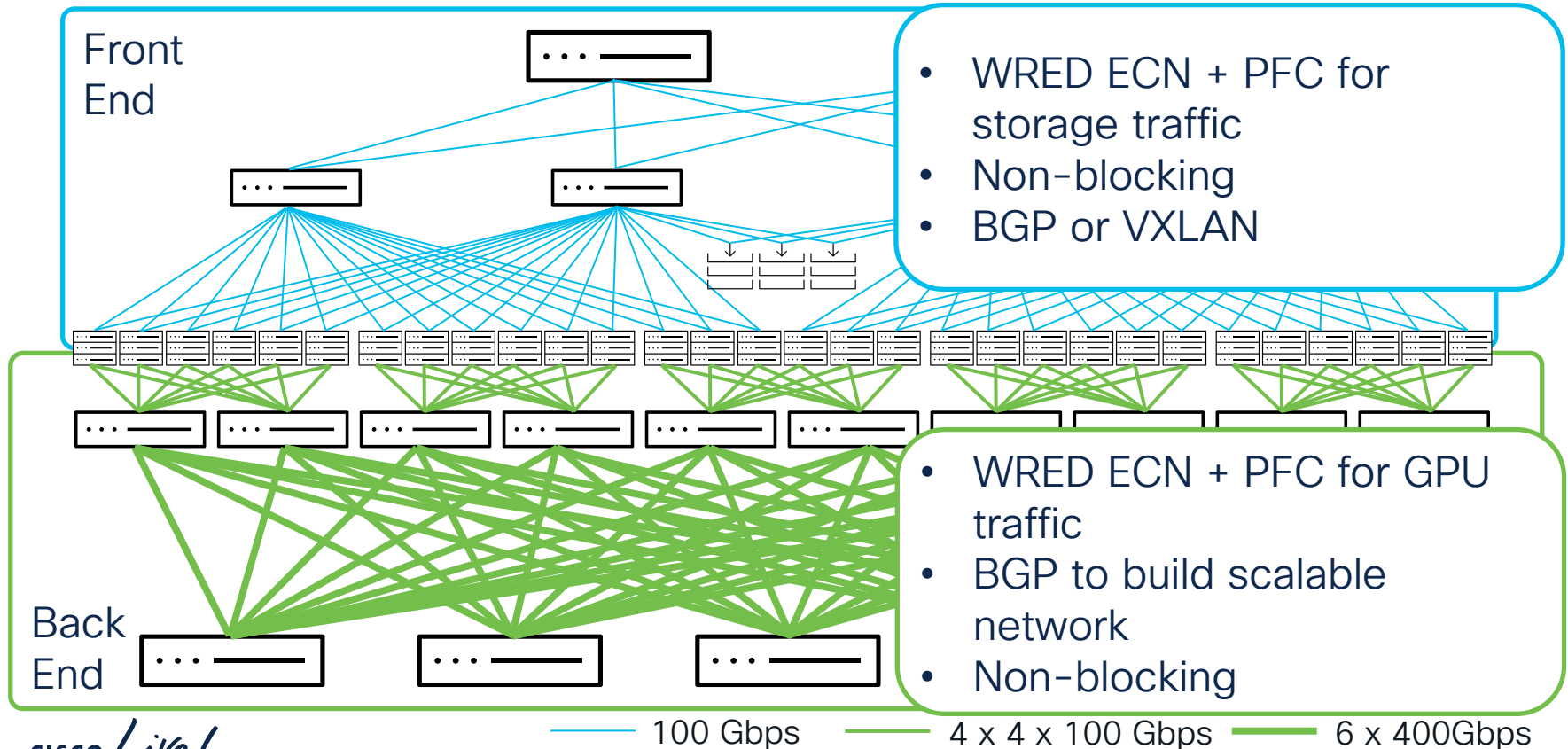


N9K-C9336C-FX2

Front-End and Back-End Cluster Network



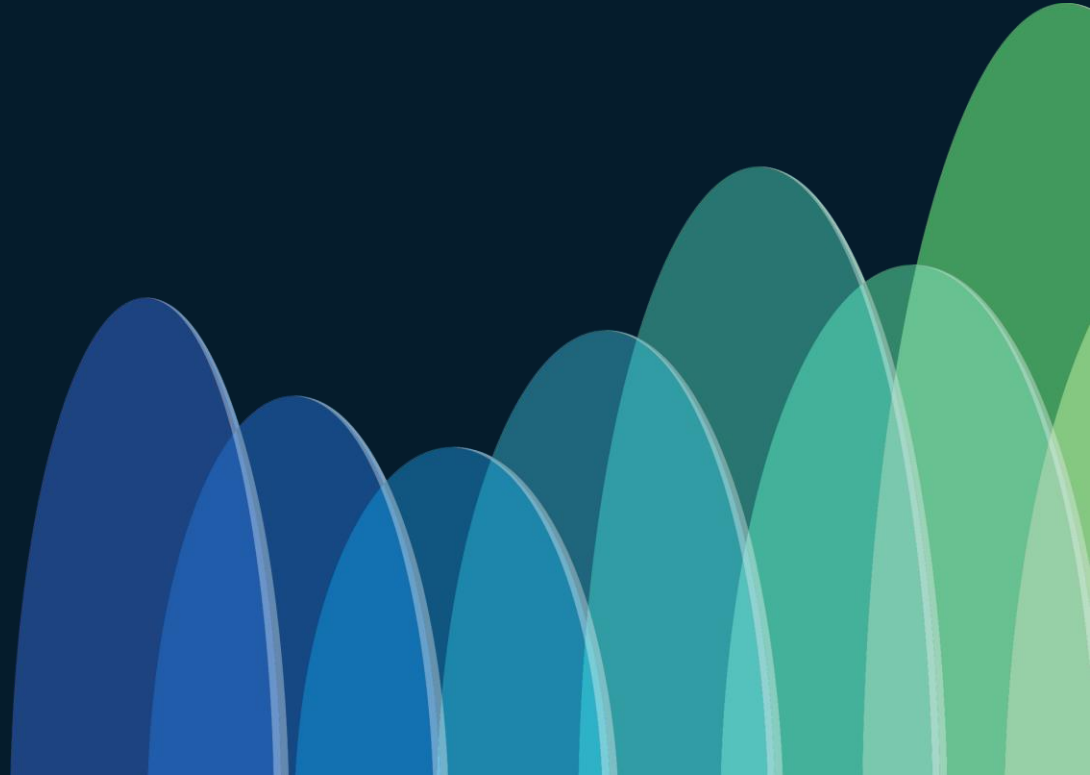
Front-End and Back-End Cluster Network



Customer request #2 - Design

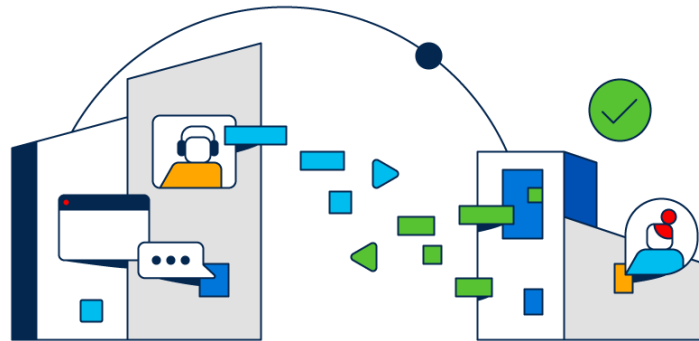
- Back-End Fabric is running BGP
 - Non-Blocking design
 - Congestion Management through ECN and PFC for optimal performance
 - Efficient traffic load-balancing in the fabric with UDF ECMP
 - Automated with Nexus Dashboard Fabric Controller
 - Visibility with Nexus Dashboard Insights
- Front-End Fabric
 - Congestion management for NFS traffic, with ECN and PFC
 - Automated with Nexus Dashboard Fabric Controller
 - Visibility with Nexus Dashboard Insights

Conclusion



Takeaways

- Robust back-end network, and flexible front-end network
- Familiar data center fabric technologies, BGP or VXLAN
- Automate network for easier bring up and operation
- Visibility in network congestion, and bottleneck to troubleshoot and optimize



The Blueprint For Today



Products and Services Solutions Support Learn Partners

Explore Cisco Search

Products & Services / Cloud and Systems Management / Cisco Nexus Dashboard Fabric Controller / White Papers /

Cisco Data Center Networking Blueprint for AI/ML Applications

Updated: May 24, 2023

Bias-Free Language Contact Cisco

Introduction

Table of Contents

Introduction

RoCEv2 as Transport for AI Clu...

AI Clusters Require Lossless N... +

How to Manage Congestion Eff... +

How Visibility into Network Be... +

Network Design to Accommod... +

Conclusion

Related Materials

Introduction

RoCEv2 as Transport for AI Clusters

AI Clusters Require Lossless Networks

Explicit Congestion Notification (ECN)

Priority Flow Control (PFC)

How to Manage Congestion Efficiently in AI/ML Cluster Networks

How ECN Works

How PFC Works

Using ECN and PFC Together to Build Lossless Ethernet Networks

Using Approximate Fair Drop (AFD)

Save Download Print



CISCO Live!

#CiscoLiveAPJC

BRKDCN-2921

© 2024 Cisco and/or its affiliates. All rights reserved. Cisco Public

108

The Blueprint For Today

 Product ▾ Solutions ▾ Open Source ▾ Pricing

Search / Sign in Sign up

allenrobel / NDFC-AIML-Fabric Public

Notifications Fork 0 Star 0

<> Code Issues Pull requests Actions Projects Security Insights

main 1 branch 0 tags

 allenrobel

Ignore .graffle files

6101d38 47 minutes ago 10 commits

doc	Update with current 'show running-config ipqos'	1 hour ago
inventory	Initial commit	5 days ago
.gitignore	Ignore .graffle files	47 minutes ago
AIML_Fabric.yml	Template name will probably change, so let's not hardcode it	2 hours ago
AI_Cluster_QOS_template.template	Adding new QOS template	2 hours ago
README.md	Add topology diagram	5 days ago
ansible.cfg	Initial commit	5 days ago

README.md

NDFC-AIML-Fabric

About

About

Ansible playbook to create an NDFC fabric which supports AI/ML workloads

- Readme
- 0 stars
- 2 watching
- 0 forks

Report repository

Releases

No releases published

Packages

No packages published



The Blueprint For Today



Products and Services Solutions Support Learn Partners

Explore Cisco Search

Preferred Networks, Inc. Eliminates Overlapping Investment and Network Bottlenecks

< Back to URL

Updated: October 15, 2021

[Bias-Free Language](#)

Challenge



Table of Contents

Preferred Networks, Inc.

Challenge

Solution

Results and the future

Learn more



Save



Download



Print

Preferred Networks, Inc. chose Cisco due to high reliability, quick response to the latest protocol, and hardware-based streaming telemetry and used Integrated Interconnect Network for deep-learning computing infrastructure into Ethernet and eliminated overlapping investment and network bottlenecks.

Executive Summary

Customer Name: Preferred Networks, Inc.

Industry: Artificial Intelligence

Location: Chiyoda-ku, Tokyo

Number of Employees: Approximately 300



Related sessions

Session ID	Session Title	Day and Time
BRKDCN-2613	Building a High-Performance AI Data Center using Nexus, UCS, and NVIDIA GPUs	Wednesday, Nov 13, 3:00 PM
BRKDCN-2999	Multi-Tier Fabric-Networks Designs for the Modern Data Center	Cisco Live On-demand Library
BRKDCN-1619	Introduction to NDFC: Simplifying Management of Your Data Center	Cisco Live On-demand Library

Complete Your Session Evaluations



Complete a minimum of 4 session surveys and the Overall Event Survey to claim a **Cisco Live T-Shirt**.



Complete your surveys in the **Cisco Live mobile app**.



Continue your education



- Visit the Cisco Stand for related demos
- Book your one-on-one Meet the Expert meeting
- Attend the interactive education with DevNet, Capture the Flag, and Walk-in Labs
- Visit the On-Demand Library for more sessions at www.CiscoLive.com/on-demand



Thank you



#CiscoLiveAPJC

CISCO *Live!*

GO BEYOND

#CiscoLiveAPJC