NVIDIA

Grounds-up LLM Development

Ayush Maheshwari, Manish Modani
Sr. Solutions Architect, Principal Solutions Architect

https://github.com/ayushbits/llm-development

ayushbits.github.io

Sessions

1. Understanding the hardware **(30 mins)**
   a) GPU vs CPU
   b) GPU communication primitives
   c) System Topology
2. Large scale data curation for LLM training **(1 hour)**
   a) Deep-dive into aspects of data curation
   b) Hands-on data curation

BREAK **(10 mins)**

3. Distributed and stable LLM training on a large-scale cluster **(1 hour)**
   a) Parallelism techniques
   b) Frameworks and wrappers
   c) Recipes and best practices
4. Inference **(15 mins)**
   a) Inference with build.nvidia.com
   b) Synthetic data generation

Register for GTC 2026

https://tinyurl.com/nvgtc2026

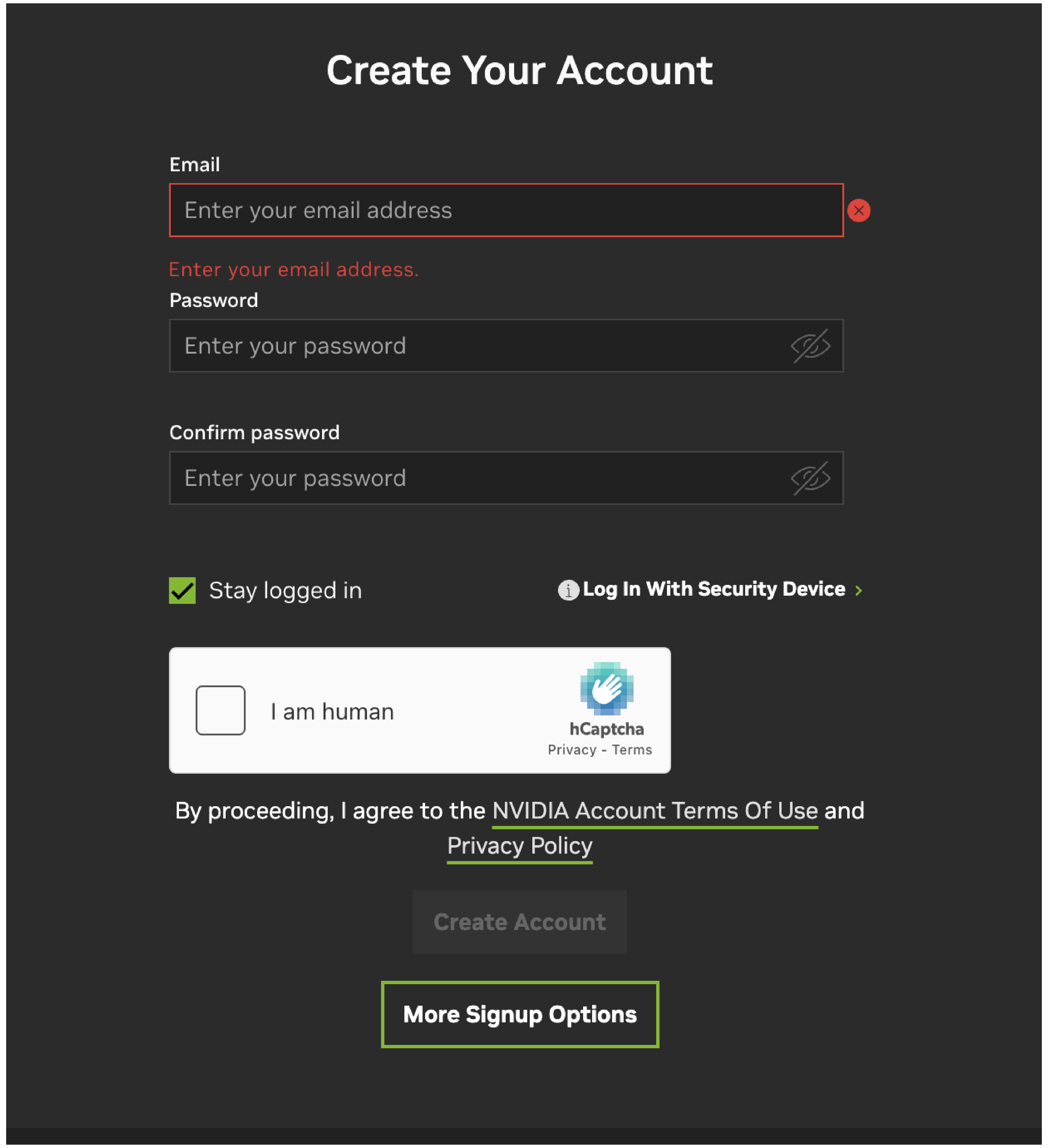
Scan QR

Logistics

brev.nvidia.com

- Go to this URL - https://tinyurl.com/casml-nvidia

- Signup with your email



Create Your Account

Email

Enter your email address

Enter your email address.

Password

Enter your password

Confirm password

Enter your password

☑ Stay logged in

Log In With Security Device ›

□ I am human

hCaptcha

Privacy - Terms

By proceeding, I agree to the NVIDIA Account Terms Of Use and Privacy Policy

Create Account

More Signup Options

Click on Launchables

NVIDIA GPUs Deployments Launchables Team Billing Docs $4,356.87 CASML-IISc-NV A

1. Click on this tab

Launchables

Create Launchable

Deploy pre-configured GPU environments with one click. Share and discover turnkey templates for AI workflows and GPU-accelerated software.

View All Metrics

Team Champion
A aymaheshwari-0

Total Views
1 +1 7d

Total Deploys
1 +1 7d

Organization Launchables

All Mine Team

Search Launchables

2. Click here

CASML-NV2 by aymaheshwari-0

1 views 1 deploys Show Configuration

Only my organization Edit

Deploy Launchable

Nemo-launchable-ayush

No description available

Container

nvcr.io/nvidia/nemo:24.05.01

Compute

NVIDIA H100 (80GiB)
1 GPUs x 16 CPUs | 200GiB
2TiB NEBIUS

Files

No file were uploaded

Exposed Ports

No ports or firewall rules were exposed

$2.84/hr

NVIDIA H100 (80GiB)
1 GPUs x 16 CPUs | 200GiB
2TiB NEBIUS

Choose a Provider: View All Options

H100
NVIDIA
NEBIUS
Creator's Choice
NEBIUS
LAMBDA-LABS
DATACRUNCH
2TiB

*Creator Originally Configured: 2TiB

Deploy Launchable

1. If instance, doesn’t spin: Refresh the page and/or Change the provider

2. Click here

NVIDIA
Deploying GPU
Install Software
Access GPU
$4,430.74
Logs
Access

CASML-IISc-NV > casml-nv2-355148

casml-nv2-355148

Created 8/12/2025, 10:48:50 am

NVIDIA H100 (80GiB)
1 GPUs x 30 CPUs | 120GiB

250GiB | helsinki-finland-2 | datacrunch | $2.26/hr | Starting

Docker Compose YAML | Waiting

This will take ~10 minutes depending on provider and number of requests.

Using Brev CLI (SSH)

Install the CLI

Windows (W...

Run this in your Windows (WSL) terminal

```
sudo bash -c "$(curl -fsSL https://raw.githubusercontent.com/brevdev/brev-cli/main/bin/install-latest.sh)"
```

Make sure you have WSL 2 installed and configured, virtualization enabled in your BIOS, and Ubuntu installed from the Microsoft Store.

Stop
Delete
Logs
Access
Share a Service
Edit Access
Delete
Allow All IPs
Expose Port

CASML-IISc-NV > casml-nv2-355148

casml-nv2-355148 Created 8/12/2025, 10:48:50 am

NVIDIA H100 (80GiB)
1 GPUs x 30 CPUs | 120GiB 250GiB 86.38.238.89 helsinki-finland-2 | datacrunch | $2.26/hr Running

Docker Compose YAML Built

Using Secure Links

Access any http application protected with your login; share it with teammates, or the public. Docs here.

Port	Shareable URL	Health	
8888	https://tunnel-20-tvmngrk4v.brevlab.com	Healthy	Edit Access Delete

In Access tab, scroll and click here

Using Ports

TCP/UDP Ports

This cloud provider doesn't allow the modifications of ports

Expose Port(s) (e.g. 2000 or 2000-2020)

Part 1

Understanding the hardware **(30 mins)**

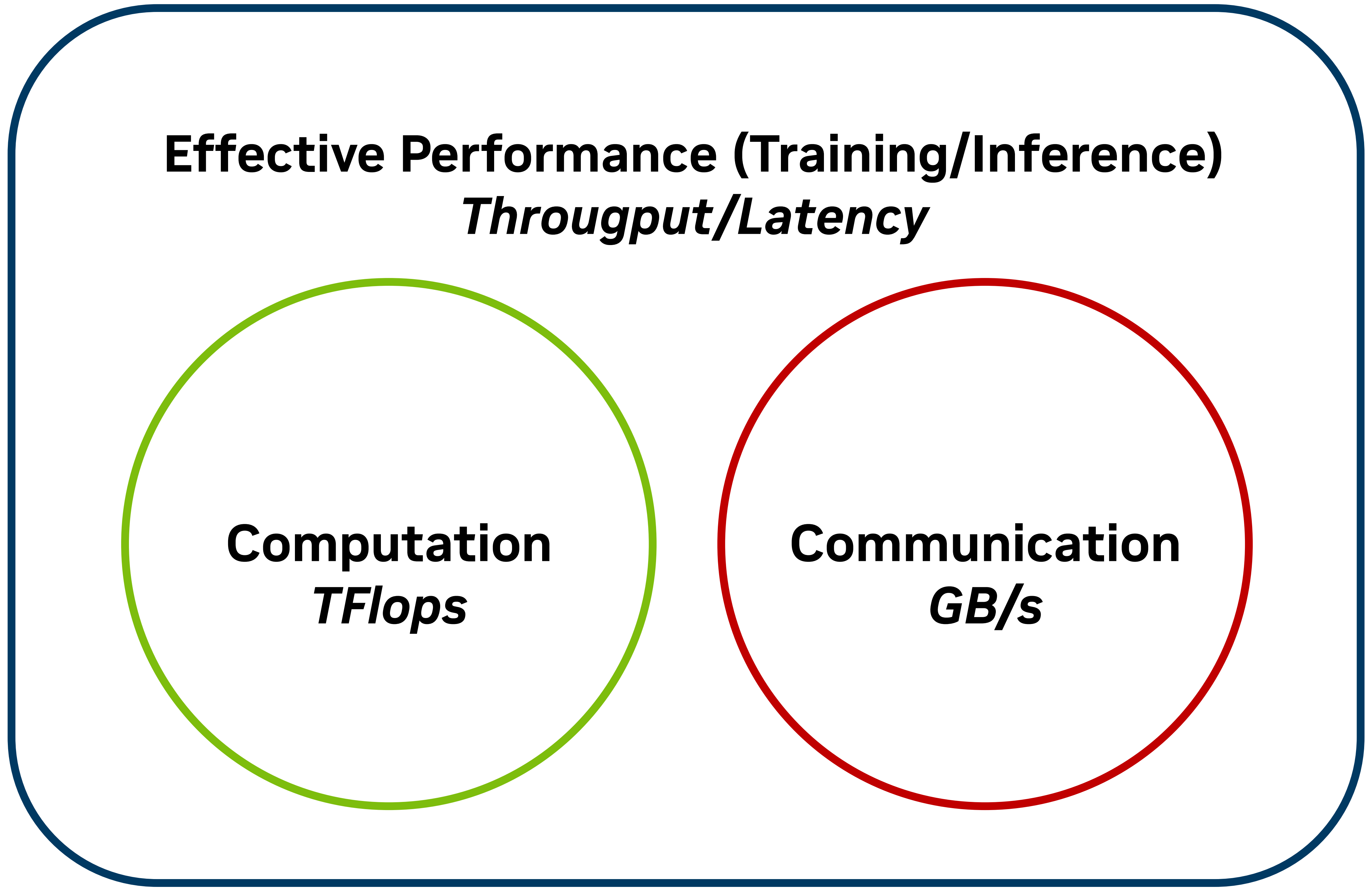
a) GPU vs CPU

b) GPU communication primitives

c) System Topology

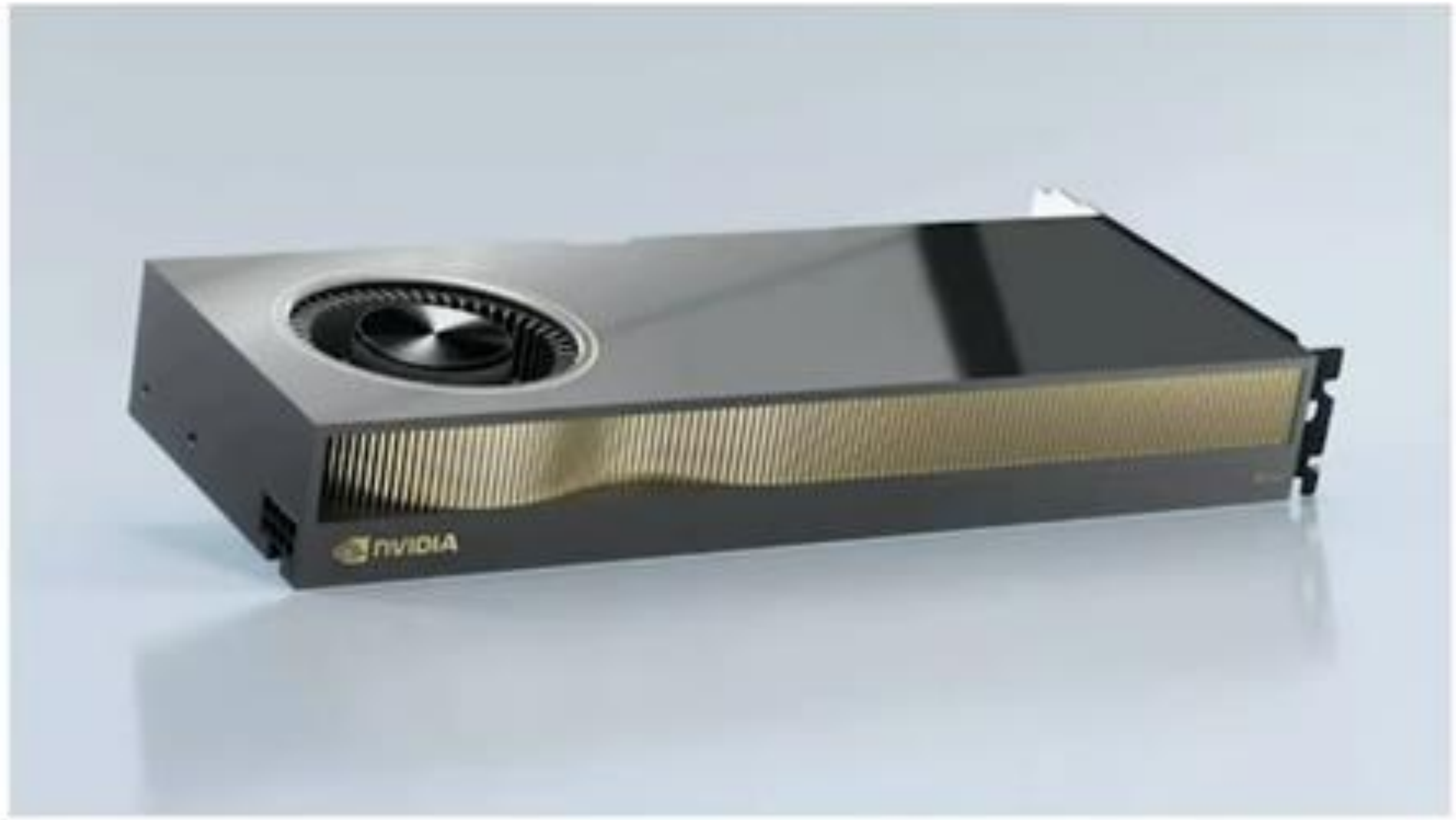
Why should you care?

- Understand the **hardware and its performance** on multiple GPUs.
- Ensure that your **training performance aligns** with the h/w benchmarks
- Evaluate the cluster to ensure platform fits **within your needs**.
- Take advantage of **new techniques** for multi-GPU computing.



1 floating point operation

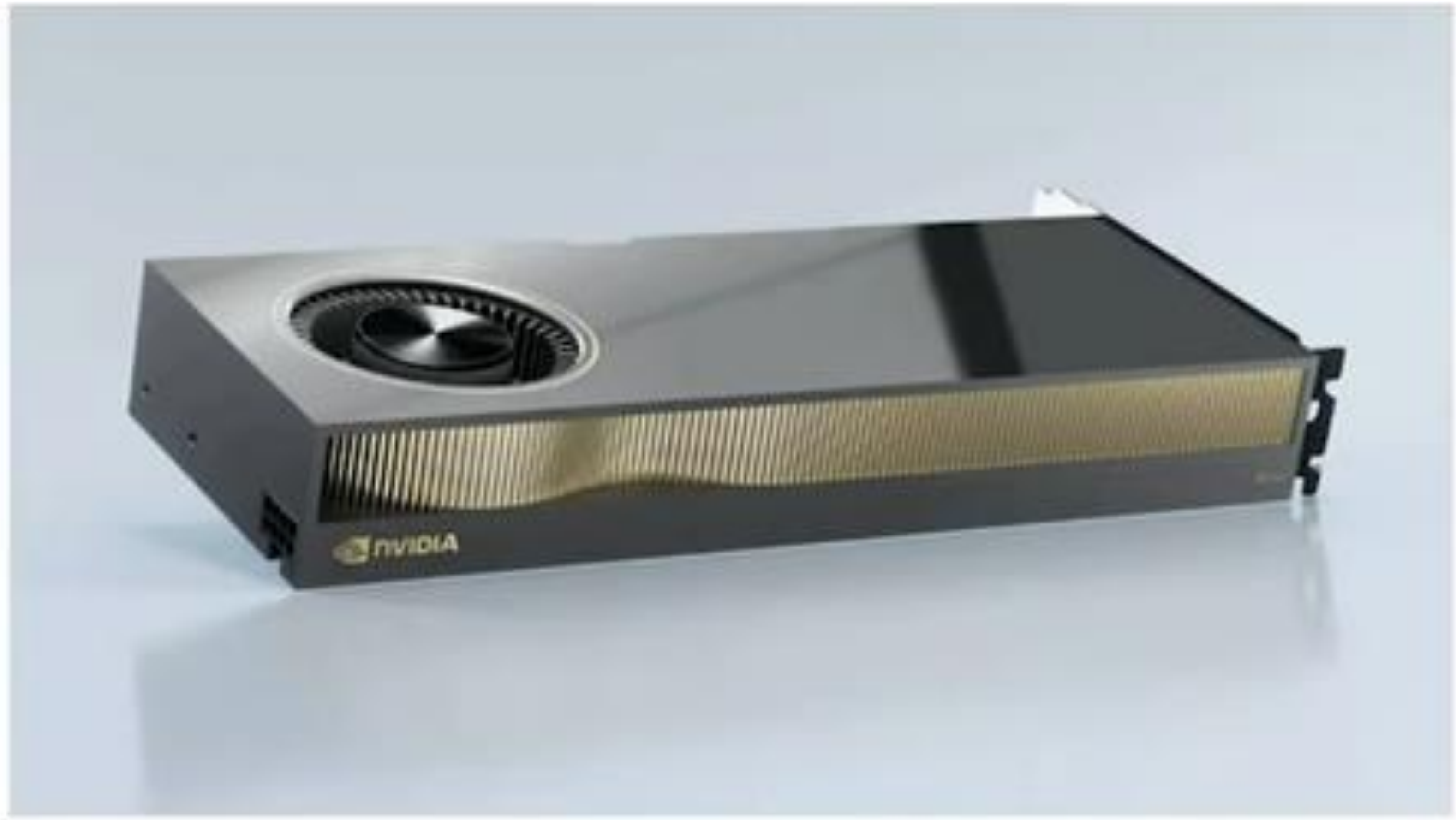
Seconds ??



Seconds ??

1 floating point operation

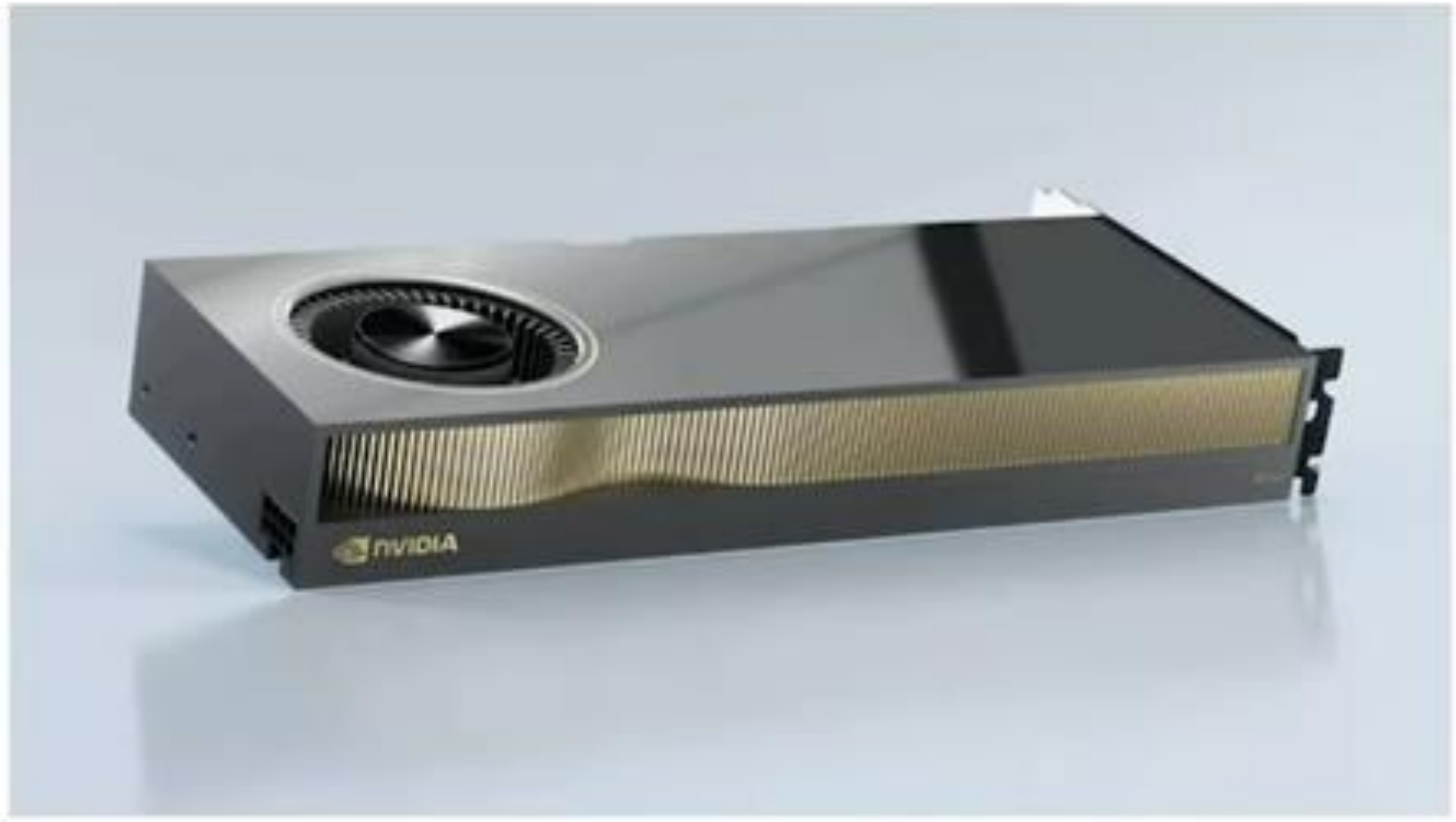
~1ns



~1μs

2048 x 2048 matmul

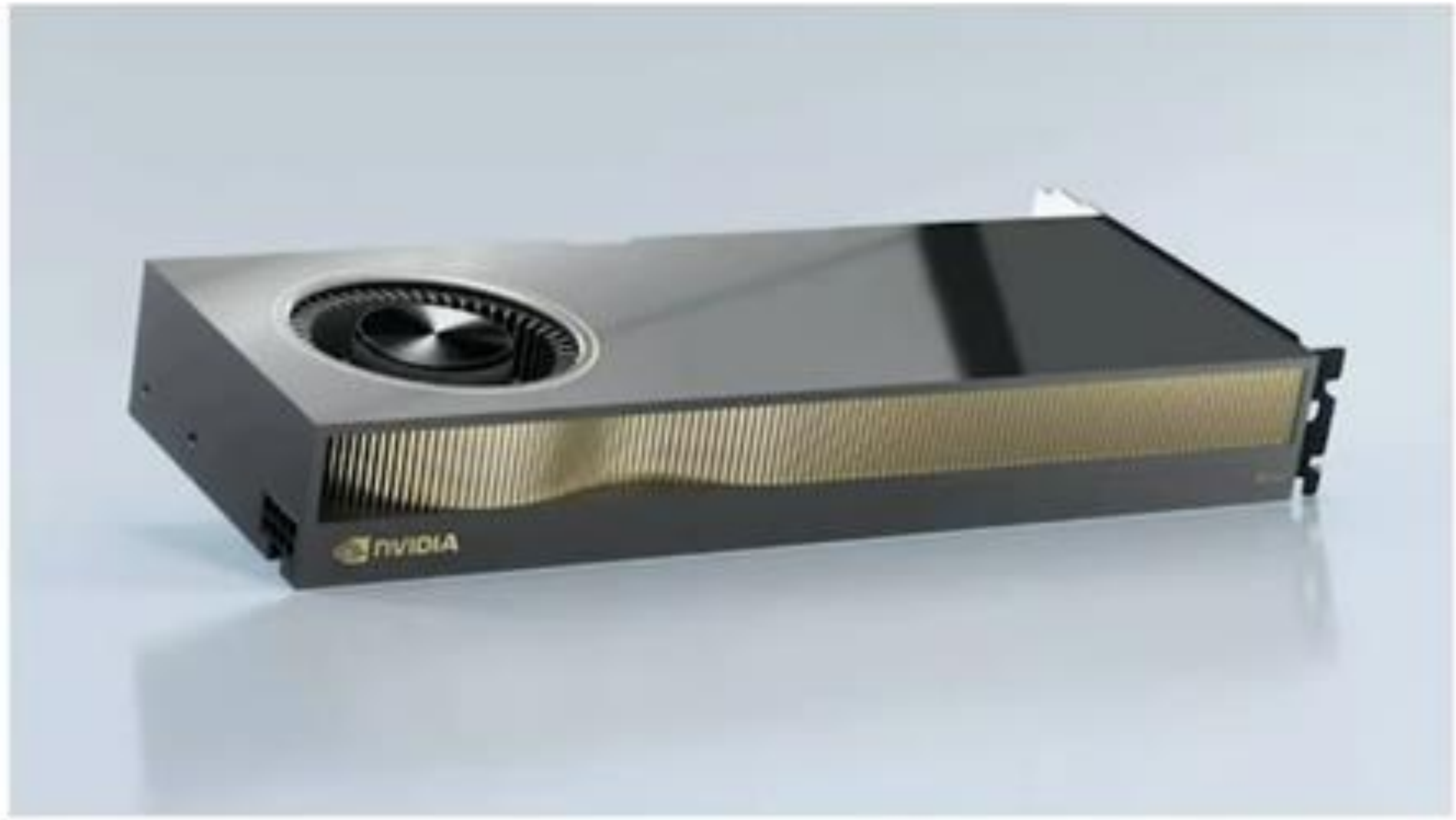
Seconds ??



Seconds ??

2048 x 2048 matmul

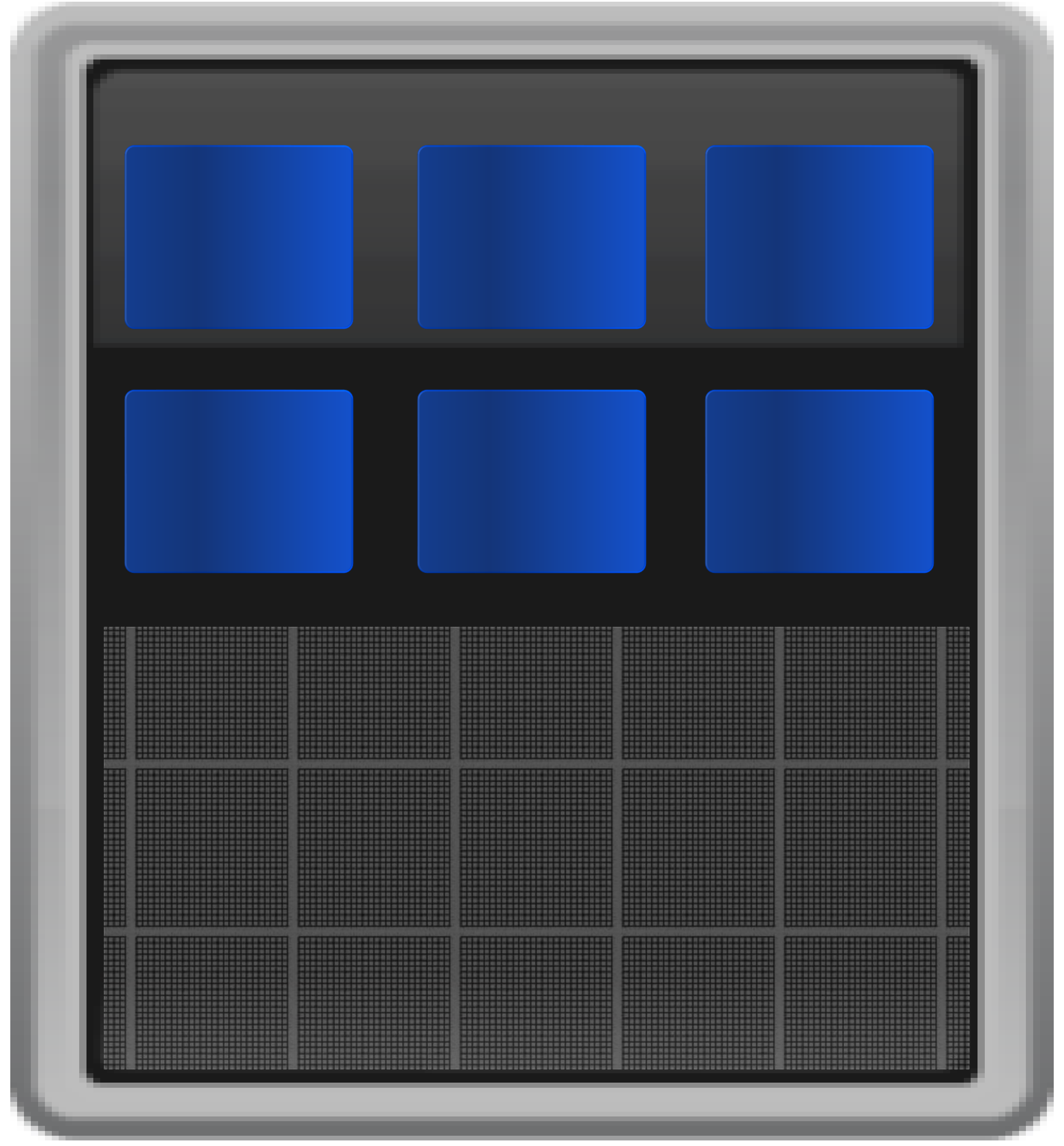
28ms



.2ms (200μs)

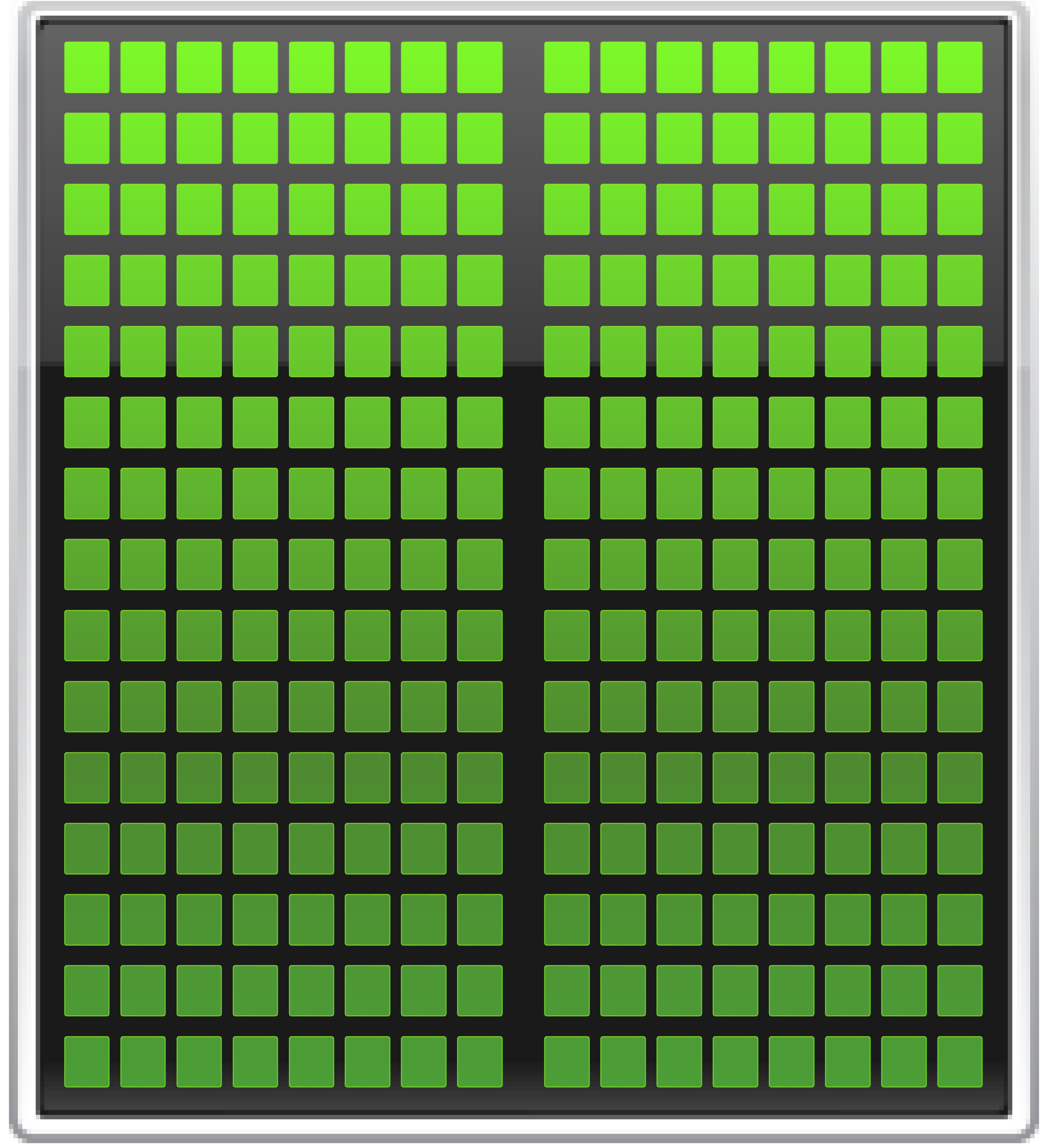
Different Objectives

CPU
Optimized for
Serial Tasks



+

GPU Accelerator
Optimized for
Parallel Tasks



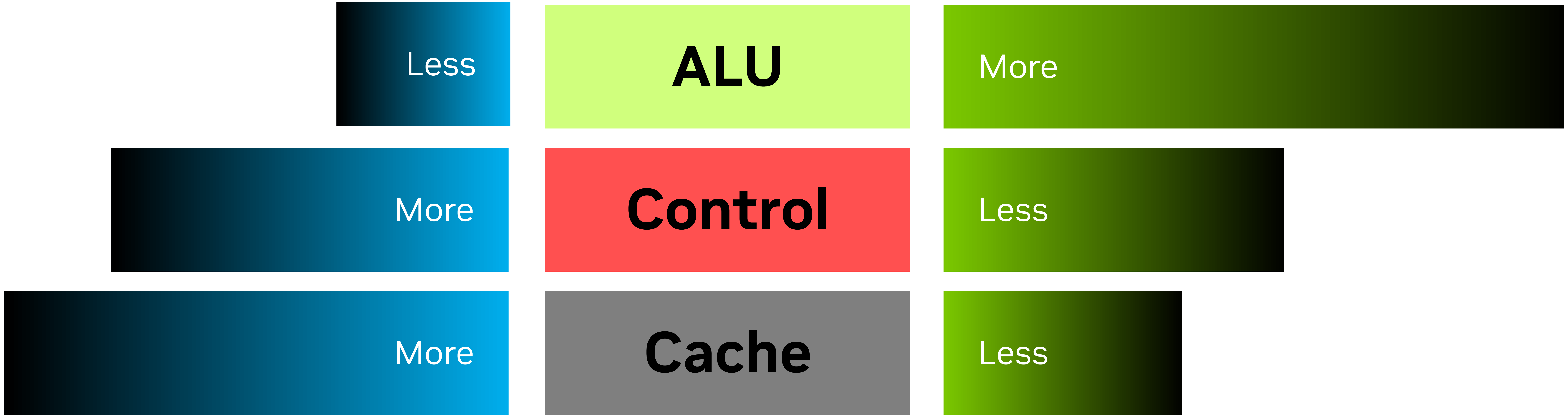
CPU vs GPU

Latency vs Throughput-oriented Design

CONTROL
ALU
ALU
ALU
ALU
CACHE

Src: modal.com

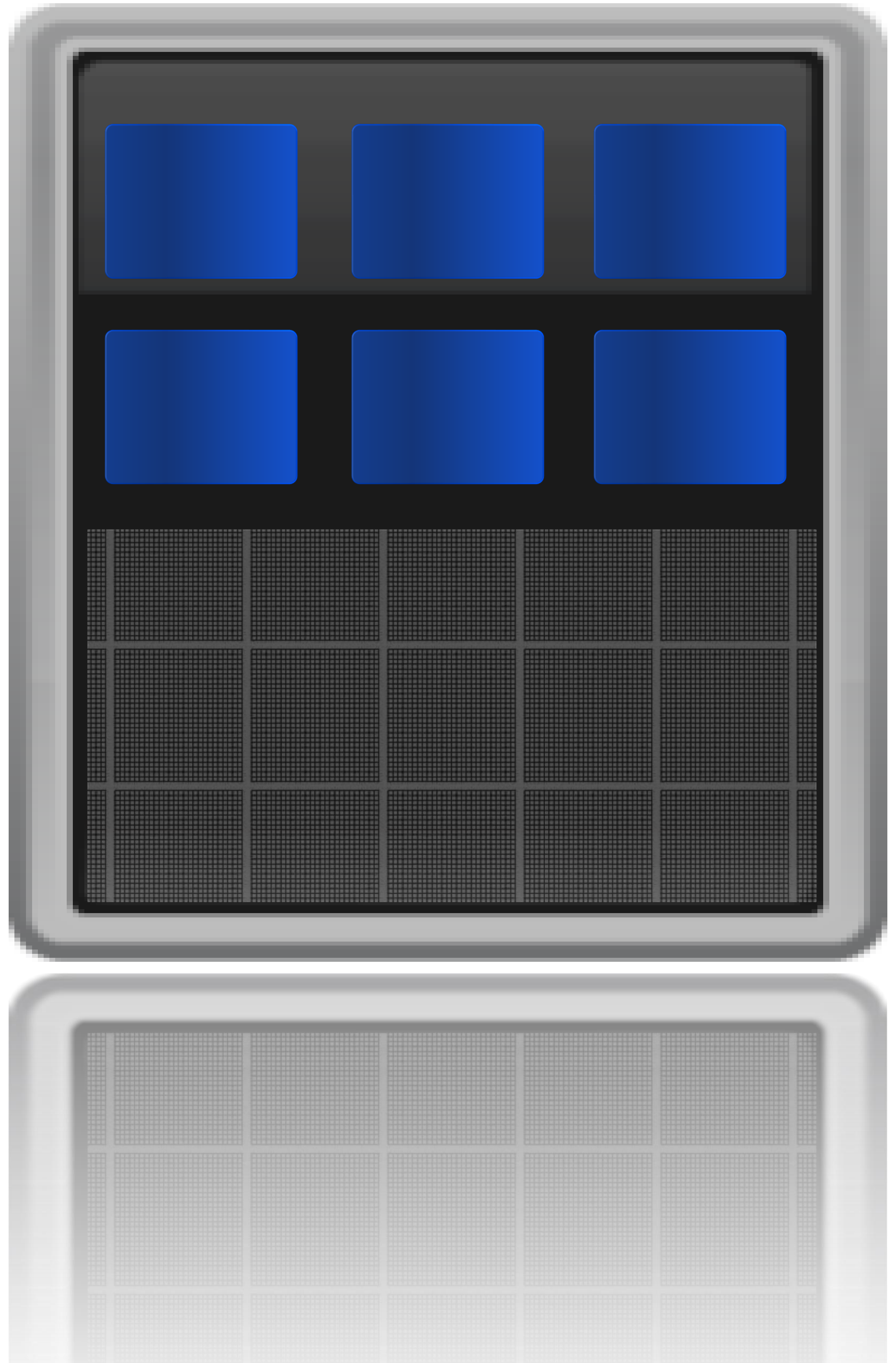
Silicon Budget

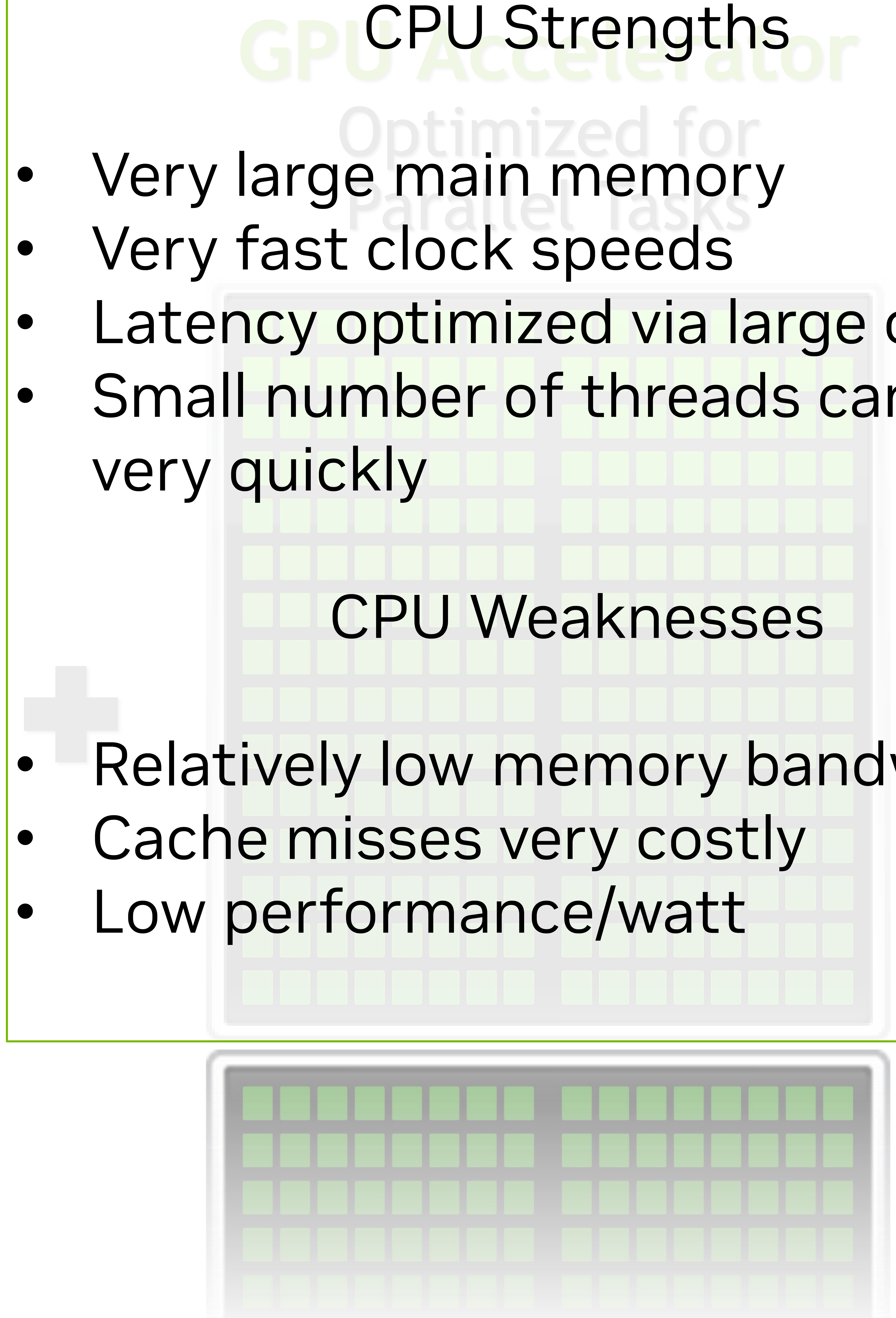


Less	ALU	More
More	Control	Less
More	Cache	Less

CPU

Optimized for Serial Tasks





CPU Strengths

- Very large main memory
- Very fast clock speeds
- Latency optimized via large caches
- Small number of threads can run very quickly

CPU Weaknesses

- Relatively low memory bandwidth
- Cache misses very costly
- Low performance/watt

GPU Strengths

- High bandwidth main memory
- Significantly more compute resources
- Latency tolerant via parallelism
- High throughput
- High performance/watt

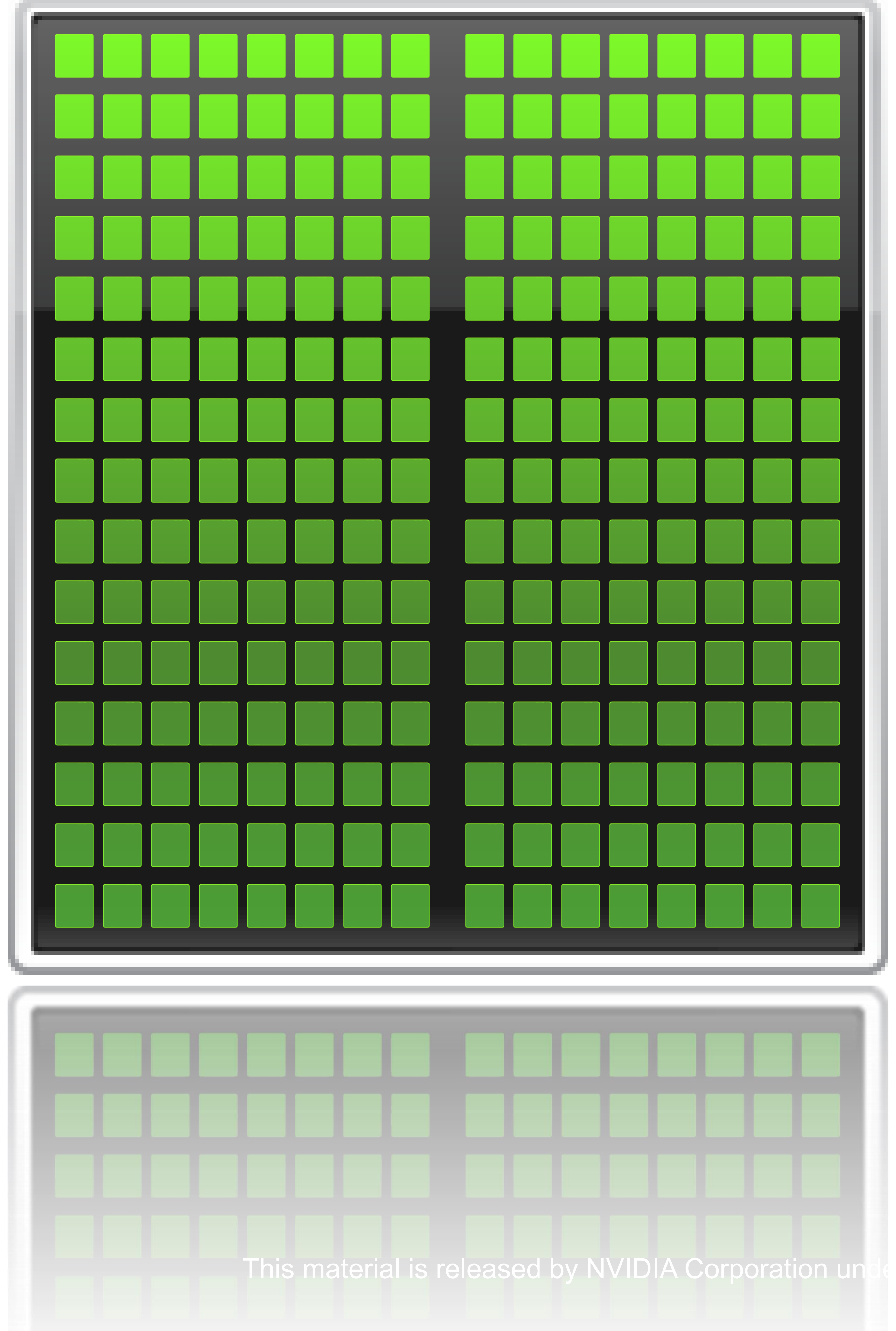
GPU Weaknesses

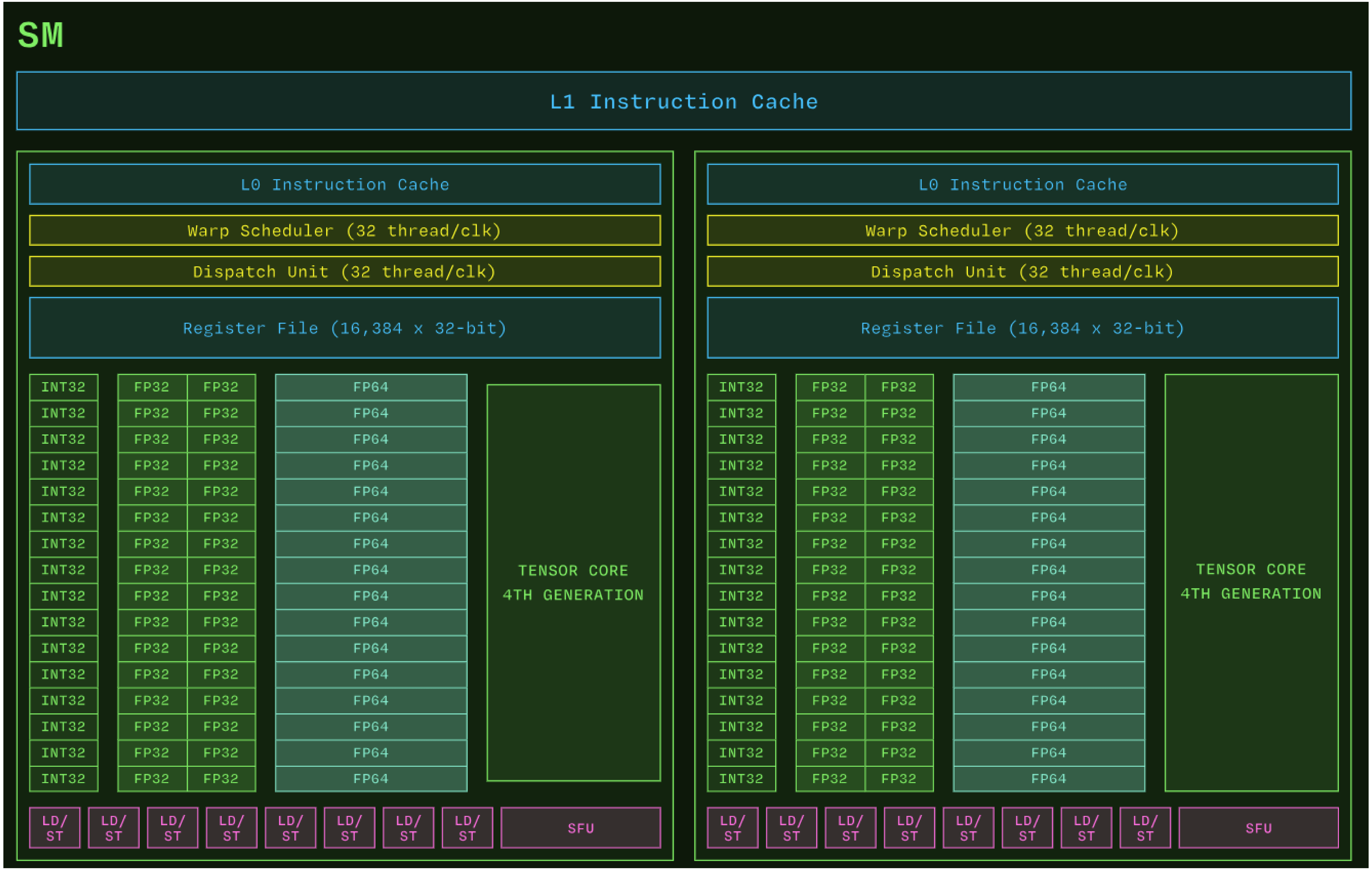
- Relatively low memory capacity
- Low per-thread performance

CPU
Optimized for
Serial Tasks

GPU Accelerator

Optimized for
Parallel Tasks



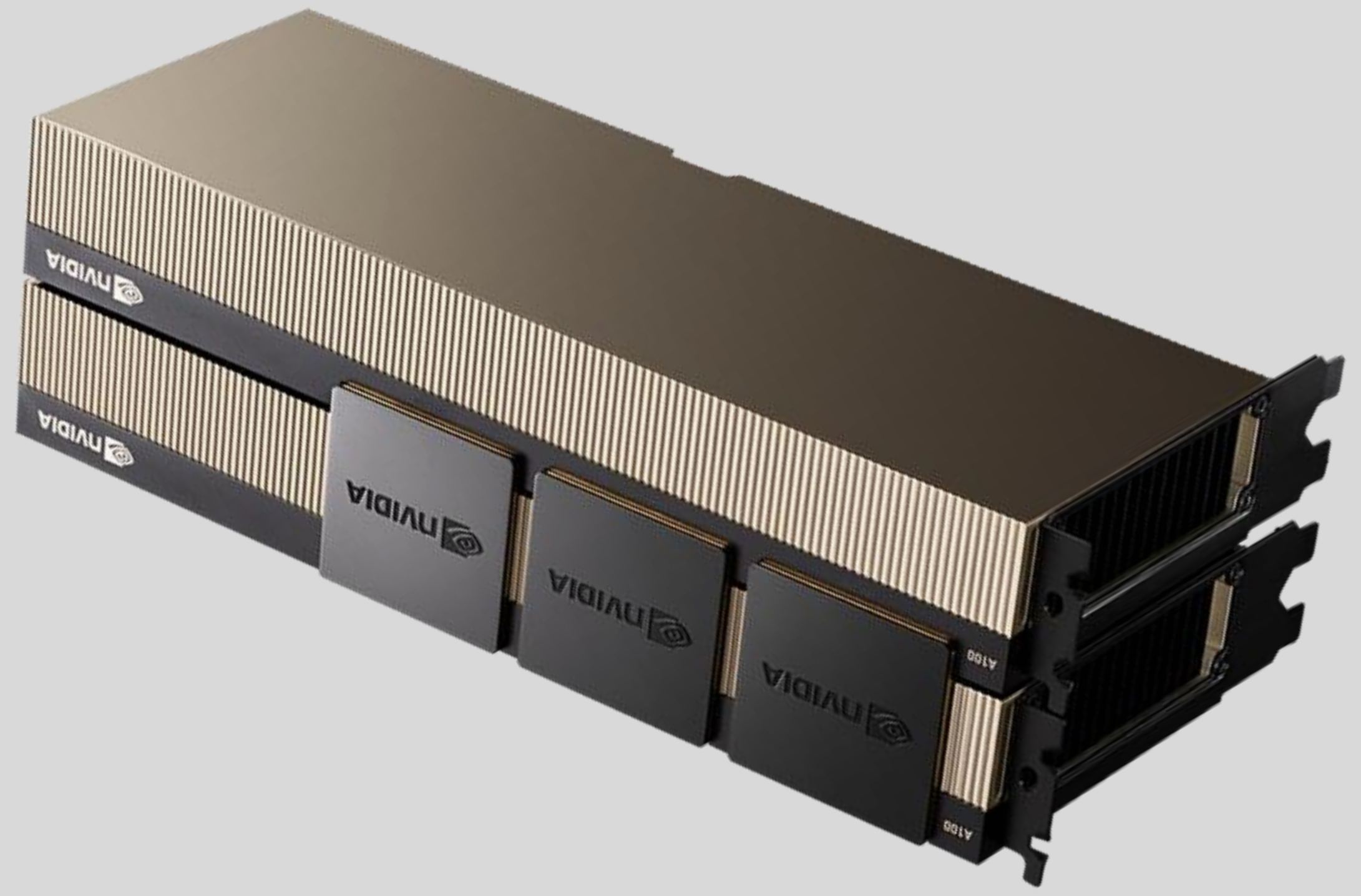

SM
L1 Instruction Cache
L0 Instruction Cache
Warp Scheduler (32 thread/clk)
Dispatch Unit (32 thread/clk)
Register File (16,384 x 32-bit)
INT32
FP32
FP64
TENSOR CORE
4TH GENERATION
LD/
ST
SFU
L0 Instruction Cache
Warp Scheduler (32 thread/clk)
Dispatch Unit (32 thread/clk)
Register File (16,384 x 32-bit)
INT32
FP32
FP64
TENSOR CORE
4TH GENERATION
LD/
ST
SFU

Src: modal.com

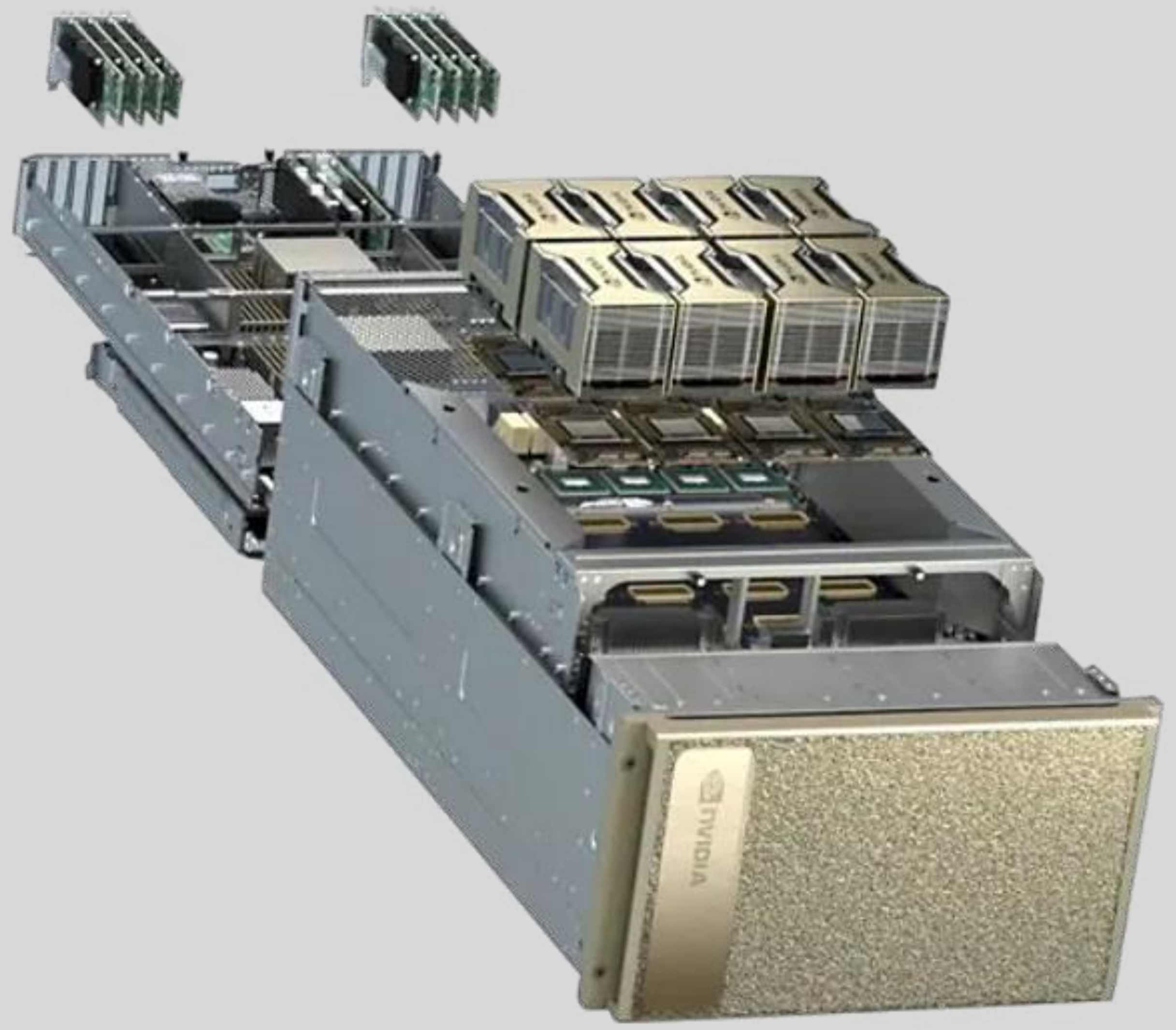
Multi-GPU Computing

NCCL: NVIDIA Collective Communication Library

Inter-GPU communication on PCI, NVLink, IB/RoCE, and other networks.



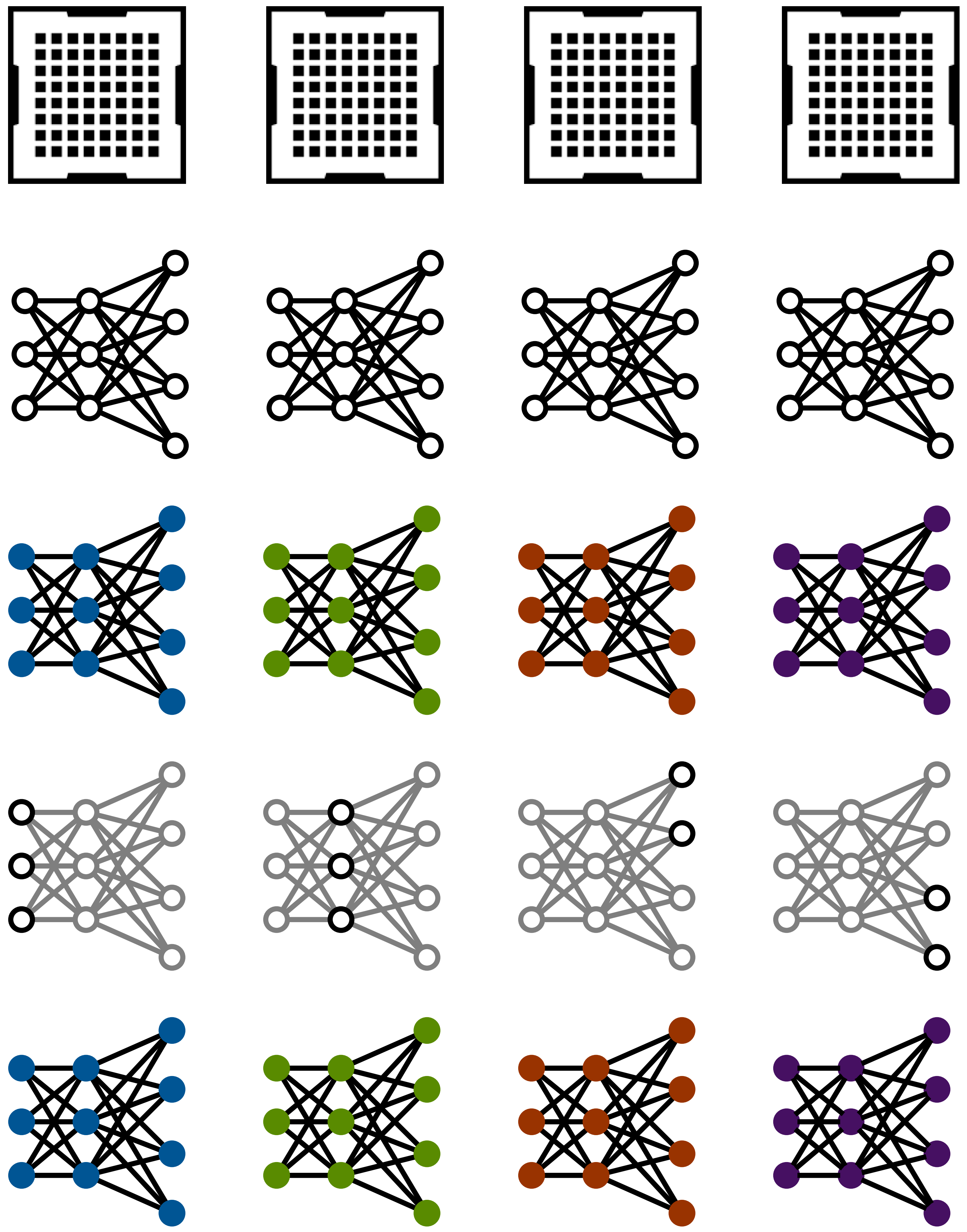
PCI Server



DGX/HGX

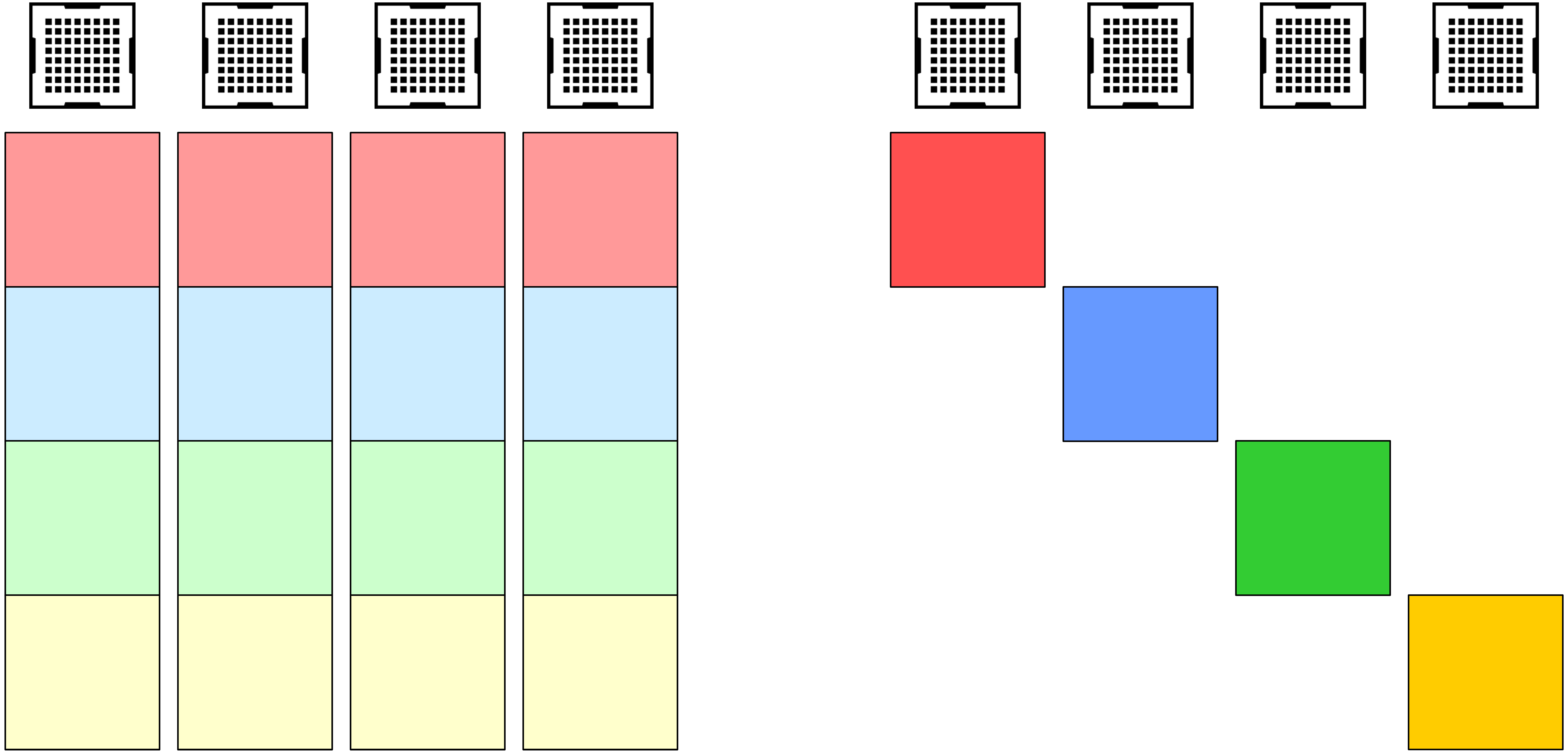
Large systems

Multi-GPU Computing in DL



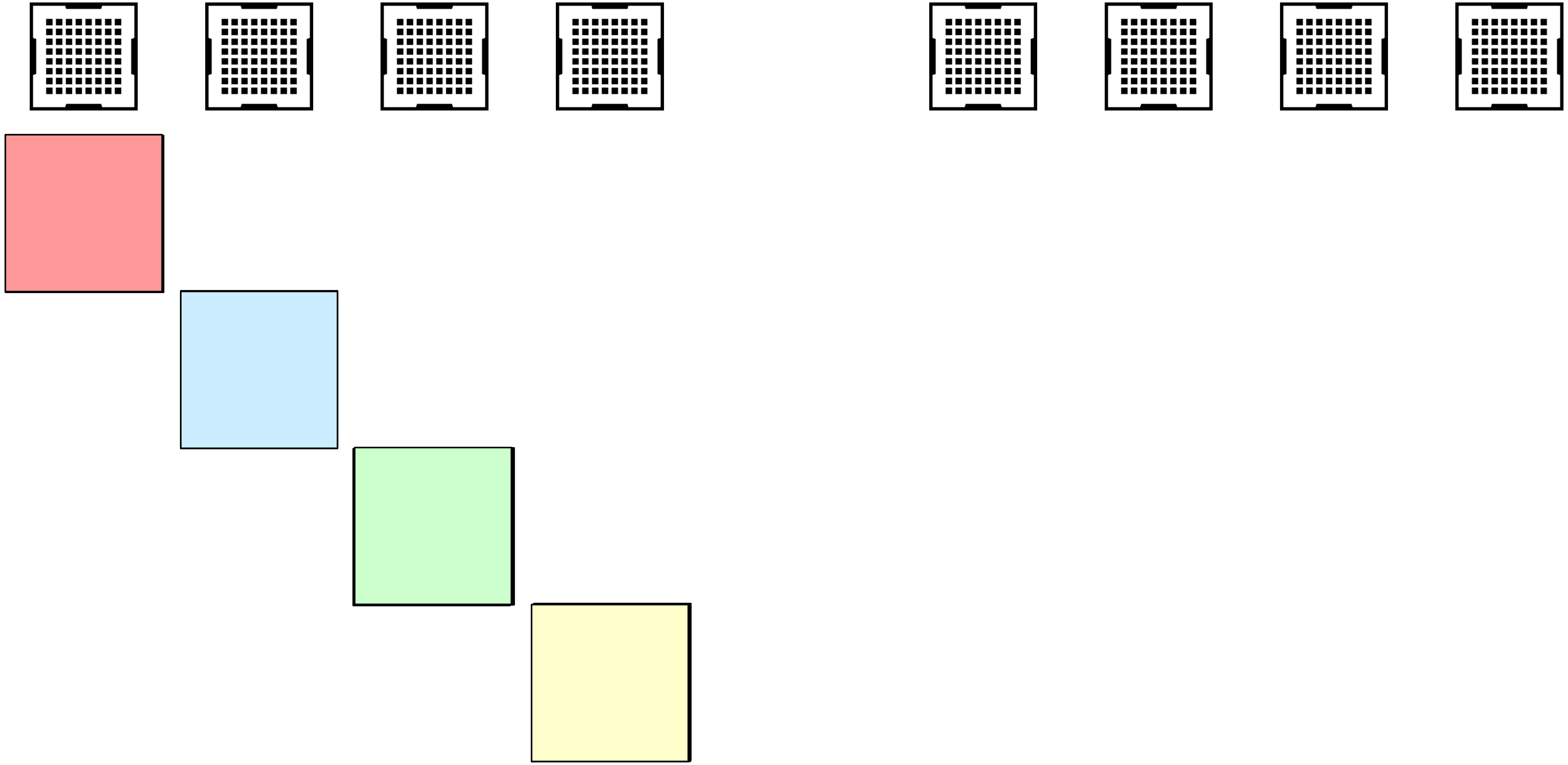
Data Parallelism / FSDP	All-reduce, all-gather, reduce-scatter
Tensor Parallelism	All-reduce, all-gather, reduce-scatter
Pipeline Parallelism	Send / receive
Expert Parallelism	All-to-all

Communication primitives



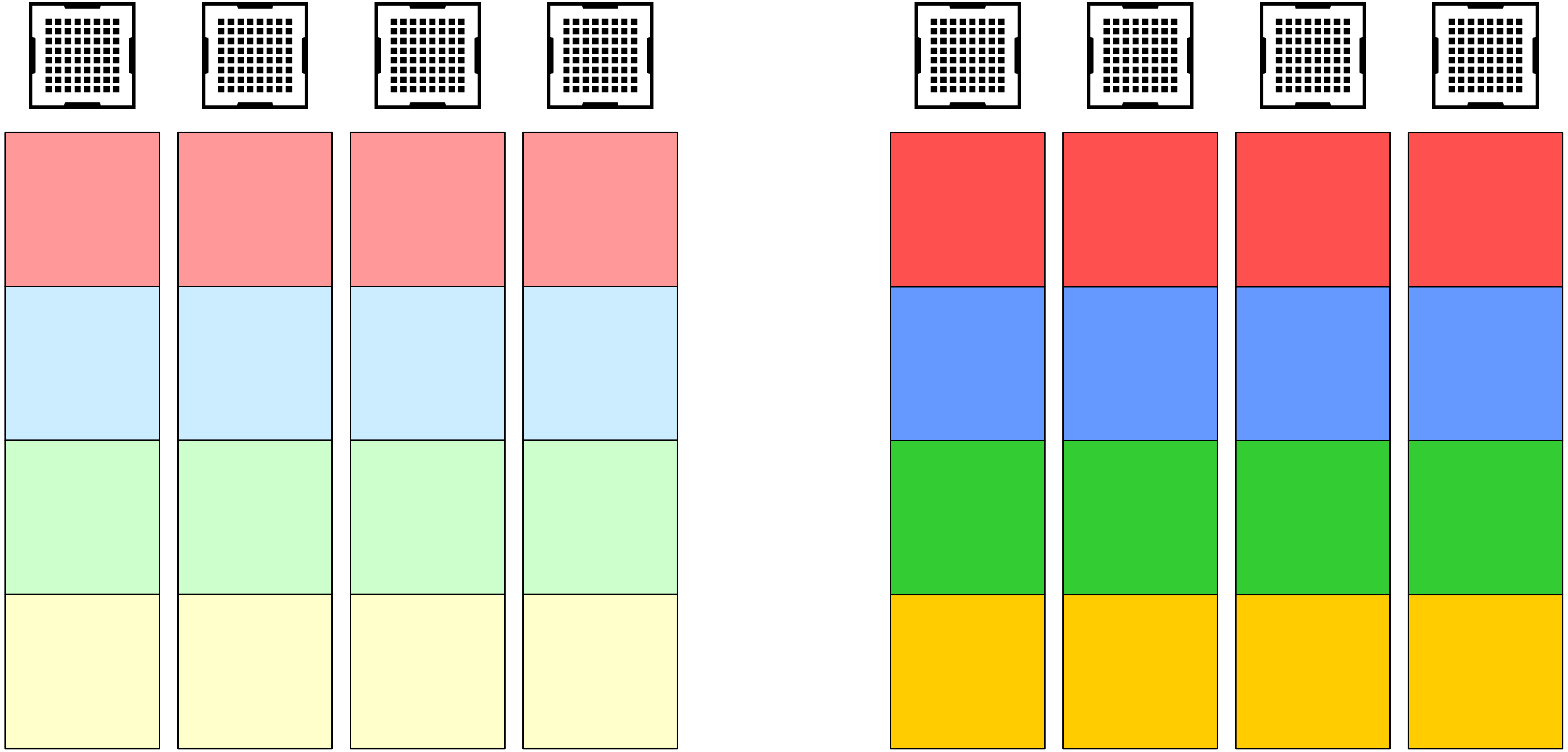
Reduce-scatter

Communication primitives



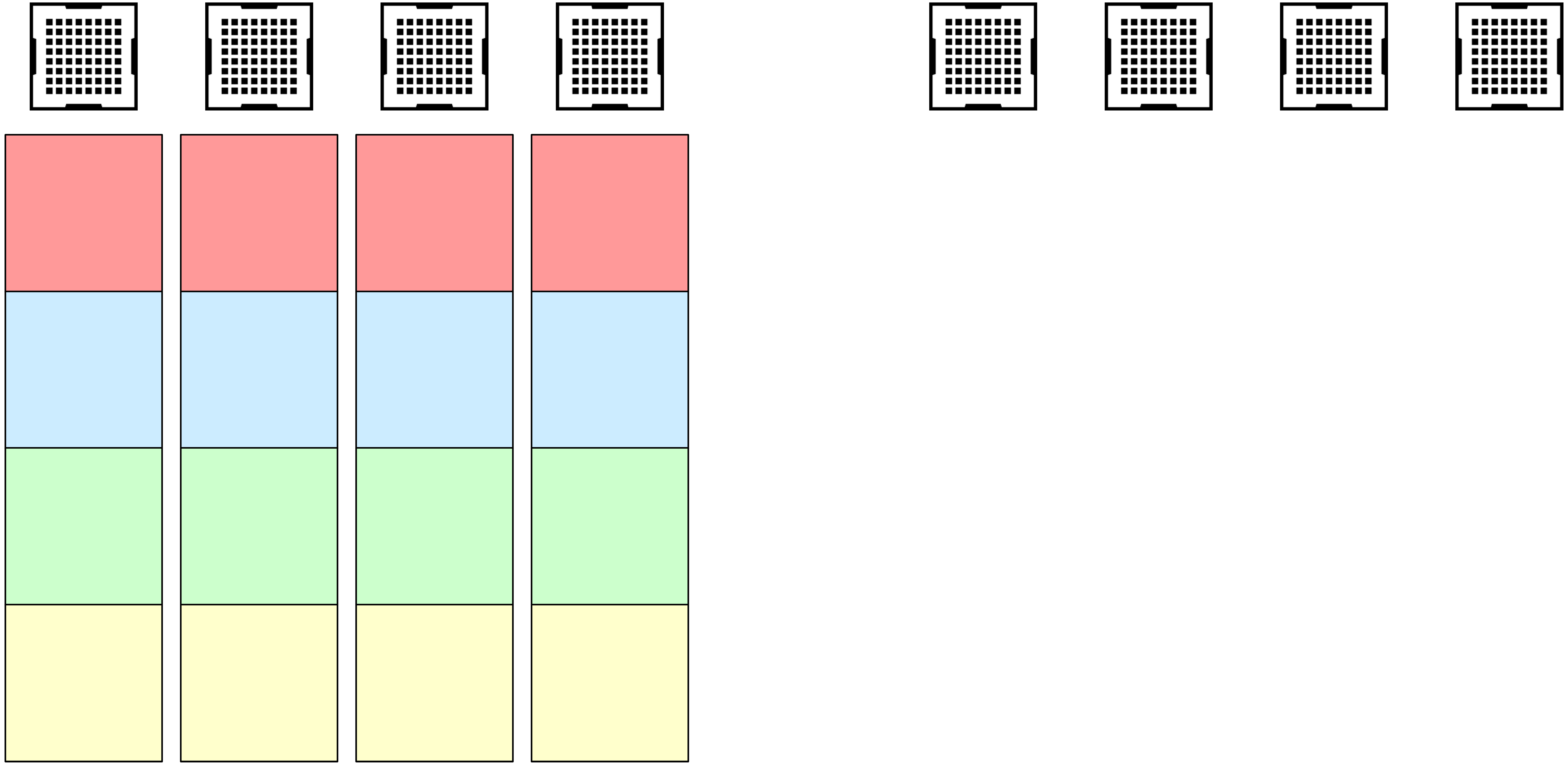
All-gather

Communication primitives



All-reduce

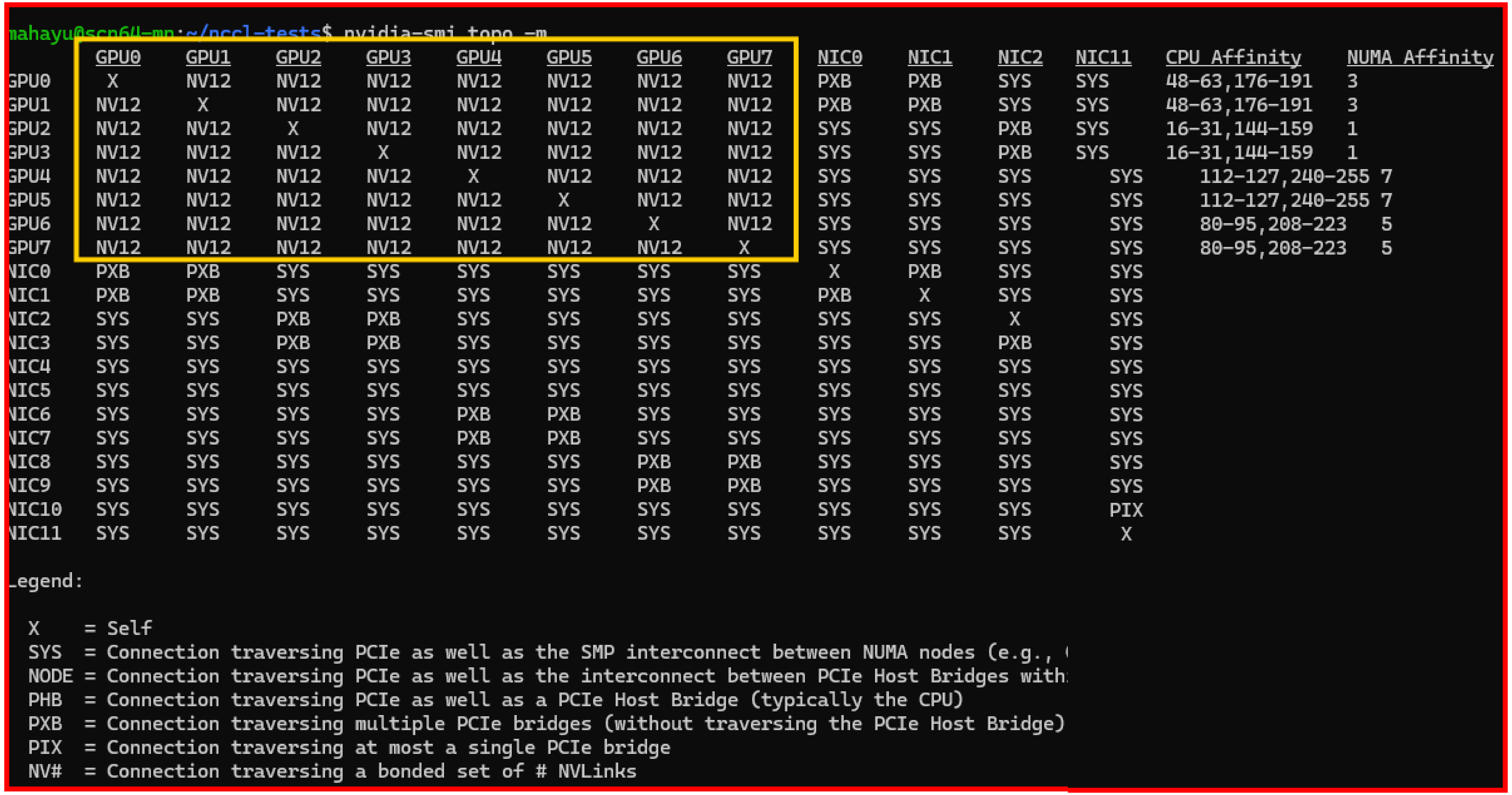
Communication primitives



All-to-all

Checking System Topology (A100)

`nvidia-smi topo –m`



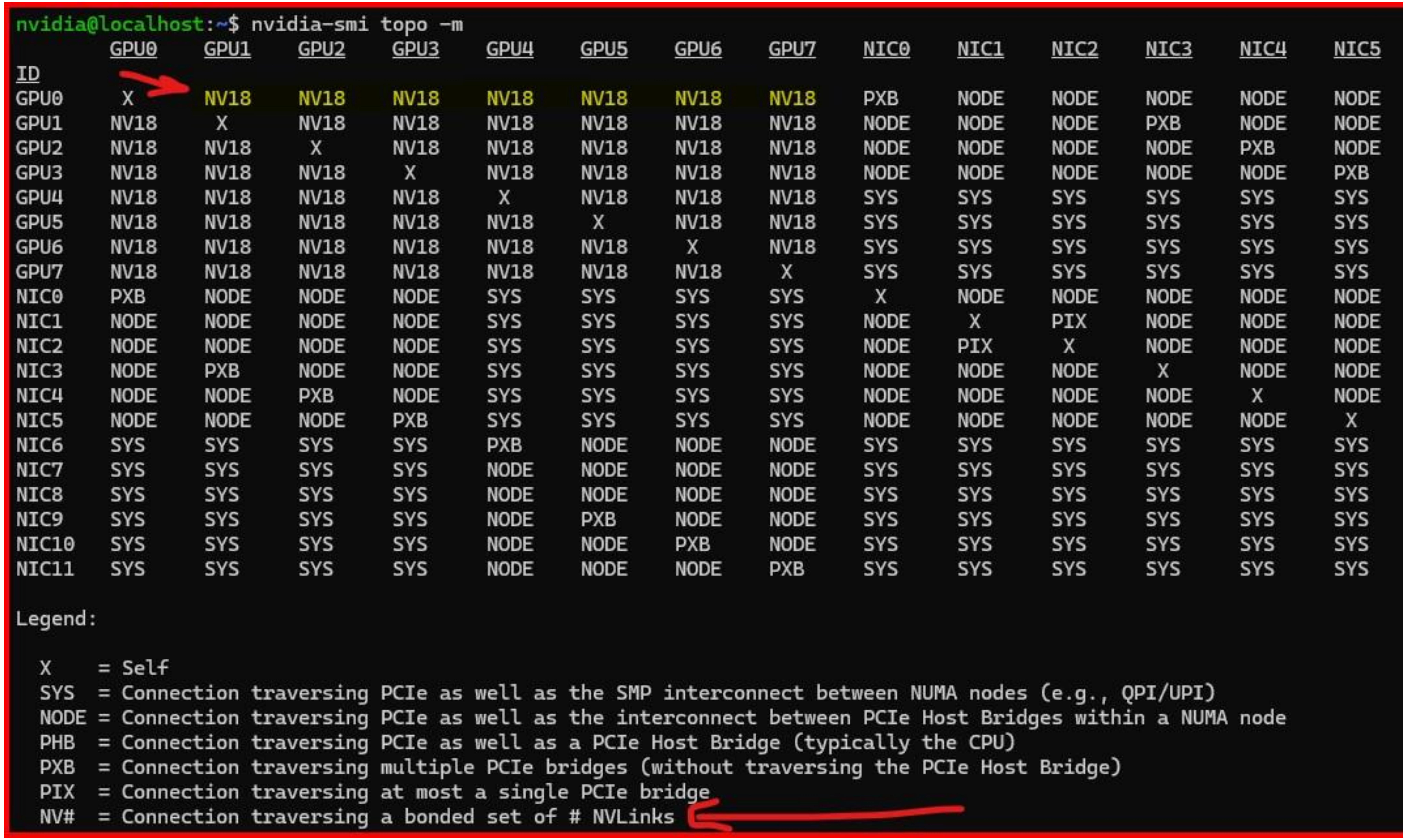
```
mahayu@scn64-mn:~/nccl-tests$ nvidia-smi topo -m
        GPU0    GPU1    GPU2    GPU3    GPU4    GPU5    GPU6    GPU7    NIC0    NIC1    NIC2    NIC11   CPU Affinity    NUMA Affinity
GPU0     X      NV12    NV12    NV12    NV12    NV12    NV12    NV12    PXB     PXB     SYS     SYS     48-63,176-191   3
GPU1    NV12     X      NV12    NV12    NV12    NV12    NV12    NV12    PXB     PXB     SYS     SYS     48-63,176-191   3
GPU2    NV12    NV12     X      NV12    NV12    NV12    NV12    NV12    SYS     SYS     PXB     SYS     16-31,144-159   1
GPU3    NV12    NV12    NV12     X      NV12    NV12    NV12    NV12    SYS     SYS     PXB     SYS     16-31,144-159   1
GPU4    NV12    NV12    NV12    NV12     X      NV12    NV12    NV12    SYS     SYS     SYS        SYS     112-127,240-255 7
GPU5    NV12    NV12    NV12    NV12    NV12     X      NV12    NV12    SYS     SYS     SYS        SYS     112-127,240-255 7
GPU6    NV12    NV12    NV12    NV12    NV12    NV12     X      NV12    SYS     SYS     SYS        SYS     80-95,208-223   5
GPU7    NV12    NV12    NV12    NV12    NV12    NV12    NV12     X      SYS     SYS     SYS        SYS     80-95,208-223   5
NIC0    PXB     PXB     SYS     SYS     SYS     SYS     SYS     SYS      X      PXB     SYS        SYS
NIC1    PXB     PXB     SYS     SYS     SYS     SYS     SYS     SYS     PXB      X      SYS        SYS
NIC2    SYS     SYS     PXB     PXB     SYS     SYS     SYS     SYS     SYS     SYS      X         SYS
NIC3    SYS     SYS     PXB     PXB     SYS     SYS     SYS     SYS     SYS     SYS     PXB        SYS
NIC4    SYS     SYS     SYS     SYS     SYS     SYS     SYS     SYS     SYS     SYS     SYS        SYS
NIC5    SYS     SYS     SYS     SYS     SYS     SYS     SYS     SYS     SYS     SYS     SYS        SYS
NIC6    SYS     SYS     SYS     SYS     PXB     PXB     SYS     SYS     SYS     SYS     SYS        SYS
NIC7    SYS     SYS     SYS     SYS     PXB     PXB     SYS     SYS     SYS     SYS     SYS        SYS
NIC8    SYS     SYS     SYS     SYS     SYS     SYS     PXB     PXB     SYS     SYS     SYS        SYS
NIC9    SYS     SYS     SYS     SYS     SYS     SYS     PXB     PXB     SYS     SYS     SYS        SYS
NIC10   SYS     SYS     SYS     SYS     SYS     SYS     SYS     SYS     SYS     SYS     SYS        PIX
NIC11   SYS     SYS     SYS     SYS     SYS     SYS     SYS     SYS     SYS     SYS     SYS         X

Legend:

  X    = Self
  SYS  = Connection traversing PCIe as well as the SMP interconnect between NUMA nodes (e.g.,
  NODE = Connection traversing PCIe as well as the interconnect between PCIe Host Bridges with
  PHB  = Connection traversing PCIe as well as a PCIe Host Bridge (typically the CPU)
  PXB  = Connection traversing multiple PCIe bridges (without traversing the PCIe Host Bridge)
  PIX  = Connection traversing at most a single PCIe bridge
  NV#  = Connection traversing a bonded set of # NVLinks
```

Checking System Topology (H100)

`nvidia-smi topo –m`



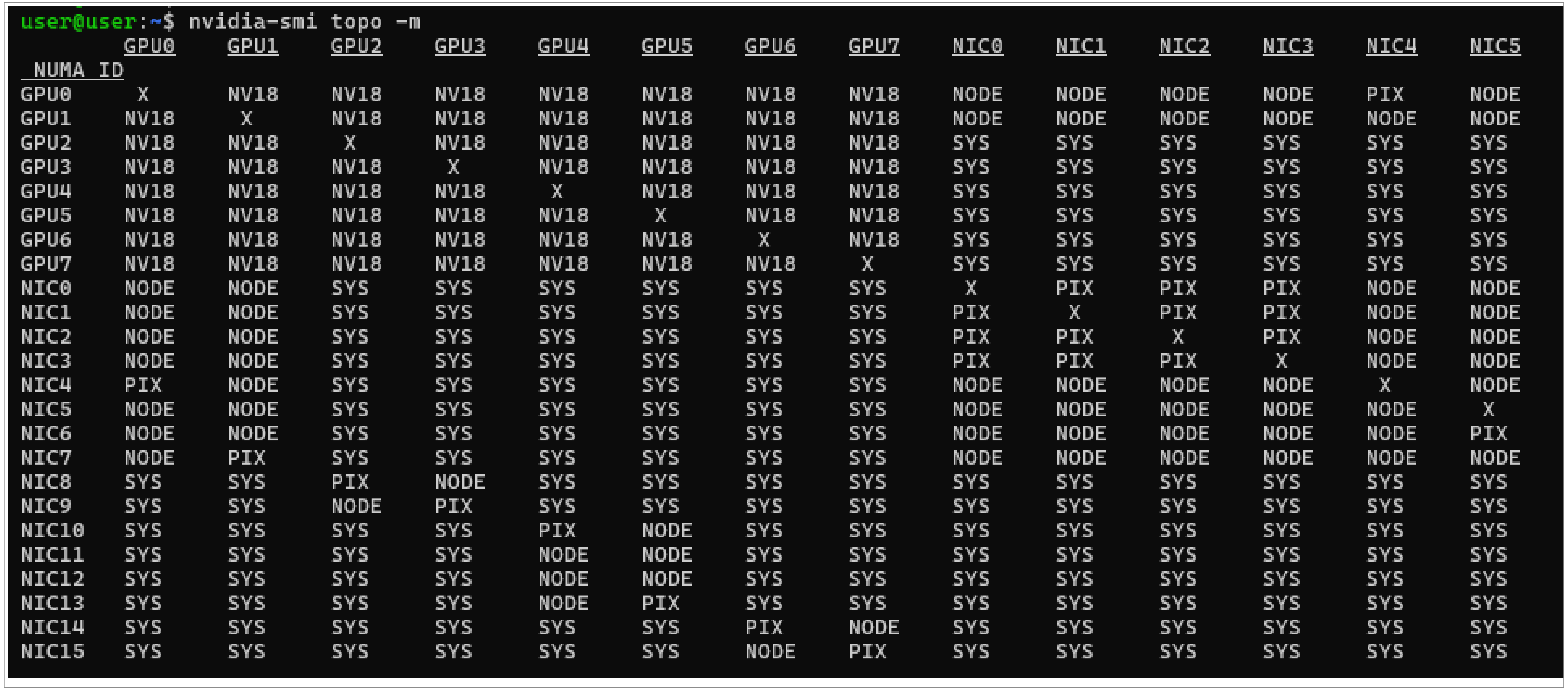
```
nvidia@localhost:~$ nvidia-smi topo -m
        GPU0    GPU1    GPU2    GPU3    GPU4    GPU5    GPU6    GPU7    NIC0    NIC1    NIC2    NIC3    NIC4    NIC5
ID
GPU0     X      NV18    NV18    NV18    NV18    NV18    NV18    NV18    PXB     NODE    NODE    NODE    NODE    NODE
GPU1    NV18     X      NV18    NV18    NV18    NV18    NV18    NV18    NODE    NODE    NODE    PXB     NODE    NODE
GPU2    NV18    NV18     X      NV18    NV18    NV18    NV18    NV18    NODE    NODE    NODE    NODE    PXB     NODE
GPU3    NV18    NV18    NV18     X      NV18    NV18    NV18    NV18    NODE    NODE    NODE    NODE    NODE    PXB
GPU4    NV18    NV18    NV18    NV18     X      NV18    NV18    NV18    SYS     SYS     SYS     SYS     SYS     SYS
GPU5    NV18    NV18    NV18    NV18    NV18     X      NV18    NV18    SYS     SYS     SYS     SYS     SYS     SYS
GPU6    NV18    NV18    NV18    NV18    NV18    NV18     X      NV18    SYS     SYS     SYS     SYS     SYS     SYS
GPU7    NV18    NV18    NV18    NV18    NV18    NV18    NV18     X      SYS     SYS     SYS     SYS     SYS     SYS
NIC0    PXB     NODE    NODE    NODE    SYS     SYS     SYS     SYS      X      NODE    NODE    NODE    NODE    NODE
NIC1    NODE    NODE    NODE    NODE    SYS     SYS     SYS     SYS     NODE     X      PIX     NODE    NODE    NODE
NIC2    NODE    NODE    NODE    NODE    SYS     SYS     SYS     SYS     NODE    PIX      X      NODE    NODE    NODE
NIC3    NODE    PXB     NODE    NODE    SYS     SYS     SYS     SYS     NODE    NODE    NODE     X      NODE    NODE
NIC4    NODE    NODE    PXB     NODE    SYS     SYS     SYS     SYS     NODE    NODE    NODE    NODE     X      NODE
NIC5    NODE    NODE    NODE    PXB     SYS     SYS     SYS     SYS     NODE    NODE    NODE    NODE    NODE     X
NIC6    SYS     SYS     SYS     SYS     PXB     NODE    NODE    NODE    SYS     SYS     SYS     SYS     SYS     SYS
NIC7    SYS     SYS     SYS     SYS     NODE    NODE    NODE    NODE    SYS     SYS     SYS     SYS     SYS     SYS
NIC8    SYS     SYS     SYS     SYS     NODE    NODE    NODE    NODE    SYS     SYS     SYS     SYS     SYS     SYS
NIC9    SYS     SYS     SYS     SYS     NODE    PXB     NODE    NODE    SYS     SYS     SYS     SYS     SYS     SYS
NIC10   SYS     SYS     SYS     SYS     NODE    NODE    PXB     NODE    SYS     SYS     SYS     SYS     SYS     SYS
NIC11   SYS     SYS     SYS     SYS     NODE    NODE    NODE    PXB     SYS     SYS     SYS     SYS     SYS     SYS

Legend:

  X    = Self
  SYS  = Connection traversing PCIe as well as the SMP interconnect between NUMA nodes (e.g., QPI/UPI)
  NODE = Connection traversing PCIe as well as the interconnect between PCIe Host Bridges within a NUMA node
  PHB  = Connection traversing PCIe as well as a PCIe Host Bridge (typically the CPU)
  PXB  = Connection traversing multiple PCIe bridges (without traversing the PCIe Host Bridge)
  PIX  = Connection traversing at most a single PCIe bridge
  NV#  = Connection traversing a bonded set of # NVLinks
```

Checking System Topology (B200)

`nvidia-smi topo –m`

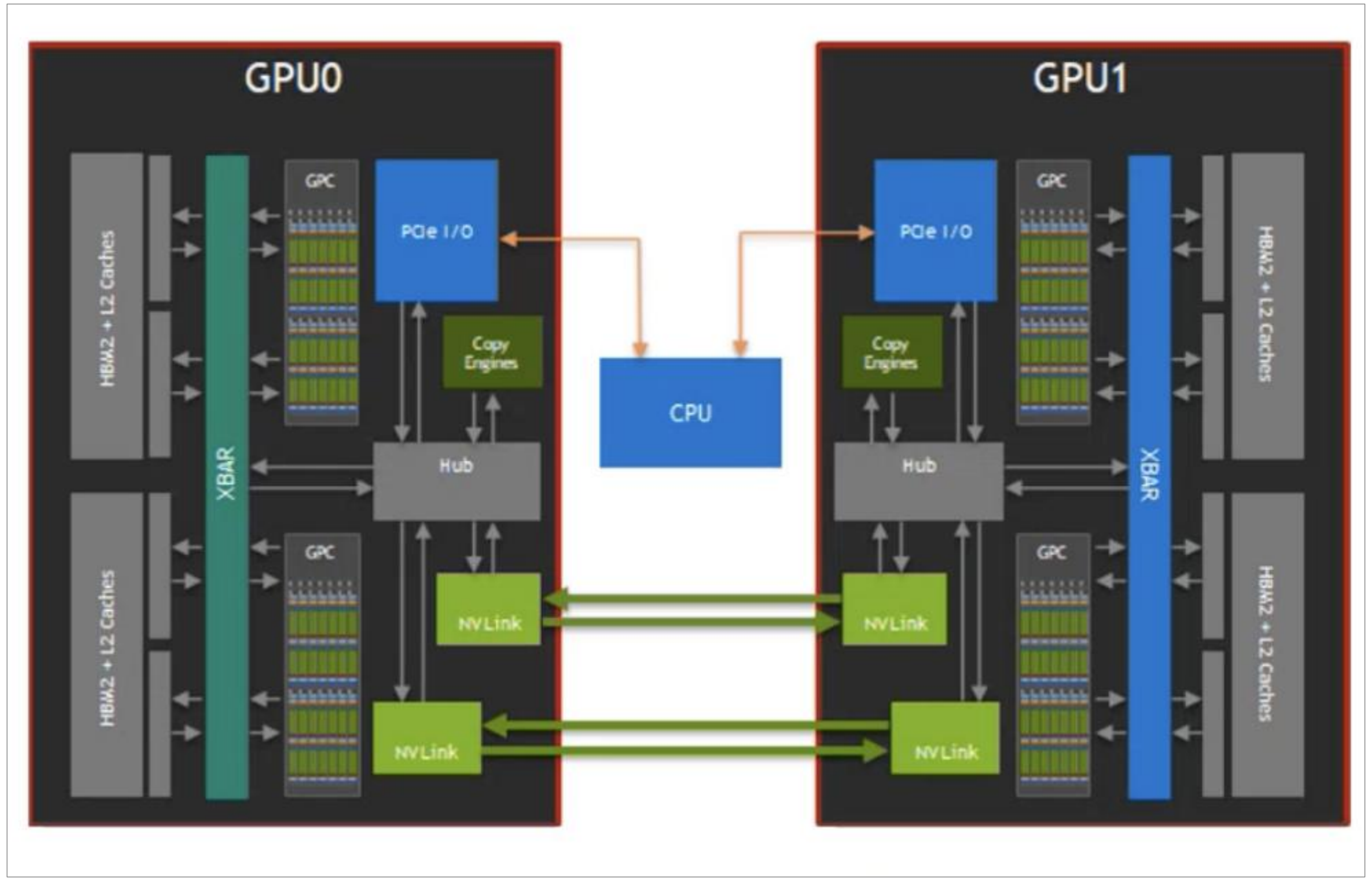


```
user@user:~$ nvidia-smi topo -m
        GPU0    GPU1    GPU2    GPU3    GPU4    GPU5    GPU6    GPU7    NIC0    NIC1    NIC2    NIC3    NIC4    NIC5
 NUMA ID
GPU0     X      NV18    NV18    NV18    NV18    NV18    NV18    NV18    NODE    NODE    NODE    NODE    PIX     NODE
GPU1    NV18     X      NV18    NV18    NV18    NV18    NV18    NV18    NODE    NODE    NODE    NODE    NODE    NODE
GPU2    NV18    NV18     X      NV18    NV18    NV18    NV18    NV18    SYS     SYS     SYS     SYS     SYS     SYS
GPU3    NV18    NV18    NV18     X      NV18    NV18    NV18    NV18    SYS     SYS     SYS     SYS     SYS     SYS
GPU4    NV18    NV18    NV18    NV18     X      NV18    NV18    NV18    SYS     SYS     SYS     SYS     SYS     SYS
GPU5    NV18    NV18    NV18    NV18    NV18     X      NV18    NV18    SYS     SYS     SYS     SYS     SYS     SYS
GPU6    NV18    NV18    NV18    NV18    NV18    NV18     X      NV18    SYS     SYS     SYS     SYS     SYS     SYS
GPU7    NV18    NV18    NV18    NV18    NV18    NV18    NV18     X      SYS     SYS     SYS     SYS     SYS     SYS
NIC0    NODE    NODE    SYS     SYS     SYS     SYS     SYS     SYS      X      PIX     PIX     PIX     NODE    NODE
NIC1    NODE    NODE    SYS     SYS     SYS     SYS     SYS     SYS     PIX      X      PIX     PIX     NODE    NODE
NIC2    NODE    NODE    SYS     SYS     SYS     SYS     SYS     SYS     PIX     PIX      X      PIX     NODE    NODE
NIC3    NODE    NODE    SYS     SYS     SYS     SYS     SYS     SYS     PIX     PIX     PIX      X      NODE    NODE
NIC4    PIX     NODE    SYS     SYS     SYS     SYS     SYS     SYS     NODE    NODE    NODE    NODE     X      NODE
NIC5    NODE    NODE    SYS     SYS     SYS     SYS     SYS     SYS     NODE    NODE    NODE    NODE    NODE     X
NIC6    NODE    NODE    SYS     SYS     SYS     SYS     SYS     SYS     NODE    NODE    NODE    NODE    NODE    PIX
NIC7    NODE    PIX     SYS     SYS     SYS     SYS     SYS     SYS     NODE    NODE    NODE    NODE    NODE    NODE
NIC8    SYS     SYS     PIX     NODE    SYS     SYS     SYS     SYS     SYS     SYS     SYS     SYS     SYS     SYS
NIC9    SYS     SYS     NODE    PIX     SYS     SYS     SYS     SYS     SYS     SYS     SYS     SYS     SYS     SYS
NIC10   SYS     SYS     SYS     SYS     PIX     NODE    SYS     SYS     SYS     SYS     SYS     SYS     SYS     SYS
NIC11   SYS     SYS     SYS     SYS     NODE    NODE    SYS     SYS     SYS     SYS     SYS     SYS     SYS     SYS
NIC12   SYS     SYS     SYS     SYS     NODE    NODE    SYS     SYS     SYS     SYS     SYS     SYS     SYS     SYS
NIC13   SYS     SYS     SYS     SYS     NODE    PIX     SYS     SYS     SYS     SYS     SYS     SYS     SYS     SYS
NIC14   SYS     SYS     SYS     SYS     SYS     SYS     PIX     NODE    SYS     SYS     SYS     SYS     SYS     SYS
NIC15   SYS     SYS     SYS     SYS     SYS     SYS     NODE    PIX     SYS     SYS     SYS     SYS     SYS     SYS
```

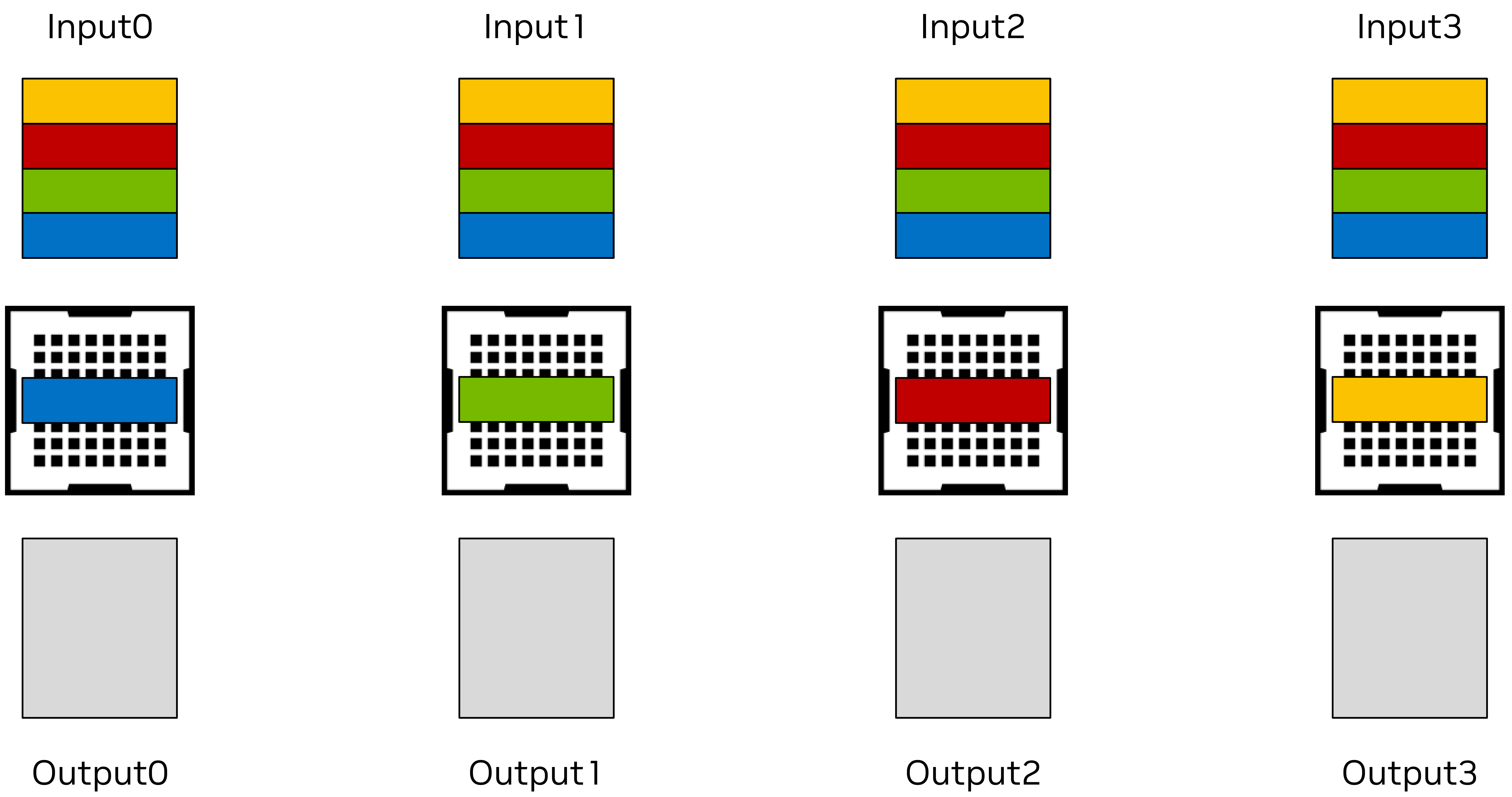
What is NVLINK?

GPU-to-GPU, CPU-to-GPU High Bandwidth Communication

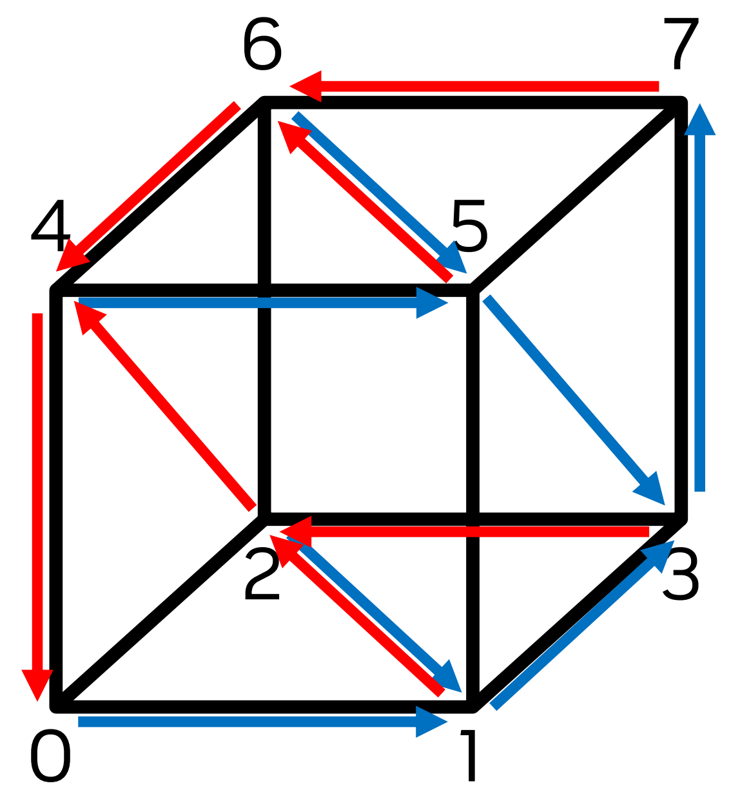
- NVLINK development start in 2013
- High speed interconnect technology enabling direct GPU-to-GPU communication, bypassing PCIe bottlenecks.
- NVLink allows faster data transfer, higher bandwidth, and lower latency between GPUs
- Supports various memory transactions
- Cacheable (coherent) / Non – cacheable (non-coherent) transaction support
- Parallelizable
- Unification of HBMs memories across a pool of GPUs
- Switchable



Ring Algorithm



Tree Algorithm



Tree #1

Tree #2

GPU A
Node 7

GPU A
Node 2

GPU A
Node 1

GPU A
Node 5

NIC A
Node 3

Input A

Input B

Input C

Input D

Output A

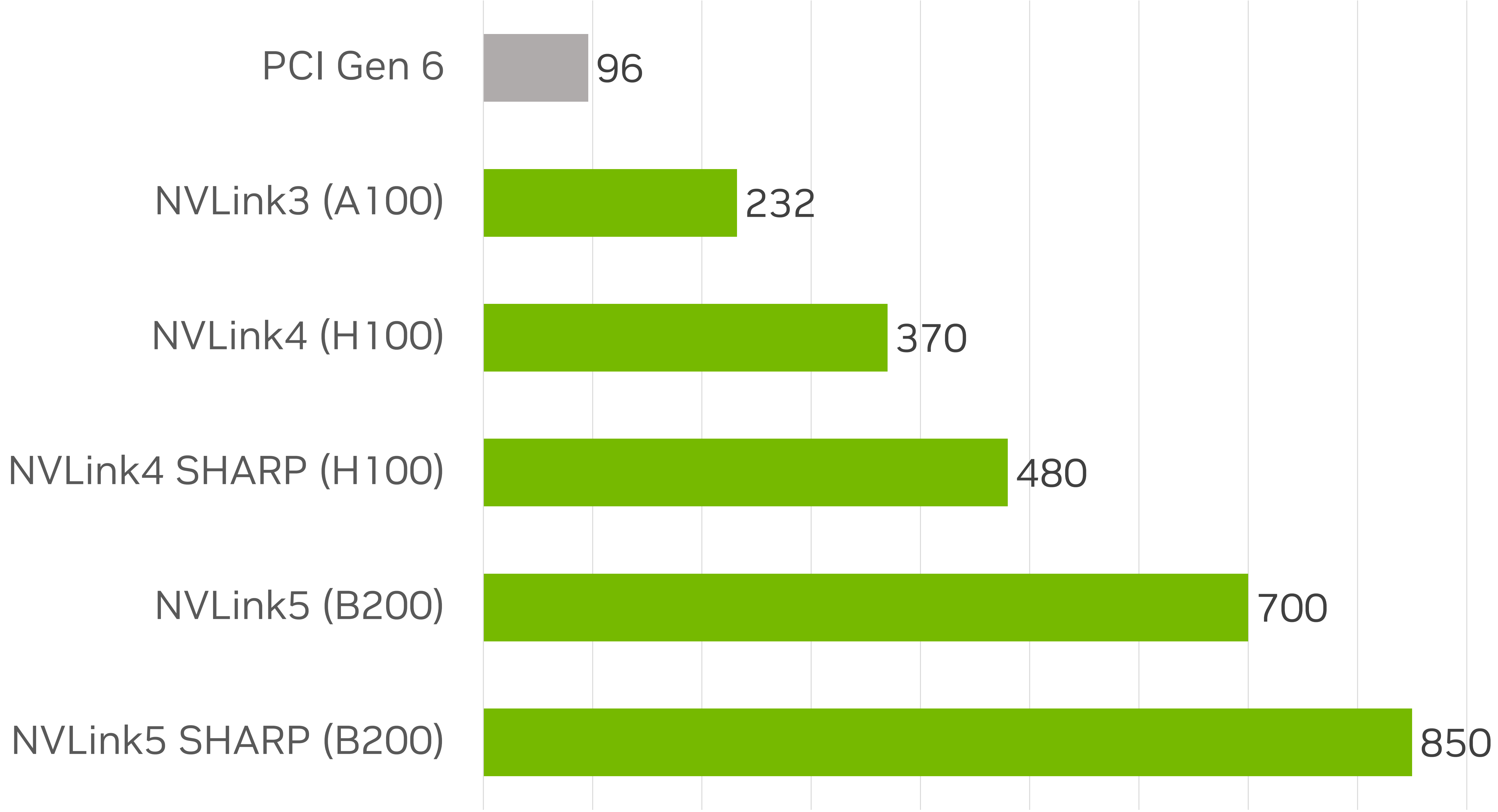
Output B

Output C

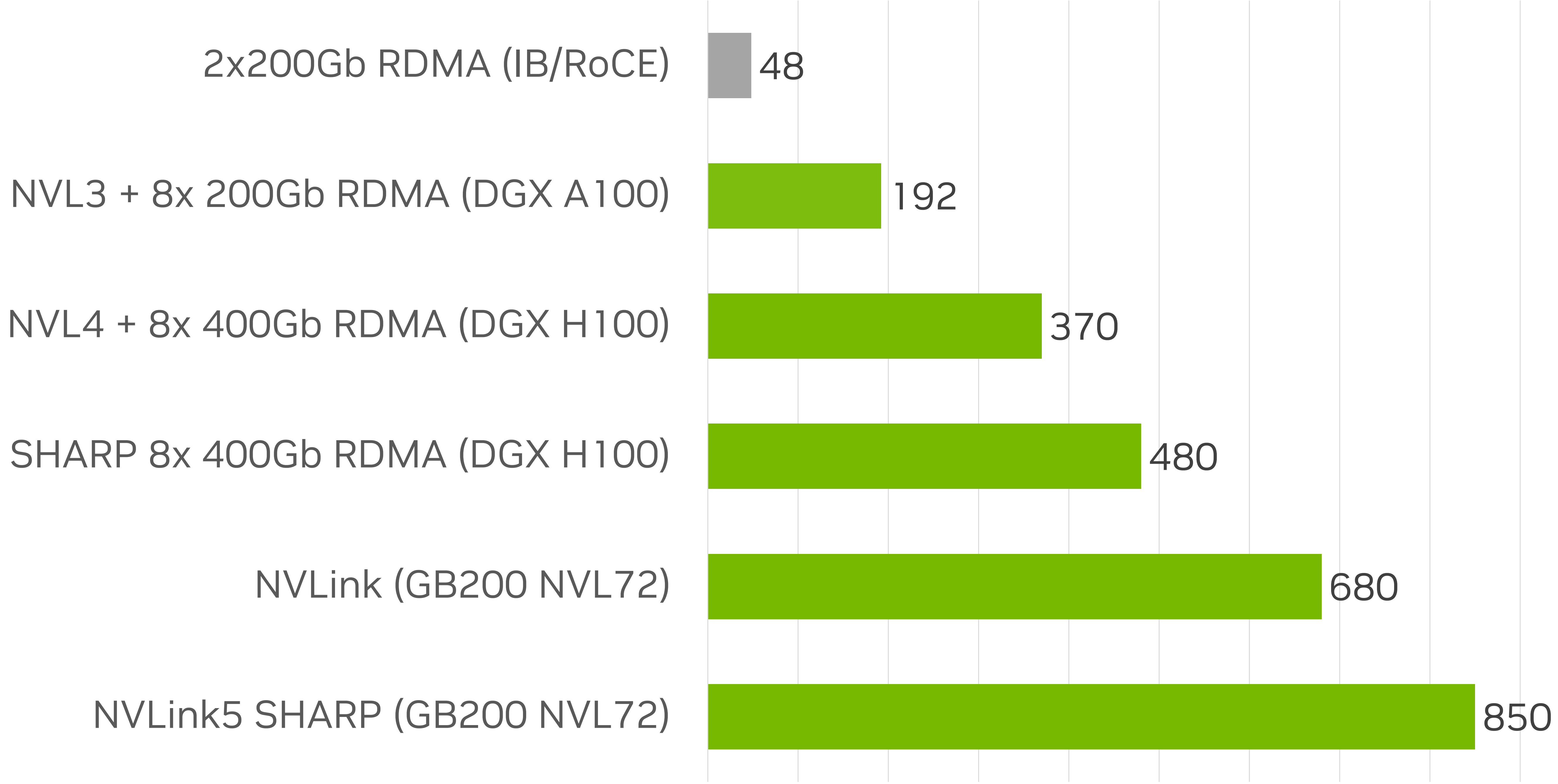
Output D

Collective Communication Bandwidth

Multi-GPU
NVLink
Multi-node
IB/RoCE
NVLink



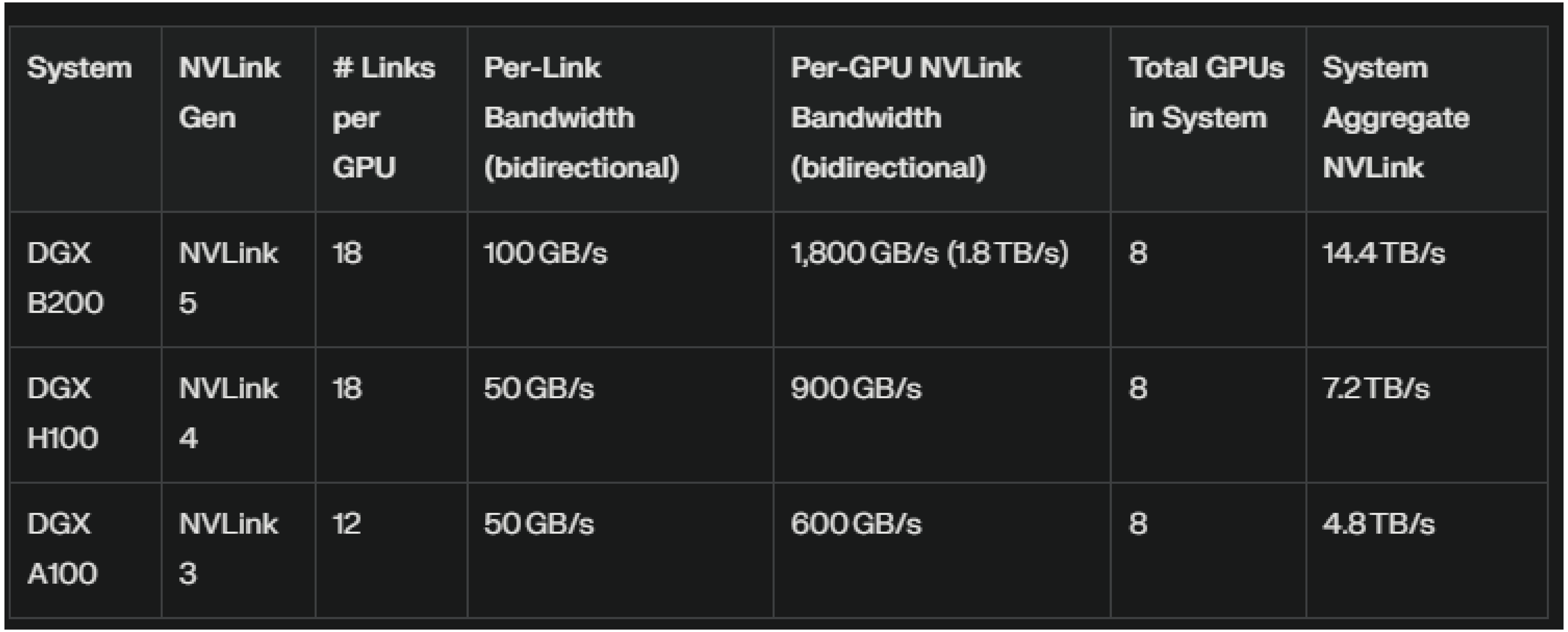
NCCL Tests Allreduce Bus Bandwidth in GB/s, 8 GPUs



NCCL Tests Allreduce Bus Bandwidth in GB/s, 32 GPUs

NVLink Evolution

Intra-node Connectivity



System	NVLink Gen	# Links per GPU	Per-Link Bandwidth (bidirectional)	Per-GPU NVLink Bandwidth (bidirectional)	Total GPUs in System	System Aggregate NVLink
DGX B200	NVLink 5	18	100 GB/s	1,800 GB/s (1.8 TB/s)	8	14.4 TB/s
DGX H100	NVLink 4	18	50 GB/s	900 GB/s	8	7.2 TB/s
DGX A100	NVLink 3	12	50 GB/s	600 GB/s	8	4.8 TB/s

Agenda

- Multi-GPU Computing in DL
- Hardware and Performance
- How-to-NCCL
- MLPerf Benchmarks
- HPL

NCCL

- The NVIDIA Collective Communications Library (NCCL, pronounced “Nickel”) is a library for inter-GPU communication.
- NCCL test is an open-source software to benchmark inter-GPU communication speed.
- When you run deep learning across multiple GPUs, you care about the communication speed among those GPUs.
- By running NCCL tests with various configs, you can check if your hardware can reach the designed performance for each config setting.

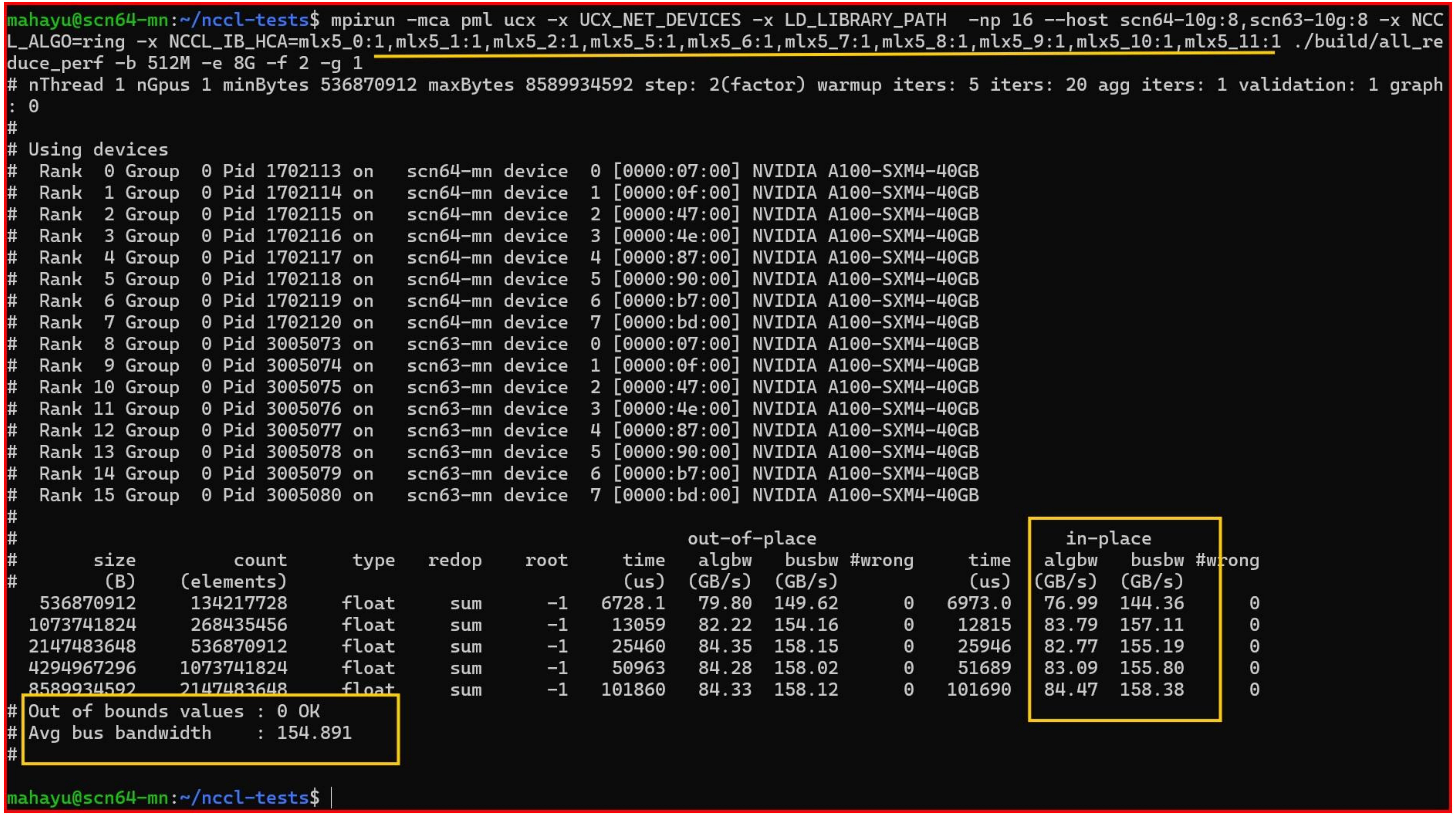
NCCL Output

A100 (Single Node)

```
mahayu@scn64-mn:~/nccl-tests$ mpirun -mca pml ucx -x UCX_NET_DEVICES -x LD_LIBRARY_PATH  -np 8 --host scn64-10g:8,scn63-10g:8 -x NCCL
_ALGO=ring -x NCCL_IB_HCA=mlx5_0:1,mlx5_1:1,mlx5_2:1,mlx5_5:1,mlx5_6:1,mlx5_7:1,mlx5_8:1,mlx5_9:1,mlx5_10:1,mlx5_11:1 ./build/all_red
uce_perf -b 512M -e 8G -f 2 -g 1
# nThread 1 nGpus 1 minBytes 536870912 maxBytes 8589934592 step: 2(factor) warmup iters: 5 iters: 20 agg iters: 1 validation: 1 graph
: 0
#
# Using devices
#  Rank  0 Group  0 Pid 1705509 on   scn64-mn device  0 [0000:07:00] NVIDIA A100-SXM4-40GB
#  Rank  1 Group  0 Pid 1705510 on   scn64-mn device  1 [0000:0f:00] NVIDIA A100-SXM4-40GB
#  Rank  2 Group  0 Pid 1705511 on   scn64-mn device  2 [0000:47:00] NVIDIA A100-SXM4-40GB
#  Rank  3 Group  0 Pid 1705512 on   scn64-mn device  3 [0000:4e:00] NVIDIA A100-SXM4-40GB
#  Rank  4 Group  0 Pid 1705513 on   scn64-mn device  4 [0000:87:00] NVIDIA A100-SXM4-40GB
#  Rank  5 Group  0 Pid 1705514 on   scn64-mn device  5 [0000:90:00] NVIDIA A100-SXM4-40GB
#  Rank  6 Group  0 Pid 1705515 on   scn64-mn device  6 [0000:b7:00] NVIDIA A100-SXM4-40GB
#  Rank  7 Group  0 Pid 1705516 on   scn64-mn device  7 [0000:bd:00] NVIDIA A100-SXM4-40GB
#
#                                                              out-of-place                       in-place
#       size         count      type   redop    root     time   algbw   busbw #wrong     time   algbw   busbw #wrong
#        (B)    (elements)                               (us)  (GB/s)  (GB/s)            (us)  (GB/s)  (GB/s)
   536870912     134217728     float     sum      -1   4275.0  125.59  219.77      0   4274.2  125.61  219.81      0
  1073741824     268435456     float     sum      -1   8293.4  129.47  226.57      0   8290.5  129.51  226.65      0
  2147483648     536870912     float     sum      -1    16420  130.78  228.87      0    16422  130.77  228.84      0
  4294967296    1073741824     float     sum      -1    32463  132.30  231.53      0    32459  132.32  231.56      0
  8589934592    2147483648     float     sum      -1    64660  132.85  232.48      0    64777  132.61  232.06      0
# Out of bounds values : 0 OK
# Avg bus bandwidth    : 227.815
#
```

NCCL Output

A100 (Multi Node)



```
mahayu@scn64-mn:~/nccl-tests$ mpirun -mca pml ucx -x UCX_NET_DEVICES -x LD_LIBRARY_PATH  -np 16 --host scn64-10g:8,scn63-10g:8 -x NCC
L_ALGO=ring -x NCCL_IB_HCA=mlx5_0:1,mlx5_1:1,mlx5_2:1,mlx5_5:1,mlx5_6:1,mlx5_7:1,mlx5_8:1,mlx5_9:1,mlx5_10:1,mlx5_11:1 ./build/all_re
duce_perf -b 512M -e 8G -f 2 -g 1
# nThread 1 nGpus 1 minBytes 536870912 maxBytes 8589934592 step: 2(factor) warmup iters: 5 iters: 20 agg iters: 1 validation: 1 graph
: 0
#
# Using devices
#  Rank  0 Group  0 Pid 1702113 on   scn64-mn device  0 [0000:07:00] NVIDIA A100-SXM4-40GB
#  Rank  1 Group  0 Pid 1702114 on   scn64-mn device  1 [0000:0f:00] NVIDIA A100-SXM4-40GB
#  Rank  2 Group  0 Pid 1702115 on   scn64-mn device  2 [0000:47:00] NVIDIA A100-SXM4-40GB
#  Rank  3 Group  0 Pid 1702116 on   scn64-mn device  3 [0000:4e:00] NVIDIA A100-SXM4-40GB
#  Rank  4 Group  0 Pid 1702117 on   scn64-mn device  4 [0000:87:00] NVIDIA A100-SXM4-40GB
#  Rank  5 Group  0 Pid 1702118 on   scn64-mn device  5 [0000:90:00] NVIDIA A100-SXM4-40GB
#  Rank  6 Group  0 Pid 1702119 on   scn64-mn device  6 [0000:b7:00] NVIDIA A100-SXM4-40GB
#  Rank  7 Group  0 Pid 1702120 on   scn64-mn device  7 [0000:bd:00] NVIDIA A100-SXM4-40GB
#  Rank  8 Group  0 Pid 3005073 on   scn63-mn device  0 [0000:07:00] NVIDIA A100-SXM4-40GB
#  Rank  9 Group  0 Pid 3005074 on   scn63-mn device  1 [0000:0f:00] NVIDIA A100-SXM4-40GB
#  Rank 10 Group  0 Pid 3005075 on   scn63-mn device  2 [0000:47:00] NVIDIA A100-SXM4-40GB
#  Rank 11 Group  0 Pid 3005076 on   scn63-mn device  3 [0000:4e:00] NVIDIA A100-SXM4-40GB
#  Rank 12 Group  0 Pid 3005077 on   scn63-mn device  4 [0000:87:00] NVIDIA A100-SXM4-40GB
#  Rank 13 Group  0 Pid 3005078 on   scn63-mn device  5 [0000:90:00] NVIDIA A100-SXM4-40GB
#  Rank 14 Group  0 Pid 3005079 on   scn63-mn device  6 [0000:b7:00] NVIDIA A100-SXM4-40GB
#  Rank 15 Group  0 Pid 3005080 on   scn63-mn device  7 [0000:bd:00] NVIDIA A100-SXM4-40GB
#
#                                                       out-of-place                       in-place
#       size         count      type   redop    root     time   algbw   busbw #wrong     time   algbw   busbw #wrong
#        (B)    (elements)                               (us)  (GB/s)  (GB/s)            (us)  (GB/s)  (GB/s)
   536870912     134217728     float     sum      -1   6728.1   79.80  149.62      0   6973.0   76.99  144.36      0
  1073741824     268435456     float     sum      -1    13059   82.22  154.16      0    12815   83.79  157.11      0
  2147483648     536870912     float     sum      -1    25460   84.35  158.15      0    25946   82.77  155.19      0
  4294967296    1073741824     float     sum      -1    50963   84.28  158.02      0    51689   83.09  155.80      0
  8589934592    2147483648     float     sum      -1   101860   84.33  158.12      0   101690   84.47  158.38      0
# Out of bounds values : 0 OK
# Avg bus bandwidth    : 154.891
#

mahayu@scn64-mn:~/nccl-tests$ 
```

NCCL Interpretation

- **Operation Time** - NCCL tests report the average time (in milliseconds) it takes to complete a collective operation
- **Algorithm Bandwidth** (algbw) - How much data (in GB) is being processed per second by the algorithm. For point-to-point operations (like Send/Receive), this is meaningful and directly reflects throughput.
- **Bus Bandwidth** (busbw) - It adjusts the algorithm bandwidth to reflect the actual hardware bottleneck (e.g., NVLink, PCIe, network), making it possible to compare results regardless of the number of ranks.
- **Verify NCCL** results by finding peak theoretical bandwidth for
  - Intra-node: NVLink
  - Inter-node: Infiniband/Connect-X Ethernet
- Run NCCL using slurm or mpirun

NVIDIA