



Research Computing Usage Overview

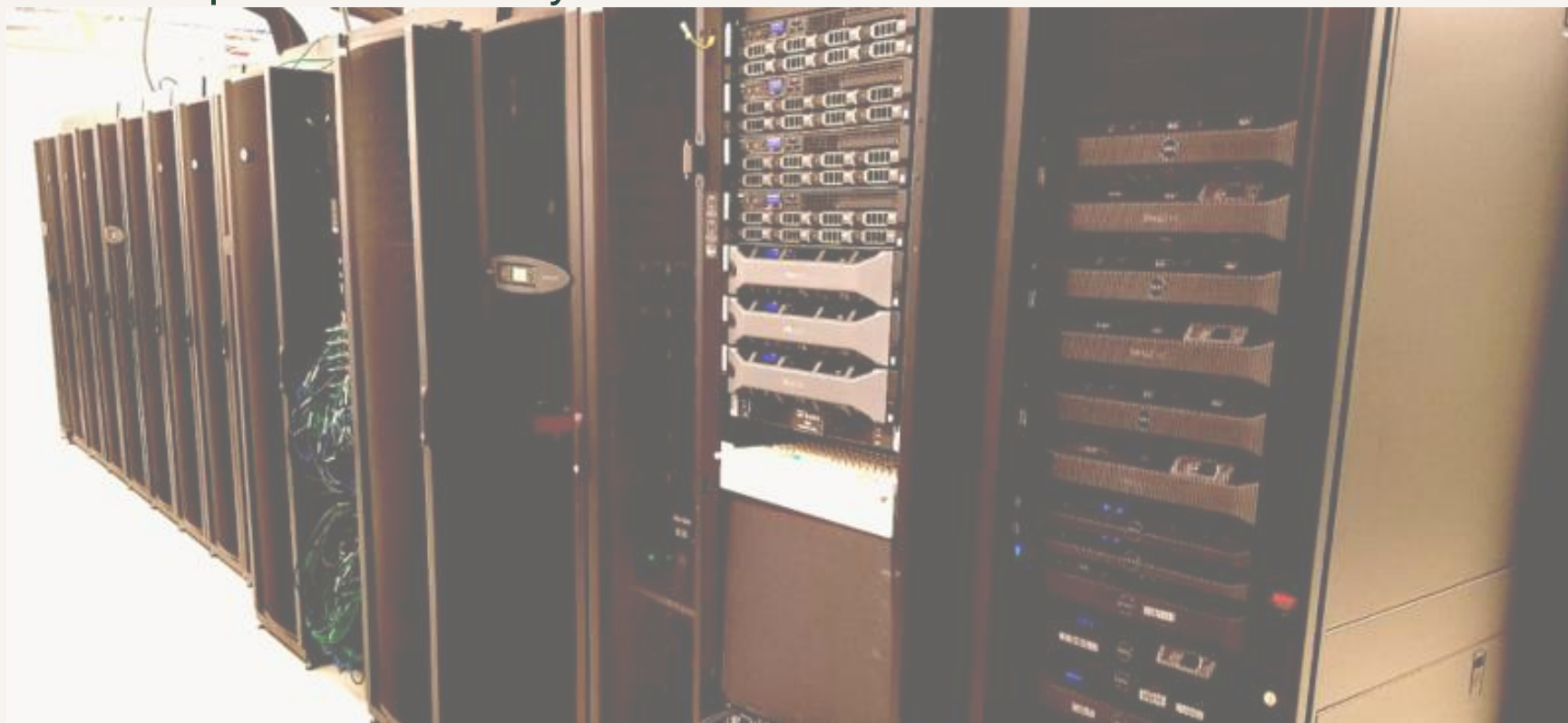
May 28th, 2026

What does Research Computing offer and how do I use it?

Research Computing main focus is operation of many clusters/servers for research at W&M/VIMS

- Prerequisites
 - Connecting / SSH
 - Linux CLI
 - Files/Folders
 - Scripting
- Filesystems
- Software / Compilers
- Batch clusters / SLURM
- Jupyter Notebooks
- Kubernetes cluster

RC currently has seven FTE staff members + multiple students



ISC 1251 – RC server room (#1)

Prerequisites and background help

Send help requests to hpc-help@wm.edu

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

Getting an account: <https://hpc.wm.edu/acctreq>

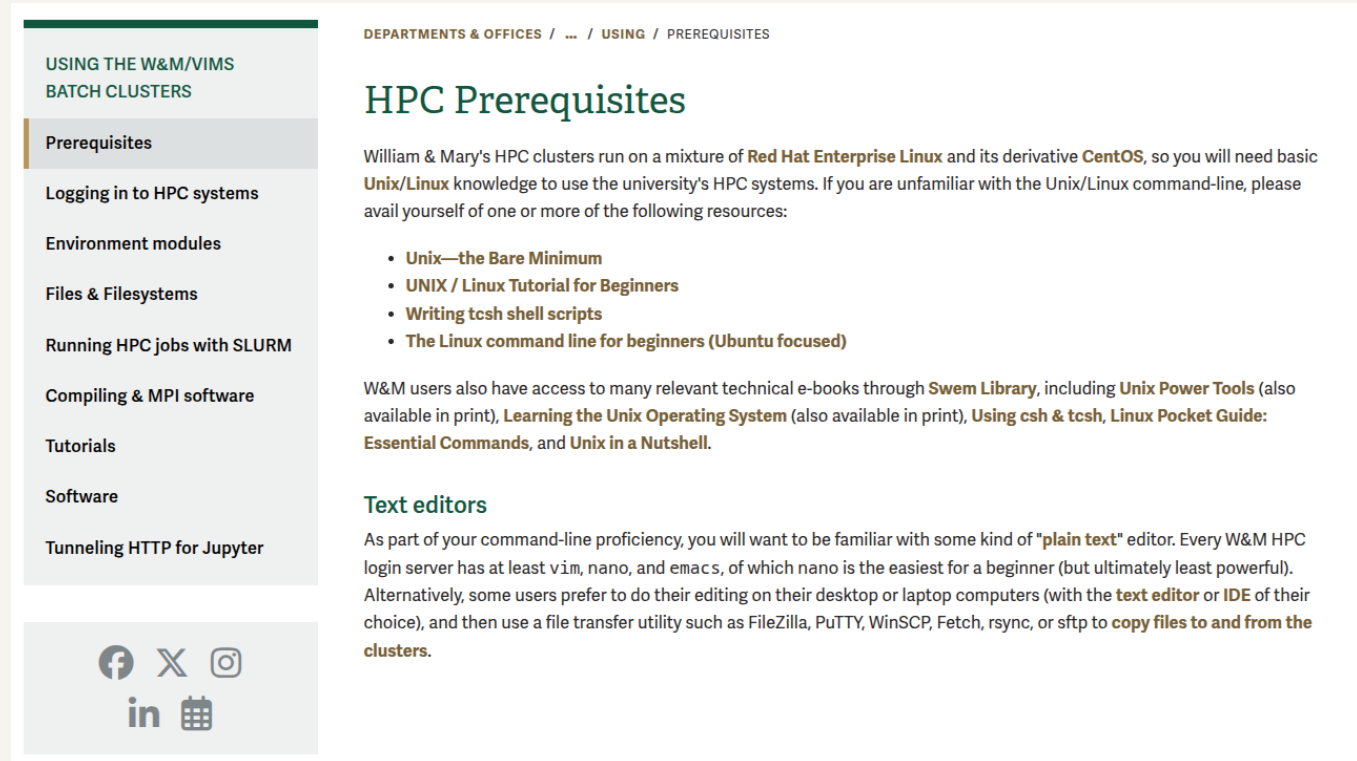
Before you can use the RC/HPC clusters effectively, you will need to have a basic knowledge of:

- Access / SSH / bastion host / graphics
- Linux command line
- Editing files
- Shell scripting
- File transferring

RC webpage:

<https://www.wm.edu/offices/it/services/researchcomputing/atwm>

<https://hpc.wm.edu>



The screenshot shows a webpage titled "HPC Prerequisites" under the heading "DEPARTMENTS & OFFICES / ... / USING / PREREQUISITES". The page content includes:

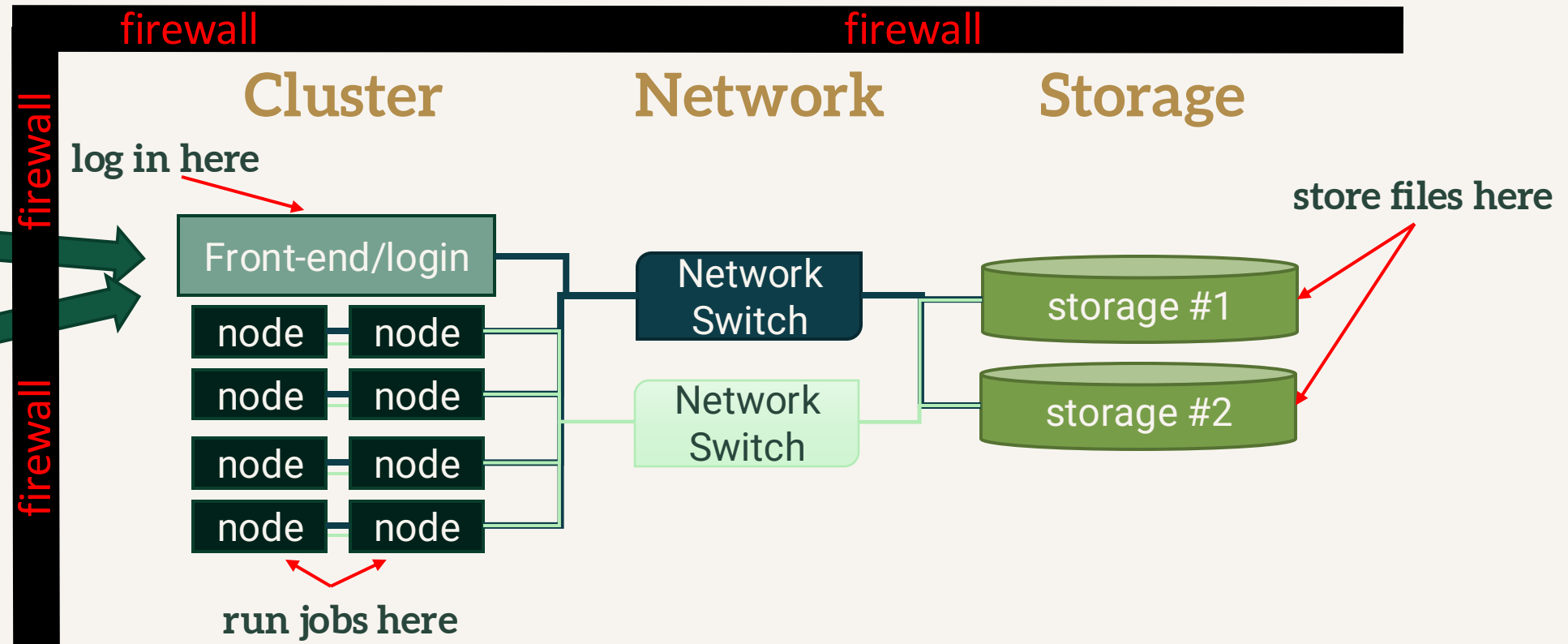
- A navigation menu on the left with items: USING THE W&M/VIMS BATCH CLUSTERS, Prerequisites (highlighted), Logging in to HPC systems, Environment modules, Files & Filesystems, Running HPC jobs with SLURM, Compiling & MPI software, Tutorials, Software, and Tunneling HTTP for Jupyter.
- Main text: "William & Mary's HPC clusters run on a mixture of Red Hat Enterprise Linux and its derivative CentOS, so you will need basic Unix/Linux knowledge to use the university's HPC systems. If you are unfamiliar with the Unix/Linux command-line, please avail yourself of one or more of the following resources:"
- A bulleted list of resources:
 - Unix—the Bare Minimum
 - UNIX / Linux Tutorial for Beginners
 - Writing tcsh shell scripts
 - The Linux command line for beginners (Ubuntu focused)
- Text: "W&M users also have access to many relevant technical e-books through Swem Library, including Unix Power Tools (also available in print), Learning the Unix Operating System (also available in print), Using csh & tcsh, Linux Pocket Guide: Essential Commands, and Unix in a Nutshell."
- Section: "Text editors" with text: "As part of your command-line proficiency, you will want to be familiar with some kind of 'plain text' editor. Every W&M HPC login server has at least vim, nano, and emacs, of which nano is the easiest for a beginner (but ultimately least powerful). Alternatively, some users prefer to do their editing on their desktop or laptop computers (with the text editor or IDE of their choice), and then use a file transfer utility such as FileZilla, PuTTY, WinSCP, Fetch, rsync, or sftp to copy files to and from the clusters."
- Social media icons for Facebook, X, Instagram, LinkedIn, and a calendar icon.

Connecting to RC clusters



bastion.wm.edu

W&M/VIMS VPN



Must get through **Firewall** to get to RC resources (W&M and VIMS)
On campus you are already within the firewall
Off campus you must hop through **bastion** host or use **VPN**
Can put in an ssh-key for bastion at <https://code.wm.edu>

How to connect to RC clusters

ssh – is the standard app to connect to a remote Linux computer
Standard software on Linux and Mac (via *terminal*)
Windows supports it in powershell, some Windows users prefer an ssh client (*putty*)

From within the firewall (on campus) or VPN:

ssh to bora: `ssh ejwalt@bora.sciclone.wm.edu`

Outside of firewall use bastion host and double hop:

ssh to bora: `ssh -J ejwalt@bastion.wm.edu ejwalt@bora.sciclone.wm.edu`

If you want to pull graphics back to your local computer (`-Y`)

ssh to bora: `ssh -Y ejwalt@bora.sciclone.wm.edu`

ssh to bora (withh bastion): `ssh -Y -J ejwalt@bastion.wm.edu ejwalt@bora.sciclone.wm.edu`

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

- log in help

<https://www.wm.edu/offices/it/services/researchcomputing/using/filesandfilesystems/xfers/>

- file transfer help

Linux Command Line Interface (CLI)

Shell – this is the program that provides the CLI **bash** or **tcsh**
commands near identical
scripting a little different
most people won't care
use command **echo \$0** to determine your shell version – default is **bash**

Prompt – This is where commands are entered

[ewalter@bora ~]\$

Prerequisites page has links to Linux command line guides

Also see tutorials page:

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

For previous year talk on Linux CLI

Useful commands:

mkdir <dir> - make directory named <dir>

cd <dir> - change to directory <dir>

rmdir <dir> - remove directory <dir>

rm <file> - remove file named <file>

cp <oldname> <newname> - copy file

mv <oldname> <newname> - move/rename file

ls - list directory contents

cat <file> - dump <file> to screen

less <file> - page through <file>

clear - clear the screen

man <command> - read the manual page

Other CLI tips:

cd by itself means **cd** to home

cd with ...

- . current directory
- .. up one directory
- ~ home directory
- previous directory

For other commands:

* wildcard symbol

tab completion

[ewalter@bora ~]\$ ls filename <tab>

This will complete the word "file" with whatever files/folders in your current directory match this name.

up arrow

Hitting the up-arrow key will cycle back through your current history

Editing files

```
File Edit View Search Terminal Help
GNU nano 7.2 list
Media Barcode: HPC154L7
Partition: Partition1
Drive Error Code: 0x7060
Reason Details: Undetermined
Tape Drive SN: 10WT048960
Drive Reported SN: C3EB997148

Media Barcode: HPC538L7
Partition: Partition1
Drive Error Code: 0x7076
Reason Details: Drive
Tape Drive SN: 10WT023352
Drive Reported SN: C3EB997008

Media Barcode: HPC574L7
Partition: Partition1
Drive Error Code: 0x7825
[ Read 72 lines ]
^G Help      ^O Write Out ^W Where Is  ^K Cut       ^T Execute   ^C Location
^X Exit      ^R Read File ^\ Replace   ^U Paste     ^J Justify   ^_ Go To Line
```

To create / edit files, need to use a **text editor**. All HPC systems have: **vi/vim, emacs, and nano**

Nano is most useful for beginners

<https://www.howtogeek.com/42980/the-beginners-guide-to-nano-the-linux-command-line-text-editor/>

List of commands at bottom of screen: i. e. **cntrl-X** = exit, **cntrl-O** = write file

Can also connect via **vscode** and edit files locally within **vscode** interface
Don't open too many windows or run jobs directly on front-end login machines!
vscode doesn't work with 2FA (use key via <https://code.wm.edu>)

Three types of parallelism

Two bora nodes



How parallel the calculation can run is application dependent
How many cores a parallel application can use efficiently also varies
Try test calculations if not sure.

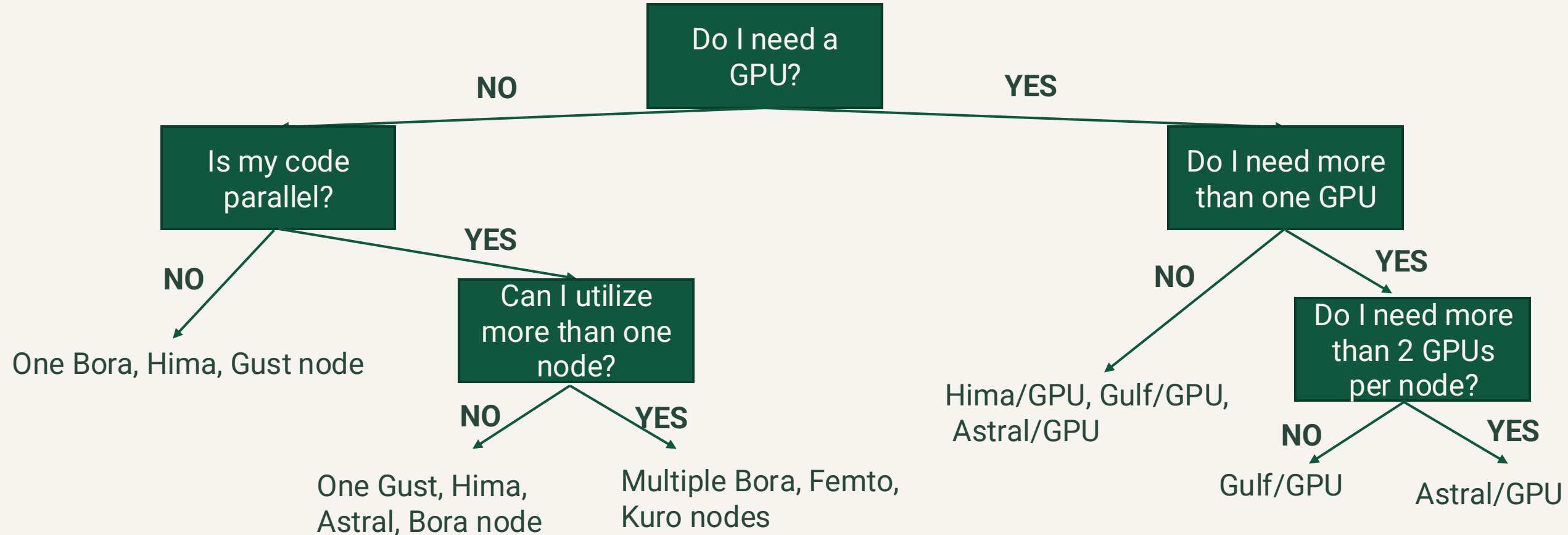
Batch cluster resources

* - not all nodes functional
 ** - hi04, hi05 have 1x P100
 hi07 has 1x V100

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 / ku01-ku64	AMD EPYC 9334	--	64	4096	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 / as00	Intel Xeon 6336Y	--	32	32	100	2022
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 HPC cluster						
James/jm01-jm27	Intel Xeon Silver 4112	--	20	480*	64	2018

See: <https://www.wm.edu/offices/it/services/researchcomputing/atwm/systemarchitecture/nodes>

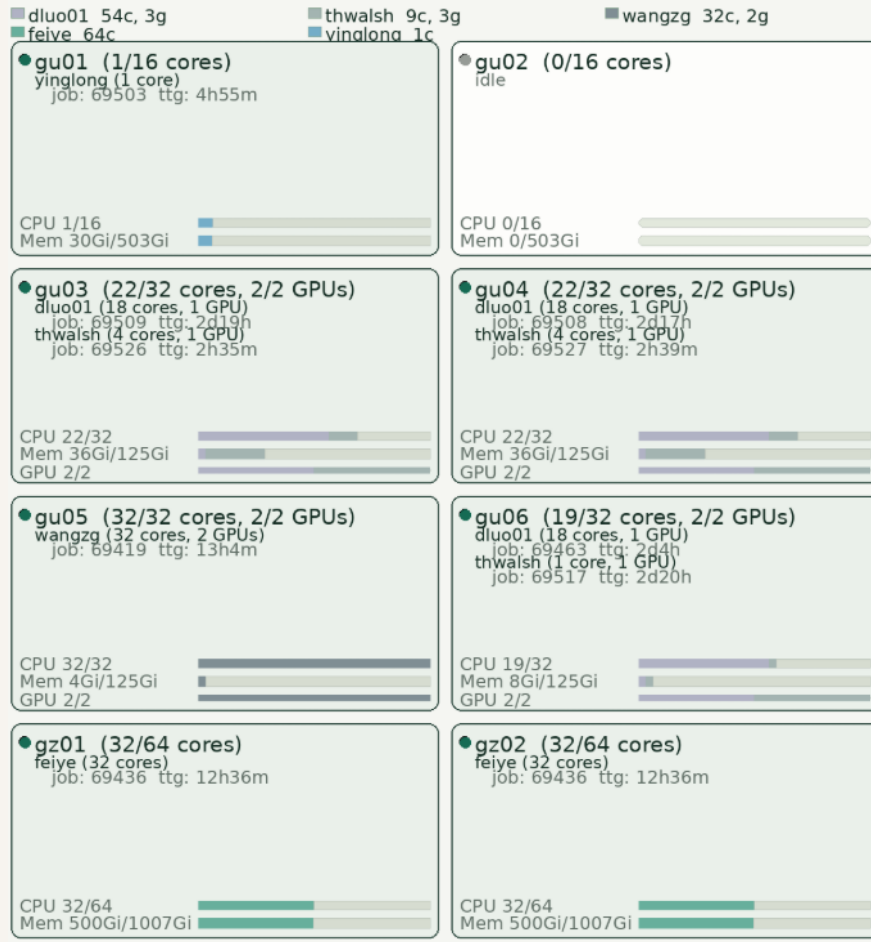
Choosing resources on the RC batch clusters



Cluster Dashboard

<https://hpc.wm.edu/grids>

Gulf • 2026-05-28 14:15:23
Next jobs: 69540(tnguyen10, 16c, 1g, 32Gi), 69539(tnguyen10, 16c, 1g, 32Gi), 69546(hnayak,...



The cluster dashboard shows the current occupation of all main-campus clusters (SLURM & Kubernetes)

Can be an effective way to quickly find an idle resources

Shows the current owners of each job and what resources each job is using, i.e. # CPUs, #GPUs and memory.

File storage targets

Four types of storage

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

Type	Location	Backed up?	Purged?	Quota
Home	/sciclone/home/<USERNAME>	YES (weeknights)	NO	Yes (50GB)
Data	/sciclone/data10/<USERNAME>	YES (weekly)	NO	NO
Scratch	/sciclone/scr{10,20,30}/<USERNAME>	NO	YES (after 90 days of inactivity)	NO
Local Scratch	(/local/scr/<USERNAME>)	NO	YES (after 14 days of inactivity)	NO

- **Home** is for scripts, configuration files, python environments, etc, ~100-200GB
- **Data** is for long-term storage of data, ~ 1-2TB
- **Scratch** is for outputs and large temporary storage (> 10TB contact RC) **Purged after 90 days of inactivity**
- **Local scratch** is sometimes used for installation or used for job outputs – **Purged after 14 days of inactivity**

Good practice is to write job outputs to *scratch* and not *home* or *data*
Then, copy outputs you want to save to the *data* file system.
Remember that *scratch* filesystems are purged!

File systems

`ls -lh` - list current dir (long, human readable)

(`/sciclone/home/ewalter`)

i.e. `~`

Symbolic links to each storage target (`->`)

```
[ewalter@bora ~ ]$ ls -lh
```

```
total 44K
lrwxrwxrwx. 1 ewalter hpcf 23 Jan 24 2024 data10 -> /sciclone/data10/ewalter
lrwxrwxrwx. 1 ewalter hpcf 17 Jun 9 2010 lscr -> /local/scr/ewalter
lrwxrwxrwx. 1 ewalter hpcf 21 Aug 1 2017 pscr -> /sciclone/pscr/ewalter
lrwxrwxrwx. 1 ewalter hpcf 22 Jun 9 2010 scr10 -> /sciclone/scr10/ewalter
lrwxrwxrwx. 1 ewalter hpcf 22 Jun 9 2010 scr20 -> /sciclone/scr20/ewalter
lrwxrwxrwx. 1 ewalter hpcf 22 Jan 24 2024 scr30 -> /sciclone/scr30/ewalter
```

```
[4 ewalter@bora ~ ]$df -h
```

Filesystem	Size	Used	Avail	Use%	Mounted on
devtmpfs	4.0M	0	4.0M	0%	/dev
tmpfs	79G	0	79G	0%	/dev/shm
tmpfs	32G	2.5G	29G	8%	/run
/dev/mapper/cl-hpcf	74G	48G	26G	66%	/
/dev/sda2	960M	224M	737M	24%	/boot
/dev/sda1	599M	7.1M	592M	2%	/boot/efi
/dev/mapper/cl-usrl	338G	43G	295G	13%	/usr/local
/dev/mapper/cl-lscr	20G	5.5G	14G	29%	/local/scr
/dev/mapper/cl-tmp	49G	666M	49G	2%	/tmp
/dev/mapper/cl-var	74G	19G	55G	26%	/var
192.168.59.130:/sciclone/nova	107T	66T	42T	61%	/sciclone/aiddata10
lu00-ib:/sciclone/data10	1.1P	905T	130T	88%	/sciclone/data10
scr20-ib:/sciclone/scr20	112T	31T	82T	28%	/sciclone/scr20
proj-ds-ib:/sciclone/proj-ds	464T	158T	307T	34%	/sciclone/proj-ds
as00-ib:/sciclone/apps	1.8T	319G	1.5T	18%	/sciclone/apps
sl00-ib:/sciclone/gluex10	107T	45T	63T	42%	/sciclone/gluex10
scr10-ib:/sciclone/scr10	219T	146T	73T	67%	/sciclone/scr10
sn00-ib:/sciclone/schism10	599T	585T	14T	98%	/sciclone/schism10
cm00:/sciclone/home	27T	24T	3.1T	89%	/sciclone/home
scr30-ib:/sciclone/scr30	502T	304T	199T	61%	/sciclone/scr30
util:/storage/REPOS	1.9T	959G	904G	52%	/storage/REPOS
scr10-ib:/sciclone/swork	42T	657G	40T	2%	/sciclone/swork
192.168.56.208@o2ib,192.168.56.209@o2ib:/pscr	297T	171T	126T	58%	/sciclone/pscr

`df -h` list all filesystems in human readable output

General use filesystems in red

Notice the **Use %** keep these below %100

Use `du` to calculate folder sizes

```
[4 ewalter@bora ~ ]$ pwd
```

```
/sciclone/home/ewalter
```

```
[5 ewalter@bora ~ ]$ du -hs .
```

```
326G .
```

```
[6 ewalter@bora ~ ]$
```

Files and Transferring files

To transfer files from one directory to another within the cluster

- `cp` or `mv` to transfer one or a few files/directories
- `rsync` for multiple folders (lets you restart)

To or from the cluster from/to outside cluster

- `rsync`, `scp`, `sftp` (from Mac, Linux CLI / Windows powershell)
- Globus – web GUI; can transfer to/from local computer
- Windows users sometimes prefer a separate client (`winscp`)

<https://www.wm.edu/offices/it/services/researchcomputing/using/filesandfilesystems/xfers/>

Running jobs through the SLURM batch system

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

All workloads are prepared on cluster front-end and jobs are submitted to the batch system

Two types of jobs:

Batch – run my job on one or more nodes (asynchronous)

Interactive – give me a session on one or more nodes (synchronous)

A batch script is needed to be prepared for a batch job (see web for examples)

A batch script can be as simple as:

```
#!/bin/tcsh
#SBATCH --job-name=serial
#SBATCH -N 1 -n1
#SBATCH -t 0:30:00
./a.out_serial
```

or a batch script can run many commands to prepare data and/or analyze outputs, etc.

Running jobs through the SLURM batch system

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

Action	Command
Start interactive job	salloc
Submit batch script	sbatch
View current jobs	squeue
Cancel job	scancel
Launch MPI job	srun
Check node status	sinfo

Interactive job:

```
[18 ewalter@bora ~ ]$salloc -N 1 -n 20 -t 30:00
salloc: Granted job allocation 257299
salloc: Waiting for resource configuration
salloc: Nodes bo14 are ready for job
[1 ewalter@bo14 ~ ]$
```

Notice that now on bo14 instead of bora

Batch script "runjob"

```
#!/bin/tcsh
#SBATCH --job-name=serial
#SBATCH -N 1 -n1
#SBATCH -t 0:30:00
./a.out_serial
```

Submit batch script to cluster:

```
[19 ewalter@bora ~ ] sbatch runjob
Submitted batch job 257300
```

Lots of explanation and examples on RC web page

Software, Compilers and Containers

Standard Linux software, utilities, compilers, parallel software installed on all front-ends and nodes of batch cluster

- Help on compilers & MPI software

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

- Help on software

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

The software page shows the list of the standard/(mostly) non-scientific software installed throughout the SLURM clusters. We also have basic instructions for using, **Stata**, **Gaussian**, **Matlab**, and **Python** on the SLURM clusters.

The majority of scientific software on the SLURM cluster is available through the modules system

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

Can add module loads to startup files or to SLURM batch scripts

Environmental Modules

```
[9] ewalter@bora:~$ module avail
----- /usr/share/Modules/modulefiles -----
dot module-git module-info modules null use.own

----- /usr/local/Modules/modulefiles -----
abcluster/3.4                iqtree/3.0.1                openmpi-ib/intel-2024.0/4.1.6  xz/gcc-11.5/5.2.2
abcluster/3.4contr          JDK/23                      pandoc/3.8.2
ampl/20210531                kraken2/gcc-11.4.1/2.1.5    proj/gcc-11.4.1/5.2.0
bcftools/gcc-11.4.1/1.22    legacy-tools/gcc-8.5.0      python/gcc-11.4.1/3.11.9
bowtie2/gcc-11.4.1/2.5.4    legacy-tools/gcc-9.5.0      qchem/6.4.0-1
bracken/gcc-11.4.1/3.1      mafft/gcc-11.4.1/7.525     qe/7.4.1
bwa-mem/2.3                 modeltest-ng/0.1.7          r/gcc-11.4.1/4.4.0
bwa/gcc-11.4.1/0.7.18       mummer/gcc-11.5/4.0.1      r/gcc-11.5.0/4.6.0
clustal-omega/gcc-11.4.1/1.2.4  muscle/5.3                 ruby/gcc-11.4.1/3.4.4
cuda/12.3                   NCIPLOT/gcc-11.4.1/4.3     samtools/gcc-11.4.1/1.21
dmtcp/gcc-11.4.1/4.0.0      ncl/gcc-11.4.1/6.6.2 openmpi
dorado/1.4.0                nco/intel-2024.0/5.2.4     seqkit/2.13.0
Flye/gcc-11.4.1/2.9.6       netcdf-c/gcc-11.4.1/4.9.2 openmpi
gap/gcc-11.4.1/4.14.0       netcdf-c/intel-2024.0/4.9.2 intelmpi
gatk/4.6.1.0                netcdf-c/intel-2024.0/4.9.2 openmpi
gromacs/gcc-11.4.1/2024.5    netcdf-fortran/gcc-11.4.1/4.6.1 openmpi
hdf5/gcc-11.4.1/1.14.3 openmpi                      netcdf-fortran/intel-2024.0/4.6.1 intelmpi
hdf5/intel-2024.0/1.14.3 intelmpi                    netcdf-fortran/intel-2024.0/4.6.1 openmpi
hdf5/intel-2024.0/1.14.3 openmpi                      openblas/gcc-11.4.1/0.3.27
htslib/gcc-11.5/1.23.1      openmpi-ib/gcc-11.4.1/4.1.6  vcftools/gcc-11.4.1/0.1.17

----- /sciclone/apps/modulefiles -----
comsol/6.2                  hyperworks/24               intel/compiler-2022.2.1      intel/mpi-2021.11           matlab/R2023a               python/3.12.7
comsol/6.3                  hyperworks/24.1             intel/compiler-2024.0        intel/tbb-2021.7.1         matlab/R2024a               silcsbio/2026.1-alpha
gaussian/g16                 intel/2019                   intel/mkl-2024.0             julia/1.9.4                 miniforge3/24.9.2-0         stata/18.0
gaussian/g16_vC02           intel/2019-mpi               intel/mpi-2021.7.1           matlab/R2020b               nvidia/nsight-compute-2024.3

Key:
loaded  default-version  modulepath
[10] ewalter@bora:~$ module list
Currently Loaded Modulefiles:
  1) modules  2) slurm/23.11.9  3) intel/compiler-2024.0  4) intel/mkl-2024.0  5) intel/mpi-2021.11

Key:
default-version
[11] ewalter@bora:~$ module load matlab/R2024a
[12] ewalter@bora:~$ module list
Currently Loaded Modulefiles:
  1) modules  2) slurm/23.11.9  3) intel/compiler-2024.0  4) intel/mkl-2024.0  5) intel/mpi-2021.11  6) matlab/R2024a

Key:
default-version
[13] ewalter@bora:~$ which matlab
/sciclone/apps/matlab-R2024a/bin/matlab
[14] ewalter@bora:~$
```

`module avail(able)`

- which modules are available

`/usr/local/Modules/modulefiles`

- specific for this cluster

`/sciclone/apps/modulefiles`

- available on all SLURM clusters

`module list`

- what modules do I have loaded?

`module load matlab/R2024a`

- load matlab version R2024a into my environment

Available Software

- Users encouraged to install own software if it is not already installed
- Users are encouraged to create their own **conda**/python environments
- Batch system also supports images using the **Singularity** container system – (must request access to use --fakeroot)

See: <https://github.com/baumgach/tue-slurm-helloworld/blob/master/instructions/singularity-workflow.md>

- **Docker** NOT used on RC systems
- **Singularity** can pull and use docker images

On <https://www.wm.edu/offices/it/services/researchcomputing/using/software/> there is a list of software packages installed throughout the SLURM clusters. This list is mainly non-technical software.

Contact us (hpc-help@wm.edu) if you have any questions using/installing/debugging any software.

Research Computing Notebooks

URL: <https://notebooks.sciclone.wm.edu>

NOTE: THIS IS NOT <https://jupyterhub.wm.edu> - this site is reserved for classroom use, has no access to RC filesystems and contains no GPU resources. It is also not administered by RC.

See: <https://www.wm.edu/offices/it/services/researchcomputing/jupyter/> for more information.

To use the system, you just need your W&M login and an HPC account.

The RC JupyterHub allows you to use our default containers:

- Basic Python (Python v3.11.8)
- Data Science Stack (Python v3.11.8, R v4.3.3, and Julia v1.10.3)
- Pytorch (Python 3.11.8 and Pytorch v2.3.1)

You will have access to all filesystems within the SLURM clusters (*home, data, scratch* etc.)

You can also create a new Anaconda Python environment within your session and create a new launcher button for it. You can also use Anaconda environments that you use on the SLURM clusters.

Kubernetes cluster / RC JupyterHub

Recent (~2023) **Kubernetes (k8s)** cluster deployed to support data-science workloads

Unlike the **Batch** cluster where most software runs right on bare-metal (some use containers)

K8s system uses containers exclusively -- container orchestration

Allows for time limited **jobs** (like batch cluster) and persistent **pods** to run microservices.

Jobs can be run by any user ; **pods** can only be run within a project **namespace**
requires permission from Data Science professor.

The majority of the RC GPUs exist in the k8s cluster. Some are for public use (26x Nvidia A40, 12x Tesla T4, 8x Quadro RTX 6000), the rest are reserved for individual DS faculty groups.

See: <https://www.wm.edu/offices/it/services/researchcomputing/atwm/systemarchitecture/nodes>

Help on k8s system:

<https://www.wm.edu/offices/it/services/researchcomputing/k8s/>

THANK YOU!

