

HIGH PERFORMANCE RESEARCH COMPUTING

HPRC Primers

Grace

January 27, 2023



High Performance
Research Computing
DIVISION OF RESEARCH



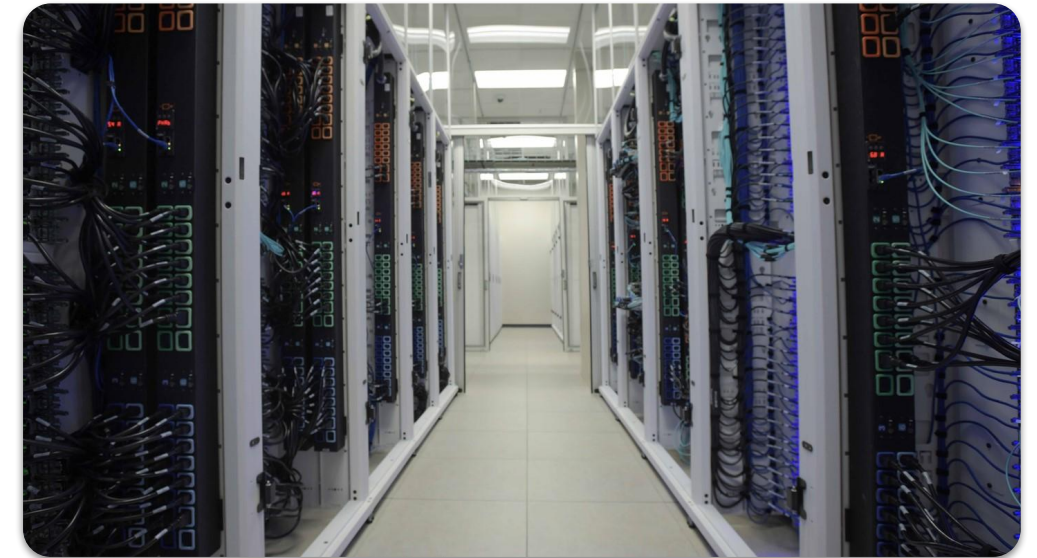
Grace: HPRC's Flagship Cluster

Grace is a 925-node Intel cluster from Dell with an InfiniBand HDR-100 interconnect, A100 GPUs, RTX 6000 GPUs and T4 GPUs. The 925 nodes are based on the Intel Cascade Lake processor.

48 cores/node

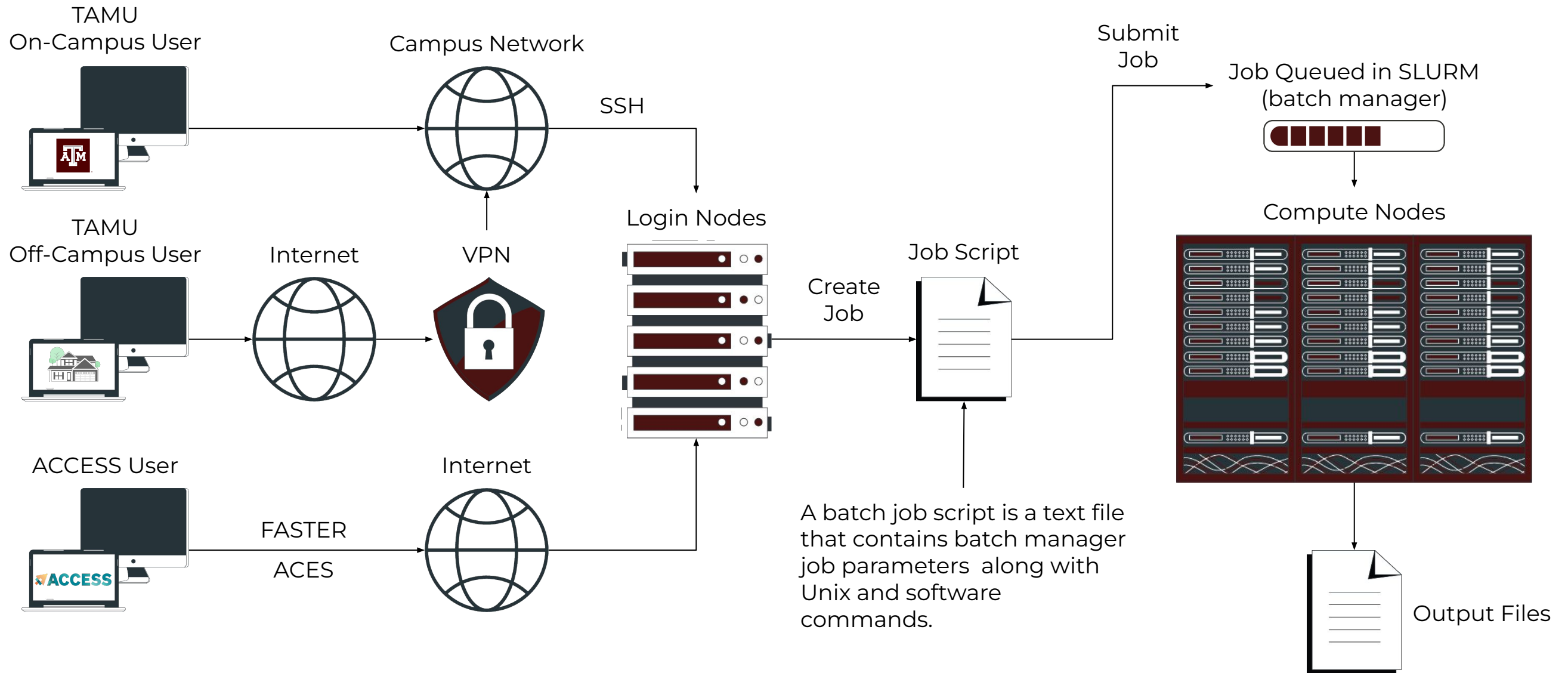
3TB Large Memory-80 cores/nodes
Login Nodes: 10 GbE TAMU network connection

Login Nodes	5
384GB memory general compute nodes	800
GPU - A100 nodes with 384GB memory	100
GPU - RTX 6000 nodes with 384GB memory	9
GPU - T4 nodes with 384GB memory	8
3TB Large Memory	8



For more information:
hprc.tamu.edu/wiki/Grace:Intro

Batch Computing on HPRC Clusters



Accessing Grace: via the Portal

- Access through (most) web browsers:

portal.hprc.tamu.edu

Top Banner Menu “Clusters” -> “Grace Shell Access”

TEXAS A&M HIGH PERFORMANCE RESEARCH COMPUTING

Home User Services Resources Research Policies Events About Portal

Terra Portal
Grace Portal
FASTER Portal
FASTER Portal (ACCESS)

Quick Links
New User Information
Accounts

TAMU HPRC OnDemand (Grace) Files Jobs Clusters Interactive Apps Dashboard My Interactive Sessions

>_grace Shell Access

OnDemand provides an integrated, single access point for all of your HPC resou

Message of the Day

IMPORTANT POLICY INFORMATION

- Unauthorized use of HPRC resources is prohibited and subject to criminal prosecution.
- Use of HPRC resources in violation of United States export control laws and regulations is prohibited. residents.
- Sharing HPRC account and password information is in violation of State Law. Any shared accounts wi
- Authorized users must also adhere to ALL policies at: <https://hprc.tamu.edu/policies>

!! WARNING: THERE ARE ONLY NIGHTLY BACKUPS OF USER HOME DIRECTORIES. !!

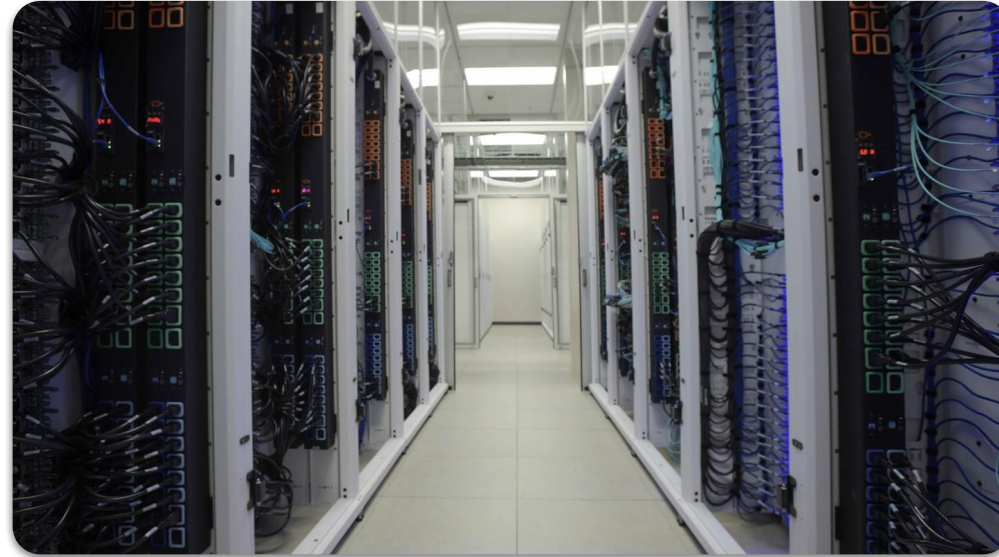
hprc.tamu.edu/wiki/HPRC:Access

Accessing Grace: via SSH

- SSH command is required for accessing Grace / Terra:
 - On campus: `ssh NetID@grace.tamu.edu`
 - Off campus:
 - Set up and start VPN (Virtual Private Network): u.tamu.edu/VPnetwork
 - Then: `ssh NetID@grace.tamu.edu`
 - *Two-Factor Authentication* enabled for CAS, VPN, SSH
- SSH programs for Windows:
 - MobaXTerm (preferred, includes SSH and X11)
 - PuTTY SSH
 - Windows Subsystem for Linux
- Grace has 5 login nodes. Check the bash prompt.
Login sessions that are idle for 60 minutes will be closed automatically
Processes run longer than 60 minutes on login nodes will be killed automatically.
Do not use more than 8 cores on the login nodes!
Do not use the sudo command.

hprc.tamu.edu/wiki/HPRC:Access

Pop Quiz



How many login nodes are on Grace?

- A. 1
- B. 5

- C. 3
- D. 10

Two-Factor Authentication

- [Duo](#) NetID two-factor authentication to enhance security
 - All web login (howdy, portal.hprc.tamu.edu, Globus) through CAS
 - VPN to TAMU campus
 - SSH/SFTP to HPRC clusters
- See instructions in two-factor [wiki page](#)
- SSH clients work with Duo
- SFTP clients work with Duo
 - Filezilla is not recommended
- Not all software supports SSH+Duo:
SFTP in Matlab

hprc.tamu.edu/wiki/Two_Factor

Example: SSH login with Duo

```
$ ssh grace.tamu.edu
*****
*****
... warning message (snipped) .....
*****
*****

Password:
Duo two-factor login for UserNetID

Enter a passcode or select one of the following
options:

  1. Duo Push to XXX-XXX-1234
  2. Phone call to XXX-XXX-1234
  3. SMS passcodes to XXX-XXX-1234 (next code starts
with: 9)

Passcode or option (1-3): 1
Success. Logging you in...
```

File Systems and User Directories

Directory	Environment Variable	Space Limit	File Limit	Intended Use
/home/\$USER	\$HOME	10 GB	10,000	Small to modest amounts of processing.
/scratch/user/\$USER	\$SCRATCH	1 TB	250,000	Temporary storage of large files for on-going computations. Not intended to be a long-term storage area.

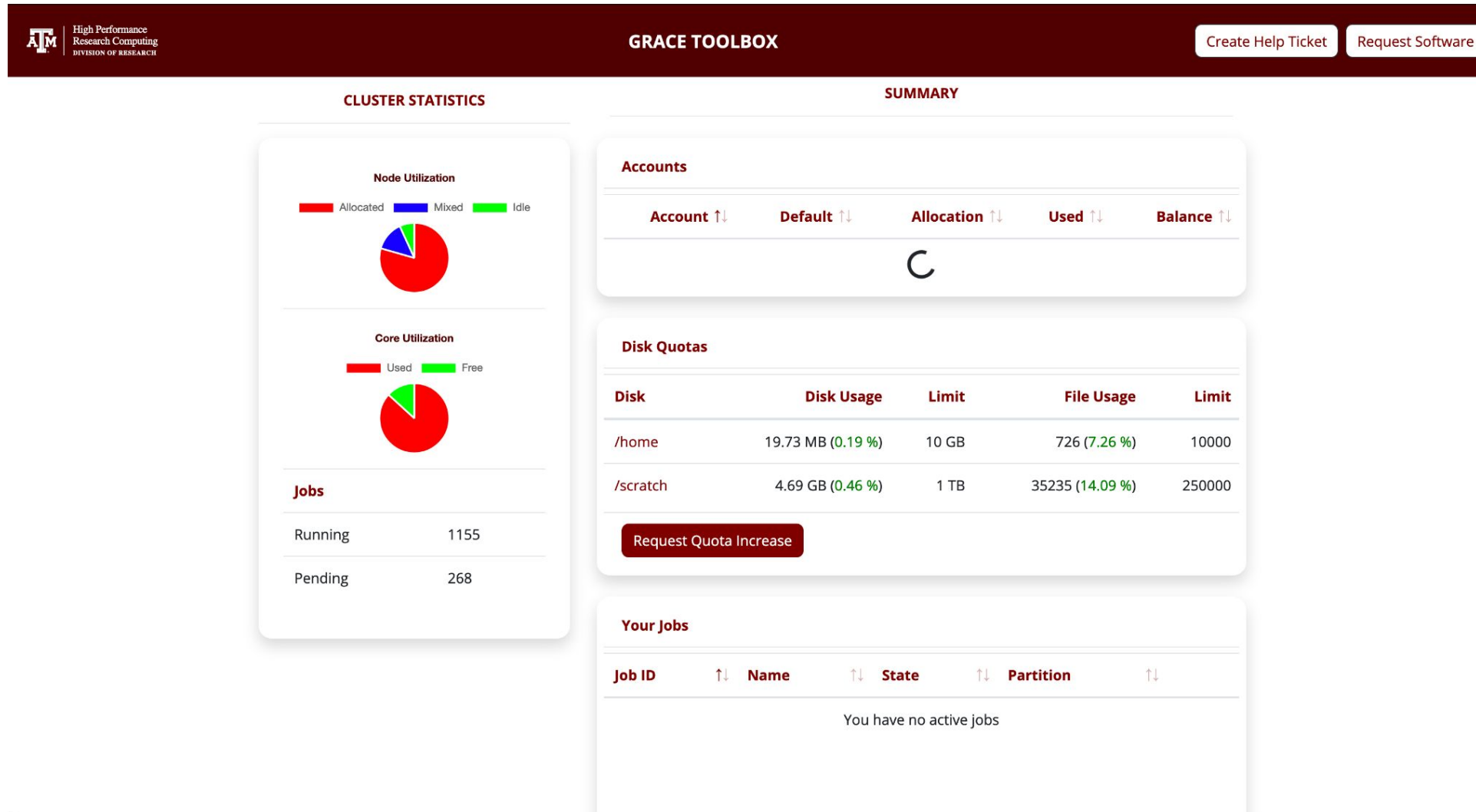
- View usage and quota limits using the command:
- Request a group directory for sharing files.
- Do not share your home or scratch directories.

showquota

hprc.tamu.edu/wiki/Grace:Filesystems_and_Files

OOD Dashboard: Grace

Easily view Cluster utilization, Storage Quotas & Allocation Balances



Quota and file limit increases will only be considered for scratch directories

Preferred way to request *Quota Increases*

Hands-On Activity - 2 Minutes

Please try to access the terminal on Grace now either through SSH or portal.hprc.tamu.edu

What message do you see when you log on?

Software

- See the Software wiki page for instructions and examples
 - hprc.tamu.edu/wiki/SW
 - hprc.tamu.edu/software/grace
- License-restricted software
 - Contact license owner for approval
- Contact us for software installation help/request
 - User can install software in their home/scratch directory
 - Do not run the "*sudo*" command when installing software

Software: Application Modules

Installed applications are made available with the module system

Grace uses a *software hierarchy* inside the module system

In this hierarchy, the user loads a compiler which then makes available Software built with the currently loaded compiler

```
module avail
```

```
# shows which software is available
```

```
module load GCC/10.3.0 OpenMPI/4.1.1
```

```
# load GCC compiler version 10.3.0 and OpenMPI  
version 4.1.1
```

```
module avail
```

```
# show which software is now available
```

```
module load PyTorch/1.12.1
```

```
# load PyTorch version 1.12.1
```

Software: Modules and Toolchains

- Toolchains are what we call groups of compilers & libraries
- There's a variety of toolchains available on the clusters:
 - intel/2022a
 - iomkl/2020a
 - foss/2022a
 - GCCcore/11.3.0

```
module purge
```

```
# removes all loaded modules
```

Hands-On Activity - 2 Minutes

1. Please search for and load the following module:

`OpenMPI/4.1.4`

(Tip) Type **module spider OpenMPI/4.1.4** to show which compiler needs to be loaded.

2. Next remove (unload) all your current modules.

Consumable Computing Resources

- Resources specified in a job file:
 - Processor cores
 - Memory
 - Wall time
 - GPU
- Service Unit (SU) - Billing Account
 - Use "**myproject**" to query

hprc.tamu.edu/wiki/HPRC:AMS:Service_Unit

```
myproject
```

```
=====
List of YourNetID's Project Accounts
=====
```

Account	FY	Default	Allocation	Used & Pending SUs	Balance	PI
1228000223136	2023	N	10000.00	0.00	10000.00	Doe, John
1428000243716	2023	Y	5000.00	-71.06	4928.94	Doe, Jane

- Other resources:
 - Software license/token
 - Use "**license_status**" to query
 - hprc.tamu.edu/wiki/SW:License_Checker

```
license_status -a
```

Find available license for "ansys":

```
license_status -s ansys
```

```
License status for ANSYS:
```

```
-----
| License Name           | # Issued | # In Use | # Available |
|-----|-----|-----|-----|
| aa_mcad                 | 50       | 0        | 50         |
| aa_r                    | 50       | 32       | 18         |
| aim_mp1                 | 50       | 0        | 50         |
| .....                  |          |          |           |
|-----|-----|-----|-----|
```

Find detail options:

```
license_status -h
```

Check your Service Unit (SU) Balance

- List the SU Balance of your Account(s)

```
myproject
```

```
=====
List of YourNetID's Project Accounts
-----
| Account | FY | Default | Allocation | Used & Pending SUs | Balance | PI |
-----
| 1228000223136 | 2023 | N | 10000.00 | 0.00 | 10000.00 | Doe, John |
-----
| 1428000243716 | 2023 | Y | 5000.00 | -71.06 | 4928.94 | Doe, Jane |
-----
| 1258000247058 | 2023 | N | 5000.00 | -0.91 | 4999.09 | Doe, Jane |
-----
```

- Run "`myproject -d Account#`" to change default project account
- Run "`myproject -h`" to see more options

hprc.tamu.edu/wiki/HPRC:AMS:Service_Unit

hprc.tamu.edu/wiki/HPRC:AMS:UI

Hands-On Activity - 2 Minutes

1. Use myproject to check the SU balance of your accounts.
2. Check the license status of ABAQUS. How many licenses are in the first row of the # Issued column?

Batch Queues

Job submissions are auto-assigned to batch queues based on the resources requested (number of cores/nodes and walltime limit)

<https://hprc.tamu.edu/wiki/Grace:Batch>

sinfo : Current Queues on Grace

```
dylan — dylan@login4:~ — ssh < c
[dylan@grace4 ~]$ sinfo
PARTITION      AVAIL  TIMELIMIT  JOB_SIZE  NODES(A/I/O/T)  CPUS(A/I/O/T)
short*         up     2:00:00    1-32      768/0/32/800    34768/1836/1796/3840
medium         up     1-00:00:00 1-128     768/0/32/800    34768/1836/1796/3840
long           up     7-00:00:00 1-64      768/0/32/800    34768/1836/1796/3840
xlong          up     21-00:00:00 1-32     768/0/32/800    34768/1836/1796/3840
vnc            up     12:00:00   1-32     49/63/5/117     2192/3184/240/5616
gpu            up     4-00:00:00 1-32     49/63/5/117     2192/3184/240/5616
bigmem         up     2-00:00:00 1-4      2/5/1/8         160/400/80/640
staff          up     infinite   1-infinite 817/63/37/917   36960/5020/2036/4401
special        up     7-00:00:00 1-infinite 817/63/37/917   36960/5020/2036/4401
[dylan@grace4 ~]$
```

For the NODES and CPUS columns:
A = Active (in use by running jobs)
I = Idle (available for jobs)
O = Offline (unavailable for jobs)
T = Total

Grace Sample Job Script Structure

```
#!/bin/bash
##NECESSARY JOB SPECIFICATIONS
#SBATCH --export=NONE
#SBATCH --get-user-env=L
#SBATCH --job-name=JobExample1
#SBATCH --time=01:30:00
#SBATCH --ntasks=1
#SBATCH --mem=5G
#SBATCH --output=stdout.%j

##OPTIONAL JOB SPECIFICATIONS
#SBATCH --account=123456
#SBATCH --mail-type=ALL
#SBATCH --mail-user=email_address

# load required module(s)
module purge
module load GCCcore/12.2.0 Python/3.10.8

./my_program.py
```

These parameters describe your job to the job scheduler

Account number to be charged

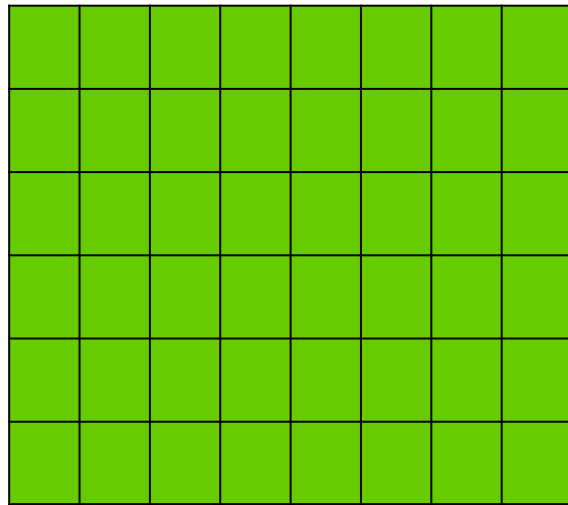
This is single line comment and not run as part of the script

Load the required module(s) first

This is a command that is executed by the job

Mapping Jobs to Cores per Node on Grace

A.

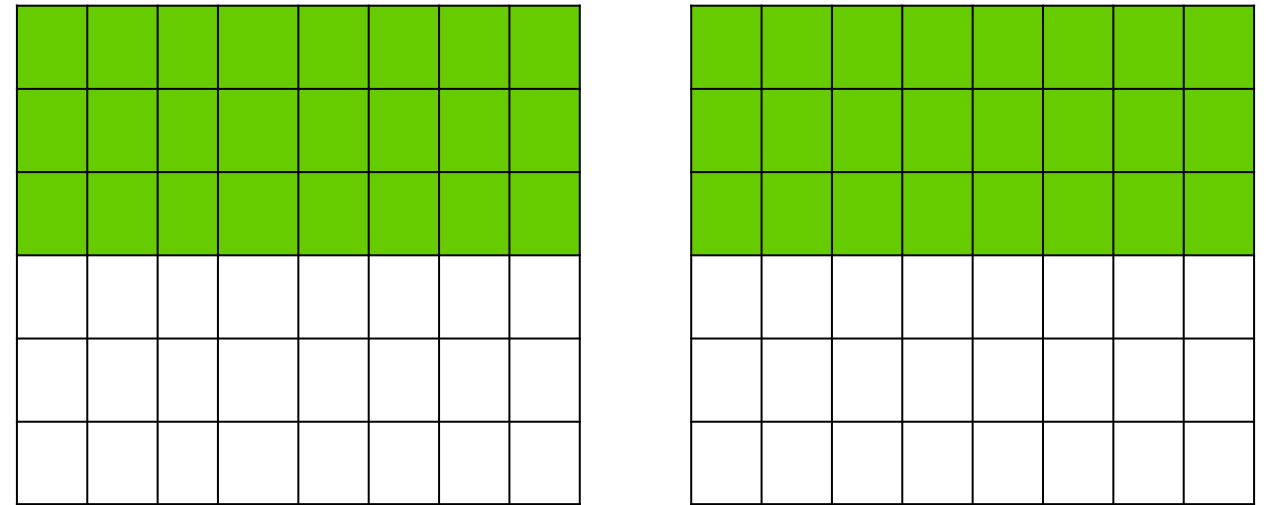


48 cores on
1 compute node

```
#SBATCH --ntasks=48  
#SBATCH --tasks-per-node=48
```

Preferred Mapping
(if applicable)

B.



48 cores on
2 compute nodes

```
#SBATCH --ntasks=48  
#SBATCH --tasks-per-node=24
```

Important Batch Job Parameters

Grace	Comment
<code>#SBATCH --export=NONE</code> <code>#SBATCH --get-user-env=L</code>	Initialize job environment.
<code>#SBATCH --time=HH:MM:SS</code>	Specifies the time limit for the job.
<code>#SBATCH --ntasks=NNN</code>	Total number of tasks (cores) for the job.
<code>#SBATCH --ntasks-per-node=XX</code>	Specifies the maximum number of tasks (cores) to allocate per node
<code>#SBATCH --mem=nnnnM</code> or <code>#SBATCH --mem=nG</code> <code>(memory per NODE)</code>	Sets the maximum amount of memory (MB). G for GB is supported on Grace

hprc.tamu.edu/wiki/HPRC:Batch_Translation

Pop Quiz

```
#SBATCH --export=NONE
#SBATCH --get-user-env=L
#SBATCH --job-name=stacks_S2
#SBATCH --ntasks=80
#SBATCH --ntasks-per-node=20
#SBATCH --mem=40G
#SBATCH --time=48:00:00
#SBATCH --output=/scratch/user/dylan/stdout.%J
#SBATCH --error stderr.%J
```

How many nodes is this job requesting?

- A. 1600
- B. 80
- C. 20
- D. 4

Grace Job File (Serial Example)

```
#!/bin/bash  
#
```

SUs = 1.5

```
##NECESSARY JOB SPECIFICATIONS
```

```
#SBATCH --job-name=JobExample1
```

```
#SBATCH --time=01:30:00
```

```
#SBATCH --ntasks=1
```

```
#SBATCH --mem=4G
```

```
#SBATCH --output=stdout.%j
```

```
# Set the job name to "JobExample1"
```

```
# Set the wall clock limit to 1hr and 30min
```

```
# Request 1 task (core)
```

```
# Request 4GB per node
```

```
# Send stdout and stderr to "stdout.[jobID]"
```

```
##OPTIONAL JOB SPECIFICATIONS
```

```
#SBATCH --account=123456
```

```
#SBATCH --mail-type=ALL
```

```
#SBATCH --mail-user=email_address
```

```
# Set billing account to 123456
```

```
# Send email on all job events
```

```
# Send all emails to email_address
```

```
# load required module(s)
```

```
module purge
```

```
module load intel/2022a
```

```
# run your program
```

```
./myprogram
```


Grace Job File (multi core, single node)

```
#!/bin/bash
```

```
#
```

SUs = 156

```
##NECESSARY JOB SPECIFICATIONS
```

```
#SBATCH --job-name=JobExample2
```

```
#SBATCH --time=6:30:00
```

```
#SBATCH --nodes=1
```

```
#SBATCH --ntasks-per-node=24
```

```
#SBATCH --mem=48G
```

```
#SBATCH --output=stdout.%j
```

```
#SBATCH --error=stderr.%j
```

```
# Set the job name to "JobExample2"
```

```
# Set the wall clock limit to 6hr and 30min
```

```
# Request 1 node
```

```
# Request 24 tasks(cores) per node
```

```
# Request 48GB per node
```

```
# Send stdout to "stdout.[jobID]"
```

```
# Send stderr to "stderr.[jobID]"
```

```
##OPTIONAL JOB SPECIFICATIONS
```

```
#SBATCH --account=123456
```

```
#SBATCH --mail-type=ALL
```

```
#SBATCH --mail-user=email_address
```

```
# Set billing account to 123456
```

```
# Send email on all job events
```

```
# Send all emails to email_address
```

```
# load required module(s)
```

```
module purge
```

```
module load GCC/12.2.0
```

```
# run your program
```

```
./my_multicore_program
```


Job Memory Requests on Grace

- Specify memory request based on memory per node:
`#SBATCH --mem=xxxxM` # memory per node in MB
or
`#SBATCH --mem=xG` # memory per node in GB
- On 384GB nodes, usable memory is at most 360 GB.
The per-process memory limit should not exceed ~7500 MB for a 48-core job.
- On 3TB nodes, usable memory is at most 2900 GB.
The per-process memory limit should not exceed 37120 MB for a 48-core job.

Submitting Your Job and Check Job Status

Submit job

```
sbatch example01.job
```

```
Submitted batch job 6853258
(from job_submit) your job is charged as below
      Project Account: 122792016265
      Account Balance: 1687.066160
      Requested SUs:   3
```

Check status

```
squeue -u netID
```

JOBID	NAME	USER	PARTITION	NODES	CPUS	STATE	TIME	TIME_LEFT	START_TIME	REASON	NODELIST
6853258	jobname	someuser	xlong	2	96	RUNNING	3-07:36:50	16:23:10	2023-01-23T17:27:3	None	c[180,202]
6853257	jobname	someuser	xlong	2	96	RUNNING	3-07:36:56	16:23:04	2023-01-23T17:27:2	None	c[523-524]

Hands-On Activity - 3 Minutes

1. Create a submission file for a 1 minute serial job.
Write the following below the line '#run your program':
echo 'Hello World'
2. Submit this job file using sbatch.
3. Check which queue the job is in.
4. When your job completes, check the contents of the output file.

Job Submission and Tracking

SLURM queue command	Description
<code>sbatch jobfile1</code>	Submit jobfile1 to batch system
<code>squeue [-u user_name] [-j job_id]</code>	List jobs
<code>scancel job_id</code>	Kill a job
<code>sacct -X -j job_id</code>	Show information for a job (can be when job is running or recently finished)
<code>sacct -X -S YYYY-HH-MM</code>	Show information for all of your jobs since YYYY-HH-MM
<code>lnu job_id</code>	Show resource usage for a job
<code>pestat -u \$USER</code>	Show resource usage for a running job
<code>seff job_id</code>	Check CPU/memory efficiency for a job

hprc.tamu.edu/wiki/HPRC:Batch_Translation

Job submission issue: insufficient SUs

```
$ sbatch myjob
sbatch: error: (from job_submit) your account's balance is not sufficient to submit your job
      Project Account: 123940134739
      Account Balance: 382.803877
      Requested SUs:   18218.666666667
```

- What to do if you need more SUs
 - Ask your PI to transfer SUs to your account
 - Apply for more SUs (if you are eligible, as a PI or permanent researcher)

hprc.tamu.edu/wiki/HPRC:CommonProblems#Q:_How_do_I_get_more_SUs.3F

hprc.tamu.edu/wiki/HPRC:AMS:Service_Unit

hprc.tamu.edu/wiki/HPRC:AMS:UI

CRLF Line Terminators

Windows editors such as Notepad will add hidden Carriage Return Line Feed (CRLF) characters that will cause problems with many applications

```
cd $SCRATCH/batch_examples
```

```
file dos_text.txt
```

use file command to check

```
dos_text.txt: ASCII English text, with CRLF line terminators
```

```
cat -v dos_text.txt
```

use cat command to see CRLF characters

```
dos2unix dos_text.txt  
file dos_text.txt
```

use dos2unix command to correct

```
dos_text.txt: ASCII English text
```

Need Help?

First check the [FAQ](#)

- [Grace User Guide](#)
- Email your questions to help@hprc.tamu.edu

Help us, help you -- we need more info

- Which Cluster
- Username
- Job id(s) if any
- Location of your jobfile, input/output files
- Application used if any
- Module(s) loaded if any
- Error messages
- Steps you have taken, so we can reproduce the problem

Continued Learning

[Intro to HPRC Video Tutorial Series](#)

[HPRC's Wiki Page](#)



High Performance
Research Computing
DIVISION OF RESEARCH

Thank you.

Any questions?