### Quick Links

<table>
<thead>
<tr>
<th>Category</th>
<th>Links</th>
</tr>
</thead>
<tbody>
<tr>
<td>HPC Home</td>
<td></td>
</tr>
<tr>
<td>Getting an account</td>
<td></td>
</tr>
<tr>
<td>Getting started on Prince</td>
<td></td>
</tr>
<tr>
<td>Prince How-to Articles</td>
<td></td>
</tr>
<tr>
<td>Logging in</td>
<td></td>
</tr>
<tr>
<td>Windows</td>
<td></td>
</tr>
<tr>
<td>Mac / Linux</td>
<td></td>
</tr>
<tr>
<td>Clusters and Storage</td>
<td></td>
</tr>
<tr>
<td>Prince (HPC)</td>
<td></td>
</tr>
<tr>
<td>Dumbo (Hadoop)</td>
<td></td>
</tr>
<tr>
<td>Dalmata (NYU Abu Dhabi)</td>
<td></td>
</tr>
<tr>
<td>Transfering data to/from the clusters</td>
<td></td>
</tr>
</tbody>
</table>
Transferri ng data to/from Prince cluster using Globus

Submittin g jobs with sbatch

Available software

Licensed Software Available on the HPC Cluster

Building Software

Slurm Tutorial

Tutorials

FAQs

Scratch Area Cleanup

Programming for Biologists

Ackno wledge Statement

Research Gallery

HPC People

HPC Policies
Something went wrong!

Why does running "ls" on /scratch take so long?

I can't login

When trying to login, I get warning s about "HOST IDENTIFICATION HAS CHANGED"

What happened to my data on /scratch?

In the library, my wireless connection keeps dropping out. How can I fix it?

I'm getting a "module : command not found" error
Warning:
no access to tty
(Bad file descriptor),
Thus no job control
in this shell

I get an error
"Warning: no display specified." when
I use -X flag with ssh

Who killed my job, and why?

I got an email
"Please do not run jobs on login nodes"

Running jobs

What resources can and should I request?

Can I make sure a job gets executed only after another one completes?

How do I log in to a specific node?

How can I make sure my job is running smoothly?
My job will take longer than 48 hours, what should I do?

My job needs (MySQL, some other service) to be running

I want to run a job at 9am every day

Using software

How do I run ... (esp, needs a license)

A S T A T A job?

A Gaussian job?

A Matlab job?
a parallel non MPI job (e.g. Julia)?

I can't find (some software package)

Can you install (some software package)?

How can I view a PDF file on Mercer?

Managing data

How much of my file/space quota have I used?

How do I give my colleague access to my files?

How do I get the best transfer speed to or from BuTinah?
I have a huge amount of data that I want to compress for storage or transfer.

Where am I?

- How can I show the cluster name in the command prompt?

  The login nodes of each cluster have names like "login-0-1". You can add the following idiom to your $HOME/.bashrc file to set the prompt to the name of the cluster:

  ```
  # what cluster am I on?
  CLUSTERNNAME=NYU-HPC
  ip=$(host $(hostname))
  [[ $ip =~ 10.0.254.[0-9]+$ ]] && CLUSTERNNAME=Mercer
  [[ $ip =~ 10.0.255.[0-9]+$ ]] && CLUSTERNNAME=Dumbo
  PS1='["$CLUSTERNNAME": \u@\h \W]$ ' 
  ```

Something went wrong!

- My account expired! What should I do? Is my data gone forever?

  You can renew your account even after it has expired, see Getting or renewing an HPC account for how to renew. Data on /home, /archive and /work is deleted 90 days after account expiration. Data on /scratch is deleted after 60 days of not being used, whether your account is current or expired.

- Why is "ls" on /scratch so slow?

  Lustre stores the file itself and the file metadata (its name, size, etc) separately. When you issue a simple 'ls' command, a remote procedure call (RPC) is made to the metadata server (MDS), which returns a list of the files in the current directory. If you add certain options, such as -l or --color=ttty, then for each file in the list, ls will call stat() on that file. The stat() call involves an RPC to the MDS and an RPC to the object storage server (OSS) which holds the file itself. These RPCs, especially those to the OSS, can take a long time.

- I can't login

  I get a message about bad permissions

  SSH is fussy about permissions on the $HOME/.ssh directory. It should have:

  ```
  [ab123@myworkstation ~]$ ls -la $HOME
  total 545460
  drwx------ 11 ab123 users 4096 Sep  4 11:19 .
  drwxr-xr-x  3 root  root 0 Aug 28 14:44 ..
  drwx------  2 ab123 users 4096 Aug  8 11:19 .ssh
  ```

  Note that .ssh has rwx permission for the owner, and no permission at all for anyone else. You can set these permissions with the command:

  ```
  [ab123@myworkstation ~]$ chmod 700 $HOME/.ssh
  ```
When trying to login, I get warnings about "HOST IDENTIFICATION HAS CHANGED"
Recent versions of OSX and Ubuntu have a newer version of ssh than the NYU HPC clusters. You can prevent the warnings with two steps:
1. Update .ssh/config on your workstation according to the example here.
2. Delete your .ssh/known_hosts fileYou will then be asked about connecting to a new host on the first time, you can safely answer "yes"
What happened to my data on /scratch?
The /scratch filesystem is a short-term filesystem providing fast I/O for running jobs. It is not backed up, moreover files which remain unused for a period of time are flushed. See Storage July 2017 - especially /scratch policy for further information.
Q: In the library, my wireless connection keeps dropping out. How can I fix it?
A1: If you are using a Mac:
OSX roams "aggressively", which means that if it can see multiple access points it will abandon a working connection to pursue another one which might be better. The Bobst library supports a lot of wireless users and thus has many wireless access points, and Mac laptops behave like an undisciplined child in a candy store, authenticating to one point only to then disconnect and try another. Some actions which might help:
   • Under Network preferences->WiFi->Advanced, remove all NYU networks except "nyu" from the "preferred networks" list, and move "nyu" to the top of the list
   • If that fails, you can disable the aggressive roaming from the terminal, with the command:

     sudo defaults write /Library/Preferences/com.apple.airport.opp roam disabled -bool true

A2: If you are using Windows:
   • If you are running the PeerGuardian personal firewall software, switch it off (it disables DHCP). Otherwise:
Recent versions of Windows take a supposedly-more-secure but also less reliable approach to authenticating to a wireless network, which causes network connections to be dropped unnecessarily. A pop-up bubble in the bottom corner of the screen which says "please re-enter your password" is an indication that this is happening.
   • Instead of using the Windows-supplied wifi drivers, download and install the most recent driver from the manufacturer of your wireless-network-interface card
A3: If all this fails:
Come see the DS helpers - they may have another trick or two up their sleeve

Q: I used "module load" and it failed with a "module: command not found" error
Normally the location of the module command is set up when the shell is started, but under some circumstances that startup procedure can be bypassed. If you get this error you can explicitly prepare your environment for modules with one of the following commands:
   • If your script (or interactive environment) uses bash (the default) or sh or ksh:

     source /etc/profile.d/env-modules.sh

   • If your script (or interactive environment) uses csh or tcsh:

     source /etc/profile.d/env-modules.csh

In the case of a PBS job script, add one of the above lines before the first "module" command in your script.
If you are seeing the error in an interactive shell, run one of the above commands at the prompt, then attempt the "module load" command again.

Warning: no access to tty (Bad file descriptor), Thus no job control in this shell
It's harmless and does not indicate an error. This is an innocuous warning that simply means you are running a script (rather than a binary) under a job that has no access to the TTY. In other words, you cannot interrupt it (^C), suspend it (^Z) or use other interactive commands because there is no screen or keyboard to interact with it. It can be safely ignored.
Preparing your Mac for X

If you wish to use any software with a graphical interface, you will need an X server. This is a software package that draws on your local screen windows created on a remote computer (such as an NYU HPC cluster).

- Download and install XQuartz

Preparing your Windows workstation for X

If you wish to use any software with a graphical interface, you will need an X server. This is a software package that draws on your local screen windows created on a remote computer (such as an NYU HPC cluster). There are a couple of options out there:

- We recommend Cygwin/X. Instructions for downloading and installing it can be found here. Before starting PuTTY you will need to have the X server running, by double-clicking the "XWin Server" shortcut under Cygwin-X on the Start Menu. You may wish to add this to your Windows Startup folder so it runs automatically after starting Windows
- Another good option is Xming. Installation instructions can be found on its web site. As per Cygwin/X, you will need to launch Xming before starting PuTTY.

You will also need to download and install PuTTY SSH if you have not already.

Who killed my job, and why?

The most likely culprit is the batch system. The other prime suspect is us, the HPC system administrators. If the batch system killed your job, it probably did so because it exceeded the amount of memory or CPU time requested (which may have been some default, if you did not explicitly request these). See Submitting a job - Resource limits for help on this, and this overview of scheduling for a more general introduction. If we killed your job, it was probably to prevent the system from crashing, which can happen when a job's runtime behavior puts a certain type of load on the system. The most common trouble is non-optimal use of the /scratch filesystem, described at the bottom of the table on the Storage page. The next most common reason is that you were running your job on the login node rather than through the batch system - see Running jobs - Nodes for more on this. There are a few circumstances where the login node is the only option - especially archiving files to your $ARCHIVE directory. If you experience trouble with this please contact us. The /scratch filesystem is configured for large-block I/O, such as sequential reading and writing of large files. However, individual I/O operations are relatively costly, so programs using frequent, small I/O accesses will put a heavy load on the metadata servers, which in extreme cases can cause the system to become unstable. The system administrators generally detect this quickly and may kill a job whose I/O characteristics are stretching the capabilities of the filesystem (if this happens, we will contact you to help configure your job for better stability and performance).

I got an email "Please do not run jobs on login nodes"

The login nodes are a shared resource intended for editing scripts, compiling small amounts of code and moving data about. This is enforced via process size limits - if a command on a login node runs for too long or uses too much memory, the system will kill it and send you this email. The likely causes are:

- You are trying to run a simulation interactively instead of via the batch system. Please read Running jobs on the NYU HPC clusters, especially Writing and submitting a job for how to do this. If you need interactive use, read Submitting a job - Working interactively
- You are running a command on the login node that takes longer or needs more memory than you expected. Common causes include:
  - Opening a large (more than 1GB) file with Vim (try using “less” instead)
  - Compiling a large and complex source file
  - Compressing a large file with gzip

In most cases the best solution is to start an interactive session on a compute node, requesting sufficient resource for the task at hand - see Submitting a job - Working interactively

Running Jobs

What resources can and should I request?

At a minimum, the scheduler needs to know:

- How many CPU cores you need, and whether they must be on the same node
  - If you don’t know, the answer is probably “1 core”. If the program supports parallel, it probably supports multithreading - multiple cores on a single node. To use multiple nodes, a program generally needs MPI, so you should only request multiple nodes if you are sure the program can use them.
- How much memory you need
• NYU has nodes with 23GB, 46GB, 62GB, 90GB, 120GB and 189GB available to jobs. (The remaining memory is needed by the operating system)
• How long the job is expected to take
• NYU HPC users can request up to 168 hours (1 week) for a single job. But priority is given to jobs requesting less time

**CPUs - nodes and cores**

HPC is associated with parallel processes - but it's not magic! To use multiple CPUs, the program must have been written with either threading (eg OpenMP) or message passing (MPI).

A multithreaded job (OpenMP) can run on two CPUs in the same node, and share the same memory

If in doubt:
  • try with 1 core first
  • check the documentation of the software, and next try with multiple cores on 1 node
  • when using multiple nodes, check whether the job is actually running on all of the nodes. Contact us for help with this.

**How much do I need?**

The HPC cluster is not magic, its CPUs are only as fast as any other contemporary CPU. In fact, some nodes on Mercer are a few years old, and may be slower than your desktop (see Clusters July 2017 for a table of node types and when we installed them).

The performance of the HPC cluster comes from its scale. Most nodes in the cluster have 12 or 20 cores, 48GB up to 192GB of RAM, access to a large fast parallel filesystem and there is a 40Gb/s dedicated network link between any two nodes in each of the main groups. And there are thousands of nodes.

So although the resources your job needs depend very much on your job, and there is no simple rule for estimating requirements, you can make some initial guesses based on why you need the HPC cluster:

• My desktop has not enough RAM
  You should request at least as much RAM as your desktop possesses. The time required will probably be similar to the time required on your desktop. Be aware though that many problems scale with $O(n^2)$ or more, so doubling the number of data points might require 4x the RAM and 8x the compute time.
• Each run takes 4 hours on my 32GB desktop, and I have 1000 experiments to run
  Each experiment will probably take 32GB of memory and 4 hours on the HPC cluster too - but you can submit 1000 of them at once and a few hundred might run simultaneously
For a few one-off jobs, you can safely request much more than you need. But use these qsub/PBS options:

```
#PBS -mae               # send an email if the job aborts (a) and when it ends (e)
#PBS -M $USER@nyu.edu   # send the email to this address
```

In the email you will see something like:

```
PBS Job Id: 3205348
Job Name: testme.q
Exec host: compute-1-10.local/7
Execution terminated
Exit_status=0
resources_used.cput=00:32:01
resources_used.mem=3052mb
resources_used.vmem=25644mb
resources_used.walltime=00:42:03
Error_Path: login-0-2.local:/home/ab123/testing/testme.q.e3205348
Output_Path: login-0-2.local:/home/ab123/testing/testme.q.o3205348
```

From which your can deduce that the job took 42 minutes of wallclock time and about 3GB of memory. So a sensible resource request for the next job is (see RESOURCES for more about the options):

```
qsub -l walltime=1:00:00 -l mem=4GB testme.q
```

For further details, see Submitting a job - Resource limits

Can I make sure a job gets executed only after another one completes? A: Yes

Options for delaying starting a job:

- `-W depend=afterok:jobid`
  - Delay starting this job until jobid has completed successfully.
- `-a [MM][DD]hhmm`
  - Delay starting this job until after the specified date and time. Month (MM) and day-of-month (DD) are optional, hour and minute are required.

If you have jobs which must not be started until some other job has begun, completed or failed, you can set up a job dependency with `-W depend=dependency`. The most common dependency is "afterok", the job can start only after another job has completed with an exit code of zero (ie no errors). For example, to wait until job 12345 has completed successfully before starting:

```
$ qsub -W depend=afterok:12345 testme.q
```

Other types of job dependencies, including dependencies on job arrays, can be found with "man qsub".

Job dependencies are useful when you have one job postprocessing the results of other jobs. Job dependencies also allow you to reserve resources more efficiently: such as single-CPU postprocessing of the result of a large parallel job.

Similarly, you can instruct Torque to not start a job before a given date and time by the command line option or PBS directive `-a hhmm`. The date/time format accepts day, month and year too. For example, to delay starting a job until midday on the first day of the next month:
Where did my job output go?

On earlier NYU HPC clusters, the output of batch jobs appeared as it happened, in a file like "my_job.o12345" in the job submission directory. When jobs are submitted from /scratch, this can cause a heavy small-block-I/O load on the Lustre filesystem, which impacts /scratch performance.

On Mercer, job stdout and stderr are instead written to hidden files in your $HOME area, under $HOME/.pbs_spool. When the job completes, the my_job.o12345 and my_job.e12345 files are moved to their final location.

You can monitor your job's progress by looking for its output in this hidden directory.

If your job writes too much to stdout or stderr, these temporary files might (temporarily) fill up your $HOME space allocation. If this is occurring, first check if you can reduce the amount of stdout and stderr (they can slow down your program, contact us if you would like assistance with this). Another option is to redirect stdout and stderr to a file in some other location:

- With bash:
  ```
  my_program > my_output_file 2>&1
  ```

- With csh/tcsh
  ```
  my_program >& my_output_file
  ```

How do I use GPUs?

To request GPU nodes:

- `-l nodes=1:ppn=1:gpus=1`
  1 node with 1 core and 1 GPU
- `-l nodes=1:ppn=1:gpus=1:titan`
  1 node with 1 core and 1 GPU, specifically a Titan Black GPU
- `-l nodes=1:ppn=1:gpus=1:k80`
  1 node with 1 core and 1 GPU, specifically an Nvidia K80 GPU
- `-l nodes=1:ppn=1:gpus=4:titan`
  1 node with 4 Titan GPUs. Note that we request ppn=4 too, it is always best to request at least as many CPU cores are GPUs

The available GPU node configurations are shown here.

When you request GPUs, the system will set two environment variables - we strongly recommend you do not change these:

- `CUDA_VISIBLE_DEVICES` has a comma-separated list of the device IDs this job is allowed to use (eg "2,3"). The CUDA library within the application will use this to prevent multiple GPU jobs on the same node from interfering with each other
- `CUDA_DEVICES` has a zero-based sequence of the "logical device IDs" for your job (eg "0 1"). So, if your application expects a list of GPU IDs starting at zero, and you have been allocated GPU numbers 2 and 3, then you can pass $CUDA_DEVICES to your application and it will see 2 devices, named "0" and "1", which happen to correspond (via $CUDA_VISIBLE_DEVICES) to the GPUs whose physical IDs are "2" and "3"

To your application, it will look like you have GPU 0,1... (up to as many GPUs as you requested). So if for example, you request 2 GPUs, and are allocated GPU 2 and GPU 3, you will have:

```bash
# echo $CUDA_VISIBLE_DEVICES
2,3
# echo $CUDA_DEVICES
0,1
```
Now if your application calls "cudaSetDevice(0)", you will use the GPU that appears as device 0, but is actually device 2.

And a call to "cudaSetDevice(3)", will return an error, because as far as the application can see, the node only has 2 GPUs, numbered 0 and 1.

How do I log in to a specific node?

A1: You can ssh to a specific login node. The login nodes on bowery are named login-0-0, login-0-1, login-0-2, login-0-3.

```bash
$ ssh login-0-0
```

A2: You can ssh to a specific compute node if you have a job running on it. To find out which nodes your job is running on use:

```bash
$ qstat -n jobid
```

You will see the usual qstat output followed by a list of the nodes and cores your job is allocated to. The list will look something like:

```
... core 11, core 12 on node compute-6-3
... core 13, core 14 on node compute-6-4
```

In this example, the list shows cores 11 and 10 on node compute-6-3.

How can I ensure my resource-intensive job is running smoothly?

A: After submitting jobs, you will be able to locate where your jobs are executing by running `pbstop -u NetID`. Then you can monitor these jobs by logging in to the corresponding compute nodes and running `top`. You will then see both CPU and memory consumptions. If you find little memory left (or even little swap left) due to your job, you should increase the "ppn" number in your PBS script or maybe consider taking advantage of the nodes with larger memory.

Error rendering macro 'excerpt-include': User 'null' does not have permission to view the page 'Interpreting pbstop'.

(my...)

My job will take longer than 48 hours, what should I do?

For long-running jobs, we suggest using Checkpoint-Restart to split the job into a series of shorter jobs. We realize though that this is not always possible - if you need the walltime limit for a job extended, contact us.

My job needs (MySQL, some other service) to be running

Contact us.

I want to run a job at 9am every day

(still to come)

Using Software

There are usage examples for many popular software packages in `/share/apps/examples` on Mercer:

- batch - An example batch job
- blcr - Checkpoint-Restart facility for long jobs
- csharp - Computational Fluid Dynamics
- fluent - Computational Fluid Dynamics / Multiphysics package
- gaussian - Chemistry package
- matlab - For mathematical exploration
- namd - Molecular dynamics
- qchem-amber - Molecular dynamics
- r - Interpreted language for statistics work
- resource-usage - Shows minute-by-minute CPU and memory usage of a program
- stata - Statistics package

How do I run a STATA job?

A: To run STATA jobs, you need to;

(1) Prepare a STATA do file, such as "stata-test.do" which may include,

```stata
sysuse auto, clear
summarize
graph twoway (scatter mpg weight) (lfit mpg weight)
graph export stata-test.ps, replace
```

StataMP was also installed on the USQ, which is the parallel version of Stata. To use multiple processors in StataMP, just insert "set processors X" as a line in your do file, where X is the number of processors and it should be equal to the "ppn" number in your PBS script.
(2) Create a PBS script "run-stata.pbs" to run STATA jobs in the batch mode. The content in this file can be like this,

```bash
#!/bin/csh -f
#PBS -V
#PBS -S /bin/tcsh
#PBS -N stata-test
#PBS -l nodes=1:ppn=1,walltime=01:00:00
#PBS -M NetID@nyu.edu
#PBS -m abe
source /etc/profile.d/env-modules.csh
module load stata/11
cd /scratch/NetID/hpc-tutorial/stata
stata -b do stata-test.do
```

**Note**
Be sure to substitute your own "NetID" for NetID.

You need to change the paths according to your directory. Please refer to this page for more information.

(3) Then submit the job by typing

```bash
$ qsub run-stata.pbs
```

**How do I run a Gaussian job?**

A: To run a Gaussian job, you need to prepare a Gaussian input file which might look like this,

```text
%Chk=H2.chk
#SP B3LYP/6-31+G*
No title
H 0.00000 0.00000 0.00000
H 0.75000 0.00000 0.00000
```

and save it as "input.com".

You can run it from an interactive session if you expect your jobs to finish very soon. The command is,

```bash
$ /share/apps/gaussian/G03-E01/intel/g03/run-g03.csh input.com >& output.out
```

Please use this script instead of running Gaussian by loading the module "gaussian/intel/G03-E01" and then executing `g03` directly, because otherwise Gaussian will write scratch files to the default path and the system space might be filled up.

You may copy the "run-g03.csh" script to any directory and even rename it for your convenience.

**(more...)**

**How do I run a Matlab job?**

You may run serial Matlab jobs from an interactive session or the batch mode. In both ways, you need to either load the module as,

```bash
module load matlab/2014a
$ matlab
```

**(more...)**

**How do I start a multinode parallel job that is NOT MPI (eg Julia)?**

The MPI modules on Mercer are built with support for the batch system. However, third-party MPI libraries and parallel software (such as Julia) may not be.

To launch a Julia job on multiple nodes you can use the node list provided by `$PBS_NODEFILE`

```bash
julia --machinefile $PBS_NODEFILE ./my_code.jl
```

For an MPI job that does not use the MPI modules on Mercer:
mpirun -np --machinefile $PBS_NODEFILE ./my_mpi_exec.exe

To run one multithreaded MPI process per node (hybrid MPI/OpenMP), see Running jobs - MPI.

Parallel libraries other than the MPI modules on Mercer do not normally have support for Torque, consequently they do not play nicely with other jobs. For this reason, the queue for multinode jobs sets the "#PBS -n" (node-exclusive) option. Multinode jobs therefore do not share nodes with any other job.

(thanks Spencer for the Julia tip!)

How can I view a PDF file on Mercer?

\texttt{evince myfile.pdf}

You need to have logged in with X forwarding enabled, as evince is an X application. See Logging in to the NYU HPC Clusters for how to do this.

Managing Data

How much of my file/space quota have I used?

On Mercer, enter 'myquota' at the prompt to see how much space you have used and available on each filesystem

How do I give my colleague access to my files?

An access control list (or ACL) gives per-file, per-directory and per-user control over who can read, write and execute files. \textbf{You can see the ACL for a file or directory with the }\texttt{getfacl} command:

\begin{verbatim}
$ getfacl myfile.txt
\end{verbatim}

To modify permissions for files or directories, use \texttt{setfacl}. For a detailed description, see 'man setfacl'. In the example below, I give read permission on \texttt{myfile.txt} to user \texttt{bob123}:

\begin{verbatim}
$ setfacl -m u:bob123:r myfile.txt
\end{verbatim}

For setting execute permission on files \texttt{chmod} (more...)

How do I get the best transfer speed to or from BuTinah?

For faster transfer between the HPC clusters at NYU in NYC and the BuTinah cluster at NYUAD, use \texttt{scp} over port 922. This will route the transfer over a high bandwidth ANKABUT link rather than the default low bandwidth MLPS link. The speed difference is greatest when pulling files from BuTinah to NYU NY.

Transferring many small files will still be slow - \textbf{you will get better performance if you tar small files into a single archive}, and transfer the tar file.

The default user environment on bowery sets an alias for \texttt{scp} which does this automatically, so in most cases you can skip over this section. If you are finding that file transfers between NYUAD and NYU are slow, you can check whether you are using the alias with 'which scp'. If the response is not '/usr/local/bin/scp_wrapper.sh', you should follow the instructions below.

You can \texttt{scp} over port 922 directly with the following commands, initiated from any of the NYU HPC clusters in NYC:

\textbf{Pushing to BuTinah:}

\begin{verbatim}
$ scp -P 922 filename NetID@butinah.abudhabi.nyu.edu:/
\end{verbatim}

\textbf{Pulling from BuTinah:}

\begin{verbatim}
$ scp -P 922 NetID@butinah.abudhabi.nyu.edu:~/filename
\end{verbatim}

(more...)

I have a huge amount of data that I want to compress for storage or transfer

Mercer has 'pigz', which is a parallel version of gzip. To use it:
My workflow uses thousands of small files, how should I manage them

Managing large numbers of files

Filesystems generally - and high-performance filesystems such as Lustre especially - perform best with a small to moderate number of medium to large files. Some specific issues to be aware of are:

On $SCRATCH

- Lustre ($SCRATCH) gets performance mainly by striping distributing a large file across several disks and several "object storage servers". File metadata operations, on the other hand, do not have much parallelism available. So a few large read or write operations is vastly faster than many small reads or writes. This is true for reads and writes within a single files as well as for reads or writes on many files.
  - If your job does many small I/O operations, it might be better to copy the file to $PBS_JOBTMP or $PBS_MEMDISK at the start of the job, and open the local copy of the file.
  - (But for large reads and writes, $SCRATCH is likely to be faster than local disk)

- The default stripe count on $SCRATCH is 4, so each file is striped across disks on 4 object storage servers. If you have a folder filled with files each smaller than 1MB, it is better not to stripe them. You can set the stripe count on a folder (under $SCRATCH) to 1 with:

  lfs setstripe -c 1 $SCRATCH/folder_with_small_files/

- Finding a file within a folder is a serial operation. And the more files in a folder, the longer it takes. With several thousand files, even 'ls' on that folder can take several minutes and may affect responsiveness of the filesystem for other users.
  - If you have more than about 1000 files in one folder, distribute them over a number of subfolders. The best performance will be when the number of subfolders is the square root of the total number of files (eg, for 1 million files, 1000 subfolders each containing 1000 files)

On $ARCHIVE

- The backup regime on /archive is optimized for small numbers of large files - 1000 files of 1kb each take 1000 times as long to backup as 1 file of 1MB! Too many small files can prevent the backup from completing in time
  - when archiving a collection of small files, please tar the files first. You can send a collection of files to /archive with the command:

  # to archive $SCRATCH/my_run_dir
tar cvf $ARCHIVE/simulation_01.tar -C $SCRATCH my_run_dir

  And fetch it again with:

  # to unpack it back into $SCRATCH/my_run_dir
tar xvf $ARCHIVE/simulation_01.tar -C $SCRATCH

I want to keep a folder on the HPC cluster in sync with a folder on my workstation

- To replicate on Mercer a data directory you have on your workstation - assuming you are entering these commands on a local terminal on your workstation, and you have an SSH Tunnel set up and running:
The host name followed by a colon tells rsync that the (in this case) destination is on another host. If your username on the other host is different to the username on the current host, you can specify the remote username with `username@remotehost`:

**Note** the backslash in `\$USER` - this instructs the shell not to expand `\$USER` to your local (on your workstation) username. An equivalent command is:

```
$ ls -F
my_input_data/
$ rsync -av my_input_data mercer:/scratch/\$USER/my_run_dir
```

- To copy in the other direction, from `/scratch` on Mercer to your workstation (again, from a local terminal on your workstation and across an SSH Tunnel):

```
$ hostname
my_workstation
$ ls -F
my_input_data/
$ rsync -av my_input_data NetID@mercer:/scratch/NetID/my_run_dir
```

Only those files not already up-to-date on your workstation will be copied.

Click here for more about using rsync on the NYU HPC clusters

**How do I transfer files to Dumbo cluster from Windows workstation?**

First, download and install WinSCP tool from [here](#). If you are inside NYU network (on campus), simply open WinSCP and fill in all the fields:

If you are outside of NYU network, one option is to set up and use VPN. After that you can use WinSCP as described above. Another option is to start an [SSH tunnel](#) on the workstation. We have instructions on how to do this for Windows workstations. Once your SSH tunnel is set up and started on dumbo, open WinSCP and fill in the fields as shown below: