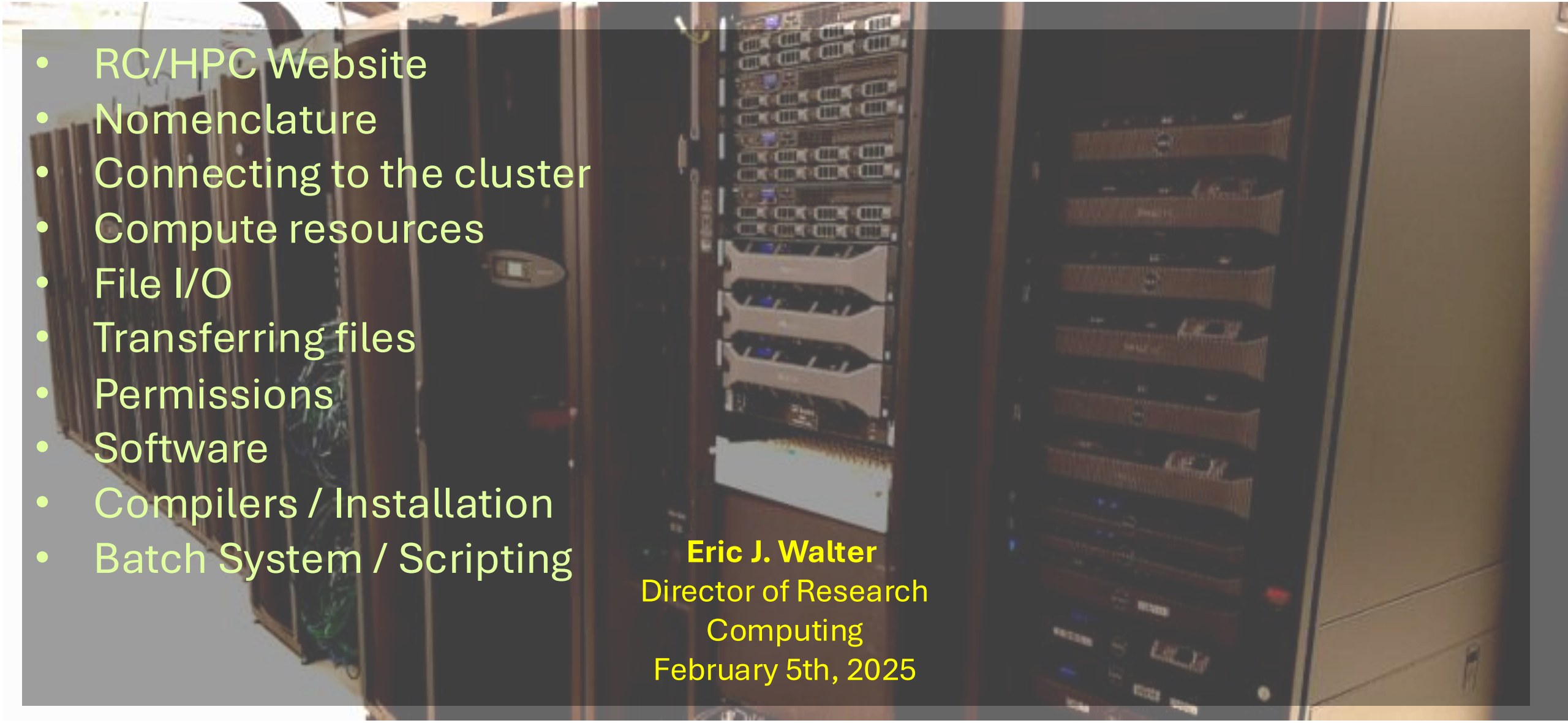
Guide to Computational Resources at W&M/VIMS



- RC/HPC Website
- Nomenclature
- Connecting to the cluster
- Compute resources
- File I/O
- Transferring files
- Permissions
- Software
- Compilers / Installation
- Batch System / Scripting

Eric J. Walter
Director of Research Computing
February 5th, 2025

Using HPC / Web Docs

What you need to get started

https://www.wm.edu/offices/it/services/researchcomputing/using/

Using HPC

Obtaining an account

Unlike many other Information Technology services, HPC access is by request only. If you have not yet obtained an account, or your account has expired, please submit an account request.

Prerequisites

You will need to be comfortable using a Unix/Linux command-line after logging in with SSH.

Logging in

The subcluster pages will tell you which "front-end" server to log in to, depending on which hardware you want to use. Generally, you must log in to the HPC systems from the campus network (at William & Mary or VIMS), via the university's VPN, or via a host that is on the campus network (i.e. the W&M bastion host) or you will see errors like `Connection timed out` or `Network is unreachable`. Chesapeake is behind VIMS' (more restrictive) firewall and from W&M must be accessed via the W&M bastion host or be logged into the main-campus cluster already. Please see "Logging in to HPC systems" for more information.

Running calculations

The login servers are called "front-ends" because you do not run your calculations there, but rather on back-end "compute" servers that the front-end server provides access to. Access compute servers via the SLURM batch system.

In order to use installed software, you must generally "load" it using Environment Modules, or you will see errors like `Command not found`. We have specific guidance for users of MATLAB, Python, and other software under our Tutorials and Software pages, as well as for users compiling software themselves.

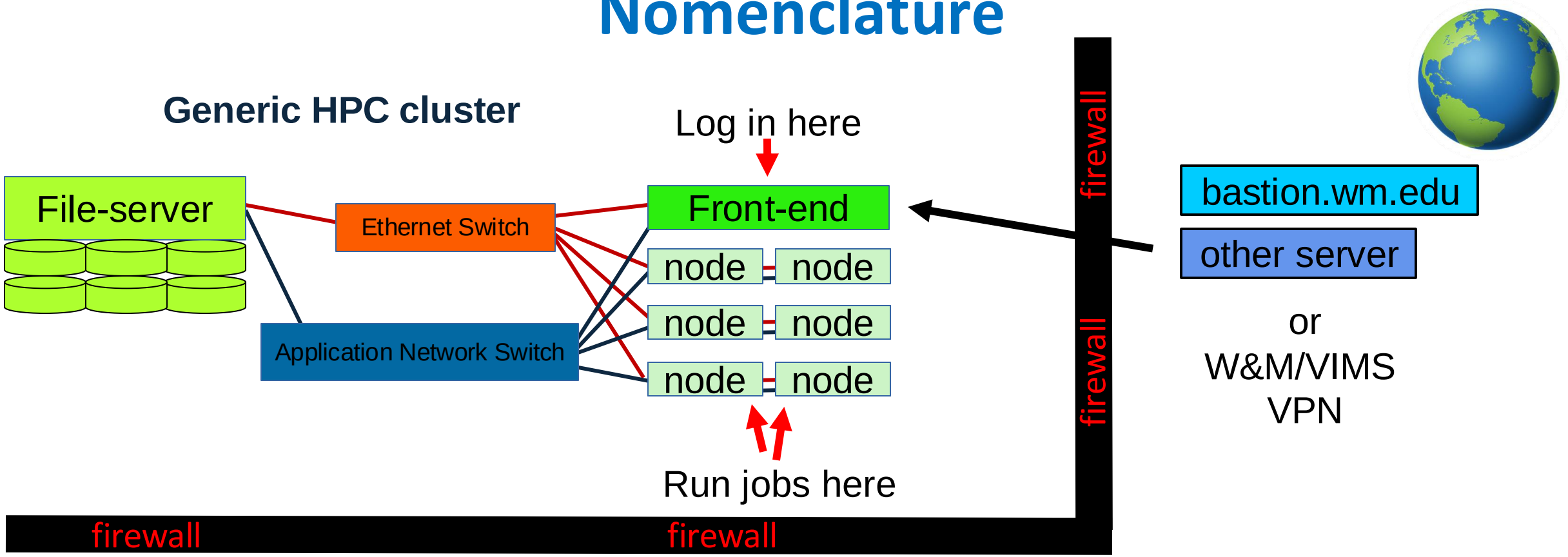
If you need to work with or produce more than a few gigabytes of data, familiarize yourself with filesystems other than your home directory, and with preventing your disk usage from disrupting others' work.

When you are finished

The HPC systems cannot provide archival or long-term storage. If files no longer need to be available for work on the system, copy them off and delete them so that the space can be used for active projects. **All files will be completely and permanently deleted after your HPC account expires**, so if your files need to remain available for work on the system, keep track of when your account will expire, and before it expires either renew your account or contact us to arrange to have your files reassigned to another user.

1. Getting an account
2. Linux command line / text editors
3. Logging into the clusters
4. Selecting/installing/requesting software
5. How to use file-systems effectively
6. How to use to use the batch system(s)
7. Compiling / installing your own applications
8. Saving your own files/projects

Nomenclature



Must get through **W&M/VIMS firewall** to get to cluster resources
On campus you are already within firewall
Otherwise use **bastion.wm.edu**, another server, or VPN

Connecting to HPC

ssh – is the standard app for connecting to a remote computer with Linux
Standard software on Linux. Mac also supports ssh
via ***terminal***. Windows users can use ***powershell*** to do this or use ssh
client program (***putty***) and a separate scp client (***WinSCP***).

For on W&M/VIMS campus or within W&M/VIMS VPN

ssh to bora - ***ssh ejwalt@bora.sciclone.wm.edu***
ssh to james - ***ssh ejwalt@james.hpc.vims.edu***

From off-campus – need to jump through bastion host

ssh to bora through bastion host - ***ssh -J ejwalt@bastion.wm.edu ejwalt@bora.sciclone.wm.edu***
ssh to james through bastion host - ***ssh -J ejwalt@bastion.wm.edu ejwalt@james.hpc.vims.edu***

Can configure your local ssh to jump through bastion host automatically:

My off-campus ~/.ssh/config file:

```
Host bora.sciclone.wm.edu
  HostName bora.sciclone.wm.edu
  ProxyJump ejwalt@bastion.wm.edu
  User ewalter
```

Is my username the same on my local machine?

If it is different use: ssh <username>@<host>.<domain>

Do I need graphics?

If yes, must log in with –Y

See website for more in depth help:

https://www.wm.edu/offices/it/services/researchcomputing/using/connecting/ - connecting/logging in to HPC
https://www.wm.edu/offices/it/services/researchcomputing/using/xfers/ - transferring files to/from HPC

Open Compute Resources

Main Campus

Name (front-end/nodes)	Processor	NVIDIA GPU	Cores /node	Total # cores	Mem/node (GB)	Deployed
Bora / bo01-bo55	Intel Xeon E5-2640	--	20	960*	128	2017
Femto / fm01-fm30	Intel Xeon 6130	--	32	960	96	2019
Kuro / ku1-ku47	AMD EPYC 9334	--	64	3008	384	2024
Bora / hi01-hi07	Intel E5-2683	P100/V100**	32	224	256	2017
Gust / gt01-gt02	AMD EPYC 7702	--	128	256	512	2020
Astral / as01	Intel Xeon 8362	8x A30	64	64	512	2022
Gulf / gu01-gu02	AMD EPYC 7313P	--	16	32	512	2024
Gulf / gu03-gu06	AMD EPYC 7313P	2x A40/node	32	128	128	2024

VIMS Campus

Chesapeake/pt01-pt30	AMD Opteron 4334	--	12	180*	128	2014
James/pm01-pm02	AMD Opteron 6380	--	64	128	256	2016
James/jm01-jm27	Intel Xeon 4114	--	20	540	64	2018

* - not all nodes functional

** - hi04, hi05 have 1x P100, hi07 has 1x V100

Files I/O

Web Documentation

https://www.wm.edu/offices/it/services/hpc/using/files/index.php

- There are multiple files-systems available
- Some are for ongoing / project storage

  data, home
- Some are for running jobs (90 day purge)

  scrXX, pscr, /local/scr
- **Only** data/home/home00 backed up
- Use **local scratch** when possible (every node has some)
- Users are responsible for using disk space **responsibly**
- Misue can disturb other jobs / cause **administrative action**
- Don't use home for writing or large reads
- Use scratch space for jobs
- Lustre (pscr / scr-lst) best practices:

https://www.wm.edu/offices/it/services/researchcomputing/using/files/lustre/index.php

Name	Appropriate for	Backups	Purged	Per-formance
`/sciclone/home` `/ches/home00`	Source code, executables, configuration files, scripts, and small (<100MB total) data files. Unless we have directed you otherwise, **you should not have a job read or write any substantial amount of data to your home directory**, as doing so is extremely likely to impact others' interactive work.	Weeknightly, on-site only	After account expiration.	Low
`/sciclone/data10` `/ches/data10`	Input data files that are needed on an ongoing basis for active projects on the cluster and cannot be easily re-created or re-uploaded. **Please do not have jobs write a substantial amount to data filesystems.** Please use the scratch filesystems (below) for job output unless already given permission from HPC staff.	Weekly, on-site only		

`/local/scr`
`/ches/scr10`
`/sciclone/scr10`
`/sciclone/scr20`
`/sciclone/scr-mlt`
	Scratch space: job outputs and working data that can be easily re-created or re-uploaded, or which will be copied elsewhere for longer-term storage.	**Never**	**Any files not accessed for 90 days**, and after account expiration.	Medium

`/sciclone/pscr`				High

Home / Data are backed up anything with **"scr" is scratch** (purged after 90 days of inactivity)

Transferring files

Web Documentation

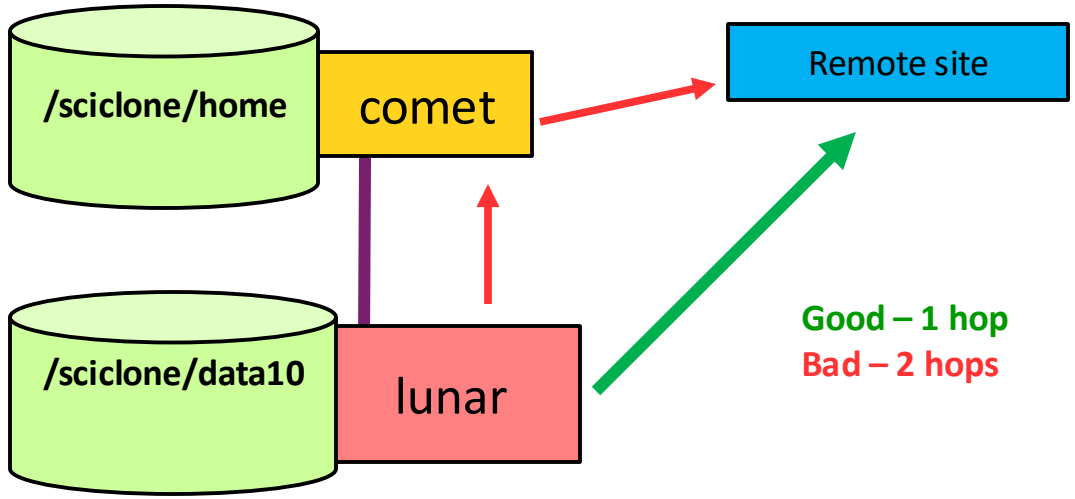
https://www.wm.edu/offices/it/services/researchcomputing/using/files/xfers/index.php

Filesystem	Hostname
/sciclone/home	comet.sciclone.wm.edu
/sciclone/data10	lunar.sciclone.wm.edu
/sciclone/pscr	bora.sciclone.wm.edu
/sciclone/scr10	polar.sciclone.wm.edu
/sciclone/scr20	orbit.sciclone.wm.edu
/sciclone/scr-lst	kuro.sciclone.wm.edu
/sciclone/schism10	snow.sciclone.wm.edu
/sciclone/gluex10	sleet.sciclone.wm.edu
/sciclone/scr-mlt	mistral.sciclone.wm.edu
/ches/home00	james.hpc.vims.edu
/ches/data10	choptank.hpc.vims.edu
/ches/scr10	rappahannock.hpc.vims.edu

Each file-system has a server that runs it
For direct access you are **STRONGLY** encouraged to use the recommended node

e.g. : Logged into **comet**; cd'd into data10 ; transfer off-site

Do this from **lunar** since files won't have to hop through **comet** to get off-site.



Globus - https://www.wm.edu/offices/it/services/researchcomputing/using/filesandfilesystems/xfers/globus/
We have endpoints for all file-systems

Permissions / sharing files

Want to allow users in the VASP group to read my results file (out)

```
44 [astral] pwd                                                    (where am I?)
/sciclone/home/ewalter

45 [astral] ls -ld results                                  (long list just <dir>)
drwx------ 2 ewalter hpcf 4096 May  6 12:27 results

                                                        (long list <dir> contents)
46 [astral] ls -l results
total 28
-rw------- 1 ewalter hpcf 25905 May  6 12:27 out

47 [astral] groups ewalter                                 (what group am I in?)
ewalter : hpcf wheel hpcstaff hpcadmin sysadmin www seadas
vasp wm wmall hugepage

                                                                (change group)
48 [astral] chgrp vasp -R results/

49 [astral] ls -ld results
drwx------ 2 ewalter vasp 4096 May  6 12:27 results

50 [astral] ls -l results
total 28
-rw------- 1 ewalter vasp 25905 May  6 12:27 out

                                                          (change group
51 [astral] chmod g+rX -R results                         permissions
                                                          recursively)
52 [astral] ls -ld results
drwxr-x--- 2 ewalter vasp 4096 May  6 12:27 results

53 [astral] ls -l results
total 28
-rw-r----- 1 ewalter vasp 25905 May  6 12:27 out
```

d-directory
r-read
w-write
x-execute/enter

pwd – print working directory
groups – print groups that user is in
chgrp – change **group** ownership
chmod – change **permissions**
chown – change user ownership

umask – controls default permissions
- change in .cshrc/.bashrc

Whole path needs to be accessible to share!

see http://linuxcommand.org/lc3_lts0090.php for more information

Software

There are many software packages available on the HPC systems!

Common packages are all available: Python, R, Gaussian16, Matlab, etc.

- Check the modules on a particular cluster with: “module avail”
- Install it yourself
- Email hpc-help@wm.edu

We encourage **users to install their own software** in their home directory if possible
We can help install, but we get **LOTS** of requests so try not to abuse
Packages used by multiple users can be considered for installing globally

Much of our support is related to getting software working on the cluster

Software Modules

Sets up environment for a software package

```
11 [bora] module avail
------------------------------------ /usr/share/Modules/modulefiles ------------------------------------
dot  module-git  module-info  modules  null  use.own

------------------------------------ /usr/local/Modules/modulefiles ------------------------------------
bowtie2/gcc-11.4.1/2.5.4           netcdf-c/gcc-11.4.1/4.9.2_openmpi          proj/gcc-11.4.1/5.2.0
cuda/12.3                          netcdf-c/intel-2024.0/4.9.2_intelmpi       python/gcc-11.4.1/3.11.9
hdf5/gcc-11.4.1/1.14.3_openmpi     netcdf-c/intel-2024.0/4.9.2_openmpi        r/gcc-11.4.1/4.4.0
hdf5/intel-2024.0/1.14.3_intelmpi  netcdf-fortran/gcc-11.4.1/4.6.1_openmpi    slurm/23.11.9
hdf5/intel-2024.0/1.14.3_openmpi   netcdf-fortran/intel-2024.0/4.6.1_intelmpi solps/3.0.8
legacy-tools/gcc-8.5.0             netcdf-fortran/intel-2024.0/4.6.1_openmpi  solps/3.0.9
legacy-tools/gcc-9.5.0             openblas/gcc-11.4.1/0.3.27                 solps/3.0.9_omp
ncl/gcc-11.4.1/6.6.2_openmpi       openmpi-ib/gcc-11.4.1/4.1.6
nco/intel-2024.0/5.2.4             openmpi-ib/intel-2024.0/4.1.6

------------------------------------ /sciclone/apps/modulefiles ------------------------------------
comsol/6.2          intel/2019              intel/mpi-2021.7.1  matlab/R2023a                  stata/18.0
gaussian/g16        intel/2019-mpi          intel/mpi-2021.11   matlab/R2024a
gaussian/g16_vC02   intel/compiler-2022.2.1 intel/tbb-2021.7.1  miniforge3/24.9.2-0
hyperworks/24       intel/compiler-2024.0   julia/1.9.4         nvidia/nsight-compute-2024.3
hyperworks/24.1     intel/mkl-2024.0        matlab/R2020b       python/3.12.7

Key:
loaded  default-version  modulepath
12 [bora]
```

Can change modules on demand: module load/unload
Also list what the module sets: module show
Can even write your own modules to make custom environments

https://www.wm.edu/offices/it/services/researchcomputing/using/modules/index.php

Startup modules

In user home directories, there are startup files which control default modules (bash users use .bashrc.XXX)

Name	Use
.cshrc	General startup options for site
.cshrc.kuro	Kuro
.cshrc.femto	Femto
.cshrc.bora	Bora & Hima
.cshrc.astral	Astral
.cshrc.gust	Gust
.cshrc.gulf	Gulf
.cshrc.james	James
.cshrc.potomac	Potomac
.cshrc.pamunkey	Pamunkey

```
11 [astral]  echo $PLATFORM
astral
```

$PLATFORM variable

This means that startup is controlled by ***.cshrc.astral*** for ***astral cluster***

Compilers and Installation

Web docs: https://www.wm.edu/offices/it/services/researchcomputing/using/compiling/index.php
https://www.wm.edu/offices/it/services/researchcomputing/using/software/index.php

- All clusters are equipped with GNU and Intel compilers
- Two popular flavors of MPI supported (Intel & OpenMPI)

Compiler web page lists suggested compiler flags for each node type and best practices guides. It is **extremely** important to check the validity of results.

Don’t assume if the job runs correctly, it has correct results!

Batch System / SLURM

All calculations should be done on a node through the batch system

Front-end/login servers are for file manipulation, transferring files, submitting jobs, etc.

There are two types of jobs:

Interactive – give me a session on a node/nodes

Batch – run a script with a set of commands on a node/nodes

sinfo – get node info

squeue – get batch queue info

```
2 [kuro] sinfo
PARTITION AVAIL  TIMELIMIT  NODES  STATE NODELIST
batch*       up 2-00:00:00      5   resv ku[05-09]
batch*       up 2-00:00:00     42  alloc ku[01-04,10-47]
debug        up      30:00      4  alloc ku[44-47]
3 [kuro] squeue
             JOBID PARTITION     NAME     USER ST       TIME  NODES NODELIST(REASON)
             18215     batch  5a3d125     ncai PD       0:00     10 (ReqNodeNotAvail, Reserved for maintenance)
             18226     batch   5a3s-2     ncai PD       0:00     10 (ReqNodeNotAvail, Reserved for maintenance)
             18227     batch  5a3s025     ncai PD       0:00     10 (ReqNodeNotAvail, Reserved for maintenance)
             18228     batch   5a3s-4     ncai PD       0:00     10 (ReqNodeNotAvail, Reserved for maintenance)
             18229     batch   5a3s05     ncai PD       0:00     10 (ReqNodeNotAvail, Reserved for maintenance)
             18230     batch    5a3s1     ncai PD       0:00     10 (ReqNodeNotAvail, Reserved for maintenance)
             18238     batch s0.25d0.     ncai PD       0:00     10 (ReqNodeNotAvail, Reserved for maintenance)
             18239     batch s0.25d0.     ncai PD       0:00     10 (ReqNodeNotAvail, Reserved for maintenance)
             18240     batch s0.25d1.     ncai PD       0:00     10 (ReqNodeNotAvail, Reserved for maintenance)
             18241     batch s0.25d1.     ncai PD       0:00     10 (ReqNodeNotAvail, Reserved for maintenance)
             18242     batch s0.25d1.     ncai PD       0:00     10 (ReqNodeNotAvail, Reserved for maintenance)
             18265     batch u05a3s07     ncai PD       0:00     10 (ReqNodeNotAvail, Reserved for maintenance)
             18268     batch  pred021   dluo01 PD       0:00      1 (ReqNodeNotAvail, Reserved for maintenance)
             18273     batch s1.5d0.9     ncai  R   22:00:10     15 ku[25-39]
             18272     batch s1.5d0.8     ncai  R   22:00:43     15 ku[10-24]
             18276     batch s1.5d1.5     ncai  R    8:21:56     12 ku[01-04,40-47]
4 [kuro]
```

https://www.wm.edu/offices/it/services/researchcomputing/using/running_jobs_slurm/

Interactive SLURM jobs

You must log into the appropriate front-end to run a job on a cluster
To get an interactive session within SLURM: salloc

`salloc –N1 –n1 –t 1:00:00`	get one node and one core for 1 hr on this cluster
`salloc –N1 –n20 –t 3:00:00`	get one node and 20 cores for 3 hrs on this cluster
`salloc –N4 –ntasks-per-node=20 –t 1-0`	get 20 cores on 4 nodes (80 cores) for 1 day on this cluster
`salloc –N1 –n32 –t 30:00 --gpus=1`	get 32 cores and a GPU on one node or 30min on this cluster

```
14 [bora] salloc -N1 -n20 -t 1:00:00
salloc: Granted job allocation 18955
salloc: Nodes bo03 are ready for job
1 [bo03] ./a.out
1 [bo03]
1 [bo03]
1 [bo03]
```

```
18 [bora] salloc -N1 -n32 -t 30:00 --gpus=1
salloc: Granted job allocation 18958
salloc: Nodes hi07 are ready for job
2 [hi07] nvidia-smi
Tue Feb  4 18:39:48 2025
+-----------------------------------------
| NVIDIA-SMI 555.58.02              Driver Ver
|-----------------------------------------+---
| GPU  Name                 Persistence-M | Bu
```

Interactive jobs are good for debugging and short calculations
Not good manners to keep resources in an interactive state if you are not actually using them
Network connection must be maintained – can use multiplexer (screen)

Batch SLURM jobs

Priorities for resources are largely controlled by ***fairshare*** principles – your priority goes down with more use
Can ssh into nodes you are running jobs on

To run a ***batch*** job you will need to create a ***batch script***:

```
#!/bin/tcsh                          "hash bang"  which shell syntax to run (here tcsh)
#SBATCH --job-name=serial            Job Name
#SBATCH -N1 -n1                      # nodes , # cores
#SBATCH -t 0:30:00                   walltime (30min)

./a.out                              run the program
```

```
#!/bin/tcsh                          "hash bang"  which shell syntax to run (here tcsh)
#SBATCH --job-name=kurotest          Job Name
#SBATCH -N 5 --ntasks-per-node 64    # nodes , # cores per node
#SBATCH -t 1-0                       Walltime (1 day)

srun ./a.out  >&  log                Run the parallel program with srun (passes topolog
                                     Also redirect stdout, stderr to a file named "log"
```

SLURM advanced example

```
24 [bora] cat run
#!/bin/tcsh
#SBATCH --job-name=get_stl2
#SBATCH --nodes=1 --ntasks=1
#SBATCH --time=1:00:00
#SBATCH --gpus=1

module load miniforge3
conda activate testenv
which python

foreach i (`cat list`)
    echo $i > INPUT
    python run.py >& out.$i
end


25 [bora]
```

Name of job
Serial job
1hr
1 GPU

Load the miniforge3 module
Activate my "testenv" environment
which python am I running? (which gives the path of the executable)

Foreach loop
`xxx` means evaluate xxx
Run the run.py script (takes INPUT as a parameter)
Also send the output of each run to out.<parameter>

```
26 [bora] cat list
1.0
1.1
1.35
2.0
27 [bora]
```

```
88 [bora] sbatch run
Submitted batch job 18965
89 [bora] squeue
             JOBID PARTITION     NAME     USER ST       TIME  NODES NODELIST(REASON)
             18920     batch   run081 bmaldona  R    6:52:38     25 bo[06-08,10-31]
             18965      hima get_stl2   walter  R       0:01      1 hi07
             18846      hima interact yacahuan  R   18:30:31      1 hi04
             18875      hima interact yacahuan  R    8:56:56      1 hi05
90 [bora] scancel 18965
91 [bora] ls
INPUT  list  out.1.0  out.1.1  out.1.35  out.2.0  run  slurm-18965.out
92 [bora]
```