HPC Basics with Xanadu

Neranjan V. Perera
Computational Biology Core

UNIVERSITY OF CONNECTICUT
Outline

• HPC Introduction

• Xanadu Cluster

• Connecting to Xanadu cluster

• Running jobs
  • Interactive sessions using srun
  • Job submission using sbatch
HPC Introduction

Development of models begins at small scale.

Working on your laptop is convenient, simple.

Actual analysis, however is slow.

“Scaling up” typically means a small server or fast multi-core desktop.

Speed exists, but for very large models, not significant.
HPC Introduction

Typical cluster (BBC)
HPC Cluster Architecture

**SLURM:**
Simple Linux Utility for Resource Management

HPC Cluster:
- **Submit nodes**
- **Login**
- **Scheduler**
- **File Server**
  - `/HOME`
  - `/data_storage`
- **Computes**
- **High-speed network**
  - (10GigE / Infiniband)

**Institute for Systems Genomics:**
Computational Biology Core
Xanadu Cluster Architecture

Queue/Partition

xeon/general

amd/general

himem1-5

gpu /xeon/ general

Login Node

cpu :  4
RAM:  8GB

User 1

User 2

User 3

User N

xanadu-submit-ext
xanadu-submit-int

cpu :  36 / 48 / 64
RAM:  128 – 512 GB
How to get an Account

https://bioinformatics.uconn.edu/contact-us/
How to Connect to the Cluster

You need to Connect to the VPN of UCHC
https://remoteaccess.uconn.edu/vpn-overview/connect-via-vpn-client-2/

1. Open Pulse secure
2. Add new connection
3. Set Server URL to : vpn.cam.uchc.edu
4. Save

Connect and login with CAM ID and Password
How to Connect to the Cluster - Windows

Putty: [https://www.chiark.greenend.org.uk/~sgtatham/putty/latest.html](https://www.chiark.greenend.org.uk/~sgtatham/putty/latest.html) or google search “putty”

Open Putty it will open window1.
1. Provide host name e.g. `username@xanadu-submit-int.cam.uchc.edu`
2. Expand SSH tab and select X11 (shown in window2)
3. Enable X11 forwarding by selecting it. (window2)
4. Scroll up the left panel and select Session.(window1)
5. Name your session e.g. Xanadu_cluster and click save tab to save.
6. Your session name should appear in saved sessions. Double click on your session name to connect to server with SSH session.
How to Connect to the Cluster

Internal submit node

```
ssh <user_name>@xanadu-submit-int.cam.uchc.edu
```

External submit node

```
ssh <user_name>@xanadu-submit-ext.cam.uchc.edu
```
Queue/Partition | Nodes
---|---
xeon/general | xanadu01-11
amd/general | shangrila01-19 xanadu20-28
himem1-5 | xanadu29-31
gpu /xeon/ general | xanadu01

CPU: 4
RAM: 8GB

CPU: 32 – 48
RAM: 128 – 512 GB
How to use the cluster

1. Interactive session: srun
2. Submission script: sbatch

Login Node

xanadu-submit-ext
xanadu-submit-int

cpu: 4
RAM: 8GB

cpu: 32 – 48
RAM: 128 – 512 GB
How to start an interactive session

`srun --mem=500M --pty bash`

`srun` starts a interactive session

`--mem` specify the real memory required per node. Different units can be specified using the suffix [K|M|G|T]

Default memory allocation for an interactive session is 128MB. If you need more memory you may need to allocated it in using the `--mem` option
Running jobs using a submission script

A submission script is composed of:

- Defining of Bash shell environment
- Parameters for the SLURM scheduler
  - Number of processors / threads
  - Allocate memory for the job (default 128MB)
- Command submission line(s)
  - Load your modules
  - commands
Running jobs using a submission script

Define the bash shell

```bash
#!/bin/bash
```

Parameters for the SLURM:

- `#SBATCH --job-name=sickle`  jobname
- `#SBATCH -n 1`  number of tasks
- `#SBATCH -N 1`  number of nodes on request for the job
- `#SBATCH -c 1`  number of processors you are requesting for the job
- `#SBATCH --mem=200M`  total memory requesting for the job
- `#SBATCH --partition=general`  partition of request (general, himem1, etc..)
- `#SBATCH --mail-type=ALL`  send mail notifications at start and end
- `#SBATCH --mail-user=email`  email the notifications will be sent to
- `#SBATCH -o sickle_%A.out`  standard output file name
- `#SBATCH -e sickle_%A.err`  error file name
Running jobs using a submission script

Command submission line(s)

```
We strongly recommend to put this line in every submission script
```

```
echo `hostname`
```

```
module load sickle/1.33
```

```
sickle pe -f ../raw_data/1A-R1.fastq -r ../raw_data/1A-R2.fastq
    -t sanger \n    -o trimmed_1A-R1.fastq -p trimmed_1A-R2.fastq -s trimmed_S.fastq \n    -q 30 -l 45
```
#!/bin/bash
#SBATCH --job-name=sickle
#SBATCH -n 1
#SBATCH -N 1
#SBATCH -c 1
#SBATCH --mem=200M
#SBATCH --partition=general
#SBATCH --mail-type=ALL
#SBATCH --mail-user@mail
#SBATCH -o sickle_%A.out
#SBATCH -e sickle_%A.err

echo `hostname`

module load sickle/1.33

sickle pe -f ../raw_data/1A-R1.fastq -r ../raw_data/1A-R2.fastq
  -t sanger \
  -o trimmed_1A-R1.fastq -p trimmed_1A-R2.fastq -s trimmed_S.fastq \n  -q 30 -l 45

Save the script as sickle.sh
Script submission and monitoring

Job submission
$ sbatch sickle.sh

Monitor the job
$ squeue -u <user_name>

Other important commands in monitoring:
$ ssact

$ scontrol show jobid <job_ID>
-for more information
$scontrol show --dd jobid <job_ID>
Understanding the Xanadu HPC Resource

1. What is a cluster
2. How to obtain an account
3. How to reset a password
4. How to access the Xanadu cluster
5. HPC resources and limits
6. Working with Slurm (Running interactive jobs and submission scripts)
7. How to load software
8. File system
9. How to Transfer Data Between Clusters

What is a cluster

A desktop or a laptop, in most cases, is inadequate for analysis of large scale datasets (e.g. Genomics) or to run simulations (e.g. protein docking). They lack both the processing power and the memory to execute these analyses. This limitation can be overcome by combining machines (computers) in a predefined architecture/configuration so that they act as a single unit with enhanced computational power and shared resources. This is the basic concept of a high performance cluster. A cluster consists of a set connected computers that work together so that they can be viewed as a single system. Each computer unit in a cluster is referred as ‘node’.

The components of a cluster are usually connected through fast local area networks (“LAN”), with each node running its own instance of an operating system. The benefits of clusters include low cost, elasticity and the ability to run jobs anytime, anywhere.

Cluster Etiquette

https://bioinformatics.uconn.edu/resources-and-events/tutorials/xanadu/