Introduction to HPC clusters
ILRI HPC infrastructure

Bernice Ngina Waweru,
Bioinformatics Research Officer, BecA-ILRI Hub

BHKi BOSS Virtual Workshop, November 2nd, 2021

What is a HPC cluster?

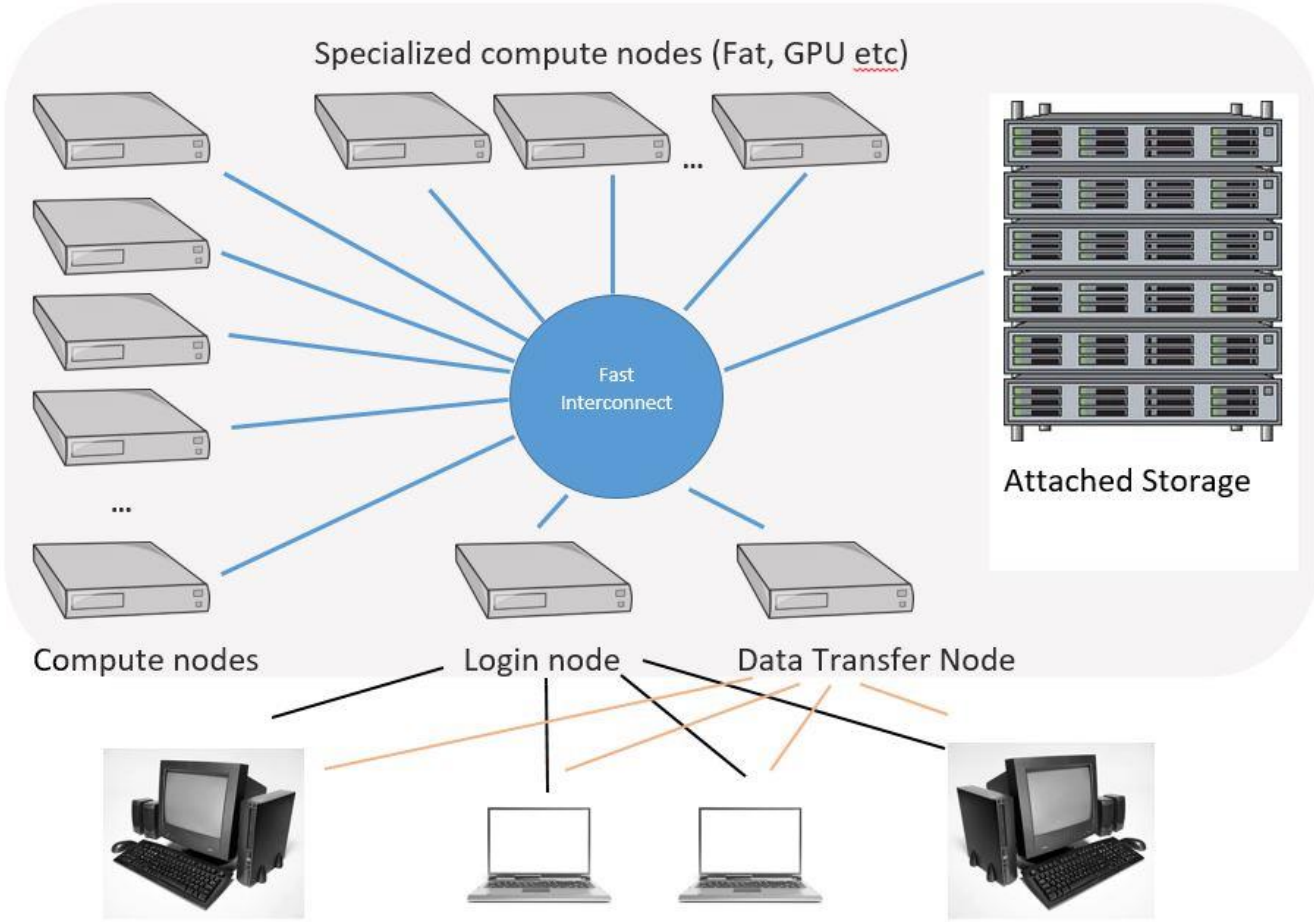
- A High Performance Computing a.k.a hpc cluster is a collection of many separate computers(servers) called nodes, connected via fast interconnect

- Different types of nodes will have capacity to do different types of tasks

- Each hpc cluster is composed of;
  - Head node or login node
  - Compute nodes (batch, highmem, GPU)
  - Storage nodes/system
  - Network fabric
  - Software

A HPC cluster

Difference between personal computer and a cluster node is in **quantity**, **quality** and **power** of the components



What is an HPC cluster | High Performance Computing (iastate.edu)

The ILRI HPC cluster

ILRI's high-performance computing "cluster" is currently composed of 7 dedicated machines:

- **hpc**: main login node, "master" of the cluster

- **compupte2**, **compute05**, **compute06**: used for batch and interactive jobs like BLAST, structure, R, etc (compute05 and compute06 have the newest AMD EPYC CPUs)

- **compute07**: used for high-memory jobs like genome assembly (mira, newbler, abyss, etc)

- **compute03**: fast CPUs, but few of them

How do we connect to a hpc cluster?

- A **terminal** is used to access to files on your computer, but more so,
- Access a server through the SSH protocol
- **SSH** short for **S**ecure **S**hell – a program for logging into a remote machine and for executing commands on a remote machine.
- To access any hpc computing server you must have login credentials, issued by the systems administrator, that includes a **username, password** and **hostname** of servers to connect to.

How do we connect to a hpc cluster?

- On linux operating machine, connect directly from the terminal
- On a windows machine, connect through' cmd', or using a linux client for windows
  - MobaXterm,
  - PuTTy
  - Cygwin
- Logging in
  - ssh bngina@hpc.ilri.cgiar.org
  - ssh user1@hpc.ilri.cgiar.org

Terminal terminology

Useful basic terminology for when asking for help or describing a problem.

- **Welcome Message:** the startup message when logging into a remote machine

- **Prompt:** The text next to where you type your commands
  - prompts can be modified to include additional information like hostname or current folder location

- **Command:** the function or script you are trying to run.

- **Argument:** added to a Command to modify the output
  - there is always a space between a command and the argument

- **Standard Out:** the result of a command

- **Standard Error:** the error of a failed command

A few useful commands …

Command	*Function*	*Syntax/example usage*
mkdir	make directory	mkdir DIRECTORY
pwd	print working directory	pwd
cd	change directory	cd ~ or cd #home directory
		cd .. #previous (parent directory)
ls	list contents	ls [OPTIONS] DIRECTORY

What is Slurm?

- **Slurm** *aka* **S**imple **L**inux **U**tility for **R**esource **M**anagement
- Slurm is an open-source workload manager for the ILRI HPC infrastructure
- HPC cluster is made up of a number of compute nodes, each with a complement of processors, memory and GPUs
- **Slurm** is responsible for finding and allocating the resources that fulfill the job's request at the soonest available time

Interactive jobs

An **interactive job** is a job that returns a command line prompt (instead of running a script) when the job runs

```
[bngina@hpc: ~]$ interactive
 salloc: Granted job allocation 4950
[bngina@taurus: ~]$
```

You can also open an interactive session on a **specific node** of the cluster by specifying it through the **-w** commandline argument;

```
[bngina@hpc ~]$ interactive -w compute06
 salloc: Granted job allocation 18940
[bngina@compute06 ~]$
```

NB: interactive jobs have **a time limit of 8 hours**: if you need more, then you should write a batch script

Anatomy of a batch job

ILRI hpc is configured with the following job queues (also called "partitions" in SLURM):

- debug
- batch
- highmem

Get info on the computing nodes use [**sinfo**], add flags to get more infor [**sinfo -lNe**]

Be mindful when running jobs that write a lot of temporary/output files!!
-use local scratch spaces on nodes i.e **/var/scratch/<user> | <JOBID>**

Data on **/var/scratch/** that has **not been open for 90 days is deleted!!!**

Anatomy of a batch job

The batch job script is composed of four main components:

- The interpreter used to execute the script
- "#" directives that convey default submission options.
- The setting of environment variables (where necessary)
- The application(s) to execute along with its input arguments and options.

```
#!/bin/bash
#SBATCH -p (batch/highmem)
#SBATCH -N 8

srun -N 8 -n 32 myApp
```

- The **sbatch** command is used to submit a batch script to Slurm
- Use **scancel** <JOBID> to cancel a job

Batch jobs ……

- Check the submitted job with **squeue**
- Check usage and flags with **man sbatch**
- Flags to add to **squeue**
  - **-u** to check only your jobs
  - **-O** username, jobid,name,nodelist,numcpus

- Add more options to sbatch for easy tracking
  - `#SBATCH -J (give you job a name)`
  - `#SBATCH -w (request a particular node)`
  - `#SBATCH -e (error log file)`
  - `#SBATCH -o (output file)`
  - `#SBATCH -email (add email for reporting of status i.e, error, ended, failed)`
- Max number of **nodes** allowed per normal user is **20**,

- More reading on using slurm: https://bioinformaticsworkbook.org/Appendix/HPC/SLURM/slurm-cheatsheat.html#gsc.tab=0

Useful links

- https://bioinformaticsworkbook.org/Appendix/Unix/unix-basics-1.html
- http://www.ee.surrey.ac.uk/Teaching/Unix/
- https://hpc.ilri.cgiar.org/list-of-software