Spectrum-X RCP & POC

April 2025

Agenda

- Spectrum-X Reference Configuration
- Reference Configuration Playbook – RCP
- Networking Features and Debug Flows
- CloudAI Benchmark Framework
- Spectrum-X PoC

Spectrum-X Reference Configuration

Spectrum-X Protocols and Network Configuration

RCP 1.1.0

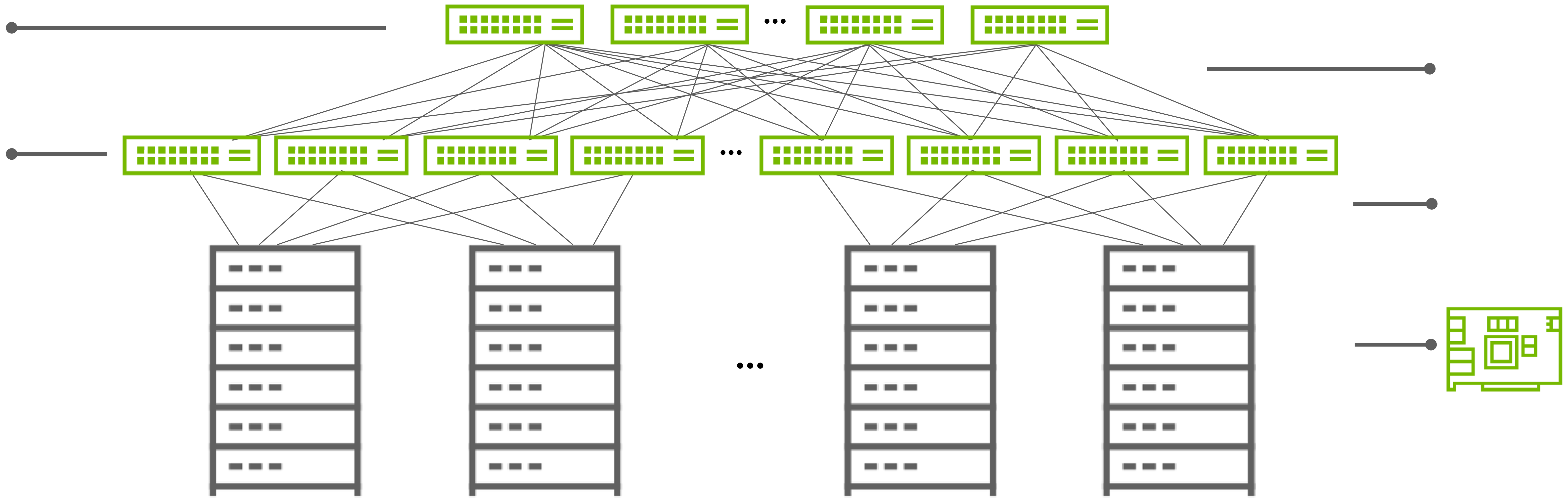
Component Versions

Component	Version
Cumulus Linux (CL) NOS	5.10.0.0037
BlueField Firmware Bundle (BFB)	2.8.0-98
DOCA Host	2.8.0-0.0.7
BlueField-3 Firmware	32.42.1000
NVIDIA Network Congestion Control (NVN-CC) Algorithm	2.8.0082-1
NCCL	2.21
NetQ	4.11

Spectrum-X Reference Architecture : https://apps.nvidia.com/pid/contentlibraries/detail?id=1123018
Spectrum-X deployment guide : https://apps.nvidia.com/pid/contentlibraries/detail?id=1129703

Spectrum-X Rail Optimized Leaf and Spine

Spectrum SN5600 + BlueField-3 GPU-to-GPU Fabric for 8K Hopper GPU Cloud



1 SU: 32 Nodes (256 GPU)
8x 400G per Node
4 Leaf Switches

Spectrum-X Protocols and Network Configuration

Host preparation

- doca-host package installation
- doca-ofed installation
- netplan, **lldpd** installation
- Restart rshim driver
- **doca-nvn-cc** package installation
- Install BF Firmware Bundle on each rshim
- Reboot host

Spectrum-X Protocols and Network Configuration

Host Devices

Device names can be different per system.

- Net devices - `enp26s0f0np0, enp27s0f0np0, enp77s0f0np0, enp94s0f0np0, enp156s0f0np0, enp188s0f0np0, enp204s0f0np0, enp220s0f0np0`
- RDMA devices - `mlx5_0,3,4,5,7,10,11`
- MST devices - `mt41692_pciconf0-1,2,3,4,5,7,8,9`
- PCI address - `1a:00.0,3c:00.0,4d:00.0,5e:00.0,9c:00.0,bc:00.0,cc:00.0,dc:00.0`

```
mst start
mst status -v
MST modules:
------------
    MST PCI module is not loaded
    MST PCI configuration module loaded
PCI devices:
------------
DEVICE_TYPE             MST                           PCI       RDMA            NET                                     NUMA
BlueField3(rev:1)       /dev/mst/mt41692_pciconf9     dc:00.0   mlx5_11         net-enp220s0f0np0                       1
BlueField3(rev:1)       /dev/mst/mt41692_pciconf8     cc:00.0   mlx5_10         net-enp204s0f0np0                       1
BlueField3(rev:1)       /dev/mst/mt41692_pciconf7     bc:00.0   mlx5_9          net-enp188s0f0np0                       1
BlueField3(rev:1)       /dev/mst/mt41692_pciconf6.1   9d:00.1   mlx5_8          net-enp157s0f1np1                       1
BlueField3(rev:1)       /dev/mst/mt41692_pciconf6     9d:00.0   mlx5_7          net-enp157s0f0np0                       1
BlueField3(rev:1)       /dev/mst/mt41692_pciconf5     9c:00.0   mlx5_6          net-enp156s0f0np0                       1
BlueField3(rev:1)       /dev/mst/mt41692_pciconf4     5e:00.0   mlx5_5          net-enp94s0f0np0                        0
BlueField3(rev:1)       /dev/mst/mt41692_pciconf3     4d:00.0   mlx5_4          net-enp77s0f0np0                        0
BlueField3(rev:1)       /dev/mst/mt41692_pciconf2     3c:00.0   mlx5_3          net-enp60s0f0np0                        0
BlueField3(rev:1)       /dev/mst/mt41692_pciconf1.1   1b:00.1   mlx5_2          net-enp27s0f1np1                        0
BlueField3(rev:1)       /dev/mst/mt41692_pciconf1     1b:00.0   mlx5_1          net-enp27s0f0np0                        0
BlueField3(rev:1)       /dev/mst/mt41692_pciconf0     1a:00.0   mlx5_0          net-enp26s0f0np0                        0
```

Spectrum-X Protocols and Network Configuration

BF3 Adaptive Routing

BF3 SuperNICs are enabled with Adaptive Routing and Congestion Control on all MST `mt41692_pciconf$i` devices

- AR and PCC (Programable Congestion Control)

```
mlxconfig -d /dev/mst/mt41692_pciconf$i -y set LINK_TYPE_P1=2 INTERNAL_CPU_OFFLOAD_ENGINE=1 ROCE_ADAPTIVE_ROUTING_EN=1 USER_PROGRAMMABLE_CC=1
TX_SCHEDULER_LOCALITY_MODE=2
```

- RoCE AR acceleration engines

```
mlxreg -d /dev/mst/mt41692_pciconf$i --reg_name ROCE_ACCL –set roce_adp_retrans_en=0x1,
roce_tx_window_en=0x1,roce_slow_restart_en=0,roce_slow_restart_idle_en=0x0,adaptive_routing_forced_en=0x1 --yes
```

```
Device #1:
----------
Device type:        BlueField3
Configurations:                              Default            Current          Next Boot
*       LINK_TYPE_P1                        IB(1)              ETH(2)             ETH(2)
        INTERNAL_CPU_OFFLOAD_ENGINE         DISABLED(1)        DISABLED(1)        DISABLED(1)
*       ROCE_ADAPTIVE_ROUTING_EN            False(0)           True(1)            True(1)
*       USER_PROGRAMMABLE_CC                False(0)           True(1)            True(1)
*       TX_SCHEDULER_LOCALITY_MODE          DEVICE_DEFAULT(0)  ACCUMULATIVE(2)    ACCUMULATIVE(2)
The '*' shows parameters with next value different from default/current value.
```

doca-host, doca-ofed, mft, bfb and nvncc must be installed on the host

Spectrum-X Protocols and Network Configuration

RoCE and QoS on BF3

RoCEv2 on all RDMA `mlx5_$i` devices

- ToS set to 96 → default to prio3

```
cma_roce_tos -d mlx5_x -t 96
cma_roce_tos -d mlx5_x
```

- ToS 96 is mapped to traffic-class 1

```
echo 96 > /sys/class/infiniband/mlx5_$i/tc/1/traffic_class
cat /sys/class/infiniband/mlx5_$i/tc/1/traffic_class
```

PFC and classification are set on all network devices

- PFC enabled on prio3
- Classification (trust state) set to DSCP

```
mlnx_qos -i enp$i --pfc=0,0,0,1,0,0,0,0 --trust=dscp
mlnx_qos -i enp26s0f0np0
```

```
root@hgx-pod00-su00-n00:~# mlnx_qos -i enp26s0f0np0
DCBX mode: OS controlled
Priority trust state: dscp
dscp2prio mapping:
        prio:0 dscp:07,06,05,04,03,02,01,00,
        prio:1 dscp:15,14,13,12,11,10,09,08,
        prio:2 dscp:23,22,21,20,19,18,17,16,
        prio:3 dscp:31,30,29,28,27,26,25,24,
        prio:4 dscp:39,38,37,36,35,34,33,32,
        prio:5 dscp:47,46,45,44,43,42,41,40,
        prio:6 dscp:55,54,53,52,51,50,49,48,
        prio:7 dscp:63,62,61,60,59,58,57,56,
default priority:
Receive buffer size (bytes): 19872,278208,0,0,0,0,0,0,max_buffer_size=4151520
Cable len: 7
PFC configuration:
        priority    0   1   2   3   4   5   6   7
        enabled     0   0   0   1   0   0   0   0
        buffer      0   0   0   1   0   0   0   0
tc: 0 ratelimit: unlimited, tsa: vendor
         priority:  1
tc: 1 ratelimit: unlimited, tsa: vendor
         priority:  0
tc: 2 ratelimit: unlimited, tsa: vendor
         priority:  2
tc: 3 ratelimit: unlimited, tsa: vendor
         priority:  3
tc: 4 ratelimit: unlimited, tsa: vendor
         priority:  4
tc: 5 ratelimit: unlimited, tsa: vendor
         priority:  5
tc: 6 ratelimit: unlimited, tsa: vendor
         priority:  6
tc: 7 ratelimit: unlimited, tsa: vendor
         priority:  7
```

Spectrum-X Protocols and Network Configuration

Host NVIDIA Network Congestion Control (NVN-CC)

BF3 SuperNICs are configured to NVN-CC and tuned for AI - the switch handles ECN and RTT

- Congestion control is enabled on all network `enp$i` interfaces

```
echo 1 > /sys/class/net/enp$i/ecn/roce_rp/enable/{0-7}
echo 1 > /sys/class/net/enp$i/ecn/roce_np/enable/{0-7}
```

```
cat /sys/class/net/enp$i/ecn/roce_rp/enable/{0-7}
cat /sys/class/net/enp$i/ecn/roce_rp/enable/{0-7}
```

- DOCA NVN-CC is enabled on all RDMA `mlx5_$i` devices - doca-nvn-cc package must be installed on the host first

```
/opt/mellanox/doca/tools/doca_nvn_cc -d mlx5_$i
```

- DCQCN (default CC) and NVN-CC-RTT (CC counters) are disabled on all MST `mt41692_pciconf$i` devices

```
mlxreg -d /dev/mst/mt41692_pciconf$i -y --set "cmd_type=2" --reg_name PPCC --indexes "local_port=1,pnat=0,lp_msb=0,algo_slot=15,algo_param_index=0"
mlxreg -d /dev/mst/mt41692_pciconf$i -y --set "cmd_type=2" --reg_name PPCC --indexes "local_port=1,pnat=0,lp_msb=0,algo_slot=1,algo_param_index=0"
```

- PCC and **NVN-CC** are enabled on all MST `mt41692_pciconf$i` devices

```
mlxreg -d /dev/mst/mt41692_pciconf$i -y --set "cmd_type=1" --reg_name PPCC --indexes "local_port=1,pnat=0,lp_msb=0,algo_slot=0,algo_param_index=0"
```

Spectrum-X Protocols and Network Configuration

Host Inter Packet Gap (IPG)

IPG is set to reduce tail latency

- In a pure L3 fabric, the IPG is set to `0x00000019` on all MST `mt41692_pciconf$i` devices

```
mlxreg -d /dev/mst/mt41692_pciconfx --set "ipg=0x00000019" --reg_name PIPG --indexes "local_port=1,lp_msb=0,ipg_cap_index=0" -y
```

- In a multi-tenant fabric (L3EVPN overlay), the IPG is set to `0x00000021` on all MST `mt41692_pciconf$i` devices

```
mlxreg -d /dev/mst/mt41692_pciconfx --set "ipg=0x00000021" --reg_name PIPG --indexes "local_port=1,lp_msb=0,ipg_cap_index=0" -y
```

- To apply the IPG, MST devices are toggled

```
mlxreg -d /dev/mst/mt41692_pciconfx --reg_id 0x5006 --set "0x0.8:4=2,0x0.16:8=1,0x4.8:1=1,0x4.31:1=1" --reg_len 16 -y
mlxreg -d /dev/mst/mt41692_pciconfx --reg_id 0x5006 --set "0x0.8:4=1,0x0.16:8=1,0x4.8:1=1,0x4.31:1=1" --reg_len 16 -y

mlxreg -d /dev/mst/mt41692_pciconf$i --show_reg PIPG
```

Spectrum-X Protocols and Network Configuration

Host IP Routing

Host Netplan configuration

- IP address and MTU
- Static routes:
  - Rail-specific (/11)
  - Inter-rail (/8)

netplan.io package must be installed on the host

```
ubuntu@ubuntu:~$ cat /etc/netplan/dpu_config.yaml
---
network:
  version: 2
  renderer: networkd
  ethernets:
    enp26s0f0np0:
      addresses: [172.0.0.0/31]
      routes:
      - to: 172.0.0.0/11
        via: 172.0.0.1
      - to: 172.0.0.0/8
        via: 172.0.0.1
      mtu: 9216
    enp77s0f0np0:
      addresses: [172.128.0.0/31]
      routes:
      - to: 172.128.0.0/11
        via: 172.128.0.1
      - to: 172.0.0.0/8
        via: 172.128.0.1
      mtu: 9216
...
```

Spectrum-X Protocols and Network Configuration

Host IPv4 Addressing Schema

Well-defined host IP schema for easier troubleshooting

- Rail ID (3 bits) = 8 rails - Mandatory
- POD ID (5 bits) – up to 32 PODs
- SU ID (8 bits) – up to 128 SUs (we aim for 64 SUs)
- Node ID (5 bits) – up to 32 nodes per SU
- NIC (1bit) – p2p address (host gets the **even**, leaf gets the **odd**)
- First octet is user-defined

X.00000000.00000000.00000000/31

172.224.3.62/31

172.11100000.00000011.00111110/31

Rail 7 of HGX host 32 in SU03 of POD00

Spectrum-X Protocols and Network Configuration

RoCE and QoS on Switch

Lossless RoCE configuration (buffer pools, PFC based on DSCP, and ECN)

- Default RoCE mode using "Do-RoCE"

```
nv set qos roce enable on
```

```
nv show qos roce
```

- Optimized traffic pools for RDMA in AI fabrics – 90% buffer memory to RDMA traffic and 10% for the rest

```
nv set qos traffic-pool default-lossy memory-percent 10
nv set qos traffic-pool roce-lossless memory-percent 90
```

```
nv show qos traffic-pool [default-lossy|roce-lossless]
```

Tuned default Cumulus Linux QoS profile in L3EVPN fabrics

- Reduced port-buffer size `(539.06 KB → 344.62 KB)`
- Increased PFC priority 3 xoff/xon thresholds `(46.88/37.88 KB → 63.38/63.38 KB)`

```
nv set qos pfc default-global port-buffer 353000
nv set qos pfc default-global switch-priority
nv set qos pfc default-global xoff-threshold 65000
nv set qos pfc default-global xon-threshold 65000
```

```
nv show qos pfc default-global
nv show interface <int> qos pfc
```

PFC-Watchdog

- Default PFC-WD configuration on all switch interfaces to eliminate PFC pause storm

```
nv set interface ${ALL_PORTS} qos pfc-watchdog state enable
```

```
nv show qos pfc-watchdog
```

Spectrum-X Protocols and Network Configuration

Switch Adaptive Routing

Custom Adaptive Routing profile tuned specially for AI fabrics

- Global and interface-level AR configuration
- AR profile configuration file placed under `/etc/cumulus/switch.d/`
- **AR must be set only on inter-switch links (not towards hosts)**

```
nv set router adaptive-routing enable on
nv set interface <int-range> router adaptive-routing enable on
nv set router adaptive-routing profile profile-custom
```

```
nv show router adaptive-routing
nv show router interface <int> router adaptive-routing
cumulus@switch:mgmt:~$ cat /etc/cumulus.switch.d/ar_profile_custom.conf
ar.p.m = 0
ar.ctl = 400
ar.ctm = 800
ar.cth = 2000
ar.srt = 10
ar.srf = 10
ar.p.bit = 0
ar.p.frt = 4
ar.p.but = 0
ar.p.sfe = FALSE
ar.p.ste = FALSE
ar.p.ef = FALSE
ar.ecs = 512
ar.ibm = ingress
```

Spectrum-X Protocols and Network Configuration

Routing Configuration Overview

BGP for routing in large-scale data centers (RFC7938)

- Numbered links (RFC3927 and RFC6598)
- Private 4-byte ASN
- Fast converging and resilient network
  - Minimal fabric state → Efficient prefix distribution
  - Underlay and overlay separation

Easy to automate and troubleshoot

- Simple and repetitive configuration
- Well-defined IP schema for each layer + Reusable IP addressing (POD underlay)
- Static identifiers (BGP peers, tenants RD, descriptions, etc)

Multi-tenancy support

- Routed EVPN
- Pure L3: Single tenant (or underlay network for host multi-tenancy)

Spectrum-X Protocols and Network Configuration

Switches IPv4 Addressing Schema

Leaf to Spine

- Private IP – `10.254.0.0/16`
- Well-defined schema of point-to-point (p2p) addresses – `10.254.<rail-group-spine-ID>.0/31`
- Spine-ID starts with 0 with max spines per POD = 256 → `10.254.0-255.0-127/31`
- Range is not advertised → Repeatable across PODs

Spine to Super-Spine

- RFC6598 - IANA-Reserved IPv4 Prefix for Shared Address Space – `100.64.0.0/10`
- Well-defined schema of point-to-point (p2p) addresses – `100.<64+SSP-Group-ID>.SSP-ID.0/31`
- SSP-Group-ID starts with 64 with max SSP groups = 64 and max SSP/group = 128 → `100.64-127.0-127.0-255/31`

Leaf to Host

- Pure L3 / L3 EVPN - p2p /31 addresses from the same supernet. Even on the host, odd on a leaf.

```
nv set interface <int< ip address x.y.z.w/31
nv show interface [int]
```

Loopbacks

- Sequential allocation based on `10.0.X.X/32` for leafs, `10.1.X.X/32` for spines, and `10.2.X.X/32` for SSP.

```
nv set interface lo ip address x.y.z.w/32
```

Spectrum-X Protocols and Network Configuration

Routing Policies – W-ECMP

W-ECMP is enabled using route-map with `ext-community-bw` set to the underlay BGP neighbors

- Originated (`mulitpaths`) on the uplinks – leaf (+ spine in 3-tier topology)

```
nv set router policy route-map w-ecmp-origin rule 10 action permit
nv set router policy route-map w-ecmp-origin rule 10 description enable_w_ecmp_origination
nv set router policy route-map w-ecmp-origin rule 10 set ext-community-bw multipaths
nv show router policy route-map w-ecmp-origin rule 10
```

- Added (`Cumulative`) on the downlinks – spine (+ super-spine in 3-tier topology)

```
nv set router policy route-map w-ecmp-cumulative rule 10 action permit
nv set router policy route-map w-ecmp-cumulative rule 10 description enable_w_ecmp_origination
nv set router policy route-map w-ecmp-cumulative rule 10 set ext-community-bw cumulative
nv show router policy route-map w-ecmp-cumulative rule 10
```

Spectrum-X Protocols and Network Configuration

Routing Policies – Prefix distribution in L3 Fabric

IP prefix-list for all HGX subnets

- Single prefix-list duplicated across the fabric

```
nv set router policy prefix-list hgx_subnets rule 10 action permit
nv set router policy prefix-list hgx_subnets rule 10 match 172.0.0.0/8 max-prefix-len 31
nv set router policy prefix-list hgx_subnets type ipv4
```

```
nv show router policy prefix-list hgx_subnets rule 10
```

- Used in a route-map (combined with loopback interface) on **leaf only** → redistributed (connected) into BGP **default** VRF

Route-map for all loopback IP addresses:

- Combined route-map on **leafs only** (with the prefix-list) → redistributed (connected) into BGP **default** VRF

```
nv set router policy route-map hgx_subnets_and_lo rule 10 action permit
nv set router policy route-map hgx_subnets_and_lo rule 10 description permit_hgx_subnets
nv set router policy route-map hgx_subnets_and_lo rule 10 match ip-prefix-list hgx_subnets
nv set router policy route-map hgx_subnets_and_lo rule 10 match type ipv4
nv set router policy route-map hgx_subnets_and_lo rule 20 action permit
nv set router policy route-map hgx_subnets_and_lo rule 20 description permit_lo_ip
nv set router policy route-map hgx_subnets_and_lo rule 20 match interface lo
nv set router policy route-map hgx_subnets_and_lo rule 20 match type ipv4
```

```
nv show router policy route-map hgx_subnets_and_lo rule [10|20]
```

- Dedicated route-map on **spines/super-spines only** → redistributed (connected) into BGP **default** VRF

```
nv set router policy route-map lo-to-bgp rule 10 action permit
nv set router policy route-map lo-to-bgp rule 10 description permit_lo_ip
nv set router policy route-map lo-to-bgp rule 10 match interface lo
nv set router policy route-map lo-to-bgp rule 10 match type ipv4
```

```
nv show router policy route-map lo-to-bgp rule 10
```

Spectrum-X Protocols and Network Configuration

Routing Policies – Prefix distribution in L3EVPN Fabric

IP prefix-list for all HGX subnets

- Single prefix-list duplicated across the fabric

```
nv set router policy prefix-list hgx_subnets rule 10 action permit
nv set router policy prefix-list hgx_subnets rule 10 match 172.0.0.0/8 max-prefix-len 31
nv set router policy prefix-list hgx_subnets type ipv4

nv show router policy prefix-list hgx_subnets rule 10
```

- Used in a route-map on **leafs only** → redistributed (connected) into BGP **tenant** VRF

```
nv set router policy route-map hgx_subnets rule 10 action permit
nv set router policy route-map hgx_subnets rule 10 description permit_hgx_subnets
nv set router policy route-map hgx_subnets rule 10 match ip-prefix-list hgx_subnets
nv set router policy route-map hgx_subnets rule 10 match type ipv4

nv show router policy route-map hgx_subnets rule [10|20]
```

Route-map for all loopback IP addresses:

- Dedicated route-map on all switches (leafs, spines, super-spines) → redistributed (connected) into BGP **default** VRF

```
nv set router policy route-map lo-to-bgp rule 10 action permit
nv set router policy route-map lo-to-bgp rule 10 description permit_lo_ip
nv set router policy route-map lo-to-bgp rule 10 match interface lo
nv set router policy route-map lo-to-bgp rule 10 match type ipv4

nv show router policy route-map lo-to-bgp rule 10
```

Spectrum-X Protocols and Network Configuration

Underlay Network

eBGP network following RFC7938 – BGP for routing in large-scale data centers

- 4-byte ASN:
  - Unique ASN for each leaf – `4200000000 and up`
  - Common ASN on all rail-group spines in the POD – `4201000000 and up`
  - Common ASN on all super-spines (all spines in 2-tier topology) - `4202000000`

```
nv set router bgp autonomous-system <ASN>
```

- Numbered neighbors:
  - All within a peer-group per-direction (`underlay_leaf|spine|super-spine`) and description

```
nv set vrf default router bgp neighbor <p2p-ip> description to-<peer_switch>-<peer-port>
nv set vrf default router bgp neighbor <p2p-ip> peer-group <peer-group-name>
nv set vrf default router bgp neighbor <p2p-ip> type numbered
nv set vrf default router bgp peer-group <peer-group-name> description underaly_switch_interconnect_to_XXX
```

  - Auto BGP

```
nv set vrf default router bgp peer-group <peer-group-name> remote-as external
```

- Fast converging and efficient network with minimal fabric state:
  - Only necessary prefixes in BGP (loopbacks and HGX networks)

```
nv set vrf default router bgp address-family ipv4-unicast redistribute connected route-map [lo-to-bgp|hgx_subnets/and_lo]
```

  - Per-rail, /26 aggregated route on leafs (/25 if single rail per-leaf) → 2 prefixes advertised to the network (summary per tenant VRF in the L3EVPN network)

```
nv set vrf default router bgp address-family ipv4-unicast aggregate-route 172.96.0.0/26 summary-only on
nv set vrf default router bgp address-family ipv4-unicast aggregate-route 172.224.0.0/26 summary-only on
```

Spectrum-X Protocols and Network Configuration

Underlay Network

- ECMP with ASPATH-ignore

```
nv set vrf default router bgp path-selection multipath aspath-ignore on
```

- W-ECMP

```
nv set vrf default router bgp peer-group <peer-group-name> address-family ipv4-unicast policy outbound route-map w-ecmp-[origin|cumulative]
```

- BFD (3x300ms)

```
nv set vrf default router bgp peer-group <peer-group-name> bfd enable on
nv set vrf default router bgp peer-group <peer-group-name> bfd detect-multiplier 3
nv set vrf default router bgp peer-group <peer-group-name> bfd min-rx-interval 300
nv set vrf default router bgp peer-group <peer-group-name> bfd min-tx-interval 300
```

```
nv show vrf default router bgp
nv show vrf default router bgp neighbor <peer-ip>
nv show vrf default router bgp peer-group <peer-group-name>
nv show vrf default router bgp neighbor brief
nv show vrf default router rib ipv4 route | grep bgp
```

Spectrum-X Protocols and Network Configuration

Virtualized Network

Overlay network with VXLAN data and EVPN control planes:

- VXLAN data plane
  - Fabric-based virtualization – **VTEP** on leafs

```
nv set nve vxlan enable on
nv set nve vxlan source address <local-lo-ip>
```

  - End-to-end QoS for RoCE traffic – DSCP value handling on encap/decap

```
nv set nve vxlan encapsulation dscp action copy
nv set nve vxlan decapsulation dscp action preserve

nv show nve
```

- EVPN control plane
  - Complete separation of the overlay and underlay network → Network Scalability
    - Multi-hop BGP over loopbacks (underlay on physical ports)

```
nv set vrf default router bgp neighbor <peer-lo-ip> description to-<peer-switch>-loopback
nv set vrf default router bgp neighbor <peer-lo-ip> peer-group overlay
nv set vrf default router bgp neighbor <peer-lo-ip> type numbered
nv set vrf default router bgp peer-group overlay update-source lo
nv set vrf default router bgp peer-group overlay multihop-ttl 3
nv set vrf default router bgp peer-group overlay description overlay_switch_interconnect
```

    - Only EVPN prefixes advertisement on the overlay sessions (IPv4 AF disabled)

```
nv set vrf default router bgp peer-group overlay address-family l2vpn-evpnipv4-unicast enable off
nv set vrf default router bgp peer-group overlay address-family enable on
```

Spectrum-X Protocols and Network Configuration

Virtualized Network

- Auto BGP

```
nv set vrf default router bgp peer-group overlay remote-as external
```

- BFD (3x1000ms)

```
nv set vrf default router bgp peer-group overlay bfd enable on
nv set vrf default router bgp peer-group overlay bfd detect-multiplier 3
nv set vrf default router bgp peer-group overlay bfd min-rx-interval 1000
nv set vrf default router bgp peer-group overlay bfd min-tx-interval 1000
```

```
nv show vrf default router bgp peer-group overlay
nv show vrf default router bgp neighbor brief | grep l2vpn-evpn
```

Spectrum-X Protocols and Network Configuration

Multi-Tenancy

Routed EVPN (L3EVPN) multi-tenancy with pure type-5 design

- Tenant = VRF → VRF-to-L3VNI mapping

```
nv set vrf tenant1 evpn enable on
nv set vrf tenant1 evpn vni 4001
```

- Physical port to tenant allocation

```
nv set interface swp1 vrf tenant1
```

- Separate BGP instance for each tenant VRF (IPv4 and EVPN AF)

```
nv set vrf tenant1 router bgp enable on
nv set vrf tenant1 router bgp address-family l2vpn-evpn enable on
```

- Static tenant RD allocation (router-id:l3vni)

```
nv set vrf tenant1 router bgp rd 10.0.0.1:4001
```

- Tenant HGX prefixes redistribution into the VRF and injection as EVPN external routes (type-5)

```
nv set vrf tenant1 router bgp address-family ipv4-unicast redistribute connected route-map hgx_subnets
nv set vrf tenant1 router bgp address-family ipv4-unicast route-export to-evpn enable on
```

- Per-rail, /26 Type-5 aggregated route (/25 if single rail per-leaf)

```
nv set vrf tenant1 router bgp address-family ipv4-unicast aggregate-route 172.96.0.0/26 summary-only on
nv set vrf tenant1 router bgp address-family ipv4-unicast aggregate-route 172.224.0.0/26 summary-only on
```

- Tenant VRF ECMP with ASPATH-ignore

```
nv set vrf tenant1 router bgp path-selection multipath aspath-ignore on
```

```
nv show vrf tenant1 evpn bgp-info
nv show vrf tenant1 evpn nexthop-vtep
nv show vrf tenant1 router bgp address-family l2vpn-evpn loc-rib | grep [5]
```

Spectrum-X Protocols and Network Configuration

Fabric Telemetry (ASIC Monitoring)

Buffer, counters, and BW telemetry using Spectrum-4 ASIC telemetry capabilities

- Available via NetQ or switch CLI
  - Telemetry service enabled on Cumulus Linux

```
nv set system telemetry enable on
```

  - Ingress (PG) and egress (TC) queue histograms
    - The TCs that are important to sample in the E-W network are **(NetQ supports only egress and only single TC → TC3 preferred)**
      - TC3 - for RoCE traffic
      - TC0 - for non-RoCE traffic
      - TC6 - for CNP and NVN-CC RTT probes
    - Priority Groups 0 and 1 are sampled for RDMA and non-RDMA traffic **(NetQ doesn't support ingress histograms)**

```
nv set system telemetry histogram [egress-buffer|ingress-buffer] bin-min-boundary 960
nv set system telemetry histogram [egress-buffer|ingress-buffer] histogram-size 2457600
nv set system telemetry histogram [egress-buffer|ingress-buffer] sample-interval 1024

nv set interface ${ALL_PORTS} telemetry histogram egress-buffer traffic-class [0,3,6]
nv set interface ${ALL_PORTS} telemetry histogram ingress-buffer priority-group [0,1]
```

  - Counter histograms

```
nv set system telemetry histogram counter bin-min-boundary 3552
nv set system telemetry histogram counter histogram-size 55008
nv set system telemetry histogram counter sample-interval 1024

nv set interface ${ALL_PORTS} telemetry histogram counter counter-type [rx-packet|tx-packet|rx-byte|tx-byte|crc]
```

  - Bandwidth gauge

```
nv set interface ${ALL_PORTS} telemetry bw-gauge enable on
```

  - Telemetry snapshots

```
nv set system telemetry snapshot-file name /var/run/cumulus/histogram_stats
nv set system telemetry snapshot-file count 120
nv set system telemetry snapshot-interval 1
```

Spectrum-X Protocols and Network Configuration

Fabric Telemetry

Cumulus use OpenTelemetry protocol (OTLP) to export telemetry data

- Enable OpenTelemetry

```
nv set system telemetry export otlp state enabled
```

- Set the mgmt. VRF for telemetry data export

```
nv set system telemetry export vrf mgmt
```

- Enable the export of histogram data:

```
nv set system telemetry histogram export state enabled
```

- Enable the export of counters:

```
nv set system telemetry interface-stats ingress-buffer priority-group [0,1]
nv set system telemetry interface-stats egress-buffer traffic-class [0,3,6]
nv set interface ${ALL_PORTS} telemetry bw-gauge enable on
```

- Configure external gRPC collector

```
nv set system telemetry export otlp grpc destination <IP|hostname> [port <port>]
nv set system telemetry export otlp grpc insecure enabled
```

Spectrum-X Protocols and Network Configuration

Fabric Telemetry (NetQ)

- Dedicated premises for E-W network
- AI fabric workbench and cards
  - Events
  - What Just happened (WJH)
  - Inventory | Switches
  - BGP
  - EVPN
- What Just Happened (WJH)
- Validation checks:
  - Agents
  - Sensors – PSU/fans
  - NTP service
  - Interfaces
  - MTU – Link MTU consistency
  - Duplicate IP – IPv4
  - RoCE
  - BGP
  - EVPN (L3EVPN deployment) – EVPN VNI consistency and VRF consistency
- RoCE and Adaptive Routing monitoring
- Threshold Crossing Events
  - Resource Utilization - CPU, disk, and memory Utilization - 90%
  - Forwarding Resources: IPv4 routes - 95%
  - Link Flaps - Physical links - 10 count, BGP Connection Drop - 1 count
  - ECMP - ECMP imbalance - 20%

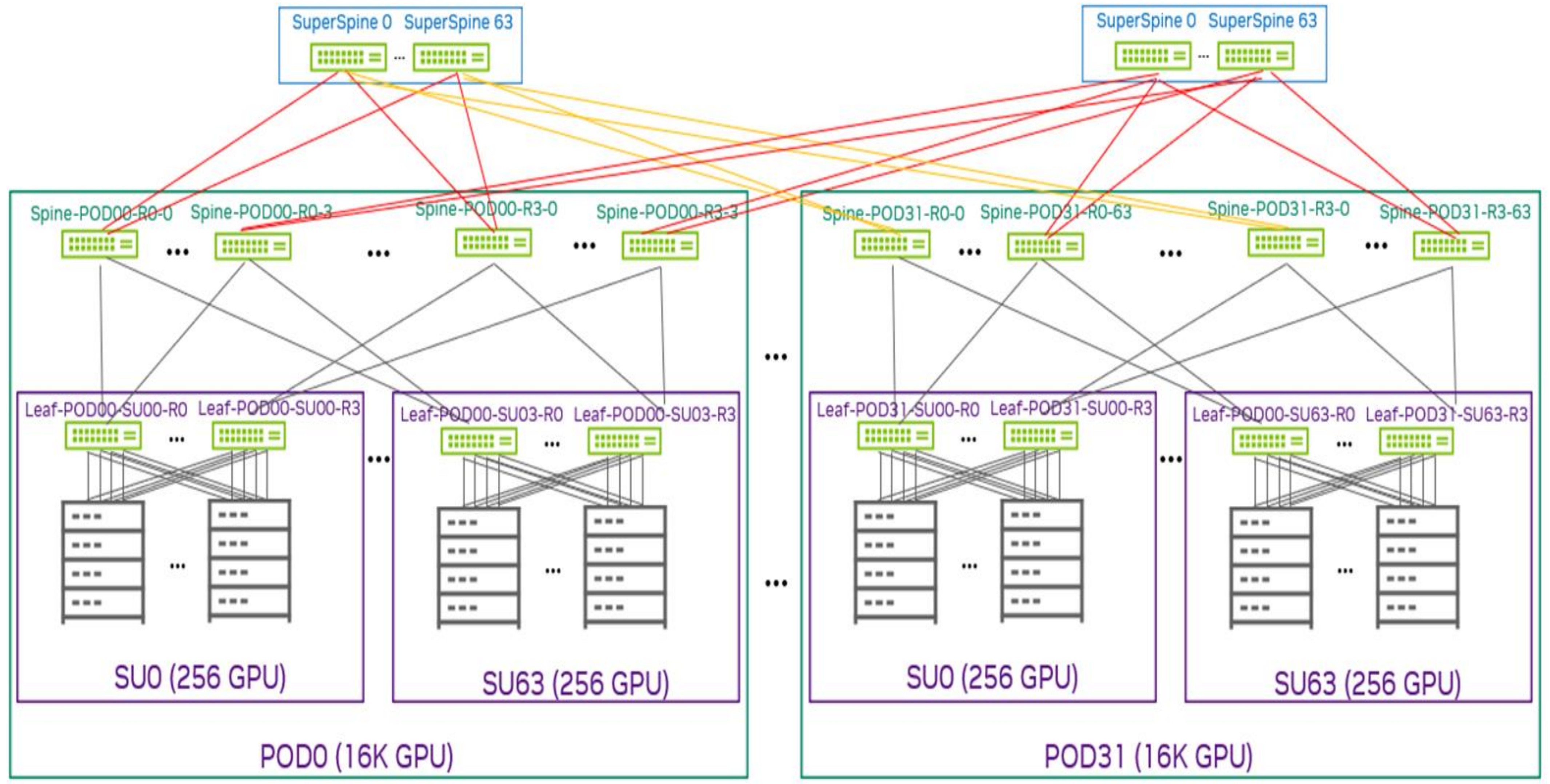
RCP – Reference Configuration Playbook

Spectrum-X Reference Configuration Playbook (RCP)

East-West Network

E-W network configuration automation tool

- Follows the Spectrum-X Reference Architecture and deployment guide 1.3.0
- Open-source (Nvonline PID: **1129714** spectrum-x-rcp-v1.3.0)
- Containerized (Docker)
- Ansible-based
  - NVUE ansible modules included
  - Easy integration to existing automation systems such as AWX
- End-to-end configuration, validation, and cleanup
  - Cumulus Linux switch (NVUE)
  - BlueField-3 SuperNIC (DOCA Management System - DMS)
- Multi-tenancy support
  - Bridged EVPN (L2VNI)
  - Routed EVPN (L3VNI)
  - OVN/HBN
- User-friendly RCP-CLI (and ansible)
- Up to 512K GPU scale cluster configuration including partial cluster
- Integration with Air (including topology generator)



RCP on NVIDIA Gitlab

Reference Configuration Playbook (RCP)

Version 1.0.1

Supported OS*

- Host: DGX OS 6.1, Ubuntu 22.04
- Switch: Cumulus Linux 5.10

Host preparation and configuration**

- Set BF3 to NIC mode
- IP allocation and routing
- DOCA for Host Installation
- BFB Installation
- Tunning for Spectrum-X (AR and NVN-CC)

Switch configuration***

- Interconnect IP allocation
- BGP (underlay and overlay)
- VXLAN-EVPN
- AR
- QoS and RoCE
- Telemetry (ingress/egress/counter histograms and snapshots)

Topology****

- Recommended
- Discovery (LLDP)
- Validation – actual vs. RA (mis-cabeling)

Validations

- Switch and host configuration (current to reference)
- Host connectivity validation (ping BF3 All-to-All)

Multi-tenancy support

- Routed EVPN (L3VNI)

*SONIC support - future

**Basic Linux configuration (DMS support future)

***Cumulus Linux configured using NVUE

****Only 3-tier topology supported (2-tier and 2-tier-poc - future)

Reference Configuration Playbook (RCP)

RCP-Tool Commands

Topology

```
# rcp-tool topology [--discover | --recommended | --validate]
```

Host preparation, configuration, and information

```
# rcp-tool host [--prepare | --configure | --clean | --show {ip-scheme}]
```

Switch configuration and information

```
# rcp-tool switch [--configure | --clean | --show {ports|brief}]
```

Configure or clean all at once

```
# rcp-tool all [--configure | --clean]
```

Validation

```
# rcp-tool validation [--host-config | --switch-config | --ping]
```

Reference Configuration Playbook (RCP)

Ansible Commands

Topology

```
ansible-playbook -i inventory topology/discover.yaml
ansible-inventory --list -i inventory -i topology/topology-plugin.yaml
```

Host preparation, configuration, and information

```
# ansible-playbook -i inventory host/prepare.yaml
# ansible-playbook -i inventory -i topology/topology-plugin.yaml host/configure.yaml
# ansible-playbook -i inventory host/clean.yaml
```

Switch configuration and information

```
# ansible-playbook -i inventory -i topology/topology-plugin.yaml switch/configure.yaml
# ansible-playbook -i inventory switch/clean.yaml
```

Validation

```
# ansible-playbook -i inventory -i topology/topology-plugin.yaml validation/[host|switch]-config.yaml
# ansible-playbook -i inventory -i topology/topology-plugin.yaml validation/ping.yaml
```

Ansible commands can be used on specific hosts using `--limit` flag

Reference Configuration Playbook (RCP)

Main Configuration File

```
# Cluster Environment
is_simulation: False # True, False

# Cluster Size
pod_num: 1 # 1-32
pod_size: 4 # 1-64

# Cluster Topology Design
# topology: "3-tier" # 2-tier, 2-tier-poc, 3-tier

# loopback first octet
loopback_first_octet: "10“

# Host Networking Parameters. List of NICS on the host from GPU0 to GPU7
host_interfaces: ["enp26s0f0np0","enp156s0f0np0","enp60s0f0np0","enp188s0f0np0","enp77s0f0np0","enp204s0f0np0","enp94s0f0np0","enp220s0f0np0"]

# Host to leaf first octet of the /8 subnet
host_first_octet: "172“

# E-W Multi-tenancy
overlay: "none" # none, l3evpn

# Configure 1 tenant for L3 EVPN
l3evpn_config: { "vrf": “tenant1”,
                 “vni”: 4001 }

# Telemetry
telemetry: "none" # none, experimental, netq

# Bluefield DPU software. This package(s) should be located on /doca
doca_for_host_pkg: “doca-host_2.7.0_204000-24.04-ubuntu2204_amd64.deb“
firmware: “bf-fwbundle-2.7.0-31_24.04-prod.bfb"
```

Reference Configuration Playbook (RCP)

Inventory

Ansible cluster inventory file

- Device order is important!
- Update device credentials
- All group names must be listed in `[switch:children]` group (even if empty)
- `[disabled]` group:
  - Used to skip device configuration by RCP
  - Devices must be listed in their groups even if are in the `[disabled]` group
- `[host]` group must list all the 32 HGX nodes of each SU (non-existing nodes must be put in the disabled group)
- Hostnames can be summarized, e.g.

  `leaf-pod00-su00-r[0:3]`

  `spine-pod00-r0-s[00:01]`

  `ssp-group00-s[00:03]`

  `hgx-pod00-su00-h[00:31]`

```
[all:vars]
ansible_user=cumulus
ansible_become_pass=cumulus
ansible_ssh_pass=cumulus

[host:vars]
ansible_user=ubuntu
ansible_bcome_pass=nvidia
ansible_ssh_pass=nvidia

[leaf]
leaf-pod00-su00-r0
leaf-pod00-su00-r1
leaf-pod00-su00-r2
leaf-pod00-su00-r3
...

[spine]
spine-pod00-r0-s00
spine-pod00-r0-s01
...

[super_spine]
ssp-group00-s00
ssp-group00-s01
...

[host]
hgx-pod00-su00-h00
hgx-pod00-su00-h01
...

[switch:children]
leaf
spine
super_spine

[disabled]
hgx-pod00-su00-h02
hgx-pod01-su03-h31
```

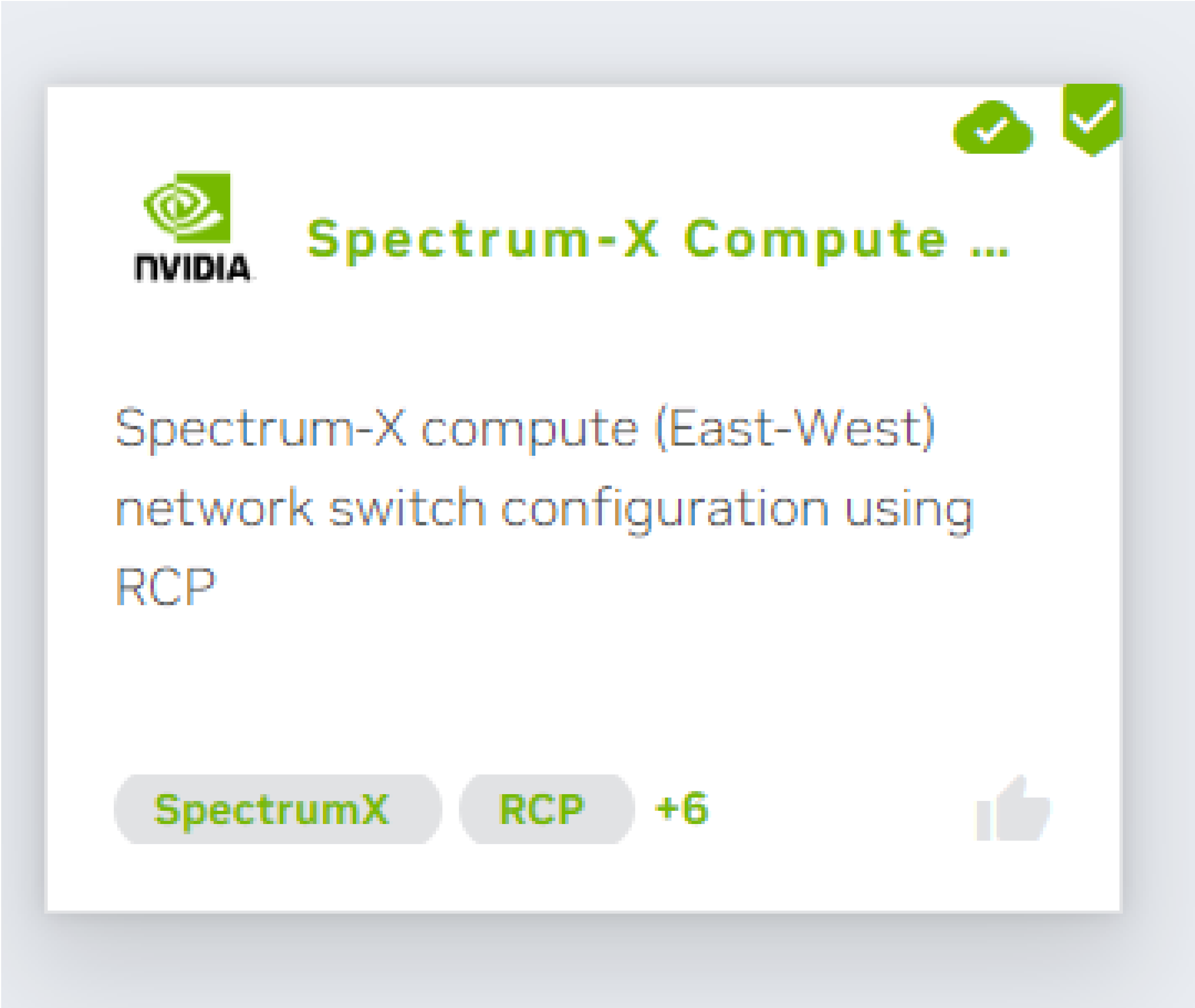
Validate RCP commands result

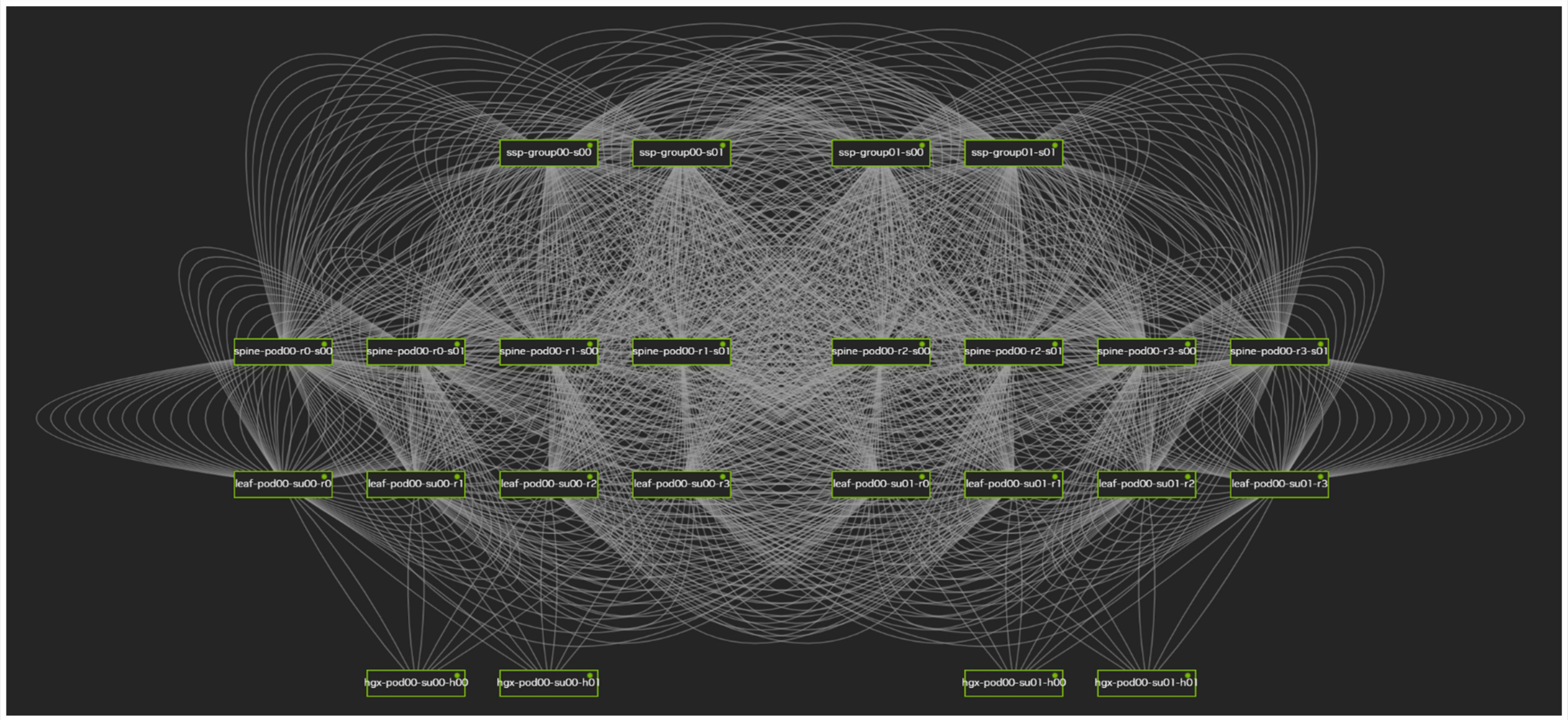
An example on Air/RCP

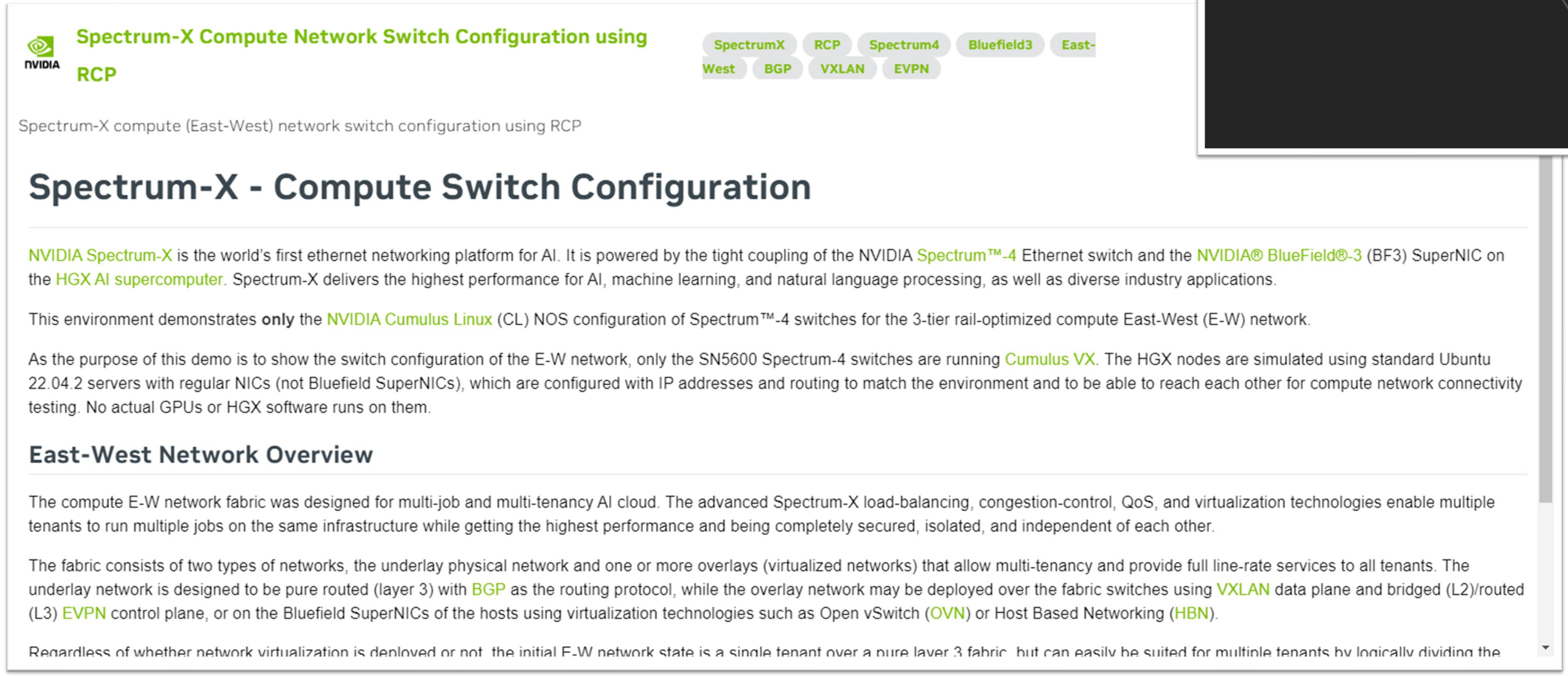
```
spectrum-x-rcp:V1.2.0 ~/spectrum-x-rcp # rcp-tool host show --ip_scheme
+--------------+---------------+---------------+------+-----------+----------+------------+--------------+
| host         | port          | ip address    | rail | pod index | su index | host index | rail_subnet  |
+--------------+---------------+---------------+------+-----------+----------+------------+--------------+
| hgx-su00-h00 | enp26s0f0np0  | 172.16.0.0/31 | 0    | XX        | 0        | 0          | 172.16.0.0/15 |
| hgx-su00-h00 | enp60s0f0np0  | 172.18.0.0/31 | 1    | XX        | 0        | 0          | 172.18.0.0/15 |
| hgx-su00-h00 | enp77s0f0np0  | 172.20.0.0/31 | 2    | XX        | 0        | 0          | 172.20.0.0/15 |
| hgx-su00-h00 | enp94s0f0np0  | 172.22.0.0/31 | 3    | XX        | 0        | 0          | 172.22.0.0/15 |
| hgx-su00-h00 | enp156s0f0np0 | 172.24.0.0/31 | 4    | XX        | 0        | 0          | 172.24.0.0/15 |
| hgx-su00-h00 | enp188s0f0np0 | 172.26.0.0/31 | 5    | XX        | 0        | 0          | 172.26.0.0/15 |
| hgx-su00-h00 | enp204s0f0np0 | 172.28.0.0/31 | 6    | XX        | 0        | 0          | 172.28.0.0/15 |
| hgx-su00-h00 | enp220s0f0np0 | 172.30.0.0/31 | 7    | XX        | 0        | 0          | 172.30.0.0/15 |
| hgx-su00-h01 | enp26s0f0np0  | 172.16.0.2/31 | 0    | XX        | 0        | 1          | 172.16.0.0/15 |
| hgx-su00-h01 | enp60s0f0np0  | 172.18.0.2/31 | 1    | XX        | 0        | 1          | 172.18.0.0/15 |
| hgx-su00-h01 | enp77s0f0np0  | 172.20.0.2/31 | 2    | XX        | 0        | 1          | 172.20.0.0/15 |
| hgx-su00-h01 | enp94s0f0np0  | 172.22.0.2/31 | 3    | XX        | 0        | 1          | 172.22.0.0/15 |
| hgx-su00-h01 | enp156s0f0np0 | 172.24.0.2/31 | 4    | XX        | 0        | 1          | 172.24.0.0/15 |
| hgx-su00-h01 | enp188s0f0np0 | 172.26.0.2/31 | 5    | XX        | 0        | 1          | 172.26.0.0/15 |
| hgx-su00-h01 | enp204s0f0np0 | 172.28.0.2/31 | 6    | XX        | 0        | 1          | 172.28.0.0/15 |
| hgx-su00-h01 | enp220s0f0np0 | 172.30.0.2/31 | 7    | XX        | 0        | 1          | 172.30.0.0/15 |
+--------------+---------------+---------------+------+-----------+----------+------------+--------------+
spectrum-x-rcp:V1.2.0 ~/spectrum-x-rcp #
spectrum-x-rcp:V1.2.0 ~/spectrum-x-rcp # rcp-tool host show --ports
+--------------+---------------+------------+--------------+-----------+-----------------+
| host         | port          | ip address | peer         | peer port | peer ip address |
+--------------+---------------+------------+--------------+-----------+-----------------+
| hgx-su00-h00 | enp26s0f0np0  | 172.16.0.0 | leaf-su00-r0 | swp1s0    | 172.16.0.1      |
| hgx-su00-h00 | enp60s0f0np0  | 172.18.0.0 | leaf-su00-r1 | swp1s0    | 172.18.0.1      |
| hgx-su00-h00 | enp77s0f0np0  | 172.20.0.0 | leaf-su00-r2 | swp1s0    | 172.20.0.1      |
| hgx-su00-h00 | enp94s0f0np0  | 172.22.0.0 | leaf-su00-r3 | swp1s0    | 172.22.0.1      |
| hgx-su00-h00 | enp156s0f0np0 | 172.24.0.0 | leaf-su00-r0 | swp1s1    | 172.24.0.1      |
| hgx-su00-h00 | enp188s0f0np0 | 172.26.0.0 | leaf-su00-r1 | swp1s1    | 172.26.0.1      |
| hgx-su00-h00 | enp204s0f0np0 | 172.28.0.0 | leaf-su00-r2 | swp1s1    | 172.28.0.1      |
| hgx-su00-h00 | enp220s0f0np0 | 172.30.0.0 | leaf-su00-r3 | swp1s1    | 172.30.0.1      |
| hgx-su00-h01 | enp26s0f0np0  | 172.16.0.2 | leaf-su00-r0 | swp2s0    | 172.16.0.3      |
| hgx-su00-h01 | enp60s0f0np0  | 172.18.0.2 | leaf-su00-r1 | swp2s0    | 172.18.0.3      |
| hgx-su00-h01 | enp77s0f0np0  | 172.20.0.2 | leaf-su00-r2 | swp2s0    | 172.20.0.3      |
| hgx-su00-h01 | enp94s0f0np0  | 172.22.0.2 | leaf-su00-r3 | swp2s0    | 172.22.0.3      |
| hgx-su00-h01 | enp156s0f0np0 | 172.24.0.2 | leaf-su00-r0 | swp2s1    | 172.24.0.3      |
| hgx-su00-h01 | enp188s0f0np0 | 172.26.0.2 | leaf-su00-r1 | swp2s1    | 172.26.0.3      |
| hgx-su00-h01 | enp204s0f0np0 | 172.28.0.2 | leaf-su00-r2 | swp2s1    | 172.28.0.3      |
| hgx-su00-h01 | enp220s0f0np0 | 172.30.0.2 | leaf-su00-r3 | swp2s1    | 172.30.0.3      |
+--------------+---------------+------------+--------------+-----------+-----------------+
spectrum-x-rcp:V1.2.0 ~/spectrum-x-rcp # client_loop: send disconnect: Broken pipe
```

Reference Configuration Playbook (RCP)

Air Marketplace Demo







Spectrum-X Compute Network Switch Configuration using RCP

SpectrumX RCP Spectrum4 Bluefield3 East-West BGP VXLAN EVPN

Spectrum-X compute (East-West) network switch configuration using RCP

Spectrum-X - Compute Switch Configuration

NVIDIA Spectrum-X is the world's first ethernet networking platform for AI. It is powered by the tight coupling of the NVIDIA Spectrum™-4 Ethernet switch and the NVIDIA® BlueField®-3 (BF3) SuperNIC on the HGX AI supercomputer. Spectrum-X delivers the highest performance for AI, machine learning, and natural language processing, as well as diverse industry applications.

This environment demonstrates **only** the NVIDIA Cumulus Linux (CL) NOS configuration of Spectrum™-4 switches for the 3-tier rail-optimized compute East-West (E-W) network.

As the purpose of this demo is to show the switch configuration of the E-W network, only the SN5600 Spectrum-4 switches are running Cumulus VX. The HGX nodes are simulated using standard Ubuntu 22.04.2 servers with regular NICs (not Bluefield SuperNICs), which are configured with IP addresses and routing to match the environment and to be able to reach each other for compute network connectivity testing. No actual GPUs or HGX software runs on them.

East-West Network Overview

The compute E-W network fabric was designed for multi-job and multi-tenancy AI cloud. The advanced Spectrum-X load-balancing, congestion-control, QoS, and virtualization technologies enable multiple tenants to run multiple jobs on the same infrastructure while getting the highest performance and being completely secured, isolated, and independent of each other.

The fabric consists of two types of networks, the underlay physical network and one or more overlays (virtualized networks) that allow multi-tenancy and provide full line-rate services to all tenants. The underlay network is designed to be pure routed (layer 3) with BGP as the routing protocol, while the overlay network may be deployed over the fabric switches using VXLAN data plane and bridged (L2)/routed (L3) EVPN control plane, or on the Bluefield SuperNICs of the hosts using virtualization technologies such as Open vSwitch (OVN) or Host Based Networking (HBN).

Regardless of whether network virtualization is deployed or not, the initial E-W network state is a single tenant over a pure layer 3 fabric, but can easily be suited for multiple tenants by logically dividing the

[Try RCP in Air](https://air.nvidia.com/)

https://air.nvidia.com/

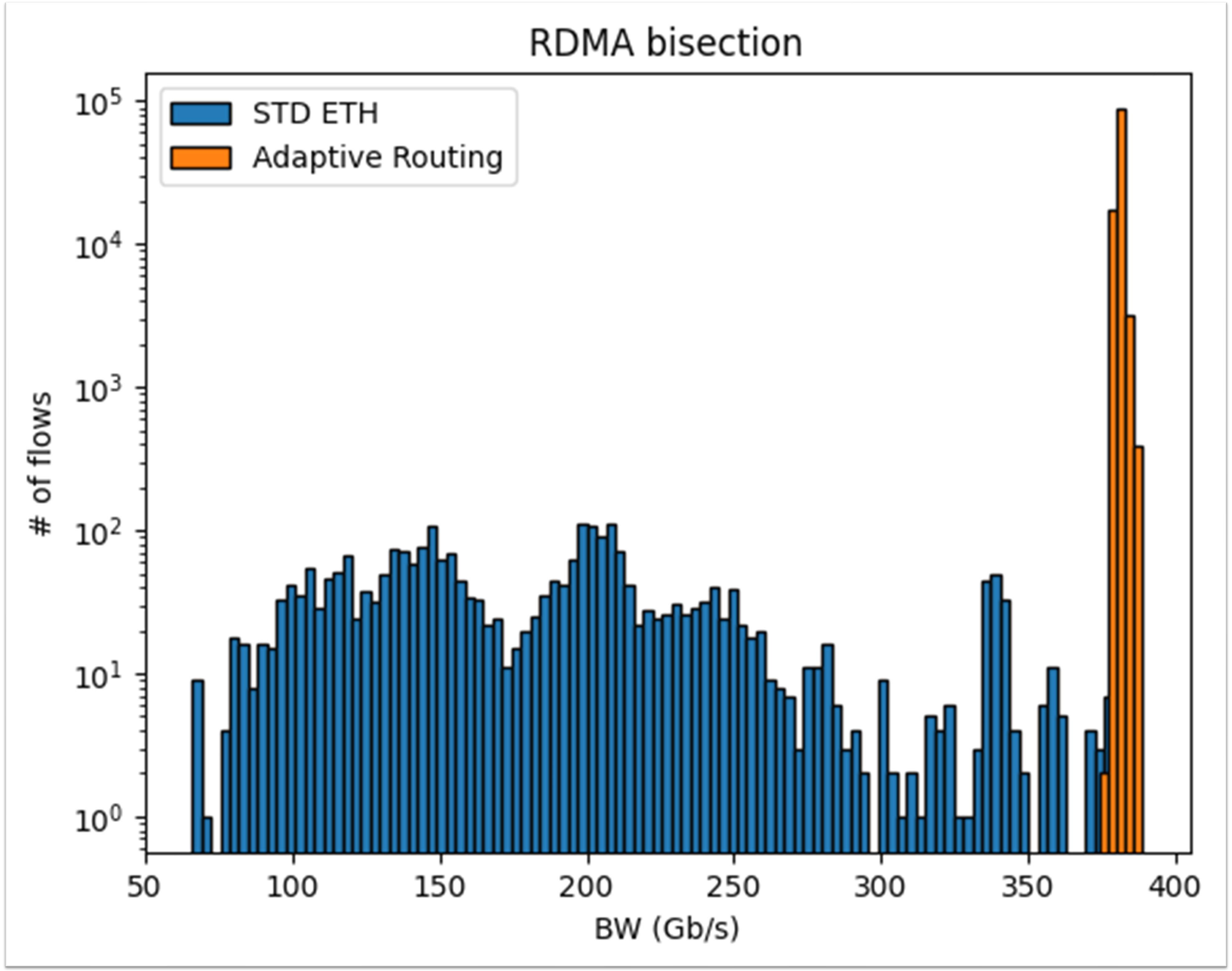
Networking Features and Debug Flows

Overview

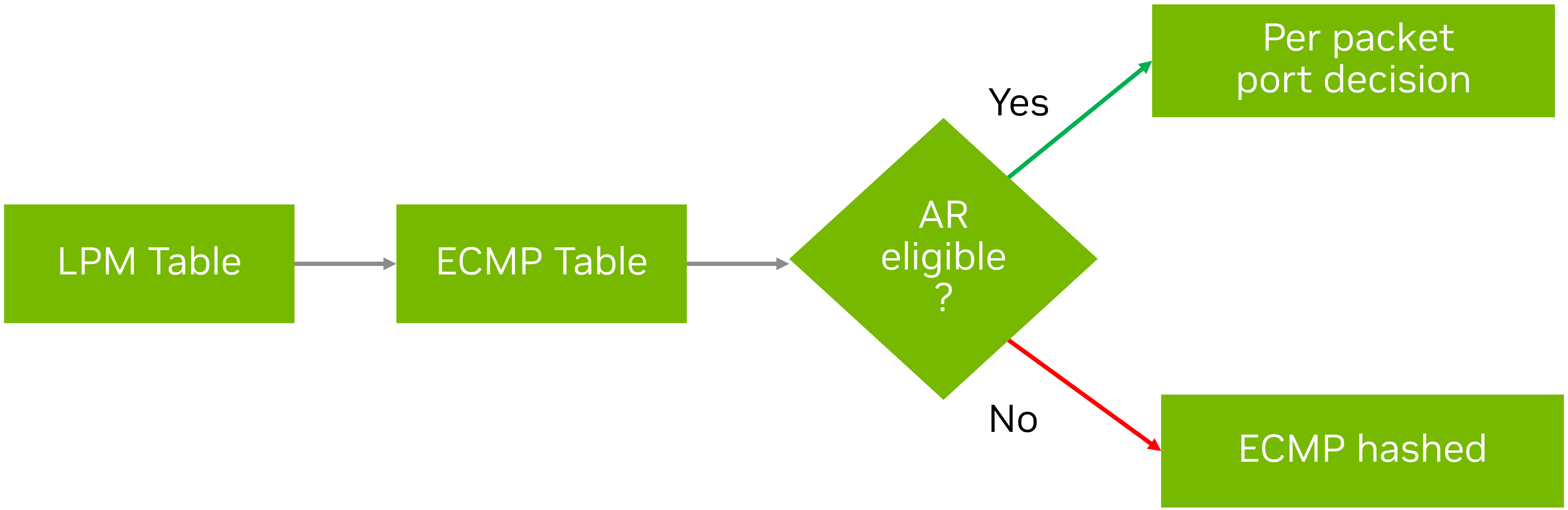
- Focus on SPC-X specific issues
- Normal ETH fabric debugging still applies
  - BER
  - link failure
  - CPU / memory / temperature
- Major features:
  - Adaptive Routing
  - Congestion Control

Feature: Adaptive Routing

- Alternative to hash-based ECMP routing
- Spreads packets evenly on all ports
- Attempts to avoid congested paths

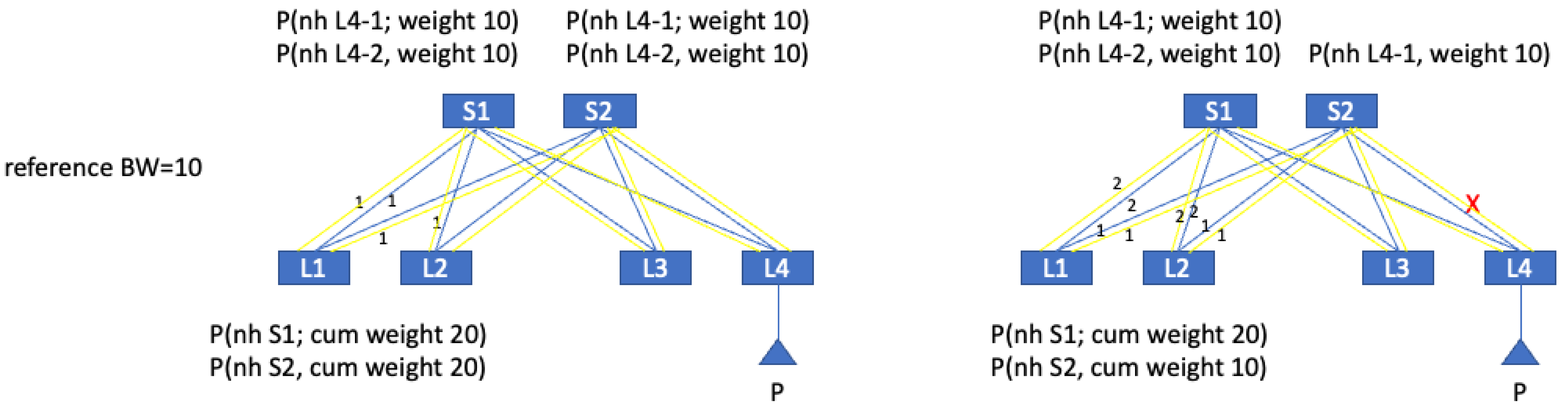


Switch Adaptive Routing Mechanism



Spectrum-X: BGP W-ECMP

- WCMP is based on IETF [draft-ietf-bess-ebgp-dmz]
- As weights are being updated in source leaf – data plane is updated accordingly



Feature: Congestion Control

- Sender (adapter) throttling mechanism
- Kicks in when latency increases
- Aims to prevent fabric congestion and pauses
- DOCA based, tailored for AI workloads
- Programmable and tunable
- Based on **RTT probes + ECN**

ZTR RTTCC

NVIDIA RTT-base congestion control algorithm

Congestion indications

- Round trip time (RTT)+ ECN/CNP

Rate update scheme

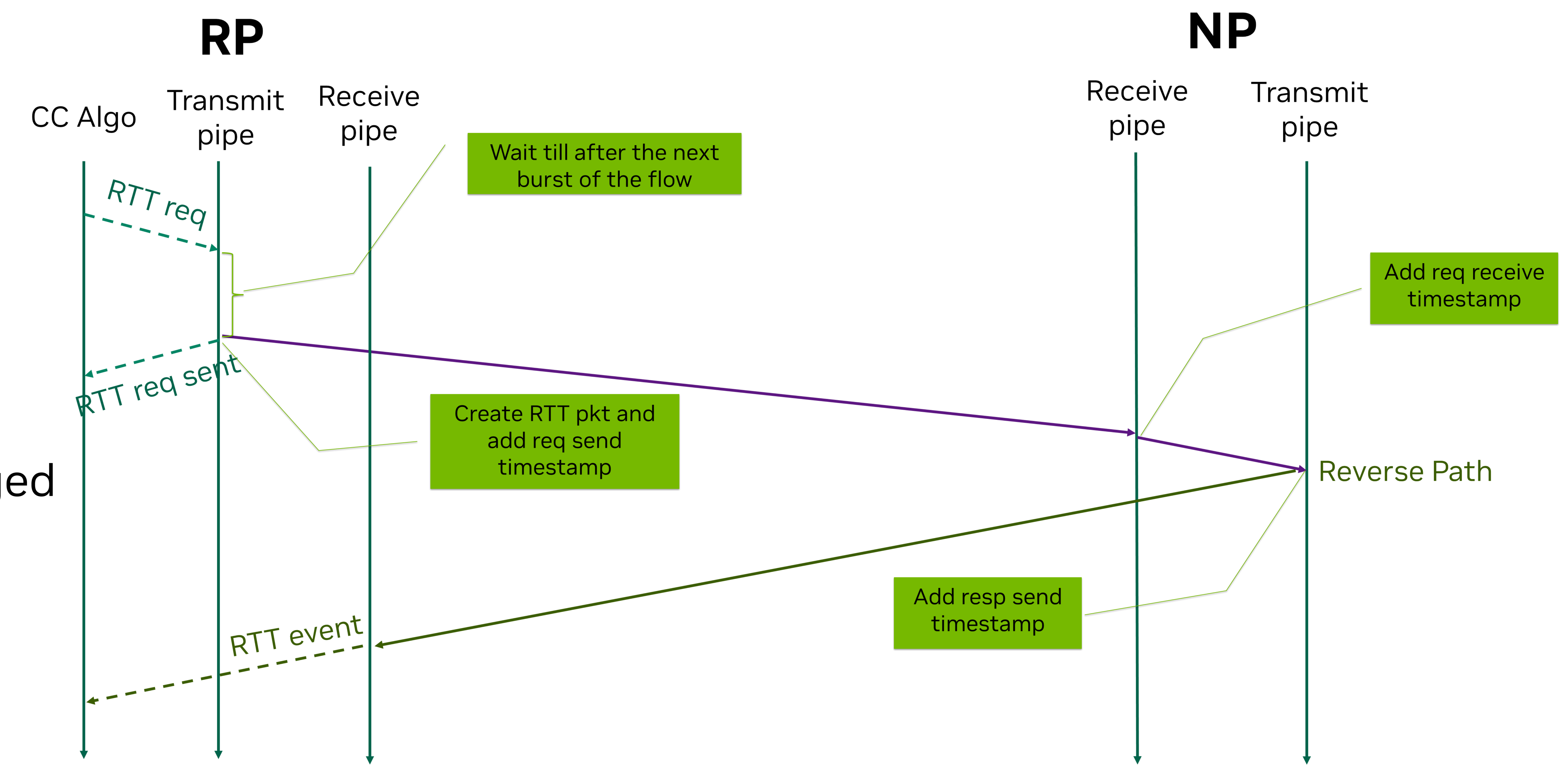
- Evaluate congestion state by comparing RTT to $\frac{1}{\sqrt{rate}}$
- Additive increase
- Multiplicative decrease

Algorithmic windowing

- Mimic window behavior by adjusting the rate when delay changed

More features

- Fast reaction on first congestion



Config Validation

- Bad config => **99% of performance issues**
- SPC-X is heavily customized
  - Lossless traffic
  - Adaptive routing enabled
  - Custom shared buffer configuration
  - Custom congestion control algorithm
- DPU + Switch

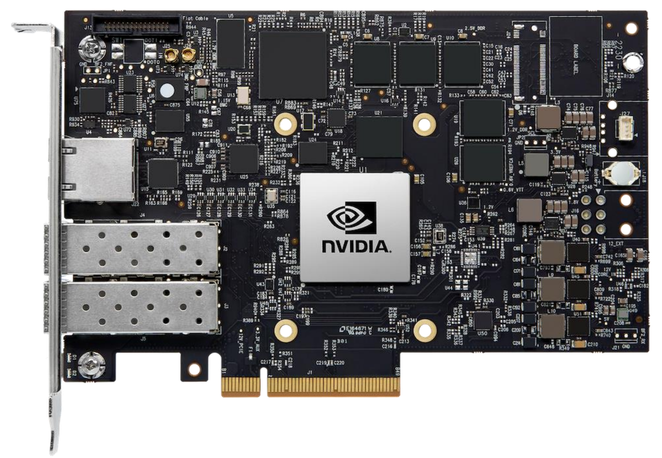
Config Validation: Examples

Switch

MTU: 4K
Link speed: 400G
RoCE mode enabled + lossless

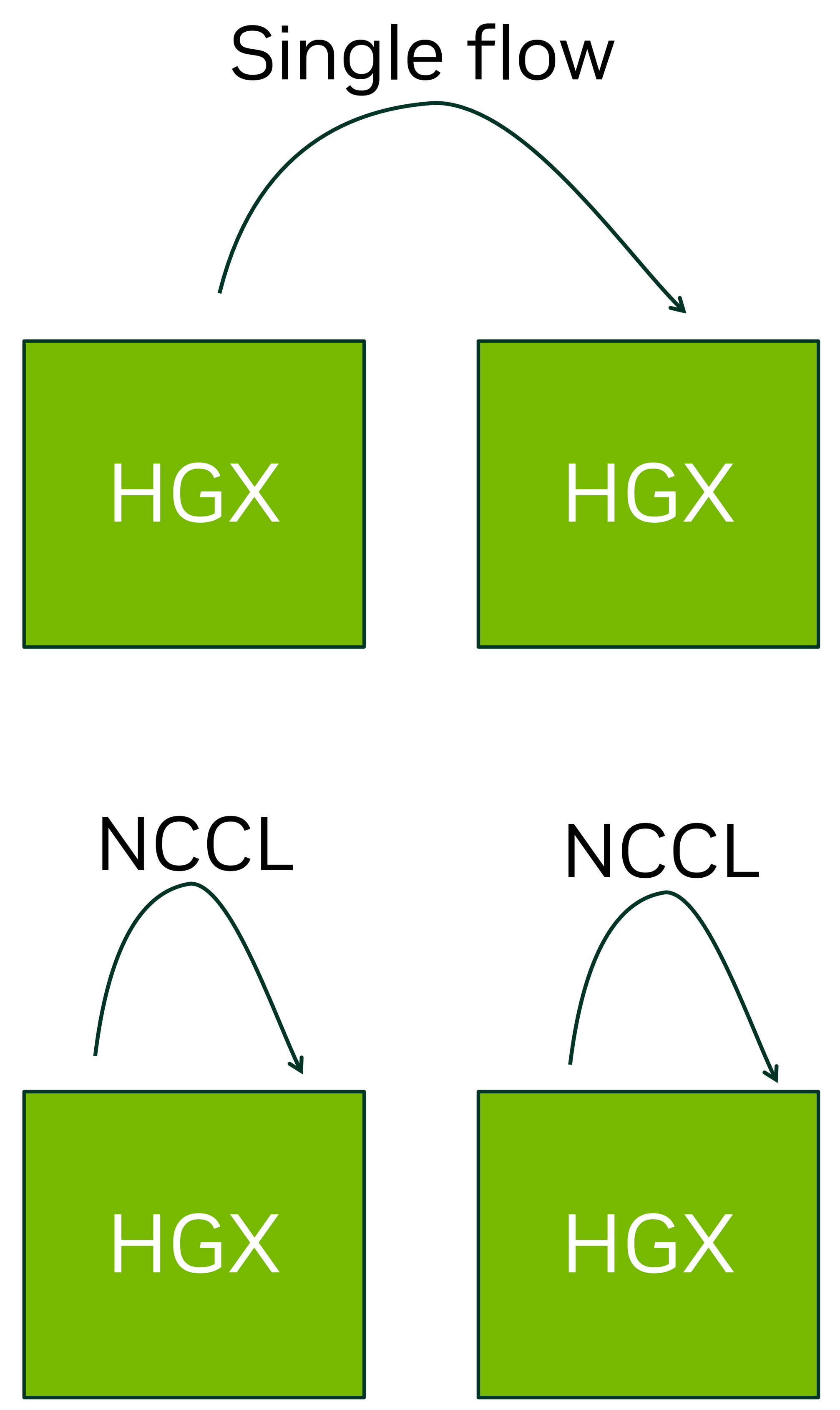
SuperNIC

DOCA PCC
Rate limiter
Traffic Class: 96



Sanity Tests

- Single ib_write_bw
  - RoCE
  - Connectivity
- Single-node NCCL on all nodes, no NVLink
  - Host configuration
  - GPU availability
  - Adaptive routing
  - NVLink disabled

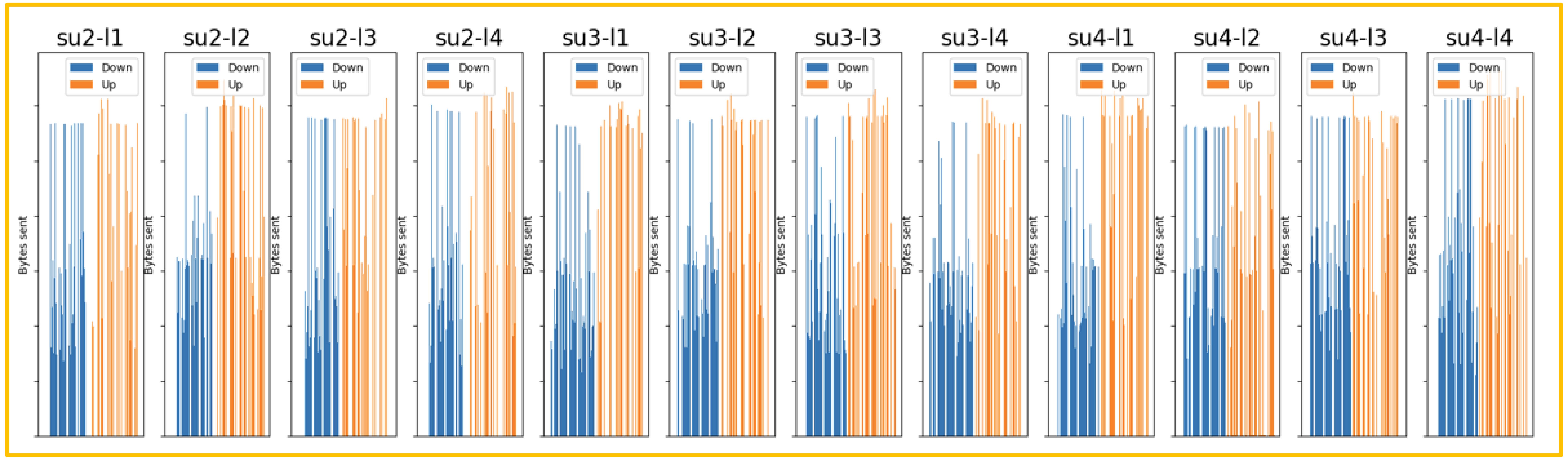


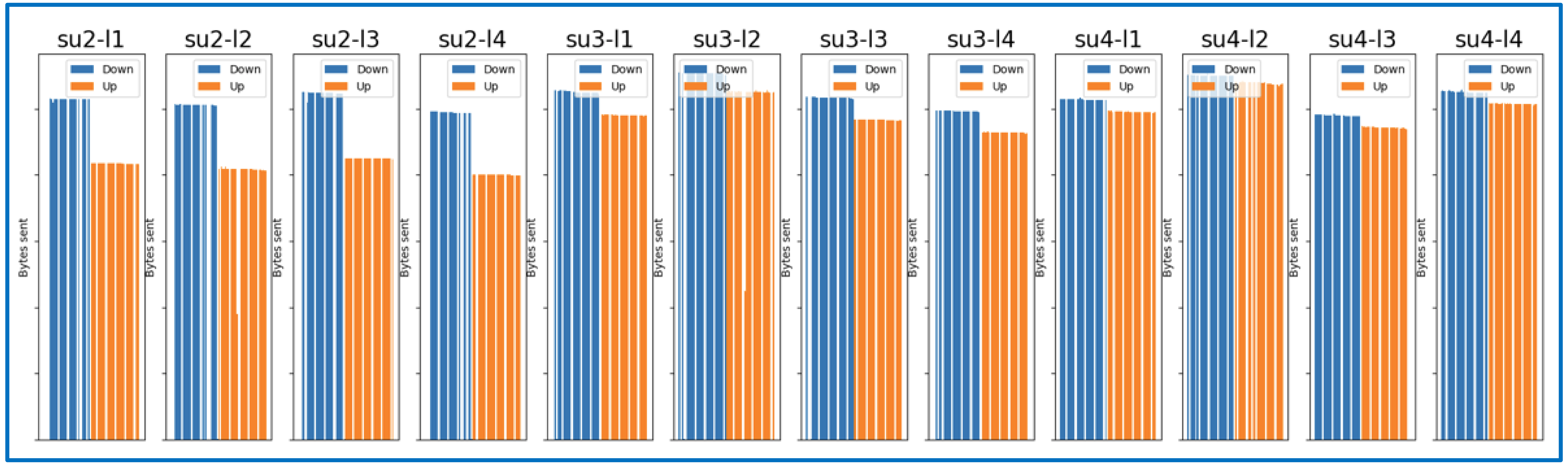
Switch Metrics

- Taken while traffic is running, non-intrusive
- Indicate Adaptive Routing / Congestion Control issues
- Queue lengths, TX bytes, PFC
- NetQ-based + High frequency telemetry

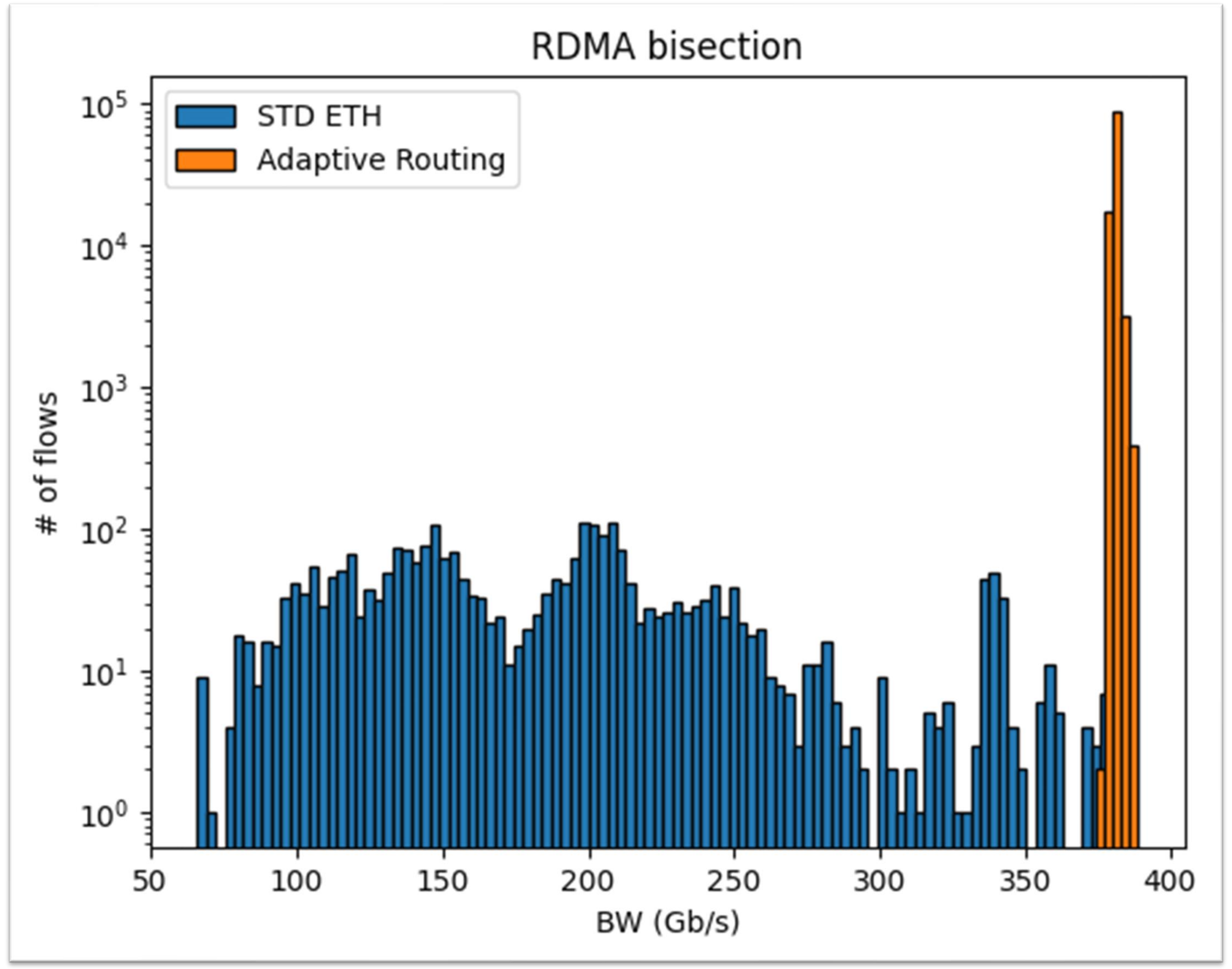
Case study: AR traffic

Infra view: TX bytes





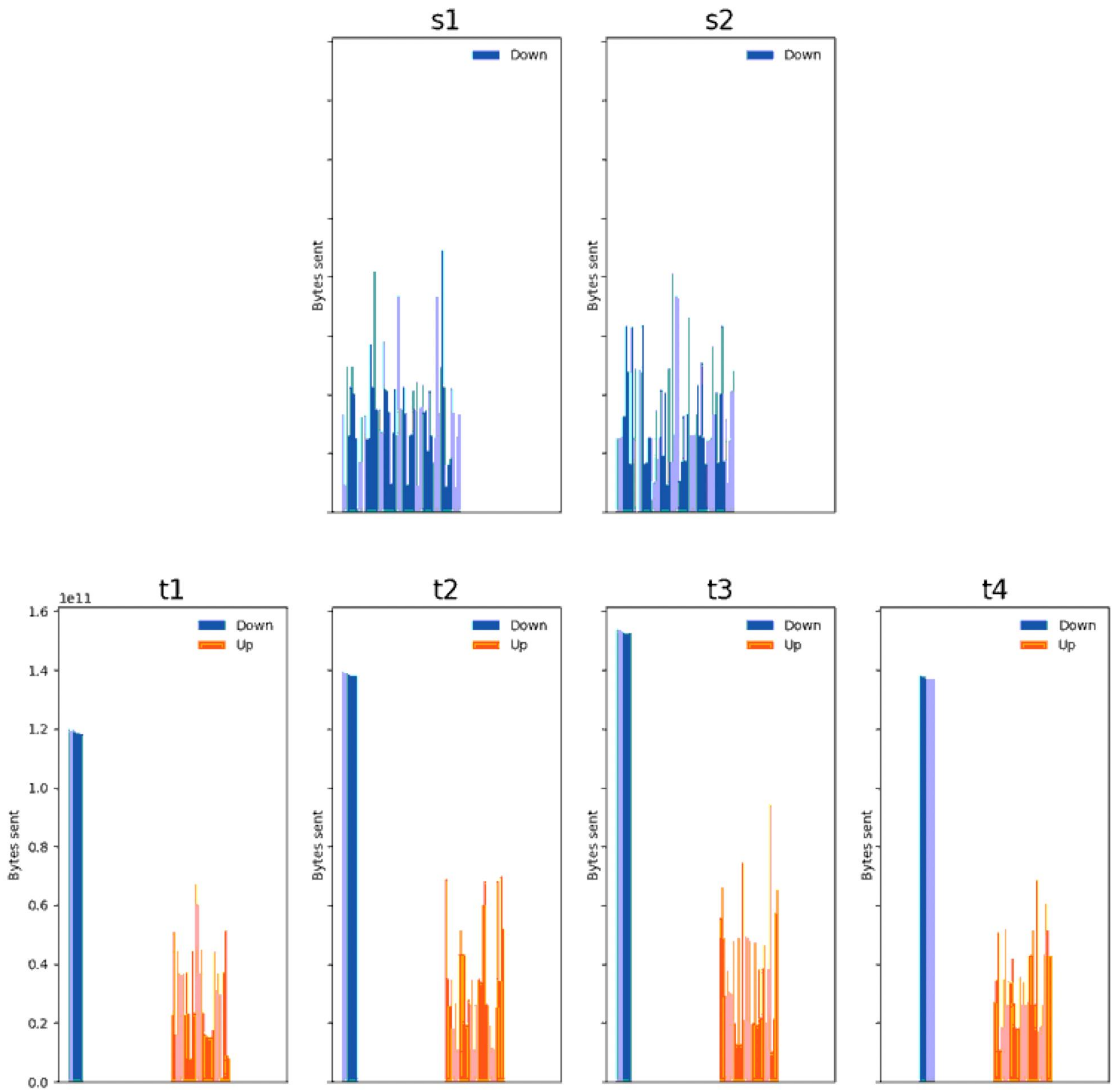
What the app sees



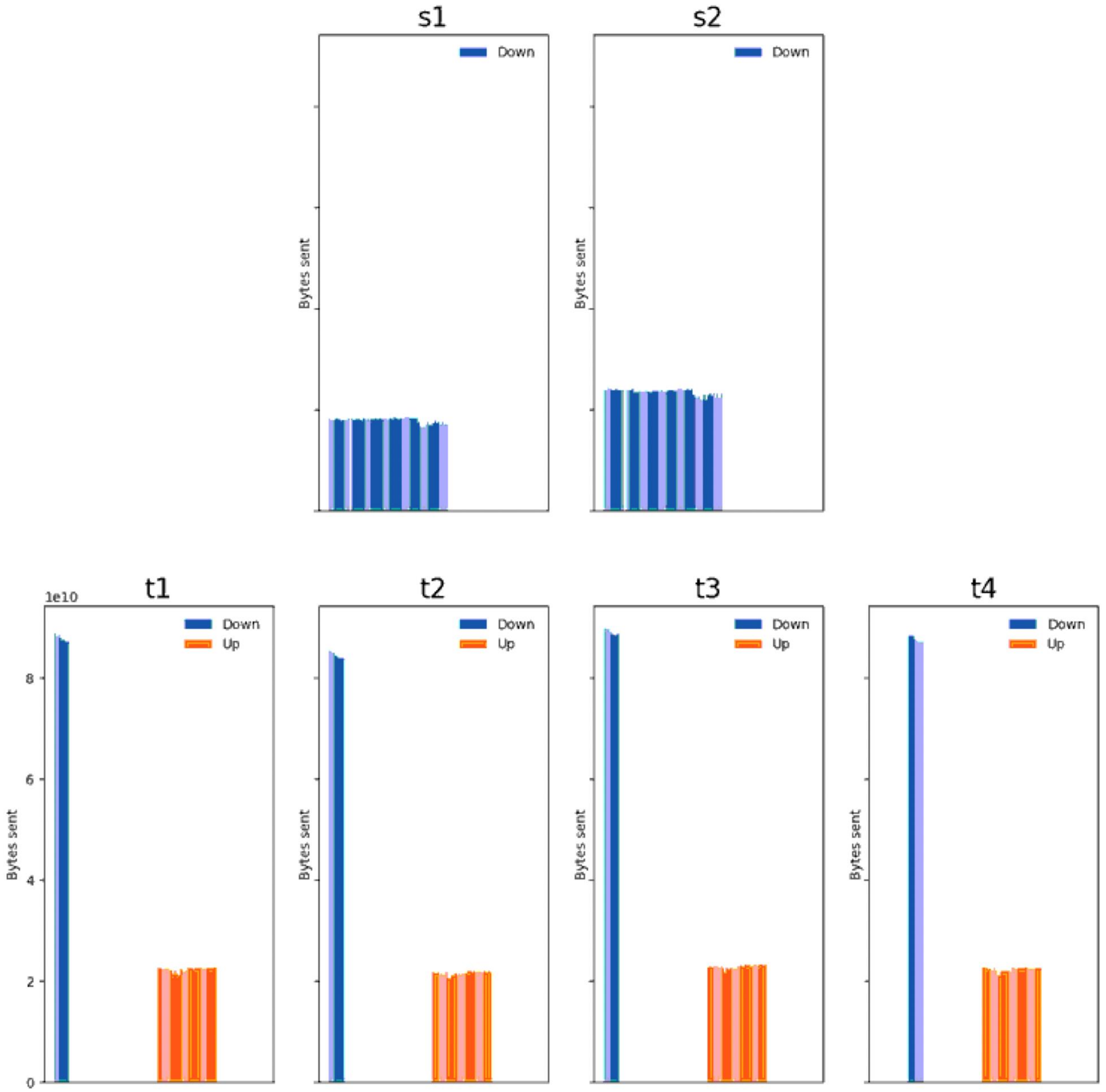
Case study: NCCL

Adaptive Routing vs Many QPs

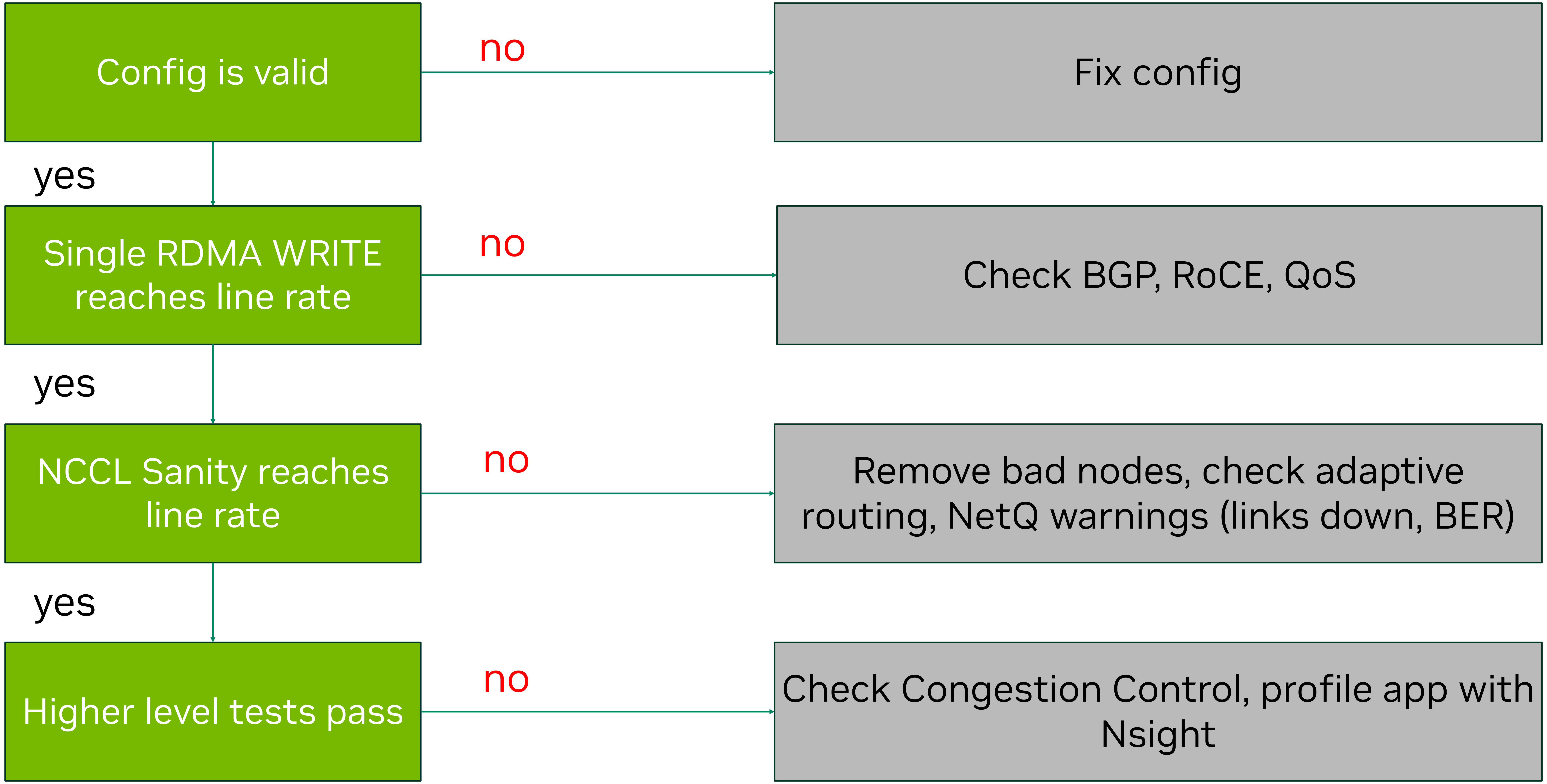
AR off, 8 QPs



AR on, 1 QP



Debug Flow Chart



NetQ

What will it do in the future?

- Positioning: Frontend for Spectrum-X Infrastructure Operations
- NetQ Fabric Builder* – deploys and validates RA configurations for Spectrum-X
- Aggregates Network and Application Infrastructure Metrics
  - At a glance dashboard for Spectrum-X Infrastructure
  - Metrics useful for validating Spectrum-X performance (AR, CC, GPU performance w/ NSight)
  - Deep performance debuggability through unparalleled **High Frequency Telemetry** (HFT)

NetQ releases	Capabilities
NetQ 4.9 (Feb 2024) - RA 1.0 (In Execution)	• E2E (Switch + DPU) RoCE Monitoring • AR Monitoring (Switch) • Queue histograms • Switch life cycle management upgrades
NetQ 4.10 (April 2024) – RA 1.1 (In Execution)	• E2E RoCE & Adaptive Routing Validation (Switch + DPU) • Link BER Collection and Visualization (Switch) • Topology Validation (Rail Optimized) • High frequency telemetry collection
NetQ 4.11 (July 2024) – RA 1.2 (In Planning)	• High Frequency Telemetry Enhancements • Buffer, Bandwidth and Latency Histograms • Visualization enhancements
NetQ 4.12 (Oct 2024) – RA 2.0 (In Planning)	• Scalability enhancements • NCCL aware telemetry • Switch & DPU provisioning*

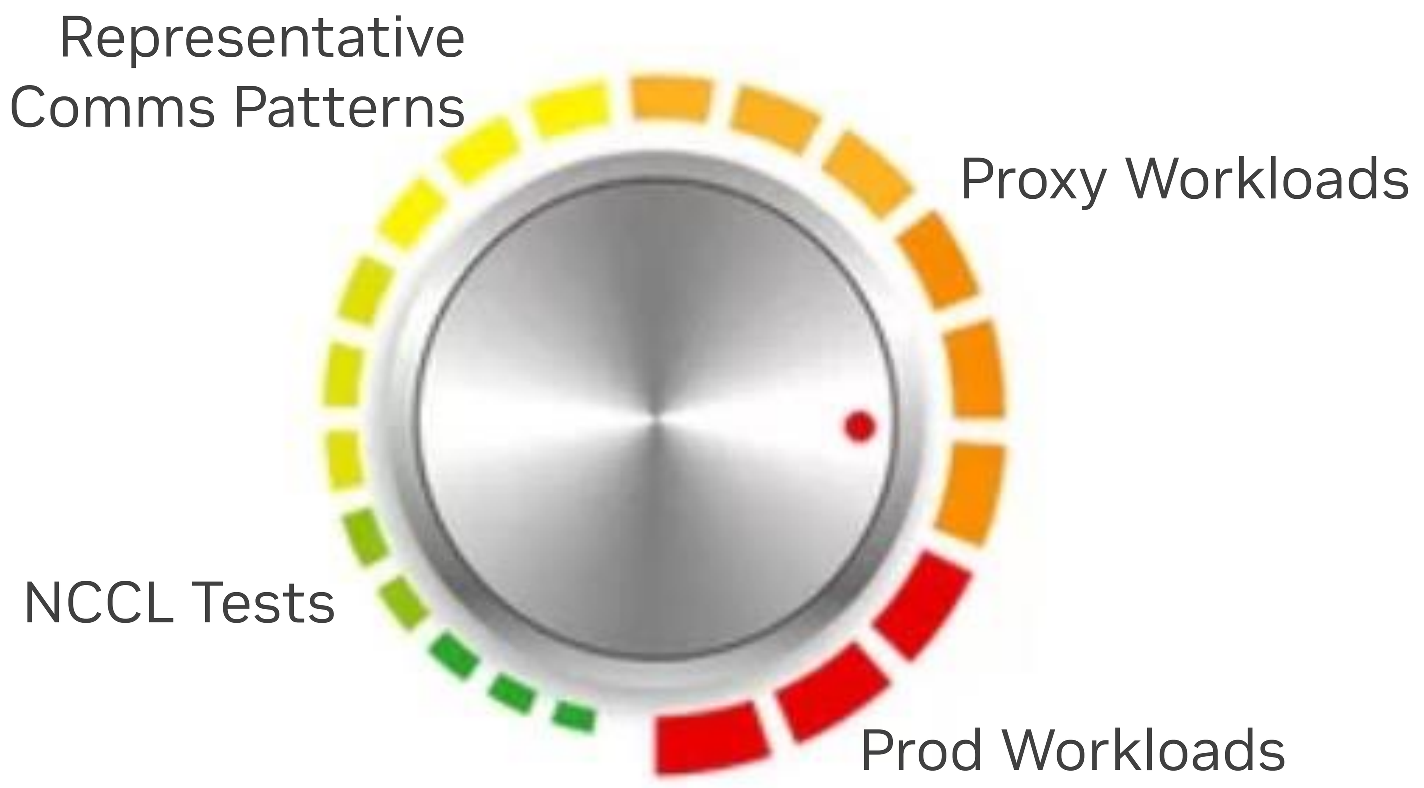
CloudAI Benchmark Framework

CloudAI Benchmark Framework

Introduction

- **Highlights**
  - Declarative benchmark and system configuration specification
  - Reproducible multi-job execution behavior - induce congestion in network switches
  - Schedulers supported: **Slurm** (w/ Docker/Enroot) or **baremetal**
  - Automated report generation
  - Using real GPU hardware to generate realistic NCCL/AI traffic on the network
  - GPU model agnostic
  - Agnostic from AI fabric/Cluster network technology
    - Ethernet (Spectrum-X or Traditional)
    - InfiniBand
    - Rail-optimized or non-railoptimized topologies
    - 2-Tier, 3-Tier etc

  - NCCL bisection, allreduce, allgather, alltoall, send/receive, reduce_scatter tests
  - Grok, GPT and Llama tests



Workload Scenarios

Production Scenarios

https://github.com/nvidia/cloudai

CloudAI: Execution Flow

Latency Metrics by Communication Type

Comm Type	Total	Max	Min	Average	p50
wait	3415065.15	1689067.54	1.35	309.78	2.17
reduce_scatter_base	1932738.86	8663.7	28.83	585.68	31.37
all_reduce	1235081.67	5383.32	27.71	1009.05	39.74
barrier	4473.34	2161.08	98.44	1118.34	1106.91

Spectrum-X PoC

Why do we need a PoC document?

Spectrum-X

- NVIDIA's secret sauce for AI datacenter fabric over Ethernet
- Holistic ethernet based end-to-end solution
- Focuses on AI use cases

PoC Document Purpose

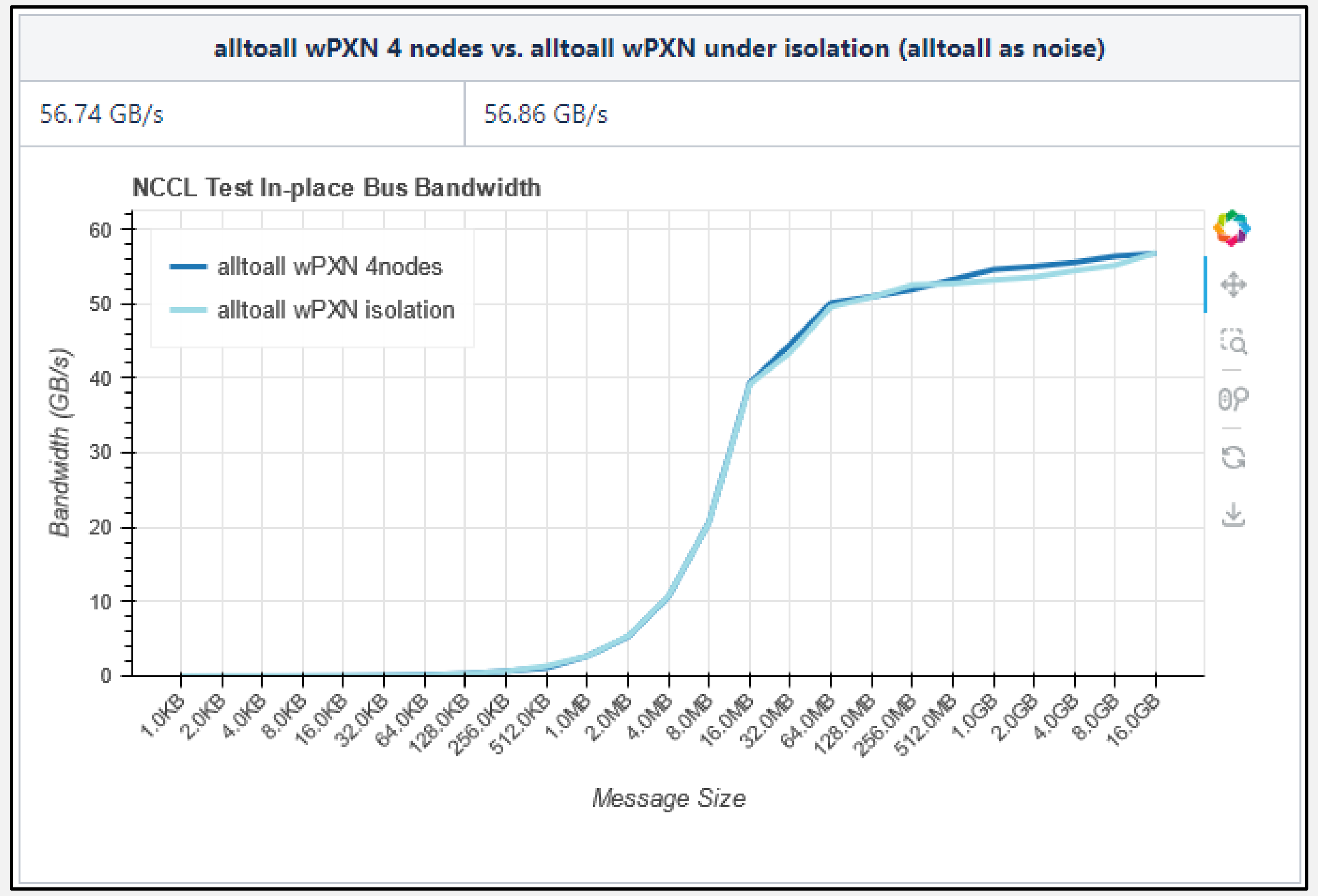
- Clarify the claims in the market about how an AI Fabric should perform
- Demonstrate the capabilities and performance of Spectrum-X
- Give valued NVIDIA customers a cook-book how to run tests
- Leverage NVIDIA in-house developed testing solution

Spectrum-X PoC shows actual performance differentiation with >= 16 GPU nodes in 2-Tier spine/leaf fabric topology, non-rail optimized fabric design

Spectrum-X Technology Performance

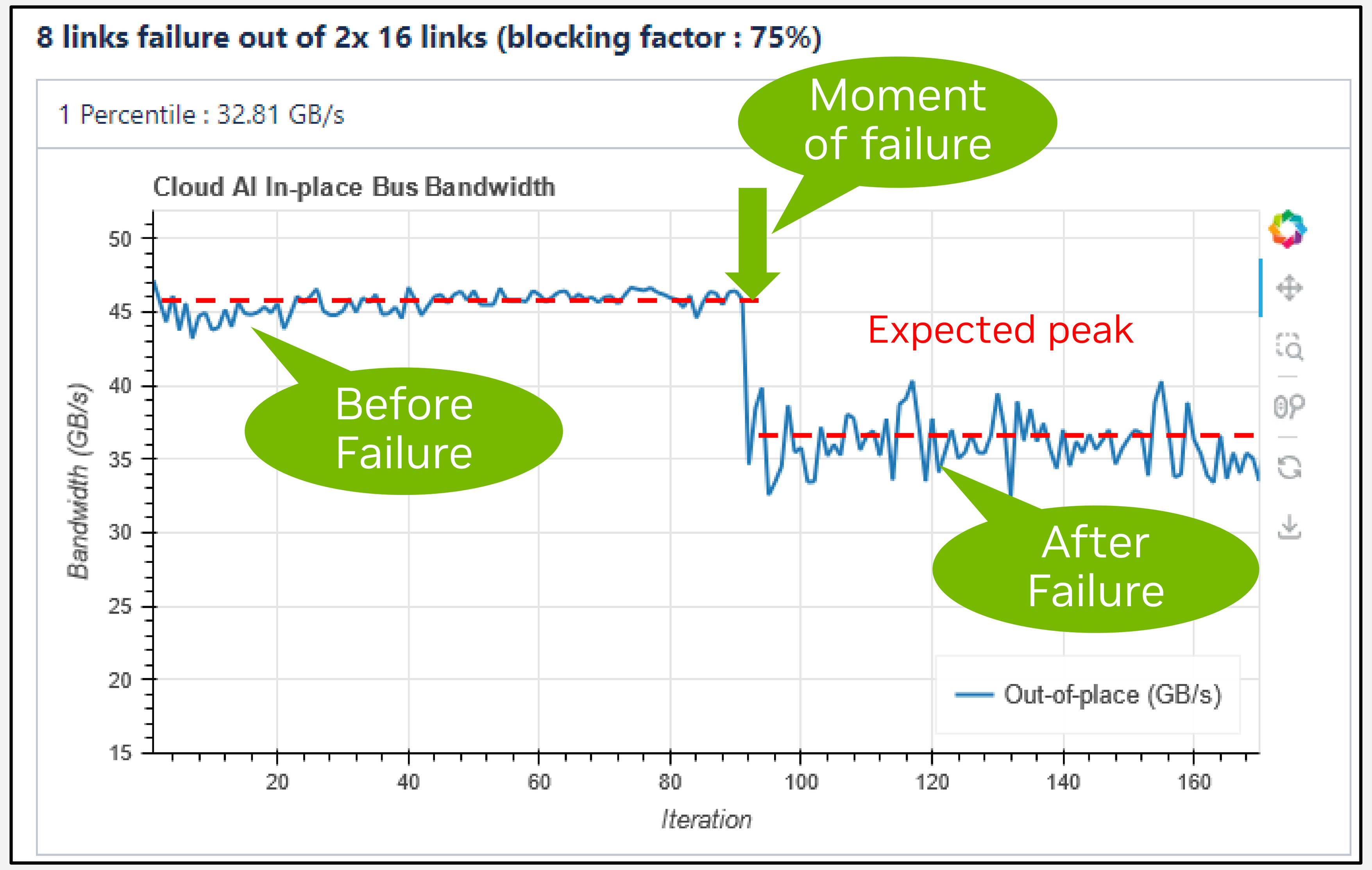
Spectrum-X Network Performance:

Optimal building block performance of RDMA and NCCL



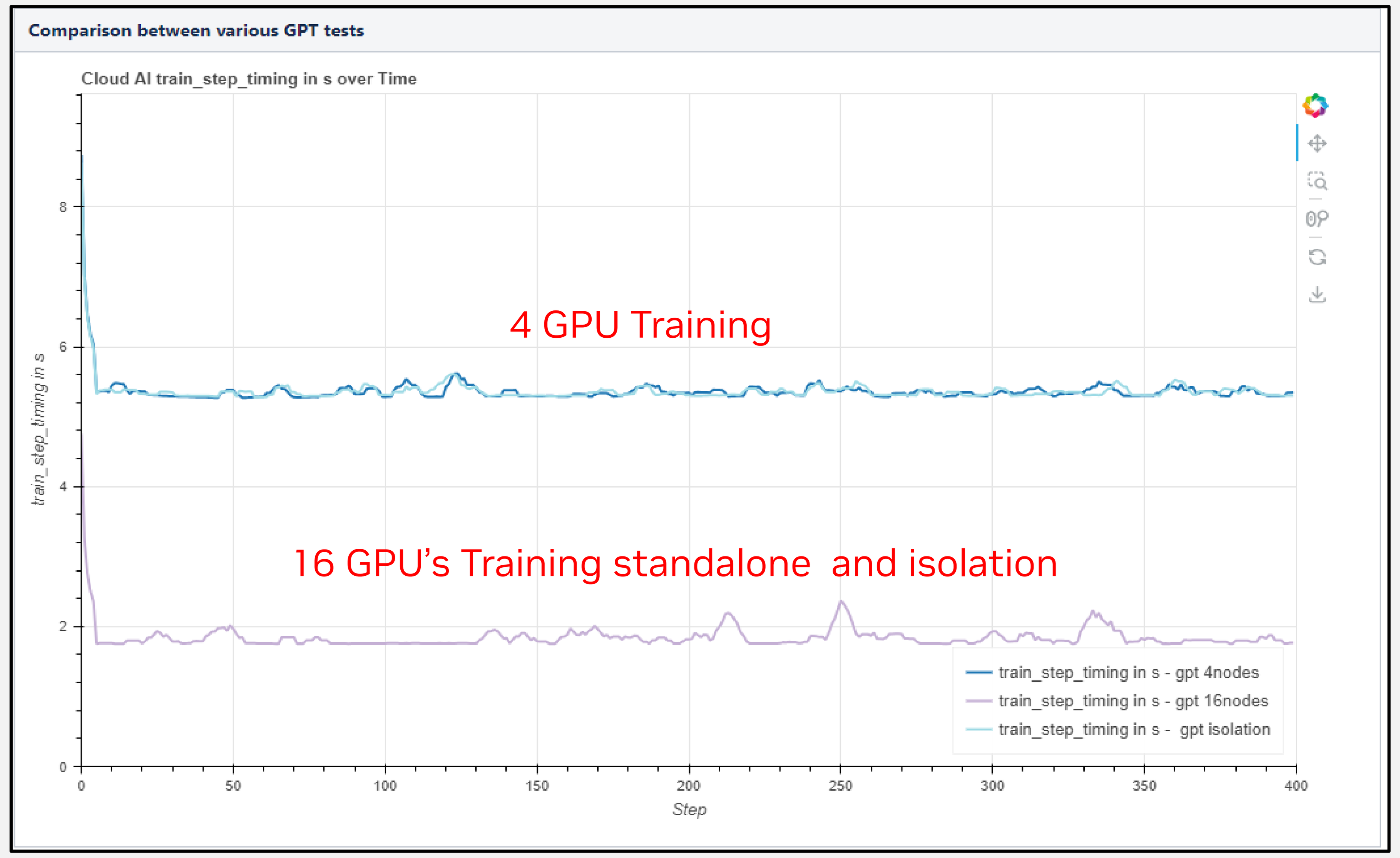
Spectrum-X Solution Resiliency:

Graceful degradation under failures utilizing global adaptive routing technologies



LLM Training Performance:

Efficient scalability from 4 to 16 node scale
Multi-job isolation



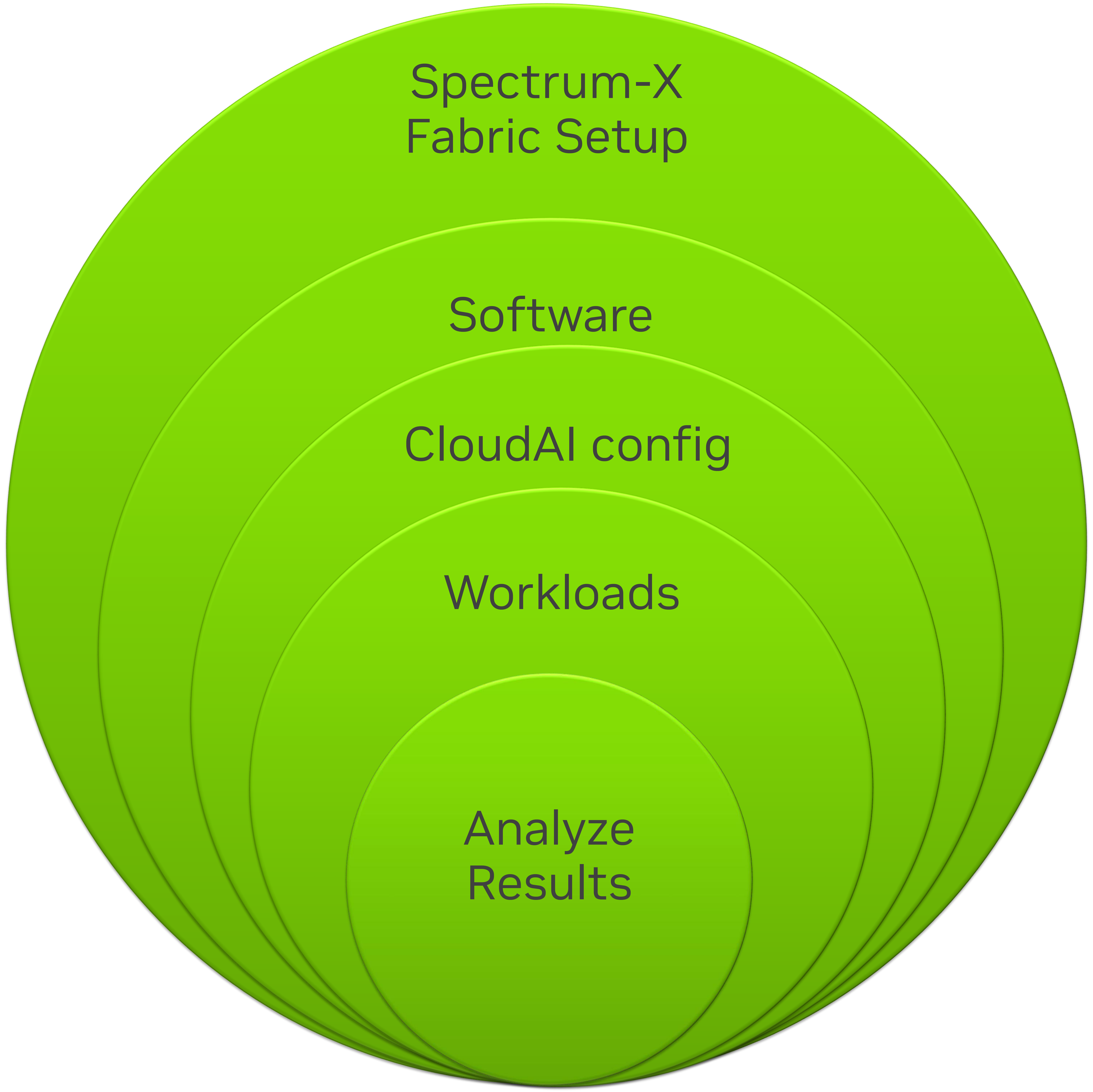
Spectrum-X PoC Components

CloudAI Test Harness

Non-Rail Optimized 2-Tier Topology

* Spectrum-X PoC document uses non-rail optimized topology to leverage benefits of Spectrum-X Adaptive Routing

POC Test Procedure



Spectrum-X E-W Fabric Setup
Building Spectrum-X PoC Fabric and configuration using **RCP**

Software
Prepare required firmware and software installation for tests

CloudAI Configuration
Test parameters and Test Schema configuration

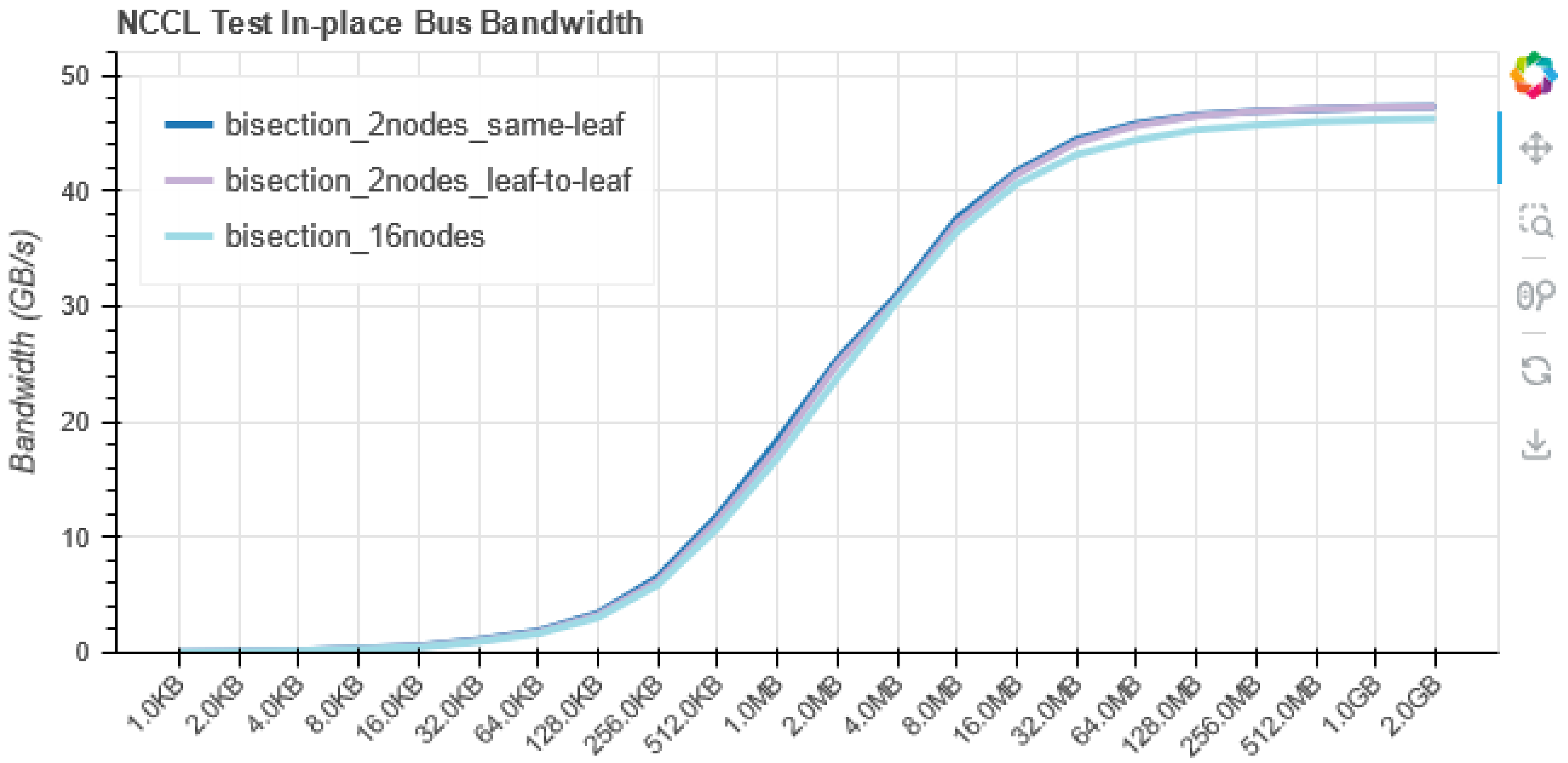
Workload and Component Tests
NEMO Megatron (LLAMA, GPT), NCCL collective operations

Analyze Results
Review results and tweak test config if necessary

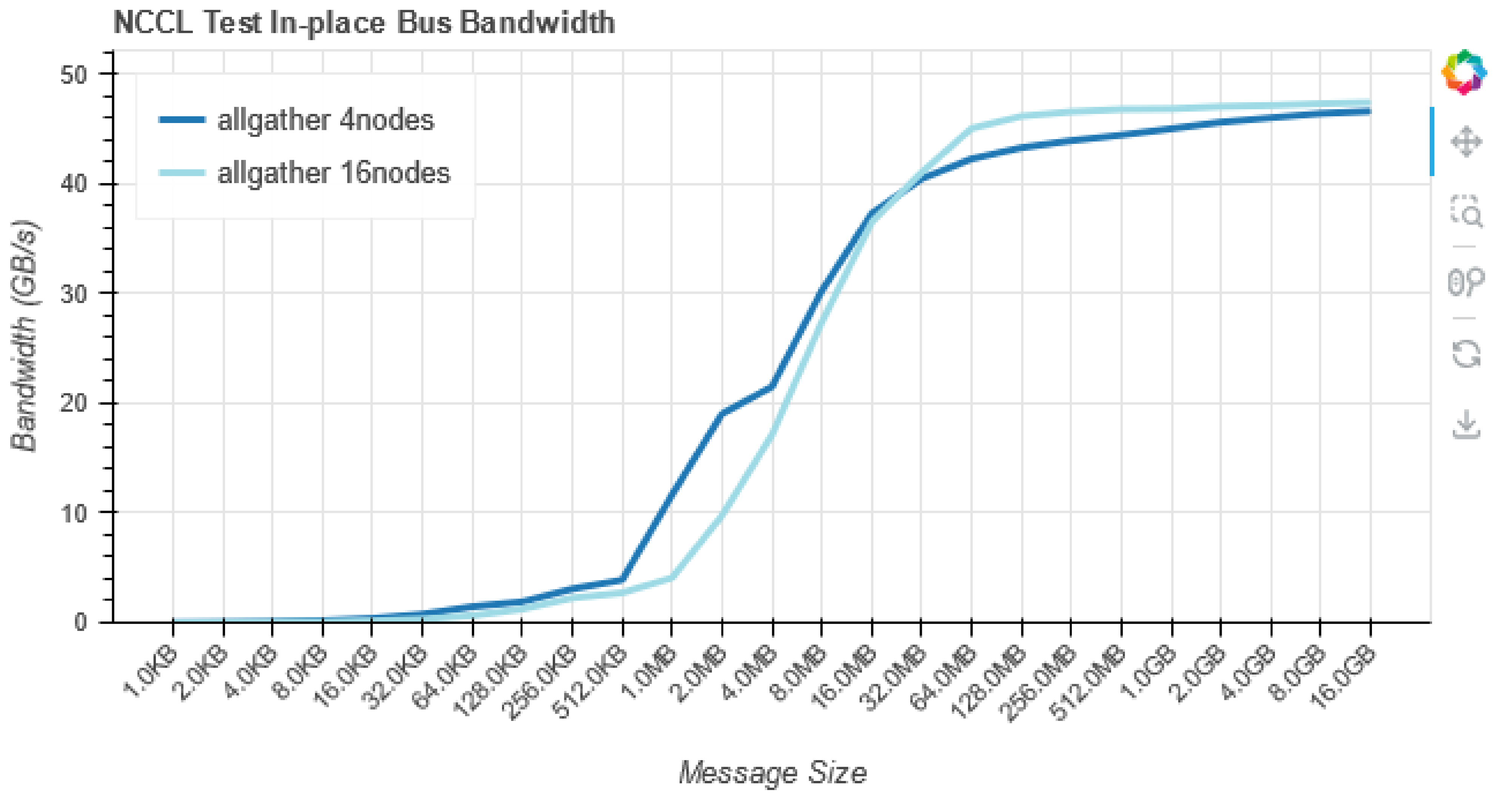
Test/Performance reports – NCCL tests

Results comparison

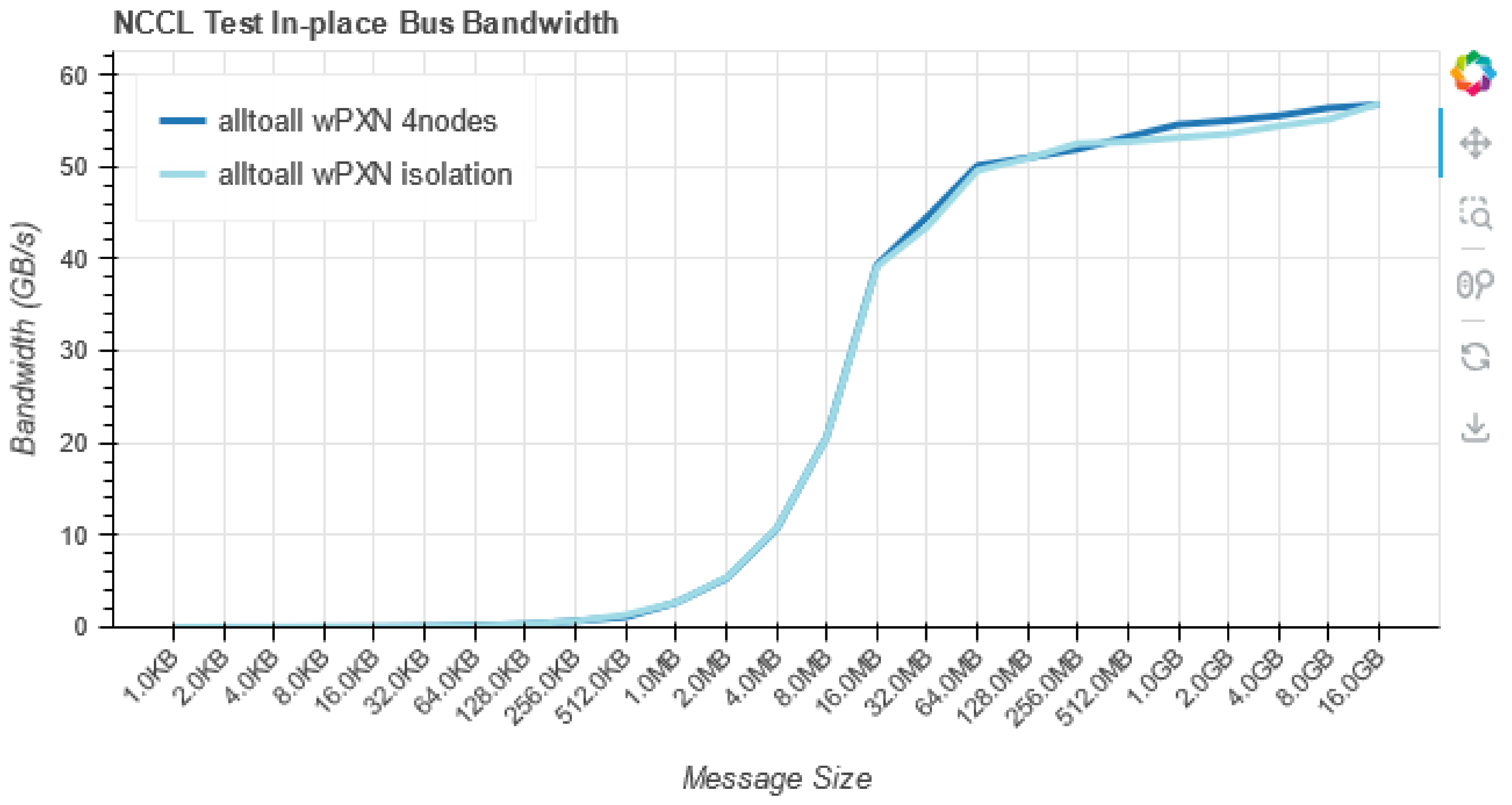
- Test results are in comparison form
- Bisection, allreduce, allgather, alltoall, send/receive, reduce-scatter NCCL collectives
- Comparing same type of tests for different:
  - Scale
  - Topology
  - Traffic conditions (single tenant, multi-tenant (isolation))



send/receive 4 nodes vs. 16 nodes (max scale)	
4nodes : 49.48 GB/s	16nodes : 49.47 GB/s



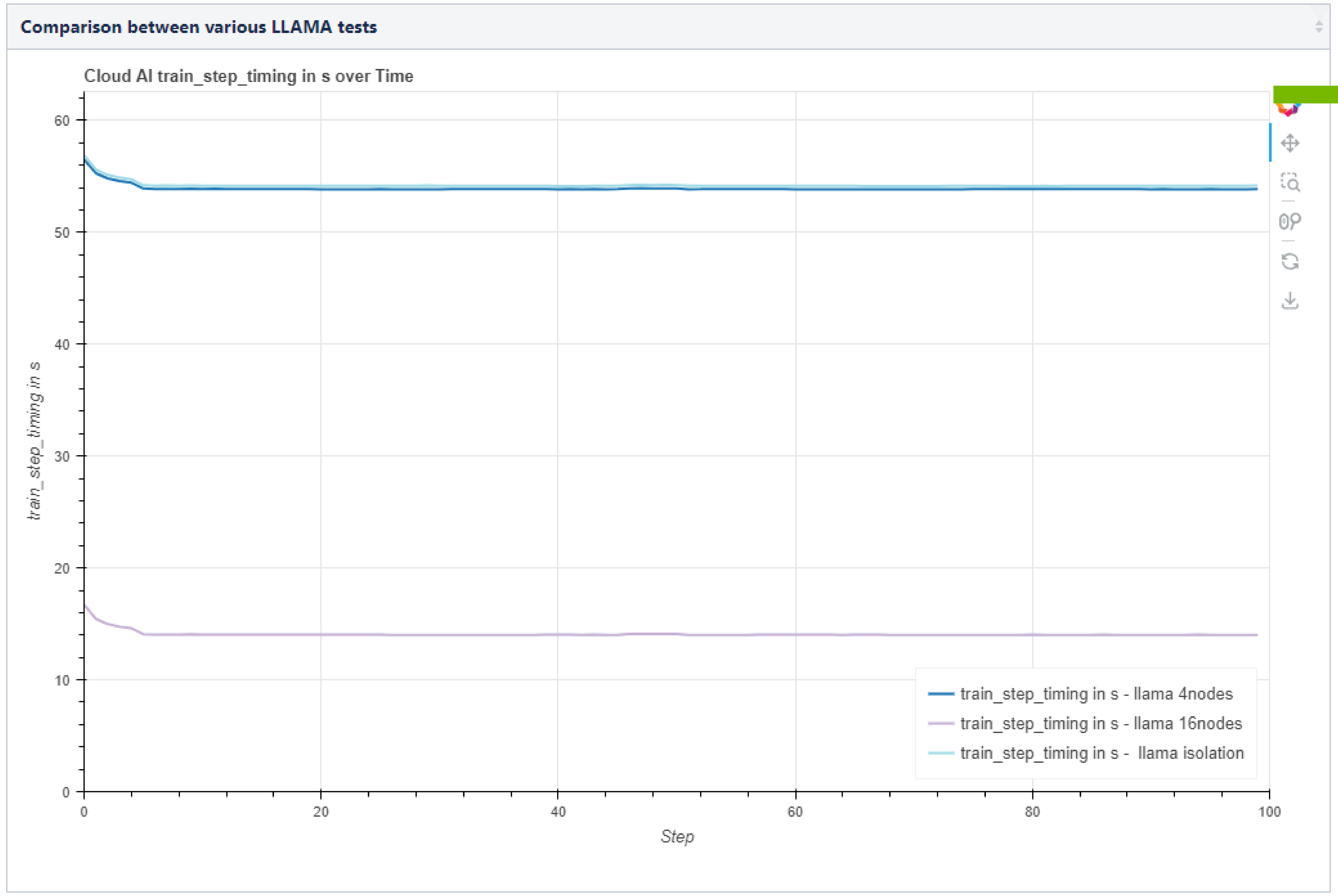
alltoall wPXN 4 nodes vs. alltoall wPXN under isolation (alltoall as noise)	
56.74 GB/s	56.86 GB/s



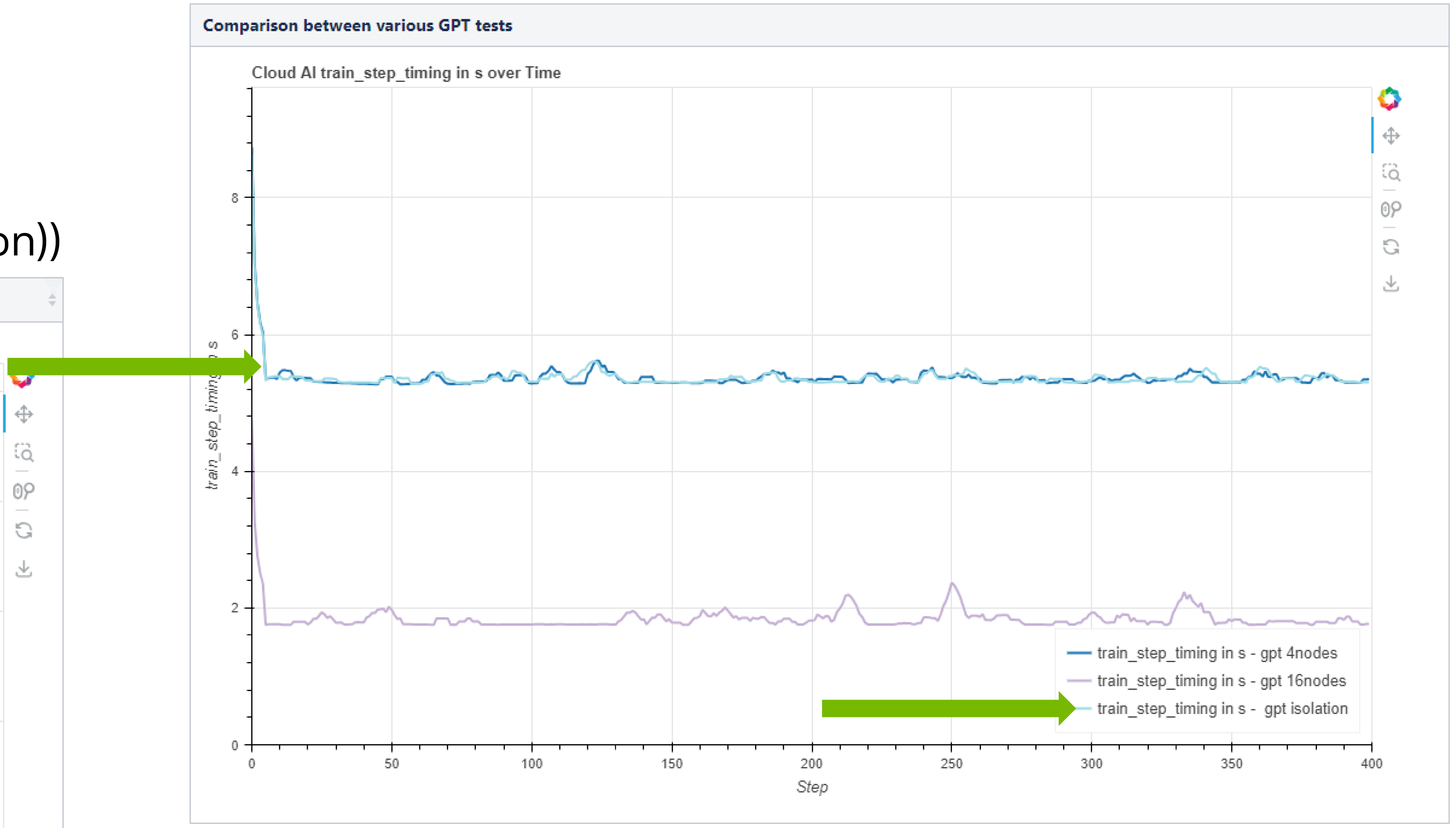
Test/Performance reports – LLM tests

Results comparison

- Test results are in comparison form
- Comparing same type of tests for different;
  - Scale
  - Traffic conditions (single tenant, multi-tenant (isolation))



LLAMA test



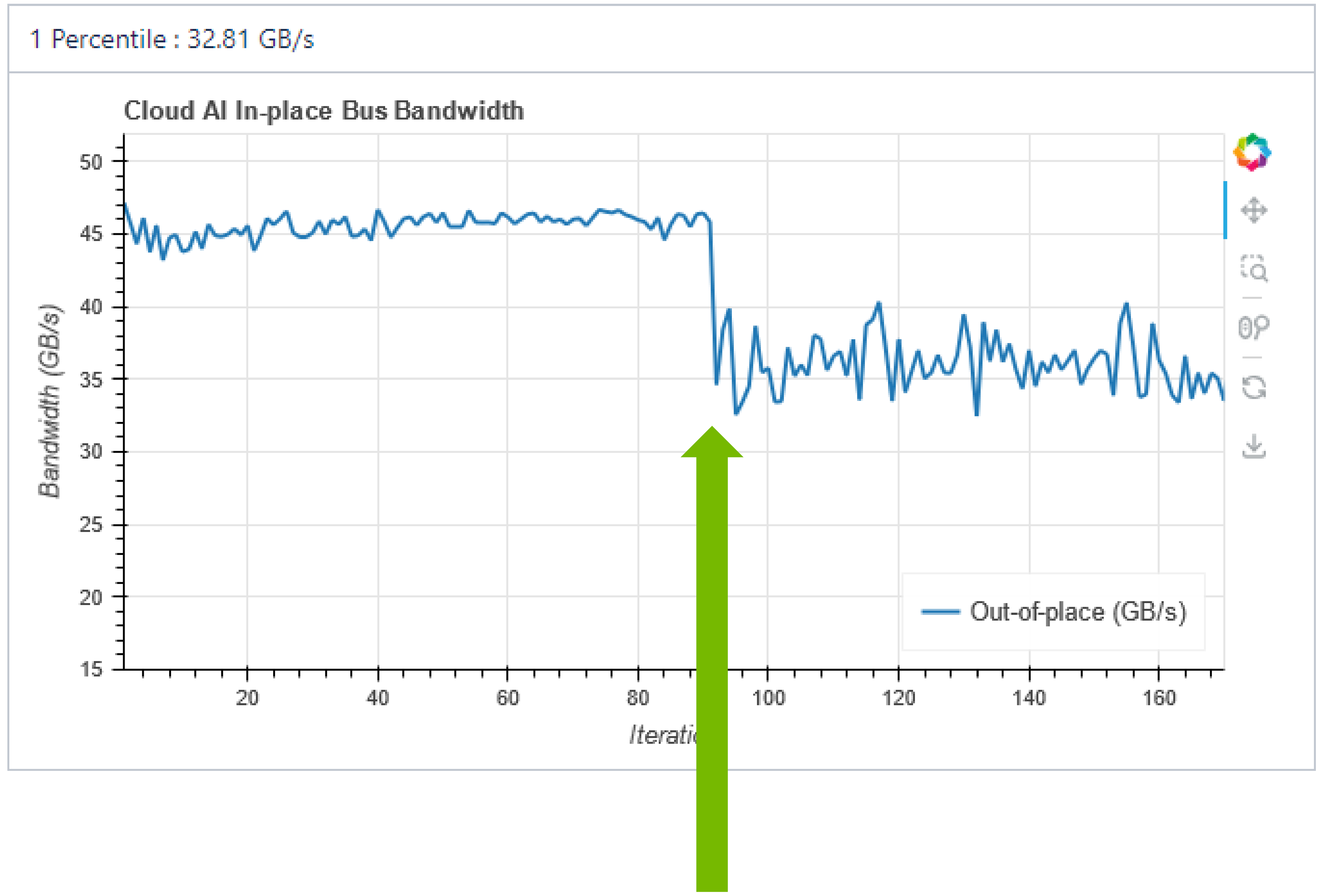
GPT test

Evaluating the results – Resiliency tests

Results comparison

- Tests are based on NCCL all-to-all collective to achieve max utilization in the fabric
- Running same test with incremental link-down events (1/4/8/12/16 links)

8 links failure out of 2x 16 links (blocking factor : 75%)



Accessing PoC Document

NVIDIA. Product Information Delivery

READ ONLY CONTENT

< Browse Library

ID: 1123419

Content Class:	Platform
Content Type:	Documentation
Product Segment:	Enterprise
Product Line:	Networking
Product Family:	BlueField-3, Spectrum-4
HW Platform/ SW Release:	ETH Switch, DPU
Product SKU:	
Brand Name:	

*Title:
Spectrum-X Performance Test Plan Spectrum-4 and BlueField-3
71 characters remaining (130 maximum)

*Description:
Spectrum-X Performance Test Plan Spectrum-4 and BlueField-3

Version:
1.0

Click-through License Agreement:
No Click-through License Agreement Required

Security Level:
Medium

Owner: Daniela Chili

Co-Owners:
Peter Rizk, Hani Salloum, Ariella Megory, Run Almog

Browse Security: OFF

Email DL:

Content expiry date:

Customer Approval - PCN: OFF

Internal Notes:

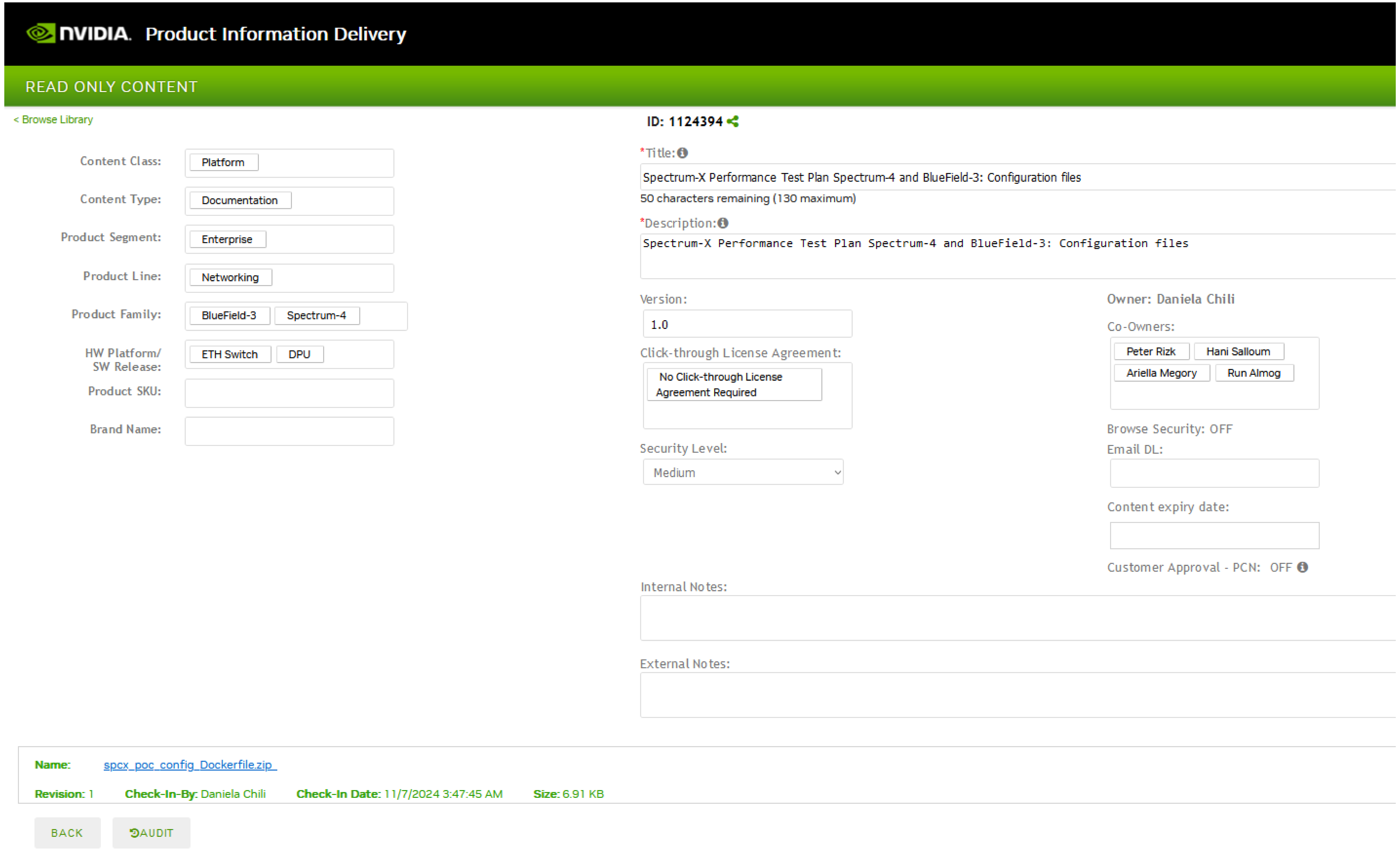
External Notes:

Name: Spectrum-X Performance Test Plan Spectrum-4 and BlueField-3 v1.0.pdf

Revision: 1 **Check-In-By:** Daniela Chili **Check-In Date:** 10/10/2024 10:46:53 AM **Size:** 1.69 MB

https://apps.nvidia.com/pid/contentlibraries/detail?id=1123419

Accessing Configuration Files



https://apps.nvidia.com/pid/contentlibraries/detail?id=1124394

Conclusion

- Spectrum-X PoC document and the PoC procedure is targeted to customers:
  - To see Spectrum-X in action with NVIDIA pre-canned tests
  - To see the performance of Spectrum-X with custom traffic parameters

- Pre-canned tests can be customized depending on specific requirements

- Spectrum-X PoC starts showing actual performance differentiation with more than 16 nodes in 2-Tier spine/leaf fabric topology, non-rail optimized fabric design

- NVIDIA experts accompany tests

THANK YOU