# Supercomputer Clusters

Victor Eijkhout

2026 PCSE

# Cluster setup

Typical cluster:

- Login nodes, where you ssh into; usually shared with 100 (or so) other people. You don't run your parallel program there!
- Compute nodes: where your job is run. They are often exclusive to you: no other users getting in the way of your program.

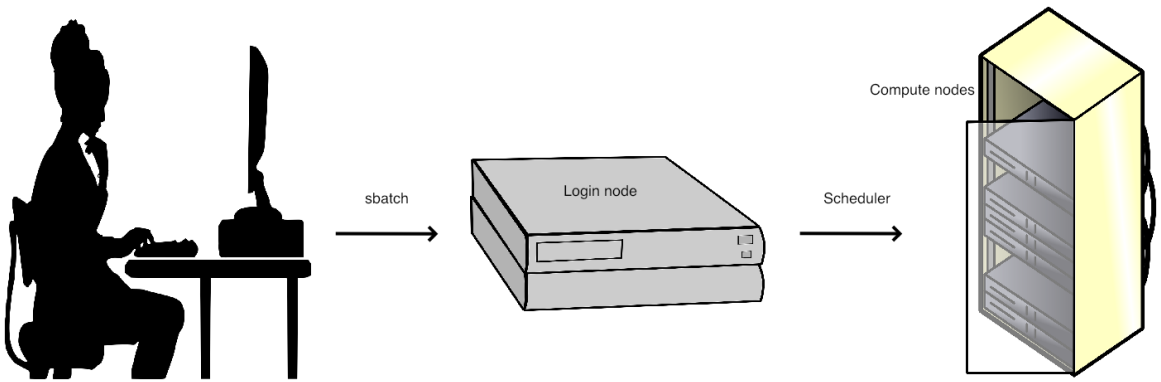
# Login nodes

Shared between many users
(how many right now?)
You are allowed to do:

- Compilation
- Post-processing
- Run very short programs (but not MPI)
- Submit jobs for batch execution (`sbatch`)
- Connect for interactive job (`idev`)

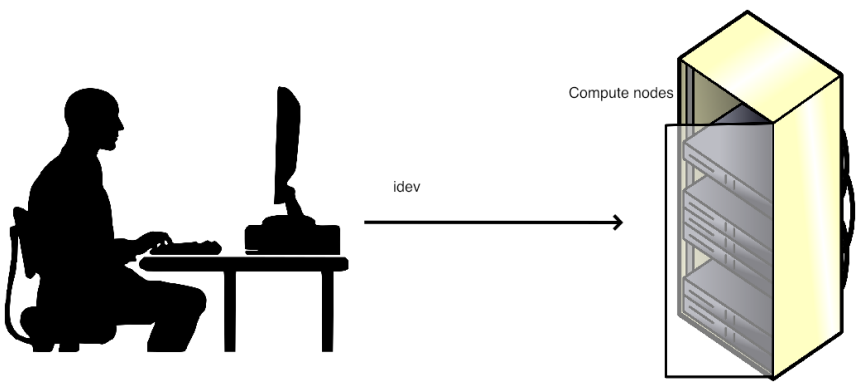
# Batch run

- Submit batch job with *sbatch*
  (on other clusters: *qsub*)
- Your job will be executed . . . Real Soon Now.
- See userguide for details about queues, sizes, runtimes, . . .



# Interactive run

- Do not run your programs on a login node.
- Acquire compute nodes with *idev*
- Caveat: only small short jobs; nodes may not be available.
  During training there will be a 'reservation':
  carefully look at messages from *idev*



# `idev` command

*`idev -t hh:mm:ss -N nodes -n cores -p queue`*

- `-t`: time
- `-N`: number of nodes
- `-n`: total number of cores
- `-p`: partition / queue

# Batch job

*sbatch batchfile.slurm*

- *sbatch*: submit
- *squeue*: job status

# Exercise 1

- Connect to your favorite cluster
  what is the hostname? how many users are logged in?
- Start an interactive session with `idev`;
  what is the hostname? how many users are logged in?
- Run: `ibrun hostname`
  also `ibrun -n 3 hostname`
- Same, but `idev` on two nodes.
- Create a job script that will run on 10 nodes;
  again let it run the `hostname` command.
  Submit with *`sbatch`*

# Course practicalities

# Languages

You can program in:

- C: use *mpicc* as compiler
- C++: use *mpicxx* as compiler
  with MPL library: additionally *module load mpl*
- Fortran: use *mpif90* as compiler
- Python: *module load python* or *python3*

# Lab setup

- Clone the repository `https://github.com/VictorEijkhout/TheArtOfHPC_vol2_parallelprogramming`
- Directory: `exercises-mpi-c` or `cxx` or `f` or `f08` or `p` or `mpl`
- Open a terminal window on a TACC cluster.
- Type `idev -N 2 -n 10 -t 2:0:0` which gives you an interactive session of 2 nodes, 10 cores, for the next 2 hours.
- Type `make exercisename` to compile it
- Run with `ibrun` or `mpiexec`

# CMake

. MPI is discoverable by CMake.