# UNIX Exercise 3

This exercise mainly deals with using HPC clusters for large scale data (Next Generation Sequencing analysis, Genome annotation, evolutionary studies etc.). These clusters have several processors with large amounts of RAM (compared to typical desktop/laptop), which makes it ideal for running programs that are computationally intensive. The operating system of these clusters are primarily UNIX and are mainly operated via command line. All the commands that you have learned in the previous exercises can be used on HPC.

## Prerequisites

ISU High Performance Computing (ISUHPC) offers shared cluster computing infrastructure for researchers and students at ISU. Brief descriptions for the available resources can be found here: <http://www.hpc.iastate.edu/systems>. To begin with, you need to request permission for accessing these resources either through your department or through your advisor. All workshop attendees will have their account setup on HPC class education cluster and they can use their ISU NetID and the password for logging-in. You should have already received a confirmation email about your account creation with instructions on how to connect to the cluster. In this exercise we will specifically teach you how to connect to a remote server (HPC), transfer files in and out of the server, and running programs by requesting resources.

### Logging in

You can log onto its front-end/job-submission system (hpc-class.its.iastate.edu) using your ISU NetID and password. Logging into HPC class requires an SSH client if you are using Windows but Mac/Linux have these built into their OS. There are several available for download for the Windows platform.

Microsoft Windows:

* **PuTTY** is an extremely small download of a free, full-featured SSH client.
* **SSH Secure Shell Client**, also a full featured client that is commercial. It is available as part of the Iowa State University site-licensed software.

Mac OS X/ Linux / Solaris or other 'nix systems

* The ssh command is pre-installed. You may start a local terminal window from "Applications > Utilities" or by searching for installed programs. Log in using

ssh username@hpc-class.its.iastate.edu

### queues

HPC class uses PBS for job scheduling and resource management. You will probably have access to the following queues, each with several nodes (1 node = 16 processors and 64 GB RAM).

|  |  |
| --- | --- |
| short | 1:00:00 |
| medium | 6:00:00 |
| long\_2node | 73:00:00 |
| large\_short | 0:15:00 |
| tiny | 0:10:00 |
| long | 72:00:00 |

### File transfer:

There are a number of ways to transfer data to and from HPC clusters. Which you should use depends on several factors, including the ease of use for you personally, connection speed and bandwidth, and the size and number of files which you intend to transfer. Most common options include scp, rsync (command line) and SCP and SFTP clients (GUI).

scp (*s*ecure *c*o*p*y) is a simple way of transferring files between two machines that use the SSH (Secure SHell) protocol. You may use scp to connect to any system where you have SSH (login) access. scp is available as a protocol choice in some graphical file transfer programs and also as a command line program on most Linux, UNIX, and Mac OS X systems. scp can copy single files, but will also recursively copy directory contents if given a directory name. scp can be used as follows:

scp sourcefile username@hpc-class.its.iastate.edu:somedirectory/

(to a remote system from local)

scp username@hpc-class.its.iastate.edu:somedirectory/sourcefile destinationfile

(from a remote system to local)

scp SourceDirectory/ username@hpc-class.its.iastate.edu:somedirectory/

(recursive directory copy to a remote system from local)

rsync is a fast and extraordinarily versatile file copying tool. It can synchronize file trees across local disks, directories or across a network

rsync -rave "ssh -l username" path/to/SourceDirectory username@hpc-class.its.iastate.edu:somedirectory/

Synchronize a local directory with the remote server directory

rsync -rave "ssh -l username" username@hpc-class.its.iastate.edu:SourceDirectory/ path/to/Destination/

Synchronize a remote directory with the local directory

User friendly (GUI) choices for file transfer:

* WinSCP (http://winscp.net): for Windows only
* FileZilla (https://filezilla-project.org): Windows/Linux/Mac
* **Cyberduck** (http://cyberduck.io): Mac and Windows
* Macfusion (http://macfusionapp.org): Mac only

## variables

When your account is setup, some standard variables (known as environment variables) that are specific to your account were created. These variables can be used to simplify your navigation (many environment variables specify storage locations and paths). Think it of as "shortcut" that you create on your desktop to open the desired application that you frequently use. Your login automatically defines these variables for you. Some standard variables are

Name Description

USER your username

HOME path to your home directory

PWD path to your current directory

PATH all directories searched for commands/applications

HOSTNAME name of the machine you are on

SHELL your current shell (bash, tcsh, csh, ksh)

SSH\_CLIENT your local client's IP address

TERM type of terminal or terminal emulator being used

To perform the action you need to use them with $ sign in front. For example:

cd $HOME

Changes your directory from the current location to home your directory

You can look up the values stored in these variables by using echo command

echo $VARIABLE\_NAME

You can add any number of such variables manually by editing the hidden file (.bashrc) in your home directory (make sure that you create a backup copy of this original file before you start editing).

## PRE installed programs

To use pre-installed applications you can use the module command. First configure it using following command:

module use /path/to/bioinformatics/modules

After that, you can use the module load command to access the software you want to use. For instance, to use FASTQC (program to check the quality of fastq reads of NGS) program,

module load fastqc

To check all available programs:

module avail

## Submitting jobs

To submit a job (running your script, starting a program etc) to the HPC-class cluster, you should use portable batch system (PBS) job scheduler. It will manage schedule jobs to run on HPC depending on the hardware requirement and other factors to efficiently use the available resources. If you run any jobs without the PBS then jobs will be executed on a front-end login hostthat is shared by all users. This will negatively impact everyone's ability to use HPC.

Usually, a submission script specifying the requirements of hardware for your job will be used to submit jobs on HPC. This script file is a simple text file where you specify:

* Memory requirement
* Desired number of processors
* Length of time you want to run the job
* Type of queue you want to use (optional)
* Additionally, you can also specify where to write output and error files as well as give name for your job while running on HPC

A simple job submission script is shown below:

#!/bin/bash

#PBS -l vmem=64Gb,pmem=8Gb,mem=64Gb

#PBS -l nodes=1:ppn=8:ib

#PBS -l walltime=48:00:00

#PBS -q long\_2node

#PBS -o BATCH\_OUTPUT -e BATCH\_ERRORS

#PBS -N JOBNAME

You can also create a script using this html utility <http://hpcgroup.public.iastate.edu/HPC/hpc-class/hpc-class_script_writer.html>

It is useful to keep a ‘template’ of a job submission file in your home directory, which can be modified every time you submit a new job. Heavily customized template submission file with some useful features is given below:

#!/bin/bash

#PBS -q <queuename>

#PBS -l vmem=64Gb,pmem=4Gb,mem=64Gb

#PBS -l nodes=1:ppn=16:compute

#PBS -l walltime=48:00:00

#PBS -N <jobname>

#PBS -o ${PBS\_JOBNAME}.o${PBS\_JOBID} -e ${PBS\_JOBNAME}.e${PBS\_JOBID}

#PBS -m ae -M userid@iastate.edu

cd $PBS\_O\_WORKDIR

ulimit -s unlimited

echo "################ STATS ##################"

SSECS=$(date +"%s")

echo ${SSECS}

START=$(date +"%r, %m-%d-%Y")

echo -e "Host\t\t: $(hostname)"

echo -e "Processors\t: $(wc -l < $PBS\_NODEFILE)"

echo -e "Nodes\t\t: $(uniq $PBS\_NODEFILE | wc -l)"

echo -e "Total memory\t: $(free | grep Mem | awk '{print $2/1048576}' OFMT="%2.2f") Gb"

echo -e "Free memory\t: $(free | grep Mem | awk '{print $4/1048576}' OFMT="%2.2f") Gb"

echo -e "Directory\t: $(pwd)"

echo "#########################################"

module use /path/to/modules

module load moduleaname

## SCRIPT-START ###############################

## SCRIPT-END #################################

echo "############# TIME STAMP ################"

DIFF=$((`date +"%s"`-${SSECS}))

printf "Start\t\t:${START}\nEnd\t\t:$(date +"%r, %m-%d-%Y")\nTIME (hh:mm:ss)\t:%02d:%02d:%02d\n" "$((${DIFF}/3600))" "$(((DIFF%3600)/60))" "$(((DIFF%3600)%60))"

echo "#########################################"

Whenever you submit a job, you have to modify: the numbers for memory/nodes/processors/walltime, program name and insert the script that you wish to run. Jobs can then be submitted using qsub command:

qsub template\_jobfile.sub

A sample job to check the quality of the reads obtained from a sequencing project is present in the jobfile.sub. It is set to run it on short queue. To start the job:

qsub jobfile.sub

You will receive a confirmation 1234.hpc-class.its.iastate.edu where 1234 is your job ID. Once you have submitted the job script, you can view status of jobs by using following commands:

qstat –f yourjobid for information about your submitted job

qstat -an1 yourqueuename current jobs running on queue you have submitted

qstat –u yourusername list all the current jobs you are running on cluster

qstat –a –u yourusername displays the status of your job

Additional resources:

<http://hpcgroup.public.iastate.edu/HPC/hpc-class>

Upon completion of the job, you will see many files in your working directory. Two of these files that start with your jobname are error log file (jobname.e1234.hpc-class.its.iastate.edu) and output log file (jobname.o1234.hpc-class.its.iastate.edu). The fastqc results for two reads will be in two separate folders (R1\_fastqc and R2\_fastqc). These folders are also saved as zip files by the program.

To view the results, change directory to one of the folder (R1\_fastqc or R2\_fastqc) and open fastqc\_report.html file. You can do this by

firefox fastqc\_report.html

## Downloading data

In order to start using the computational power of the HPC cluster, you need to first get the data there. If your data is already in your local computer, you can transfer them easily using WinSCP software or any other software (refer prerequisites). But if the data that you will be using is available in the public databases then you can directly get it from there using wget command (*W*WW *get*)

To download data from NCBI Sequence Read Archive (SRA) or genomics core website or any other website:

wget http://website.url

As an example, we will download *Glycine max* (soy bean) annotation information file from Phytozome DB.

wget http://goo.gl/CDXx15

This is a single line command and you will see ‘Gmax\_189\_annotation\_info.txt.gz’ file after few seconds. You can extract it and view it or delete it using the commands you have learnt.