


Computing Course

Roel Aaij

June 22nd, 2022

Nik|hef



Single Slide Overview

- Stoomboot
 - [CT Wiki](#)
 - [Computing Course Documentation](#)
- Exercise scripts
 - `/data/datagrid/raaij/computing_course`

Interactive nodes

stbc-i1, stbc-i2	CPU-only
stbc-g1	1 x 1080Ti
stbc-g2, wn-lot-008	1 x GV100, 2 x V100
wn-lot-001	2 x MI50

Batch queues

generic, long, etc	2000 cores
gpu-nv (1080ti)	2 x 1 x 1080Ti
gpu-nv (v100)	1 x 2 x V100
gpu-amd	6 x 2 x MI50

Scaling

- Submission overhead exists!
- Reading data is generally OK
- Writing can be trickier:
 - Writing lots of small files is bad
 - At 10s of MB output per job, you have to start thinking
 - Write temporary files to “\$TMPDIR”
 - Write the (large) output files to “/dcache”
 - Send stbc-admin@nikhef.nl a mail to request access
- O(hours) of job run time is best
- Try a few jobs on the interactive node
- Submit a few 10s of jobs to try if that works
- Submit the full run
- If things seem slow or you have questions, ask us!

Demo/Exercise

- Stoomboot
 - [CT Wiki](#)
 - [Computing Course Documentation](#)
- Exercise scripts
 - `/data/datagrid/raaij/computing_course`
 - `toy.py` does the work, try it with and without `--plot`
 - don't forget to `ssh -Y`
- `submit test job.sh` to see what it does
- doing that by hand a lot becomes painful: automate
- use `submit_jobs.py` to submit 20 jobs
- (optional) exercises:
 - Set the seeds based on the jobid
 - Specify the number of events as a command-line argument
 - Write the output to `/dcache` (if you have access)
 - Merge the output files

Frameworks

- DIY job submission takes quite a bit of development
- Bookkeeping what set of jobs did what becomes very tedious
- Frameworks are available, look for “PBS” support:
 - Nextflow: <https://www.nextflow.io/>
 - Ganga: <https://ganga.readthedocs.io/en/latest/>
 - Dask: <https://docs.dask.org/en/stable/>
 - What your experiment may be using
- We can help you install and configure these to work with stoomboot