

STUDENT CLUSTER COMPETITION

SCC Connect 26 Benchmarking

What we measure, how it is graded, and what belongs in your report.

HPL

STREAM

NCCL

FIO · OPTIONAL



Floating point operations per second of the CPU

◆ CPUs, not GPUs.

The graded result comes from the CPUs. A GPU run is welcome, but optional and ungraded.

◆ Compute theoretical peak first.

Check real frequencies in `/proc/cpuinfo` under load. Account for architecture and turbo boosting.

◆ Then build, then run.

Your choice of BLAS and MPI. Explain both choices.

YOUR INSTANCE

- ◆ Intel Xeon Gold 6548Y+
- ◆ 64 physical cores, not vCPUs
- ◆ A full, non-shared node

What you may use, and what you must declare

◆ Vendor builds are allowed.

The NGC container `nvidia/hpc-benchmarks:26.02.01` is public and fair game.

◆ Your own implementation is allowed.

Publish it to a public GitHub repo and include the link.

◆ Documented environment variables are fine

so long as the run passes the HPL residual check.

◆ Always cite your source.

Say which build you used and where you got it.

IN THE REPORT

- ◆ Peak FLOP/s calculation and methodology
- ◆ BLAS choice and rationale
- ◆ MPI implementation
- ◆ Your `HPL.DAT`
- ◆ Not within **10%** of peak? Account for it.

Measuring memory bandwidth

- ◆ **Four kernels, every time.**

Copy, Scale, Add and Triad. Report all four.

- ◆ **Size the array against cache.**

A working set that fits in cache measures cache, not memory.

- ◆ **Include your build scripts**

and any modification you made to the source.

ALSO USEFUL

STREAM is the quickest way to confirm a suspect instance. If a node looks slow, this is the evidence you bring to LaunchPad admins.

What to report

- ◆ **The array size you used**
and how it compares to total cache.
- ◆ **Results for all four kernels.**
- ◆ **Name the kernel you quote**
when you state a single bandwidth number.

AND AS ALWAYS

- ◆ Complete, unedited output
- ◆ How many runs, and the spread
- ◆ A single run is a data point, not a result

The path between the GPUs

- ◆ **Topology before measurement.**

`nvidia-smi topo -m` and `nvlink --status`. The `NV#` link count sets the peak you measure against.

- ◆ **Sweep message sizes**

across all four collectives, over the same range, so they stay comparable.

- ◆ **Two ends, two numbers.**

Latency at the small end, bandwidth at the large end.

THE FOUR COLLECTIVES

- ◆ AllReduce
- ◆ AllGather
- ◆ ReduceScatter
- ◆ Broadcast

How it is judged

- ◆ Best busbw at large message sizes.

- ◆ Latency at small message sizes.

- ◆ Your tuning writeup.

What you changed on the instance and which NCCL flags you passed. A fast number with no account of how you got it scores lower.

TWO REQUIREMENTS TO KEEP IN MIND

- ◆ Quote **busbw**, never `algbw`
- ◆ Report the **out-of-place** measurement only

The path to storage

- ◆ **Optional and unassessed.**

No benefit and no penalty. Run it if a job is slow while the CPUs and GPUs sit idle.

- ◆ **Four patterns.**

Sequential read and write at 1M for bandwidth. Random read and write at 4k for IOPS.

- ◆ **Never point FIO at a raw device.**

Write to a file in a directory you created.

DEFEAT THE PAGE CACHE

- ◆ `direct=1` in the job file

- ◆ Working set larger than RAM

- ◆ Drop caches between runs

A read far above the drive's rated speed is a cached result, not a good one.

What to report

- ◆ **Bandwidth and IOPS separately.**

They answer different questions. Do not collapse them into one number.

- ◆ **Describe the storage properly.**

Device type, local or network attached, RAID and its level, filesystem, mount options.

- ◆ **The complete job file for every run.**

PROVE THE CACHE WAS BYPASSED

- ◆ How you bypassed it
- ◆ Working set size against total system RAM

One PDF, five sections, results for HPL, STREAM, and NCCL

◆ 1 • Identification

Team, affiliation, which benchmarks you cover.

◆ 2 • Environment

One section, explicit about what you were given.

◆ 3 • Binary and tuning

Build scripts, config files, every variable you set and why.

◆ 4 • Results

Complete unedited output, run count and spread, a plot where there is a curve.

◆ 5 • Performance analysis

Your number against theoretical peak, and the gap explained.

ON THE INSTANCE

Assemble a single directory named for your team, one subdirectory per benchmark holding the build script and every results file, with the report PDF at the top level.

The report is most of your score

13%

HPL

13%

STREAM

13%

NCCL

61%

The written report

- ◆ **Spend real time on the writeup.**
Three good numbers with a thin report will not score well.
- ◆ **Your best score is the one that counts**
but include the output of every run you mention.

ALLOWED

- ◆ Any hardware tuning LaunchPad permits, including locked clocks
- ◆ LLM assistance, if you verify the output

A blue four-pointed star with a white outline, located in the upper left corner of the slide.

OVER TO YOU

Questions?

Anything unclear in the exercises, the grading, or the report, ask now. Check the Google Group discussions for other questions and responses as they come in.