Grounds-up LLM Development

Ayush Maheshwari
Sr. Solutions Architect, NVIDIA

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

ayushbits.github.io

Sessions

1. Cluster health-check using NCCL, MLPerf, HPL **(1 hour)**
   a) Understand the hardware and its performance on multiple GPUs.
   b) Ensure that your training performance aligns with the h/w benchmarks
   c) Evaluate the cluster to ensure platform fits within your needs.
2. Large scale data curation for LLM training **(1 hour)**
   a) Deep-dive into aspects of data curation
   b) Mixed-precision training
3. Distributed and stable LLM training on a large-scale cluster **(1.5 hour)**
   a) Parallelism techniques
   b) Frameworks and wrappers
   c) Recipes and best practices
4. Post-training and evaluation of pre-trained LLM **(1 hour)**
   a) Sync between training data and expected performance
   b) Algorithms and frameworks
5. Fine-tuning and deployment **(1 hour)**
   a) Dynamic and static batching, state management, inference server
   b) Best practices for optimizing model

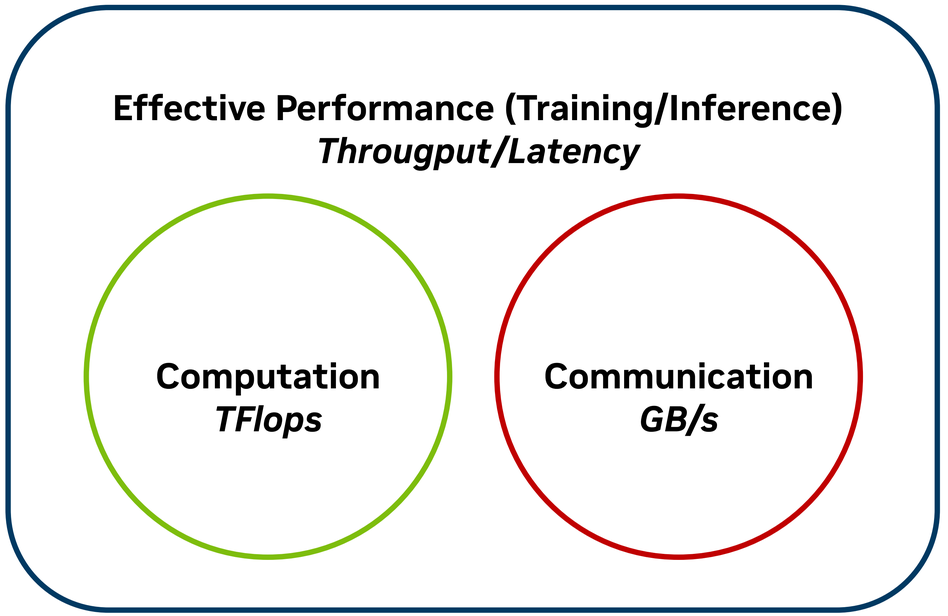
Cluster Health Check using NCCL, MLPerf and HPL

Ayush Maheshwari

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

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.

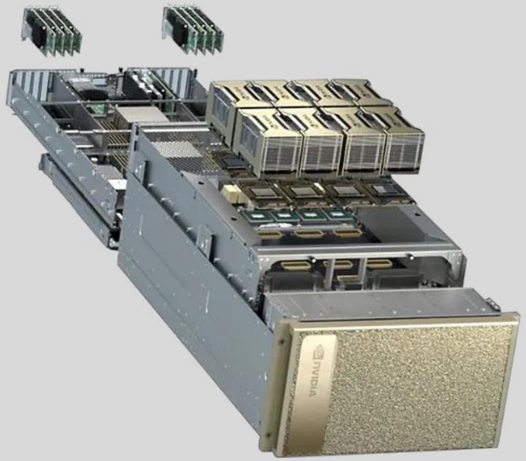


Multi-GPU Computing

NCCL: NVIDIA Collective Communication Library

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

PCI Server



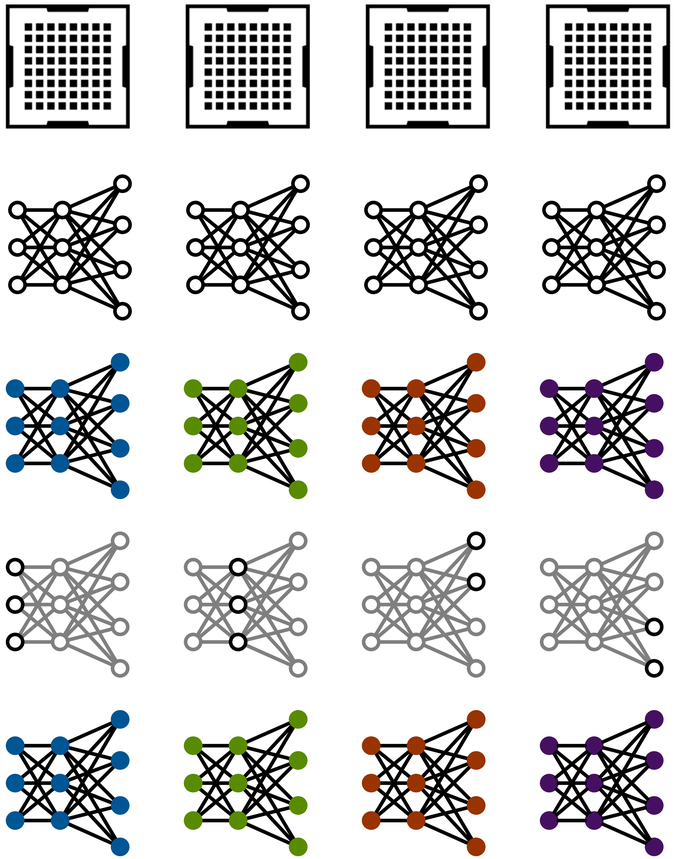
DGX/HGX

Large systems

Agenda

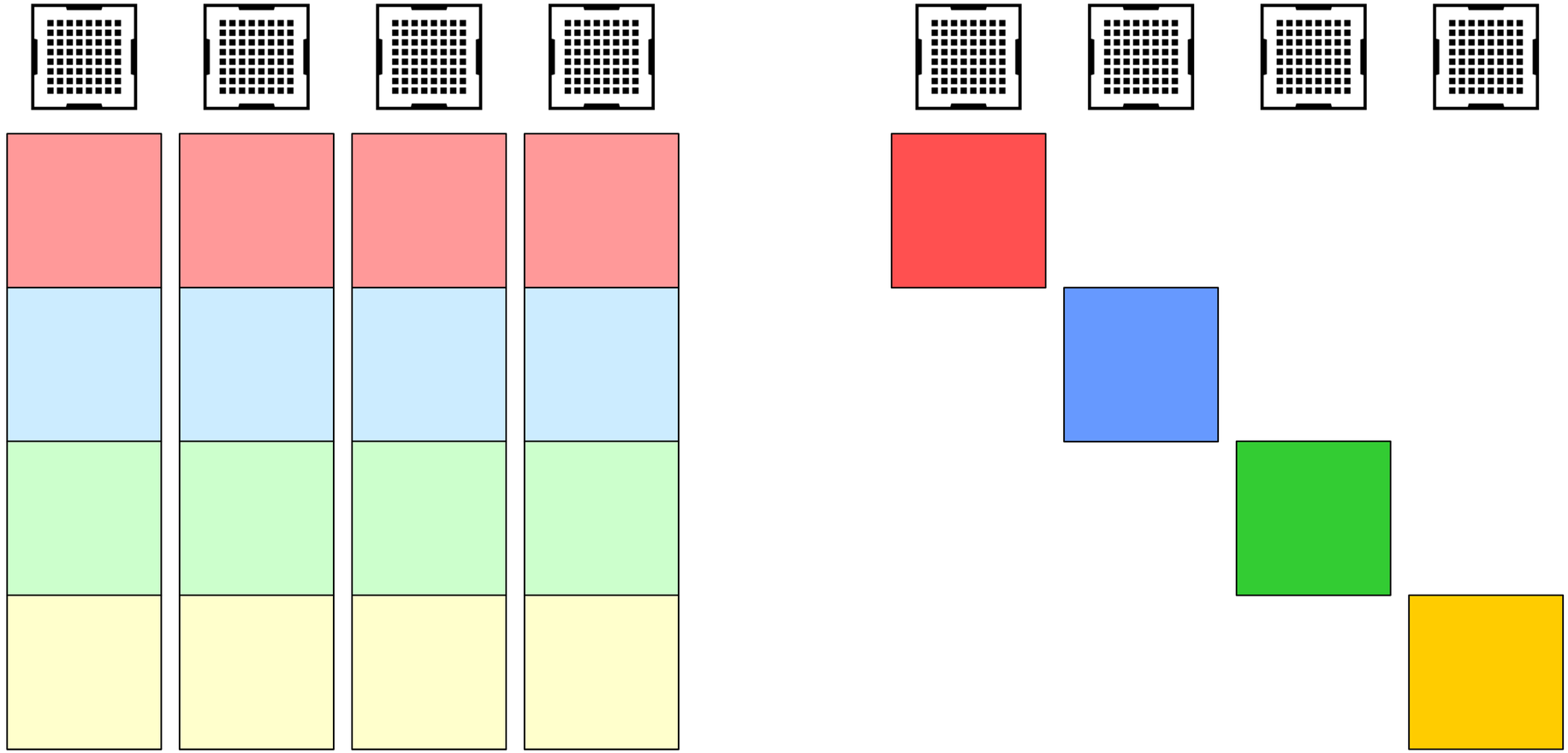
- Multi-GPU Computing in DL
- Hardware and Performance
- ML Perf Benchmarks
- HPL for effective GEMM
- Summary

Multi-GPU Computing in DL



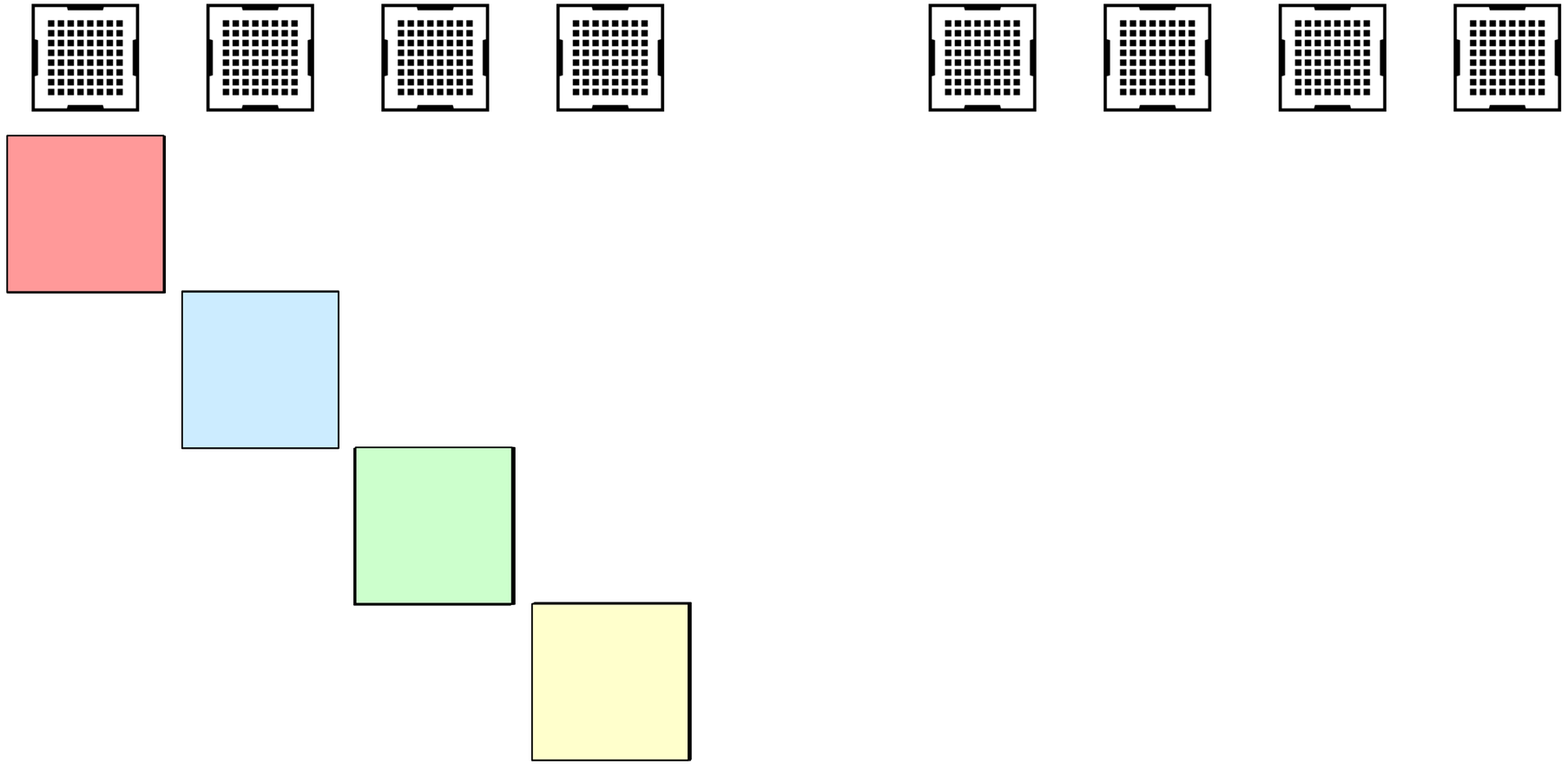
Data Parallelism / FSDP	All-reduce, all-gather, reduce-scatter
Tensor Parallelism	All-reduce, all-gather, reduce-scatter
Pipeline Parallelism	Send / receive
Expert Parallelism	All-to-all

Communication primitives



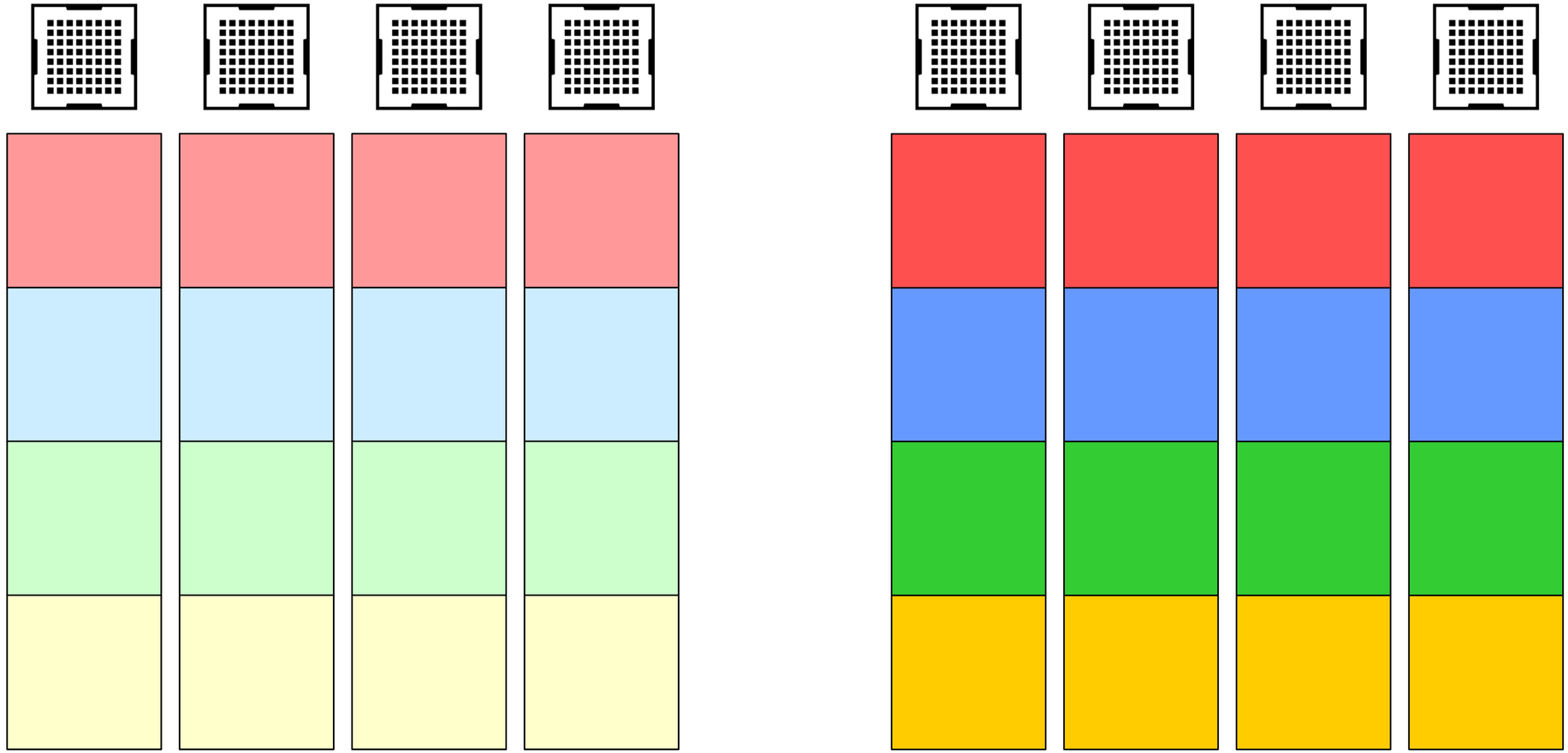
Reduce-scatter

Communication primitives



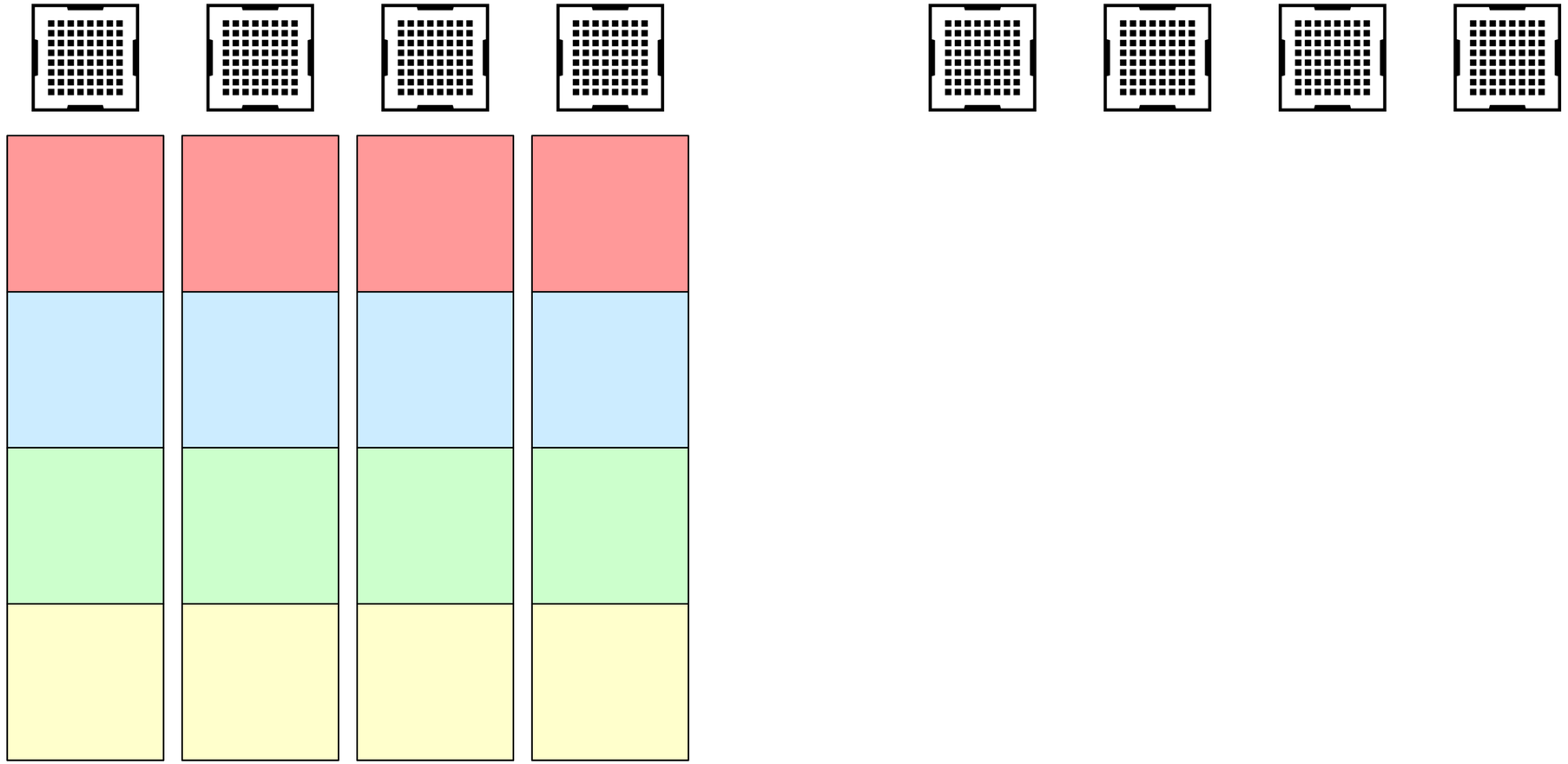
All-gather

Communication primitives



All-reduce

Communication primitives



All-to-all

Agenda

- Multi-GPU Computing in DL
- Hardware and Performance
- System design
- Recent improvements
- Future

Checking System Topology (A100)

`nvidia-smi topo –m`

```
mahayu@scp64-mp:~/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)

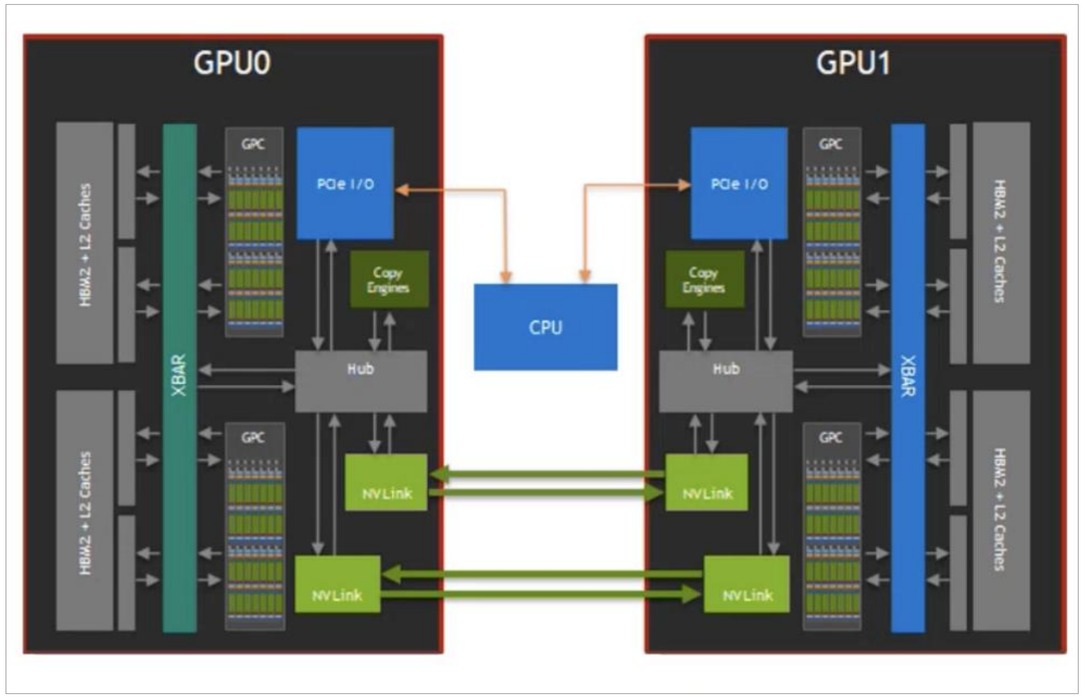
`nvidia-smi topo –m`

```
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
```

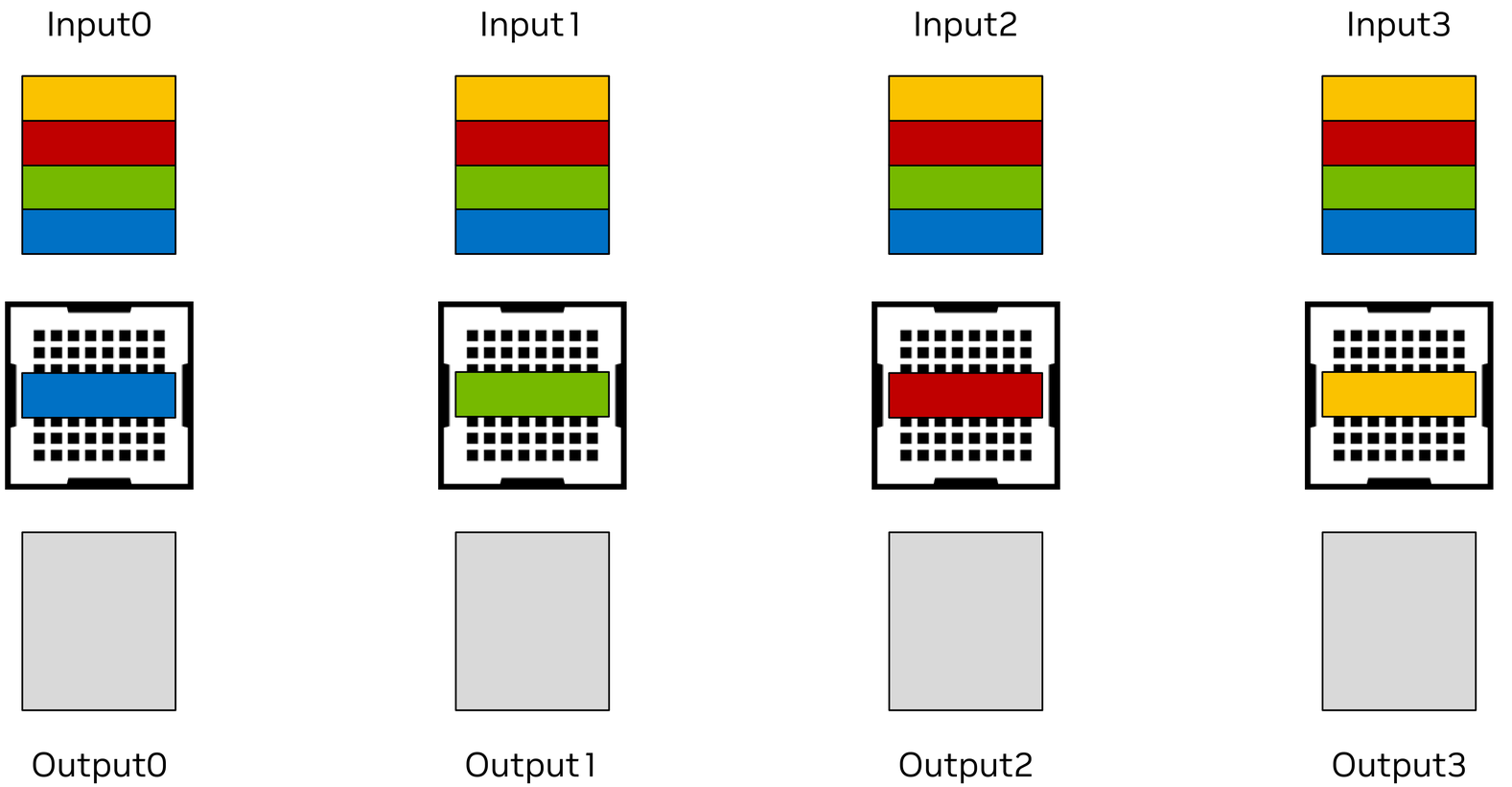
What is NVLINK?

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

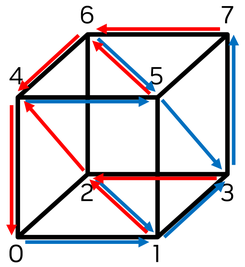
- NVLINK development start in 2013
- High speed interconnect technology enabling direct GPU-to-GPU communication, bypassing PCIe bottlenecks.
- NVLink allows faster data transfer, higher bandwidth, and lower latency between GPUs
- Supports various memory transactions
- Cacheable (coherent) / Non – cacheable (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

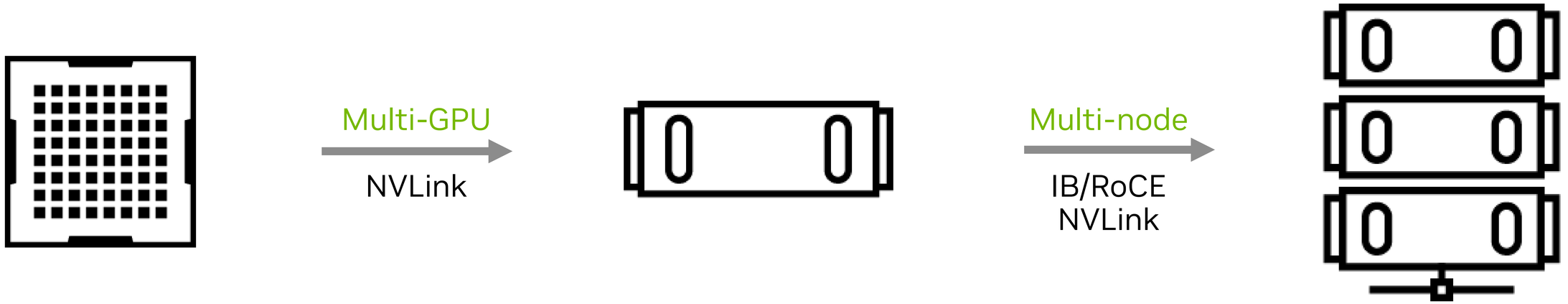
GPU A
Node 5

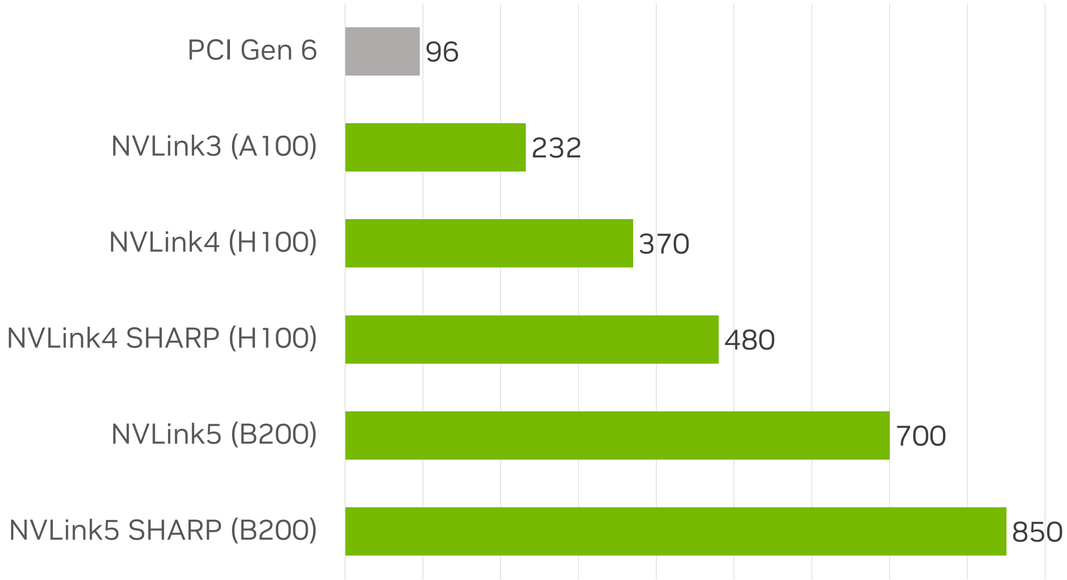
NIC A
Node 3

Input A Input B Input C Input D

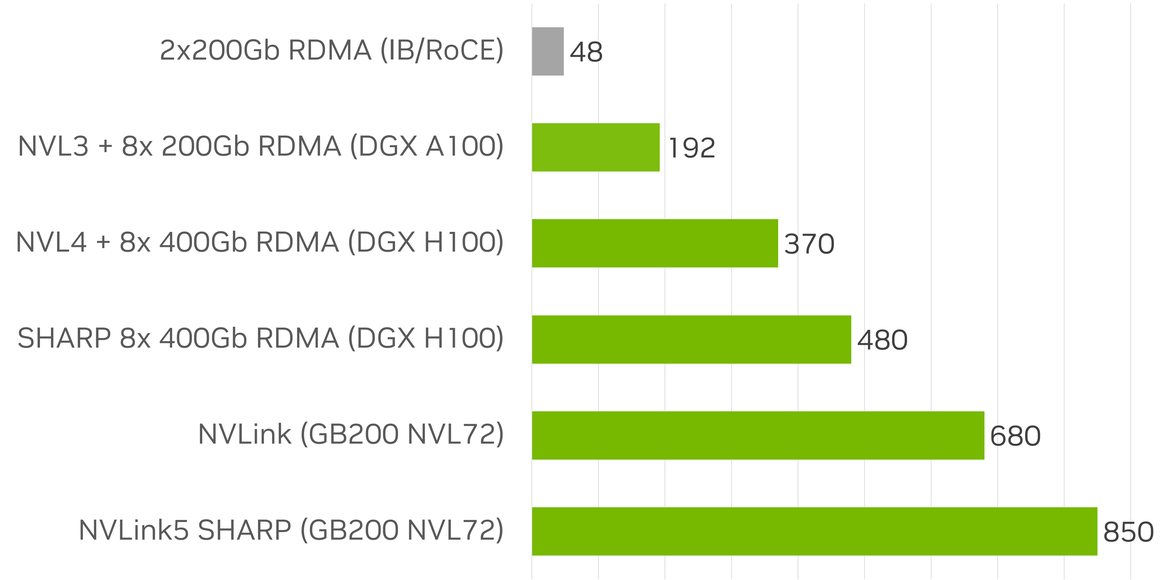
Output A Output B Output C Output D

Collective Communication Bandwidth





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

NCCL Demo

Agenda

- Multi-GPU Computing in DL
- Hardware and Performance
- **ML Perf Benchmarks**
- HPL for effective GEMM
- Summary

What is MLPerf?

Unbiased AI benchmarks negotiated by a consortium

MLPerf™ benchmarks—developed by MLCommons, a consortium of AI leaders from academia, research labs, and industry—are designed to provide unbiased evaluations of training and inference performance for hardware, software, and services. They are all conducted under prescribed conditions.

How-to-MLPerf

https://docs.mlcommons.org/inference/

- MLPerf v5.0 (latest) for training and inference
- Reported Benchmark Performance - https://mlcommons.org/benchmarks/inference-datacenter/
- Running Inference – *git clone* https://github.com/mlcommons/inference_results_v5.0
- Requirements for running NVIDIA reported benchmarks on DGX systems
  - TRT-LLM – Generate optimized TRT engine

Vision	Image classification	Server, Offline
Vision	Object detection	Server, Offline
Vision	Medical image segmentation	Offline
Speech	Speech-to-text	Server, Offline
Language	Language processing	Server, Offline
Language	Summarization	Server, Offline
Language	Question Answering	Server, Offline
Commerce	Recommendation	Server, Offline
Image generation	Text-to-image	Server, Offline
Graph	Node Classification	Offline

MLCommons

https://mlcommons.org/benchmarks/inference-datacenter/

ML Commons

Division/Power	Round	Organization	System Name	Accelerator	Processor	Software	MLC Model	# of Accelerators	Availability	Scenario
Closed	v5.0	(All)	(All)	NVIDIA B200-S...	(All)	TensorRT 10.8 ...	(All)	(All)	(All)	(All)

MLPerf Results

MLCommons data as of: 3/25/2025 10:49:49 PM

Benchmark: **Inference**
System Type: **Datacenter**
Division/Power: **Closed**
Availability: **All**
Round: **v5.0**

								Benchmark / Model MLC / Scenario / Units									
								Inference									
								llama2-70b-interactive-99		llama2-70b-interactive-99.9		llama3.1-405b		mixtral-8x7b		stable-diffusion-xl	
Public ID	Organization	Availability	System Name (click + for details)	# of Nodes	Processor	Accelerator	# of Accelerators	rver ns/s	Offline Samples/s	Server Queries/s	Offline Samples/s	Server Queries/s	Offline Samples/s	Server Queries/s	Offline Tokens/s	Server Tokens/s	Offline Samples/s	Server Queries/s
5.0-0055	NVIDIA	available	NVIDIA DGX B200 (1x B200-SXM-180GB ..	1	Intel(R) Xeon(R) Platinum 85..	NVIDIA B200-SXM-180..	1	,078.40										
5.0-0056	NVIDIA	available	NVIDIA DGX B200 (8x B200-SXM-180GB ..	1	Intel(R) Xeon(R) Platinum 85..	NVIDIA B200-SXM-180..	8	,443.30	98,858.00	59,409.30	98,858.00	59,409.30	1,526.29	845.82	128,148.00	126,845.00	30.38	28.44
5.0-0072	Supermicro	available	SYS-421GE-NBRT-LCC (8x B200-SXM-1..	1	Intel(R) Xeon(R) Platinum 85..	NVIDIA B200-SXM-180..	8	,629.84	93,857.00	59,505.30	93,857.00	59,505.30	1,538.17	1,057.52	128,795.00	129,047.00		
5.0-0074	Supermicro	available	SYS-A21GE-NBRT (8x B200-SXM-180GB..	1	Intel(R) Xeon(R) Platinum 85..	NVIDIA B200-SXM-180..	8	,552.14	92,689.20	62,265.70	92,689.20	62,265.70	1,521.74	1,080.31	125,534.00	128,961.00	30.34	28.92
5.0-0078	Google	preview	NVIDIA DGX B200 (8x B200-SXM-180GB ..	1	Intel(R) Xeon(R) Platinum 85..	NVIDIA B200-SXM-180..	8	,203.67					1,498.56	847.48	121,489.00	122,411.00	30.31	28.45

Agenda

- Multi-GPU Computing in DL
- Hardware and Performance
- ML Perf Benchmarks
- **HPL for effective GEMM**
- Summary

HPL Algorithm Overview

What is HPL?

- HPL is a software package that solves a dense linear system of equations in double precision(64-bit) arithmetic on distributed memory computers
- HPL is an implementation of the Linpack code – which measures the single machine's performance. Thus, HPL implements this at scale
- The goal is to measure the system's rate of execution of 64-bit Floating point arithmetic
- HPL result is a single number in Floating Point Operations Per Second (FLOPS)

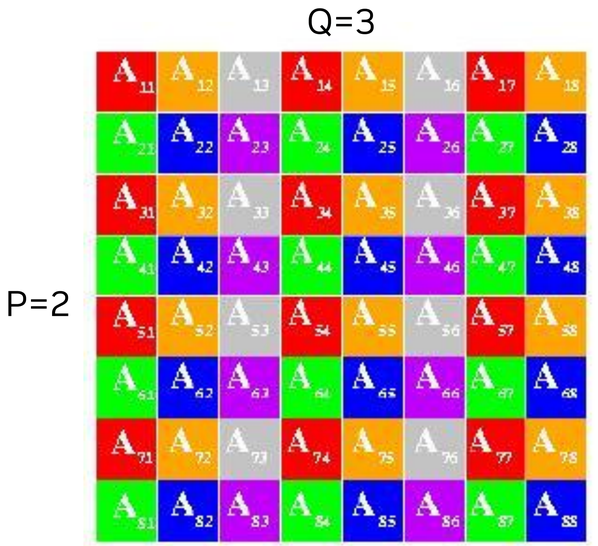
HPL Algorithm Overview

- Like in reference HPL (+other implementations) matrix is 2D block cyclic.
  - We typically prefer larger matrix blocks, i.e. NB = O(1000).

- The main computation is now **entirely on the GPU**:
  - Various kernels (factorization, NCCL comm, GEMM, TRSM share SM resources)

- We use a variety of communication libraries – NCCL/MPI/NVSHMEM.
  - For some comms operations, NCCL is the only library used (e.g. panel broadcast).
  - For other operations, there are several options (tuning for best perf depending on scale)

- Parameter file only used to specify N/NB/P/Q/PMAP
  - Most ‘advanced’ tuning done by environment variables.



HPL Performance

H100

```
================================================================================
T/V                N    NB     P     Q               Time                 Gflops (   per GPU)
--------------------------------------------------------------------------------
WC0           264192  1024     4     2              33.36          3.685e+05 ( 4.606e+04)
```

```
================================================================================
T/V                N    NB     P     Q               Time                 Gflops (   per GPU)
--------------------------------------------------------------------------------
WC0            92160  1024     1     1              11.08          4.712e+04 ( 4.712e+04)
```

HPL Performance

A100

```
================================================================================
T/V                N    NB     P     Q               Time                 Gflops (   per GPU)
--------------------------------------------------------------------------------
WC0           264192  1024     4     2             106.06          1.159e+05 ( 1.449e+04)
```

```
================================================================================
T/V                N    NB     P     Q               Time                 Gflops (   per GPU)
--------------------------------------------------------------------------------
WC0            92160  1024     1     1              30.32          1.721e+04 ( 1.721e+04)
```

Compare HPL Performance

- Refer datasheet of respective GPUs to see peak theoretical TFlops
- https://resources.nvidia.com/en-us-gpu-resources/h100-datasheet-24306

	H100 SXM	H100 NVL
FP64	34 teraFLOPS	30 teraFLOPS
FP64 Tensor Core	67 teraFLOPS	60 teraFLOPS
FP32	67 teraFLOPS	60 teraFLOPS
TF32 Tensor Core*	989 teraFLOPS	835 teraFLOPS

	A100 40GB PCIe	A100 80GB PCIe	A100 40GB SXM	A100 80GB SXM
FP64	**9.7 TFLOPS**			
FP64 Tensor Core	**19.5 TFLOPS**			
FP32	**19.5 TFLOPS**			
Tensor Float 32 (TF32)	**156 TFLOPS | 312 TFLOPS***			

Summary

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

- NCCL test for analysing intercommunication within and among multiple servers.
  - Essential for optimizing distributed training throughput and scalability.
- MLPerf: Industry-standard benchmark for end-to-end machine learning workloads.
  - Evaluates real-world AI training and inference efficiency across platforms.
- HPL : Assesses peak floating-point computing power (FLOPS) of CPU and GPU clusters.
  - Validates overall system performance for high-performance computing tasks.