

# 10 Steps to using Yale's HPC Clusters: A Beginner's Guide.

---

*Note to Reader: Yale currently has 4 different HPC clusters. Although they are similar in most respects, some details vary. In the following, pay attention to which specific details apply to the cluster you are planning to use.*

1. **Obtain an account on one of Yale's clusters:** <http://research.computing.yale.edu/account-request>.

2. **Login to the cluster**

All of Yale's clusters are accessed via ssh. You can use ssh directly, or via a graphical ssh tool. The details vary depending on the operating system of your local computer. If you want to access the clusters from outside Yale, you must use the VPN.

**Linux or MacOS:** we recommend using ssh directly:

```
$ ssh netid@cluster.hpc.yale.edu
```

Louise: You will be prompted for your netid password

Bulldogn: You will be prompted for your Multifactor Authentication, and then your netid password.

Omega and Grace: You will be prompted for the passphrase for your ssh key pair (which you provided when getting your account).

**Windows:** We recommend using putty. This is available at the ITS software library: <http://software.yale.edu/Library/Windows>. See <http://www.chiark.greenend.org.uk/~sgtatham/putty/> for more information.

3. **Orientation**

Broadly speaking, a compute cluster is a collection of computers (nodes) running the linux operating system. They are connected to a large, fast storage system. You use the cluster by allocating one or more nodes and running programs on them.

After logging into the cluster, you are on the "login node". Think of this as an entryway that is shared by all the users of the cluster. You can see all of your files, and you can edit files and submit jobs. You should NOT compile programs or run jobs here. You are in your home directory, which is where you can store small data files, scripts, etc. You have a file storage limit in your home directory that varies by cluster but is never more than a few hundred GB. We do our best to backup home directories. ***In general, nothing else is backed up.***

To check your space usage and quota, do:

Louise, Bulldogn: \$ /usr/local/cluster/bin/myquota.sh  
Omega: \$ du -s .  
Grace: \$ /usr/local/bin/groupquota.sh

Larger files and temp files should not be kept in your home directory.

#### 4. Locate the applications you want to use

We have preinstalled a large number of programs on the clusters for you to use. If you don't find a program you need, please ask by sending email to [hpc@yale.edu](mailto:hpc@yale.edu), and we can install it for you.

Grace and Omega use the Module system to help you locate programs. Each installed program has an associated module file. By loading that module file, you setup access to the program, including any dependencies.

Example:

Grace:\$ module avail python  
Omega:\$ modulefind python

(lists all python installations)

```
$ module load Langs/Python/2.7.6
```

Now if you run python, you will get the correct version.

Louise and Bulldogn do not yet use modules. You must find your application manually by looking for it in the installation directory, usually here:

Louise: /usr/local/cluster/software/installation  
BulldogN: /home/bioinfo/software.

More information about where specific applications or data are located on each cluster can be found in the cluster-specific documentation here: <http://research.computing.yale.edu/hpc-clusters>

#### 5. Rules of the road

Before you begin using the cluster, here are some important guidelines:

1. Do not run programs on the login node. Always allocate a compute node and run programs there. See details below.
2. Never give your password to anyone else.
3. Do not store any protected or regulated data on the cluster (e.g. PHI data)
4. Clean up after yourself by releasing unused node allocations and removing unneeded files.
5. Use scratch space and/or project space for large and/or numerous files, rather than using your home directory.
6. Do your best to understand your program's requirements, especially in terms of memory and IO usage. Try not to overload the nodes' memory or IO capabilities.
7. If you are uncertain about any of the above, please ask by emailing to [hpc@yale.edu](mailto:hpc@yale.edu)!

## 6. Copying files to/from the cluster

You will likely find it necessary to copy files between your local machines and the clusters. Just as with logging in, there are different ways to do this, depending on your local operating system.

Linux and MacOS: we recommend using scp. Here are some examples :

To copy a file from your local machine to your home directory on grace, do

```
$ scp myfile netid@grace.hpc.yale.edu:
```

To copy a directory and all of its contents from your local machine to omega and put it in my/subdir, do:

```
$ scp -r mydir netid@omega.hpc.yale.edu:my/subdir
```

To copy a directory and all of its contents from from the cluster to your local machine do:

```
$ scp -r netid@omega.hpc.yale.edu:my/subdir mydir
```

Note that scp will prompt you for passwords in exactly the same way as ssh.

For more information, look at the manual page for scp. (do `$ man scp`) You may also want to investigate the rsync command, which is similar to scp but much more flexible.

Windows: we recommend winscp. This is a graphical tool similar to windows explorer, and is available on the ITS software library: <http://software.yale.edu/Library/Windows>. See <http://research.computing.yale.edu/hpc/faq/transferring-files> for more information.

## 7. Introduction to the queueing system

The clusters are controlled using a queuing system that allocates the compute nodes and manages jobs running on the nodes. We currently have two different queuing systems at Yale. The two systems are very similar in spirit but differ slightly in detail.

Grace uses LSF

Omega, Louise, and Bulldogn use Torque

Both LSF and Torque have commands to let you: allocate nodes, submit batch jobs, check job status, kill jobs, etc. More detail can be found here: [LINK](#)

## 8. Allocate and use interactive allocation

All programs should be run on a proper compute node allocation. An allocation can be interactive, in which case you are logged into the node and use it by typing on the command line. This is very useful for testing as well as for tasks that are inherently interactive.

To allocate a single interactive core, do:

Torque (except Omega): `$ qsub -I`

On Omega, you must provide a queue name:

```
$ qsub -I -q fas_normal
```

LSF: `$ bsub -Is -q interactive bash`

## 9. Create batch script

Most jobs on the clusters are not run interactively. Instead, you create a batch script that contains directives to the queuing system and commands you want run. You submit that script, and the queuing system allocates resources and runs the script for you, notifying you when it finishes. The first step is to create a batch script. Here is a very simple batch script that simply prints the hostname of the allocated node and a timestamp, for both LSF and Torque:

Torque (Omega, Louise, Bulldogn)

```
#PBS -m abe -M email@yale.edu
cd $PBS_O_WORKDIR

hostname
date
```

LSF (Grace)

```
#BSUB -q shared
hostname
date
```

Using an editor, create a file called `test.sh`, and type in the commands shown above (using the correct version for your cluster). We recommend using the nano editor, but you can also use `vi` or `emacs`. Make sure to change “email” to your name.

```
$ nano test.sh
```

## 10. Run batch script

Now submit the script to the queuing system. The command to use depends on which queuing system your cluster has:

Torque: `$ qsub test.sh`

LFS: `$ bsub < test.sh`

By default, torque will leave the output in a file called `test.sh.ojobid`. LSF will include it in the notification email. These behaviors are easily modified using parameters to `qsub` or `bsub`.

By default, the queuing systems will run your job on the default queue, and use a single core. You can use different cores or multiple cores by passing parameters to `qsub/bsub`.

Assuming that there are free resources, the job should be scheduled and run immediately. You should receive an email when it finishes.

By replacing hostname and date with your own commands, you can run your own jobs in batch mode.

## Next Steps

If you have successfully completed the previous steps, you have managed the basics: getting on the cluster, editing files, copying files to and from the cluster, and running jobs.

For more information, see the other tutorials on <http://research.computing.yale.edu/hpc-support>, including the Advanced Getting Started guide and FAQ.

For specific questions or information that you cannot find on the website, please send email to [hpc@yale.edu](mailto:hpc@yale.edu).