STATS 700 Bayesian Inference and Computation   **Fall 2017**

Lecture 3: Sept 13

Lecturer: Mark Champe   *Scribe: Chun-Chen Tu*

Disclaimer: *These notes have not been subjected to the usual scrutiny reserved for formal publications. They may be distributed outside this class only with the permission of the Instructor.*

3.1 Introduction to Flux

Flux is the shared, Linux-based high-performance computing (HPC) cluster available to all researchers at the University of Michigan.

3.1.1 Flux architecture

There are three types of nodes, which are designed for different purposes:

1. Login nodes: These are the nodes that users actually connect to. Users would submit jobs or do some operations on these noes.

2. Compute nodes: Once a job is submitted, the job would be done on the compute nodes.

3. Data transfer nodes: These nodes are equipped with high bandwidth and fast network access and are designed for transferring data.

3.1.2 Flux storage

There are several places or ways we could store our files. Depends on the file size and the file lifetime, there are several options:

1. User home directory($\sim$/): Each user has 80GB of storage under their home directory. It is used for keeping small files (environment settings, scripts or codes) for long term.

2. The scratch directory(**/scratch**): There are 1.5PB of high speed temporary network storage shared by all users. Files in the scratch directory would be deleted if not access in 90 days. It is best for large files.

3. Local disk(**/tmp**): Local disks provide the fastest speed on I/O. However, files under **/tmp** directory would not share across nodes.

4. The NFS storage: This is the storage mounted from network file system.

3.2   Connecting to Flux

When you are using on-campus network, you can connect to the Flux directly. On the other hand, the off-campus connection should be established through VPN or campus login nodes:

```
ssh uniquename@login.itd.umich.edu
```

3.2.1   Connect to the login node

• For Windows user: You will need to download an ssh client (such as putty) and type in the login address for Flux as shown in Figure **??**
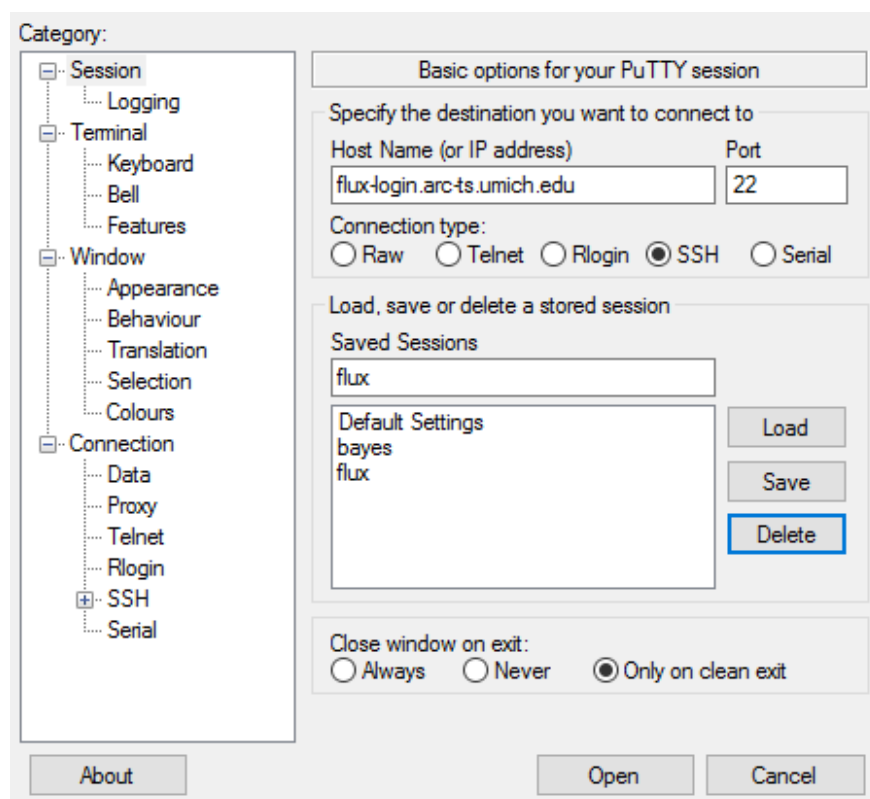

Figure 3.1: The putty configuration for connecting to Flux.

• For MaxOS or Linux user: You can simply launch terminal application and connect to Flux via ssh:

```
ssh uniquename@flux−login.arc−ts.umich.edu
```

Note that *uniquename* is required if the user account of your local machine is different from your unique name.

3.2.2 File transferring

Files can be transferred using the data transfer node. Note that the hostname is different from the login node (**flux-xfer.arc-ts.umich.edu** instead of **flux-login.arc-ts.umich.edu**).

Files transferring between your local machine and Flux serve could be done in several ways:

1. **scp**: The Linux command Secure Copy (**scp**) allows you to copy files between different hosts. The syntax is:

   ```
   ssh <path_to_source_file> <path_to_destination_file>
   ```

   As an example, the first command copy the file "hello.txt" (placed under you current working directory) to your home directory on Flux and the second one copy the "hello.txt" from the home directory of Flux to your current working directory.

   ```
   scp hello.txt uniqname@flux-xfer.arc-ts.umich.edu:~/
   scp uniqname@flux-xfer.arc-ts.umich.edu:~/hello.txt .
   ```

2. GUI applications: Applications such as FileZilla (cross-platform), Cyberduck(MacOS) and WinScp(windows) provide user-friendly interface. For details, please visit: `http://arc-ts.umich.edu/flux/using-flux/login-nodes/`, there are several video examples under **SUPPORTED SFTP AND SCP CLIENTS** section. Note: if you cannot establish a connection, first check if you are using the authorized network (on-campus network/off-campus with VPN). Next, check the connection type, it should be **sftp** using port 22.

3.3 Flux mechanics

3.3.1 The .pbs job

To submit a job you would need a **.pbs** file and the codes you want to run. A **.pbs** contain some information about your job (job name, job allocation) and settings (core number, memory capacity).

Here is a snippet of the **.pbs** file:

```
#PBS -N stats700f17_HW1
#PBS -V
#PBS -A stats700f17_flux
#PBS -l qos=flux
#PBS -q flux
#PBS -l nodes=1:ppn=12,mem=1gb,walltime=00:05:00
#PBS -m abe
#PBS -j oe

#Your Code Goes Below:
cd $PBS_O_WORKDIR
R CMD BATCH --vanilla myscript.R myscript.out
```

For more details about the options, please visit the ARC website: `http://arc-ts.umich.edu/software/torque/`.

3.3.2 Batch commands

- **qsub** *mypbs.pbs*: You can submit a job through this command.

- **qdel** *jobid*: You can delete a job through this command.

- **qstat**: This command is used for checking job status. You can either use **qstat** *jobid* to check the status of a certain job (specified by the *jobid*) or use **qstat -u** *user* to check the job status submitted by the *user*.

- **showq -w acct=stats700f17_flux**: Use this command to check the job queue of the allocation.

- **freealloc –jobs stats700f17_flux**: Use this command to check the resources and jobs of the allocation.

3.3.3   Software and module

The software provided by Flux can be loaded using the command **module**. Several useful commands are

- **module load** *moduleName*: This would load the module to your current environment.

- **module list**: Show loaded modules

- **module avail** or **module avail** *name*: Show all of the available modules or show the available modules related to *name*.

- **module save** *setName* and **module restore** *setName*: You could save your current module setting using **module save** and restore it afterwards by **module restore** *setName*. The command **module save** (note that *setName* is not specified), would save the current module setting and this setting would be loaded automatically each time you log in.

3.4   Resources

- Slides: `umich.edu/~mrchampe/stats700`

- ARC Flux user guild: `http://arc-ts.umich.edu/flux-user-guide/`

- ARC Flux page: `http://arc-ts.umich.edu/systems-and-services/flux/`