HIGH PERFORMANCE RESEARCH COMPUTING

- HPRC Primer -

Introduction to Grace: An HPRC Resource

August 30, 2024

High Performance
Research Computing
DIVISION OF RESEARCH

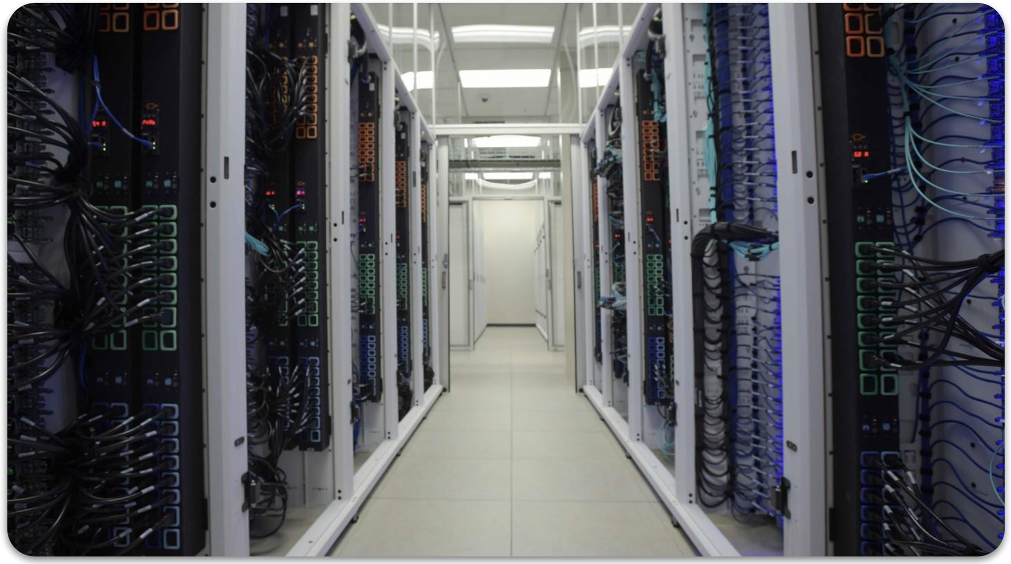
Grace Hardware

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

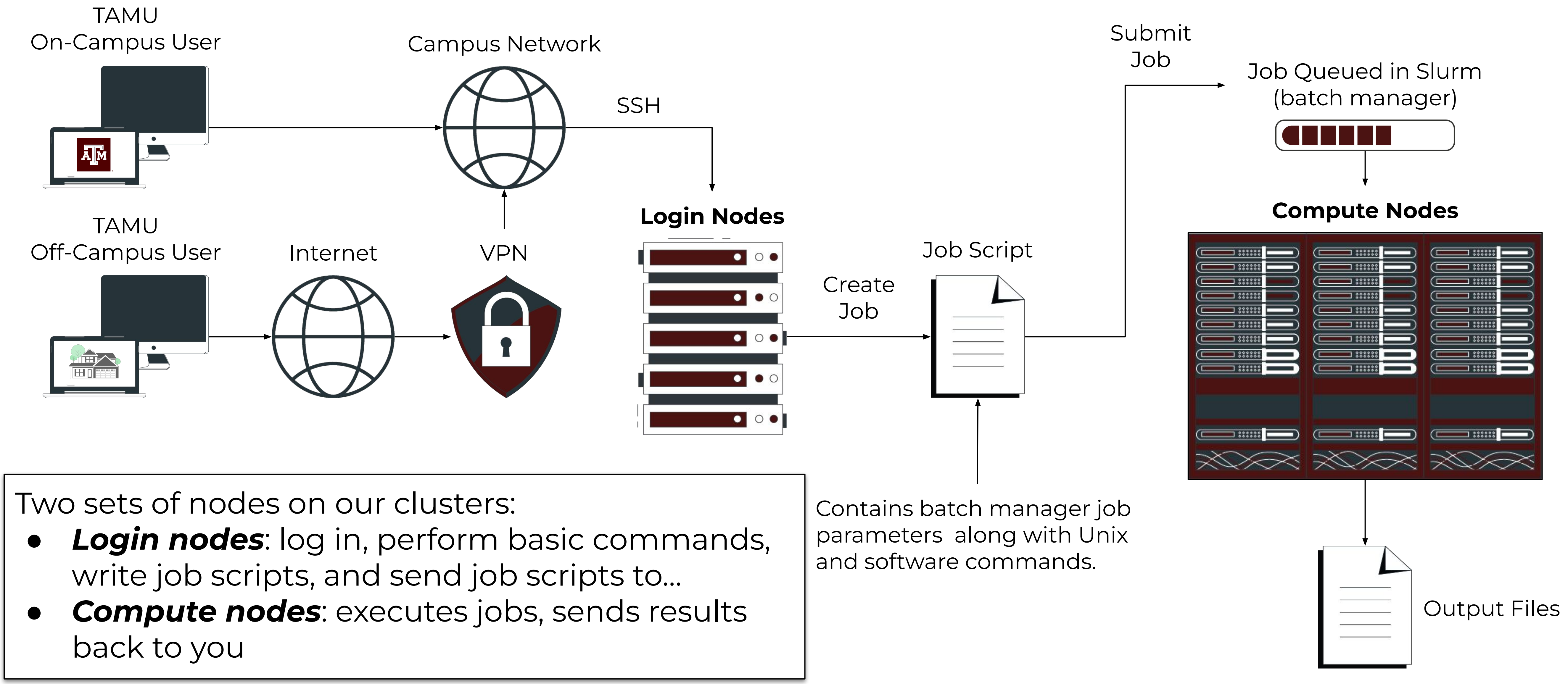
3TB Large Memory-80 cores/node
Login Nodes: 10 GbE TAMU network connection

Resource	Count
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:
https://hprc.tamu.edu/kb/User-Guides/Grace/

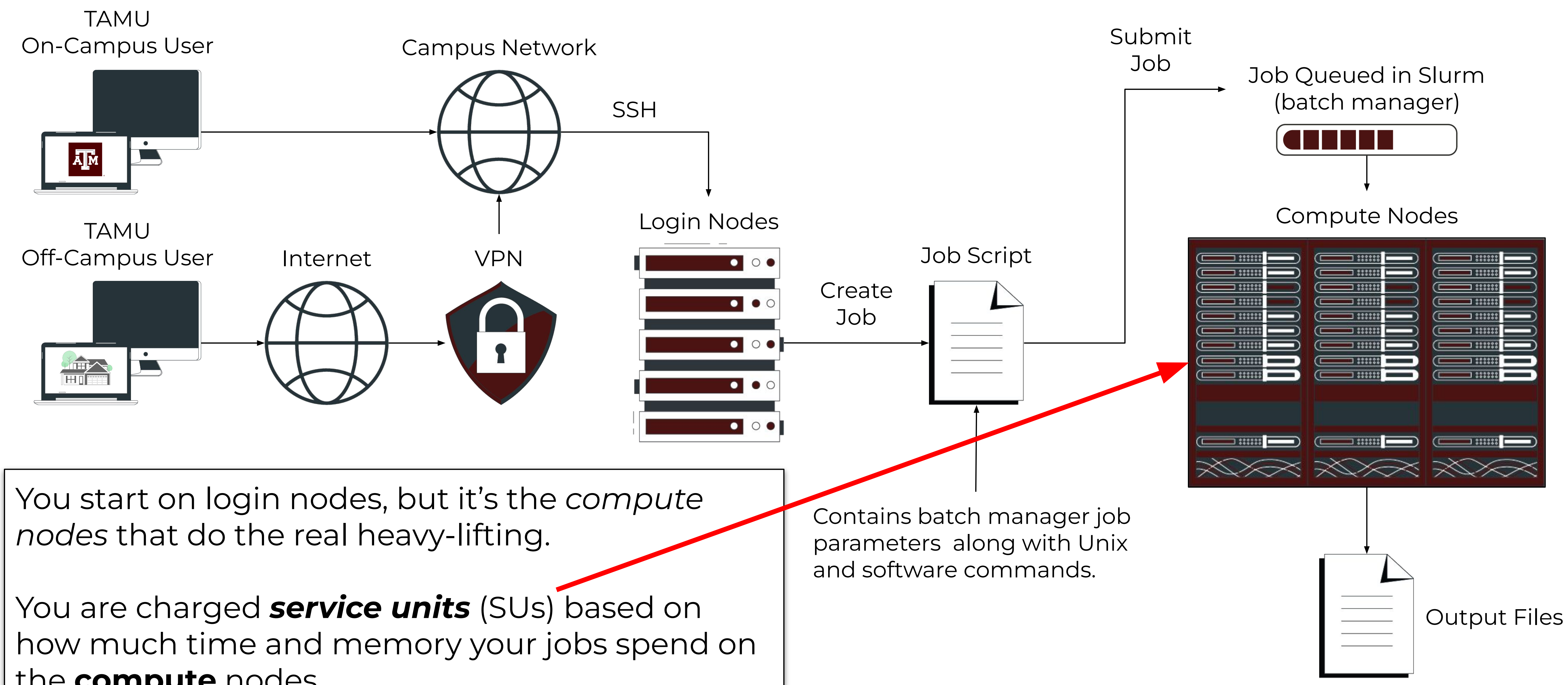
Computing on HPRC Clusters



Two sets of nodes on our clusters:

- ***Login nodes***: log in, perform basic commands, write job scripts, and send job scripts to…
- ***Compute nodes***: executes jobs, sends results back to you

Batch Jobs on HPRC Clusters



Contains batch manager job parameters along with Unix and software commands.

You start on login nodes, but it’s the *compute nodes* that do the real heavy-lifting.

You are charged ***service units*** (SUs) based on how much time and memory your jobs spend on the **compute** nodes.

File Quotas and Resource Allocations

- Two things to keep track of when computing

<u>Quota</u>

File Storage

File Count
File Space

<u>Allocation</u>

Computing Resources

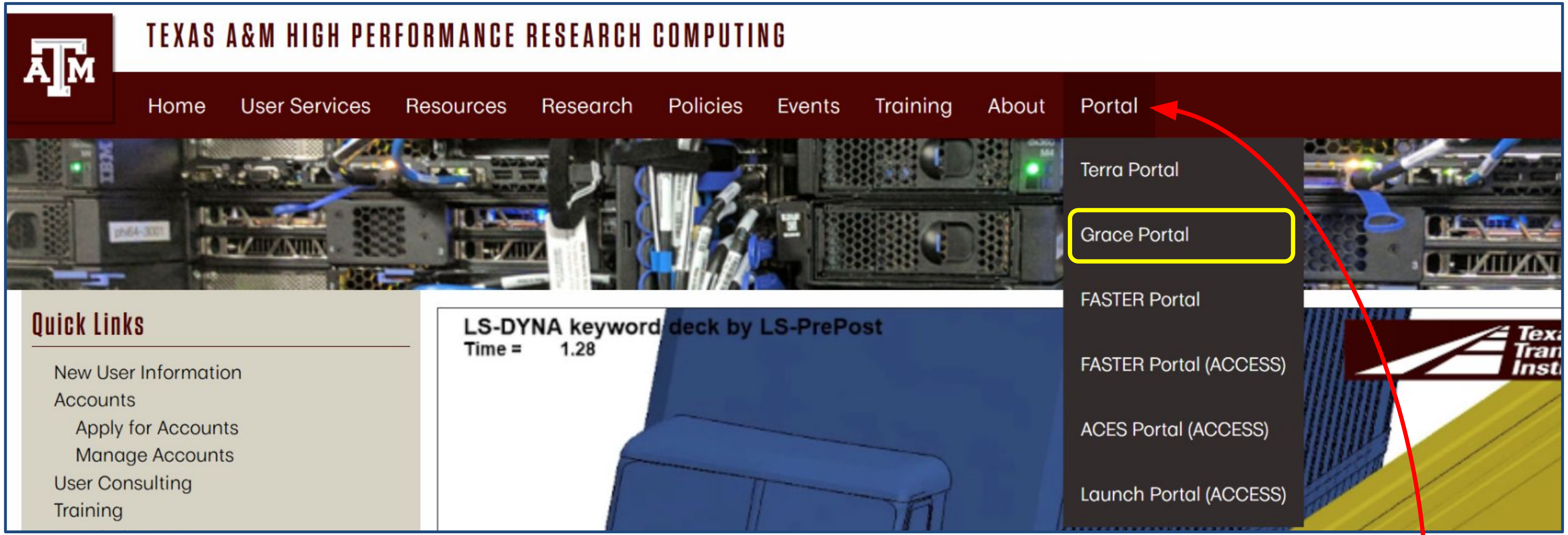
Service Units (SUs)

Accessing Grace: Setup

- If off-campus:
  Set up and start VPN (Virtual Private Network):
  u.tamu.edu/VPnetwork
- *Two-Factor Authentication* required

- Today we'll access Grace via the online Portal, but you can also use ssh.
- See https://hprc.tamu.edu/kb/User-Guides/Grace/Access/ for more details.

Accessing Grace via the Portal



TEXAS A&M HIGH PERFORMANCE RESEARCH COMPUTING

Home User Services Resources Research Policies Events Training About Portal

Terra Portal
Grace Portal
FASTER Portal
FASTER Portal (ACCESS)
ACES Portal (ACCESS)
Launch Portal (ACCESS)

Quick Links

New User Information
Accounts
Apply for Accounts
Manage Accounts
User Consulting
Training

LS-DYNA keyword deck by LS-PrePost
Time = 1.28

Access the HPRC portals through most web browsers:

1. Go to portal.hprc.tamu.edu or use the Portal dropdown menu on the HPRC homepage: https://hprc.tamu.edu/
2. Choose **Grace Portal**

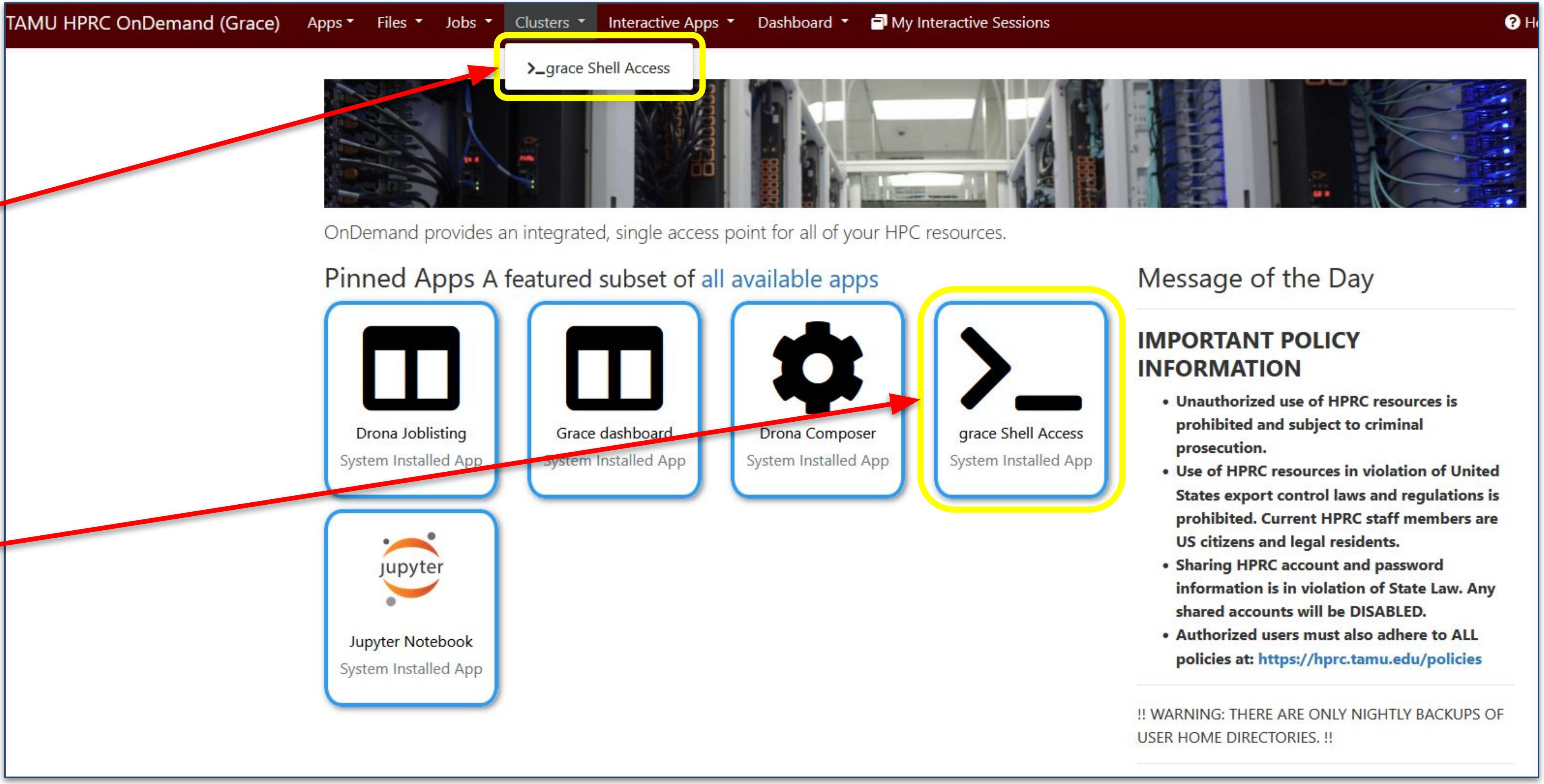
https://hprc.tamu.edu/kb/User-Guides/Grace/Access/

Accessing Grace via the Portal

Two ways to enter the Portal:

(1) Select at the top: “Clusters” → “Grace Shell* Access”

(2) Select in Main Menu: “Grace Shell Access”



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

Pinned Apps A featured subset of all available apps

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. Current HPRC staff members are US citizens and legal residents.**
- **Sharing HPRC account and password information is in violation of State Law. Any shared accounts will be DISABLED.**
- **Authorized users must also adhere to ALL policies at: https://hprc.tamu.edu/policies**

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

**shell* is also called *terminal* or *command line*

https://hprc.tamu.edu/kb/User-Guides/Grace/Access/

Hands-On Activity - 2 Minutes

Try to access a *shell** on Grace now, either through portal.hprc.tamu.edu or hprc.tamu.edu

*(also called *terminal* or *command line*)

What message do you see when you login?

Remember Grace has 5 login nodes. Which one does your command prompt say you got?

File Systems and User Directories

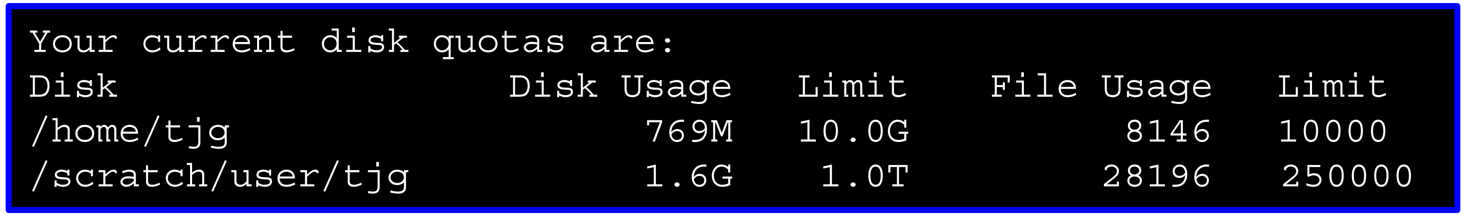
Directory	Environment Variable	Space Limit	File Limit	Intended Use
/home/$USER	$HOME	10 GB	10,000	Small to modest amounts of processing. Backed up nightly.
/scratch/user/$USER	$SCRATCH*	1 TB	250,000	Temporary storage of large files for on-going computations. Not intended to be a long-term storage area.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

*Do NOT share your home or scratch directories.

Request a group directory for sharing files.

$SCRATCH is shared between the FASTER and Grace clusters.

View file usage and quota limits in the shell using the command: **showquota**



```
Your current disk quotas are:
Disk                 Disk Usage   Limit    File Usage   Limit
/home/tjg                  769M   10.0G          8146   10000
/scratch/user/tjg          1.6G    1.0T         28196   250000
```

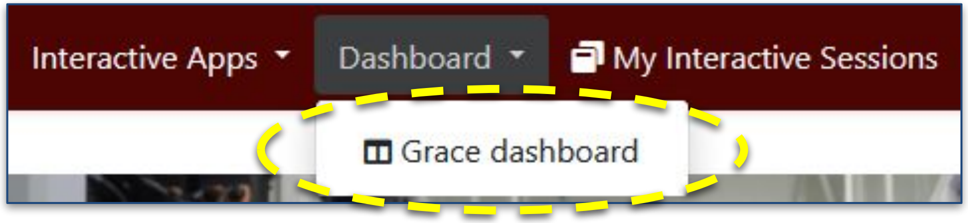
https://hprc.tamu.edu/kb/User-Guides/Grace/Filesystems_and_Files/

Portal: Grace Dashboard

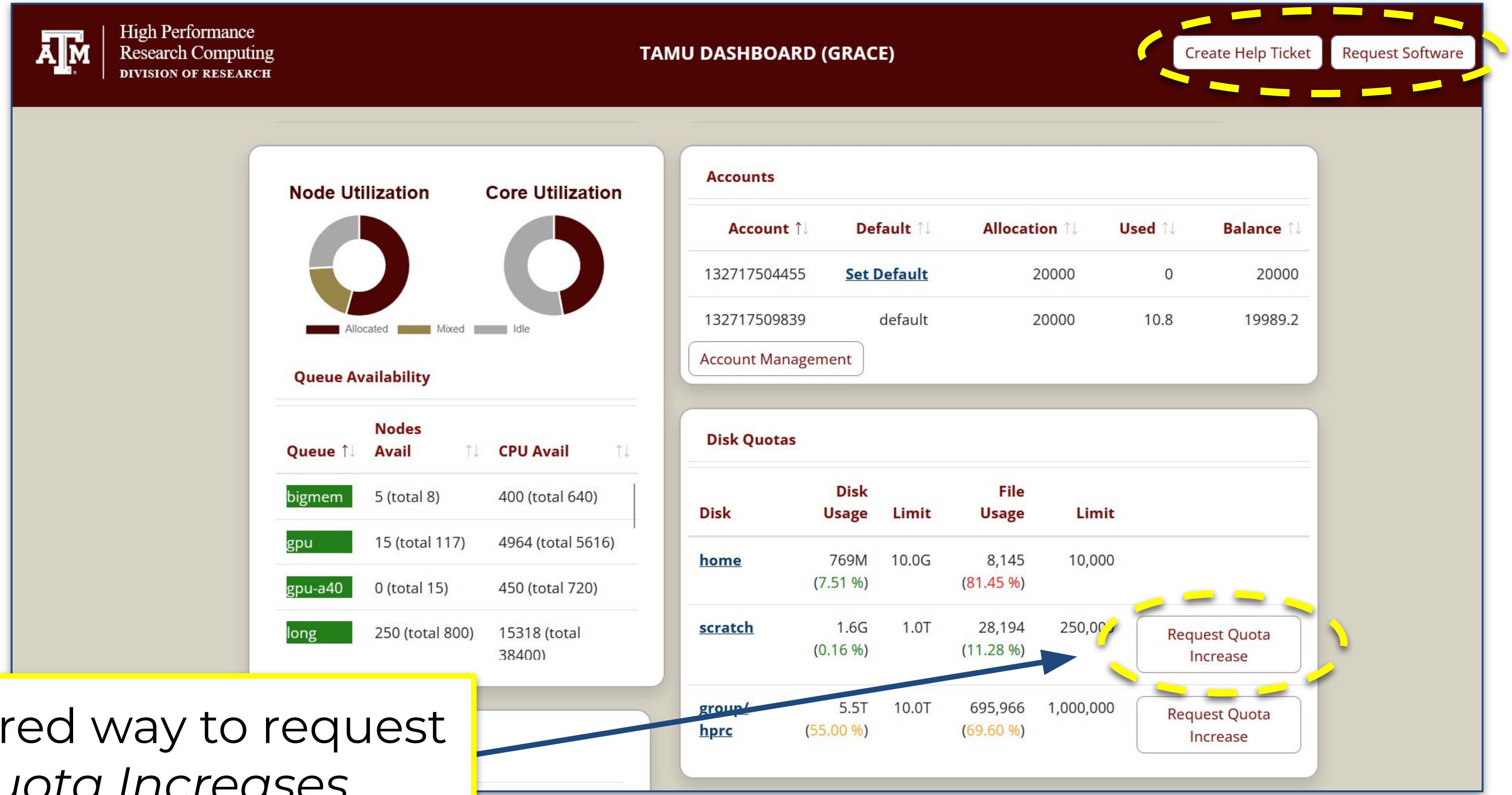
- Easily view Cluster utilization, <u>Storage Quotas</u>, & <u>Allocation Balances</u>

- Ask for help and request software

- Also View current groups and job listings in the Grace Dashboard



Request help or software



Queue Availability

Queue	Nodes Avail	CPU Avail
bigmem	5 (total 8)	400 (total 640)
gpu	15 (total 117)	4964 (total 5616)
gpu-a40	0 (total 15)	450 (total 720)
long	250 (total 800)	15318 (total 38400)

Accounts

Account	Default	Allocation	Used	Balance
132717504455	Set Default	20000	0	20000
132717509839	default	20000	10.8	19989.2

Disk Quotas

Disk	Disk Usage	Limit	File Usage	Limit
home	769M (7.51 %)	10.0G	8,145 (81.45 %)	10,000
scratch	1.6G (0.16 %)	1.0T	28,194 (11.28 %)	250,00
group/hprc	5.5T (55.00 %)	10.0T	695,966 (69.60 %)	1,000,000

Preferred way to request *Quota Increases*

Hands-On Activity - 2 Minutes

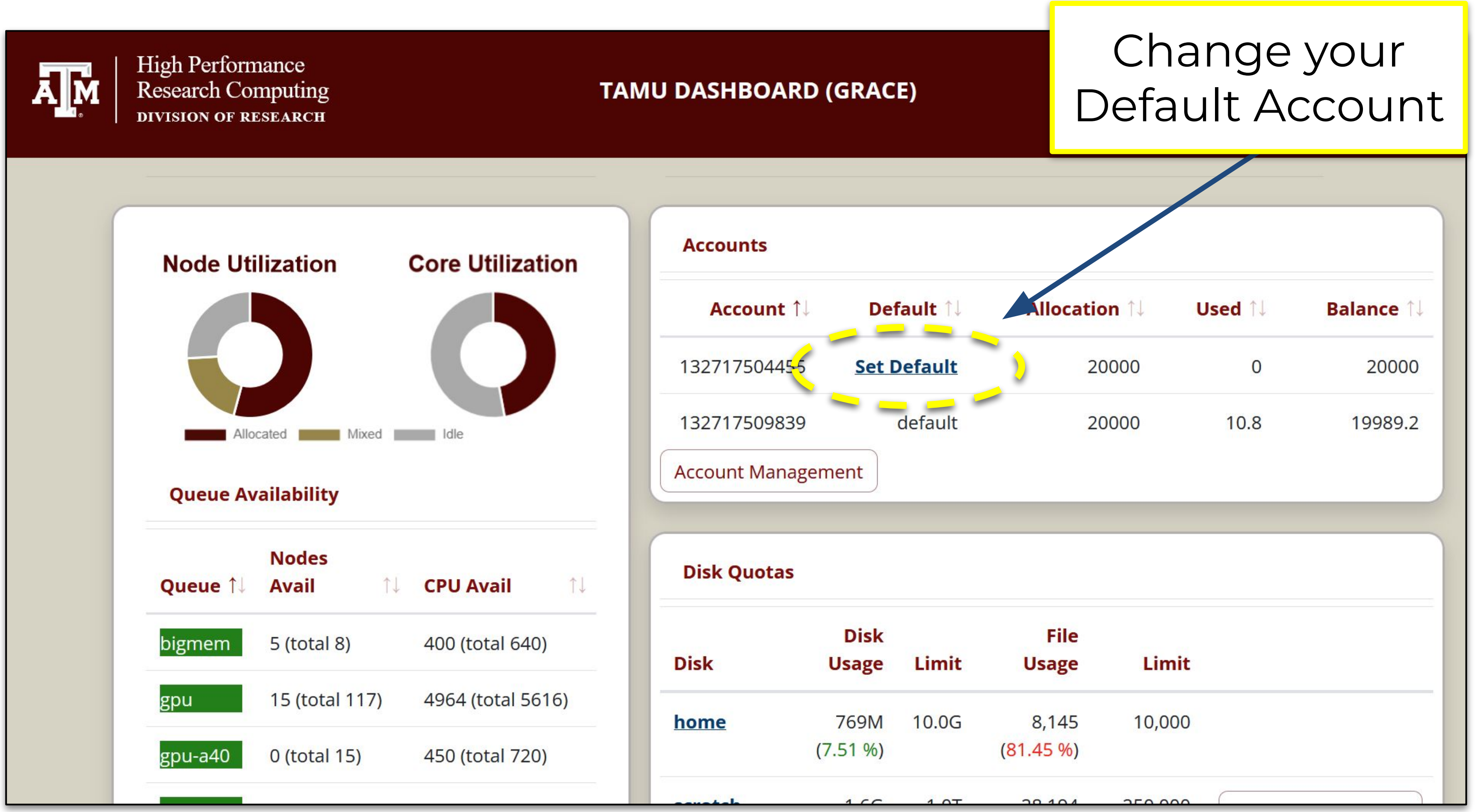
1. Please try to access dashboard now through the portal.

2. Check your quotas both on the command line and on the dashboard.

```
showquota
```

Checking Your Service Unit (SUs) Balance

- 1 SU = 1 core/hr (GPUs are more expensive per-hour!)
- SUs are charged to default account when none is specified.



Change your Default Account

High Performance Research Computing
DIVISION OF RESEARCH

TAMU DASHBOARD (GRACE)

Node Utilization

Core Utilization

Queue Availability

Queue ↑↓	Nodes Avail ↑↓	CPU Avail ↑↓
bigmem	5 (total 8)	400 (total 640)
gpu	15 (total 117)	4964 (total 5616)
gpu-a40	0 (total 15)	450 (total 720)

Accounts

Account ↑↓	Default ↑↓	Allocation ↑↓	Used ↑↓	Balance ↑↓
132717504455	Set Default	20000	0	20000
132717509839	default	20000	10.8	19989.2

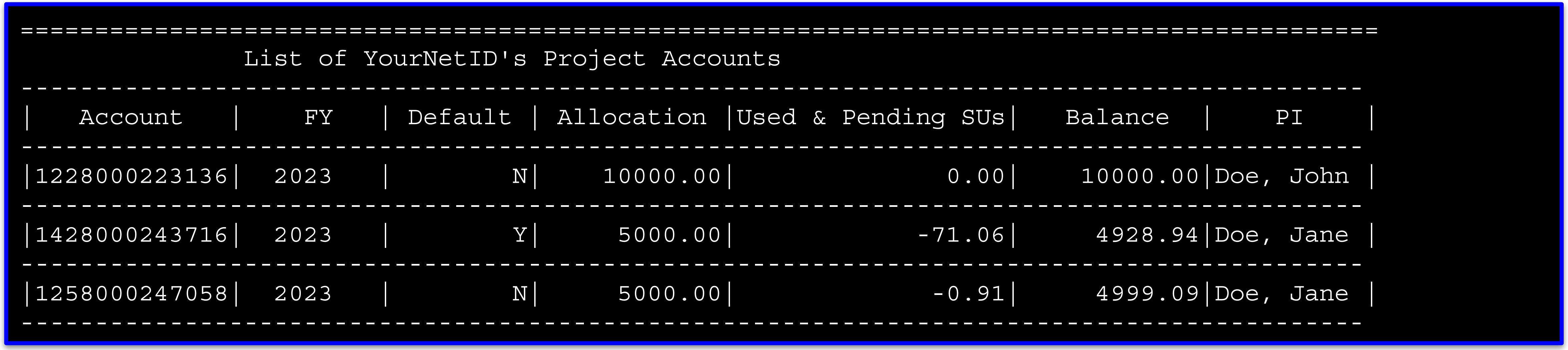
Account Management

Disk Quotas

Disk	Disk Usage	Limit	File Usage	Limit
home	769M (7.51 %)	10.0G	8,145 (81.45 %)	10,000

Checking Your SUs in the Shell

- List the SU Balance of your Account(s) with: `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

  (replace <Account#> with your number!)

- Run `myproject -h` to see more options

https://hprc.tamu.edu/kb/User-Guides/AMS/UI/
https://hprc.tamu.edu/kb/User-Guides/AMS/Service_Unit/

Hands-On Activity - 2 Minutes

1. Use **`myproject`** to check the SU balance of your accounts.

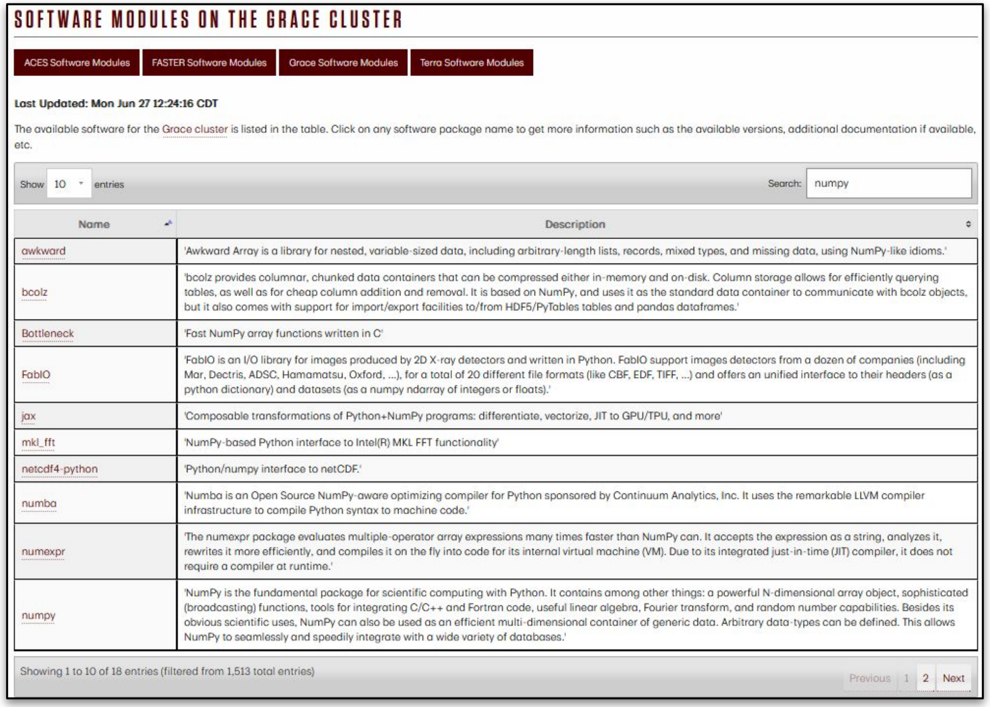
2. Use the dashboard the check the same information.

Software

- Search for software modules on https://hprc.tamu.edu/software/grace/

- See the Software Knowledge Base page https://hprc.tamu.edu/kb/Software/ for instructions and examples

- License-restricted software
  - Contact help@hprc.tamu.edu



SOFTWARE MODULES ON THE GRACE CLUSTER

Last Updated: Mon Jun 27 12:24:16 CDT

The available software for the Grace cluster is listed in the table. Click on any software package name to get more information such as the available versions, additional documentation if available, etc.

Show 10 entries   Search: numpy

Name	Description
awkward	'Awkward Array is a library for nested, variable-sized data, including arbitrary-length lists, records, mixed types, and missing data, using NumPy-like idioms.'
bcolz	'bcolz provides columnar, chunked data containers that can be compressed either in-memory and on-disk. Column storage allows for efficiently querying tables, as well as for cheap column addition and removal. It is based on NumPy, and uses it as the standard data container to communicate with bcolz objects, but it also comes with support for import/export facilities to/from HDF5/PyTables tables and pandas dataframes.'
Bottleneck	'Fast NumPy array functions written in C'
FabIO	'FabIO is an I/O library for images produced by 2D X-ray detectors and written in Python. FabIO support images detectors from a dozen of companies (including Mar, Dectris, ADSC, Hamamatsu, Oxford, ...), for a total of 20 different file formats (like CBF, EDF, TIFF, ...) and offers an unified interface to their headers (as a python dictionary) and datasets (as a numpy ndarray of integers or floats).'
jax	'Composable transformations of Python+NumPy programs: differentiate, vectorize, JIT to GPU/TPU, and more'
mkl_fft	'NumPy-based Python interface to Intel(R) MKL FFT functionality'
netcdf4-python	'Python/numpy interface to netCDF.'
numba	'Numba is an Open Source NumPy-aware optimizing compiler for Python sponsored by Continuum Analytics, Inc. It uses the remarkable LLVM compiler infrastructure to compile Python syntax to machine code.'
numexpr	'The numexpr package evaluates multiple-operator array expressions many times faster than NumPy can. It accepts the expression as a string, analyzes it, rewrites it more efficiently, and compiles it on the fly into code for its internal virtual machine (VM). Due to its integrated just-in-time (JIT) compiler, it does not require a compiler at runtime.'
numpy	'NumPy is the fundamental package for scientific computing with Python. It contains among other things: a powerful N-dimensional array object, sophisticated (broadcasting) functions, tools for integrating C/C++ and Fortran code, useful linear algebra, Fourier transform, and random number capabilities. Besides its obvious scientific uses, NumPy can also be used as an efficient multi-dimensional container of generic data. Arbitrary data-types can be defined. This allows NumPy to seamlessly and speedily integrate with a wide variety of databases.'

Showing 1 to 10 of 18 entries (filtered from 1,513 total entries)

- Contact HPRC (can use the dashboard) 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 available to use with the loaded modules
`module load PyTorch/1.12.1`	← load PyTorch version 1.12.1
`module list`	← see what software you've loaded already

Module System Youtube Video → www.youtube.com/watch?v=drxpbrOCPFw

Software: Modules and Toolchains

- Toolchains are what we call groups of compilers & libraries
- There’s a variety of toolchains available on the clusters:
    - intel/2023b
    - iomkl/2020a
    - foss/2023b
    - GCCcore/13.2.0

(more than just these versions)

- Other module commands:

```
module spider
module purge
```

← search for modules and their dependencies
← removes all loaded modules

Module System Youtube Video → www.youtube.com/watch?v=drxpbrOCPFw

Hands-On Activity - 5 Minutes

Remember: `module load` `module purge`

1. Please search for and load the following module:

```
OpenMPI/4.1.4
```

(Tip) Type this to show which compiler needs to be loaded:

```
module spider OpenMPI/4.1.4
```

(Tip) And check that it's been loaded with:

```
module list
```

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

Module System Youtube Video → www.youtube.com/watch?v=drxpbrOCPFw

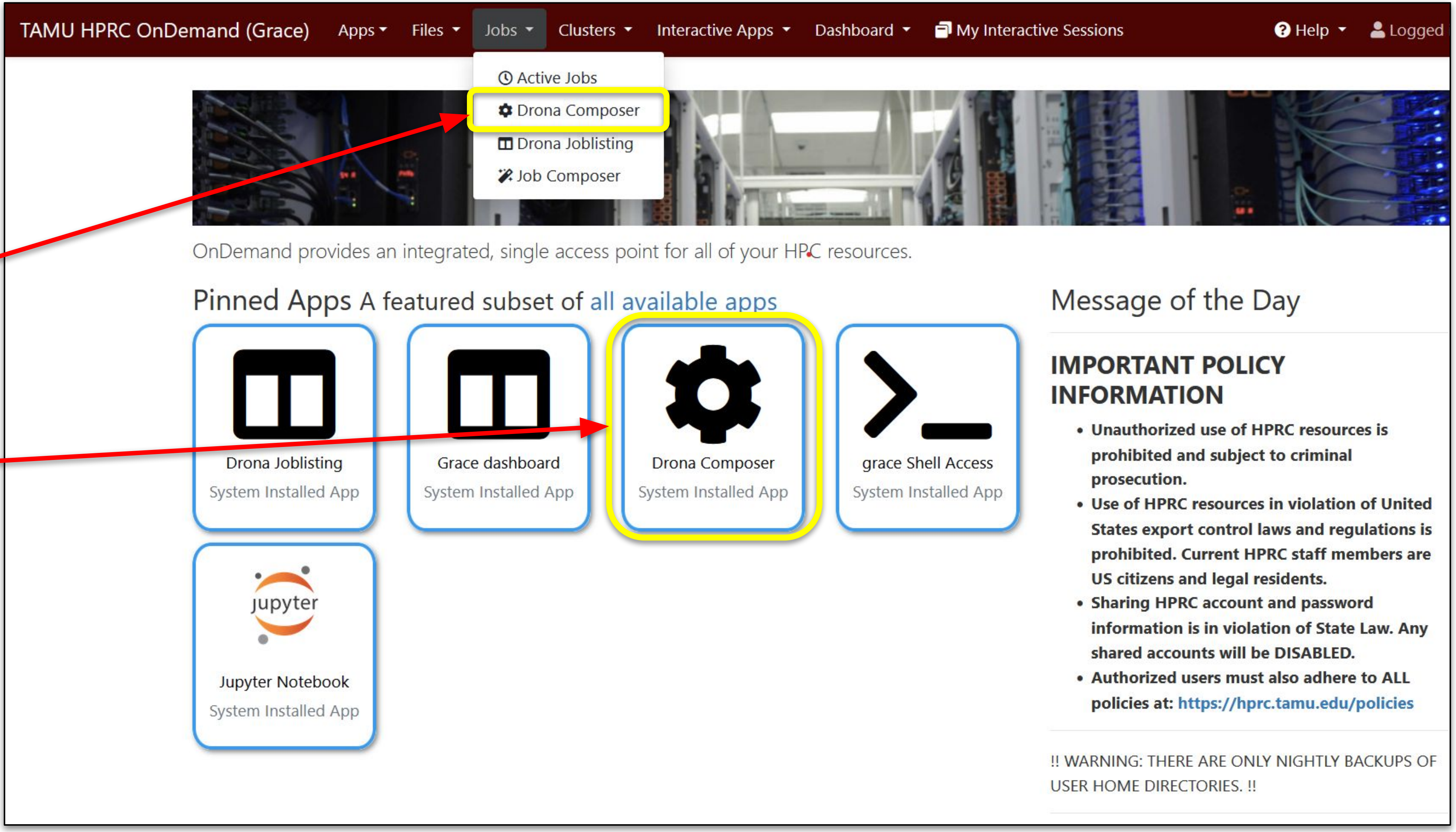
The Drona Composer

- A simple app to assist you with composing jobs

Two ways to enter the Portal:

(1) Select under “Jobs” tab

(2) Select in Main Menu



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

Pinned Apps A featured subset of all available apps

Message of the Day

IMPORTANT POLICY INFORMATION

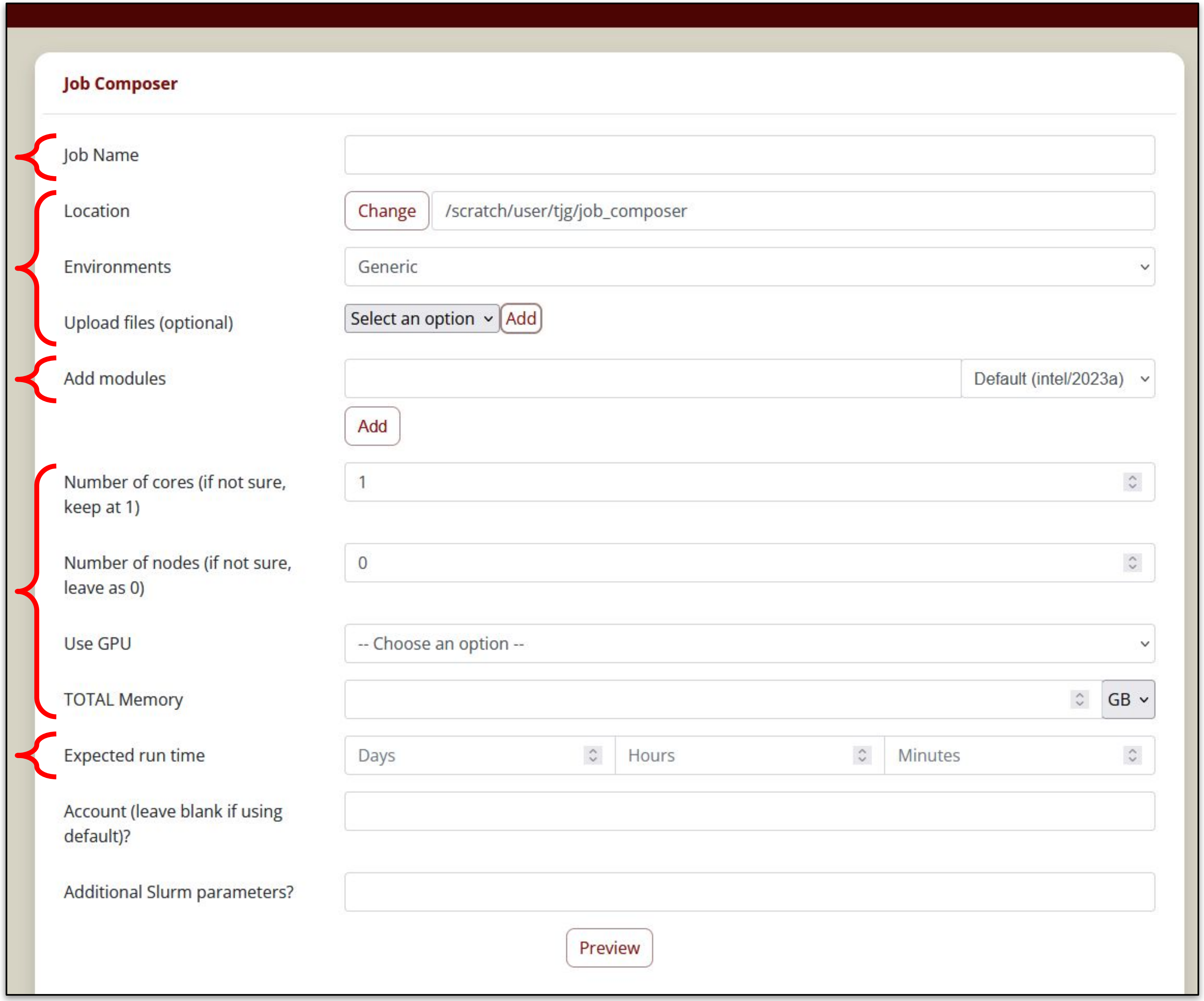
- **Unauthorized use of HPRC resources is prohibited and subject to criminal prosecution.**
- **Use of HPRC resources in violation of United States export control laws and regulations is prohibited. Current HPRC staff members are US citizens and legal residents.**
- **Sharing HPRC account and password information is in violation of State Law. Any shared accounts will be DISABLED.**
- **Authorized users must also adhere to ALL policies at: https://hprc.tamu.edu/policies**

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

The Drona Composer

Create your own jobs through the app:

(1) Pick a job name
(2) Chose location and environments
(3) Add modules
(4) Chose Resources
(5) Chose Time



Job Composer

Job Name	
Location	Change /scratch/user/tjg/job_composer
Environments	Generic
Upload files (optional)	Select an option Add
Add modules	Default (intel/2023a) Add
Number of cores (if not sure, keep at 1)	1
Number of nodes (if not sure, leave as 0)	0
Use GPU	-- Choose an option --
TOTAL Memory	GB
Expected run time	Days Hours Minutes
Account (leave blank if using default)?	
Additional Slurm parameters?	

Hands-On Activity - 3 Minutes

Using the Drona composer, create a job script with the following criteria:

- Generic Environment
- Modules Loaded: **`foss/2023b`**, **`Python/3.11.5`**
- 6 Cores
- 2 Nodes
- No GPU
- 30 GB of Memory
- 3 Minute Runtime

Preview the job script when you are done

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=2G
#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/11.3.0 Python/3.10.4

# Run your program
python my_program.py
```

These *parameters* describe your job to the Slurm job scheduler.
The lines starting with #SBATCH are NOT comments!
See the Knowledge Base for more info

← Account number to be charged

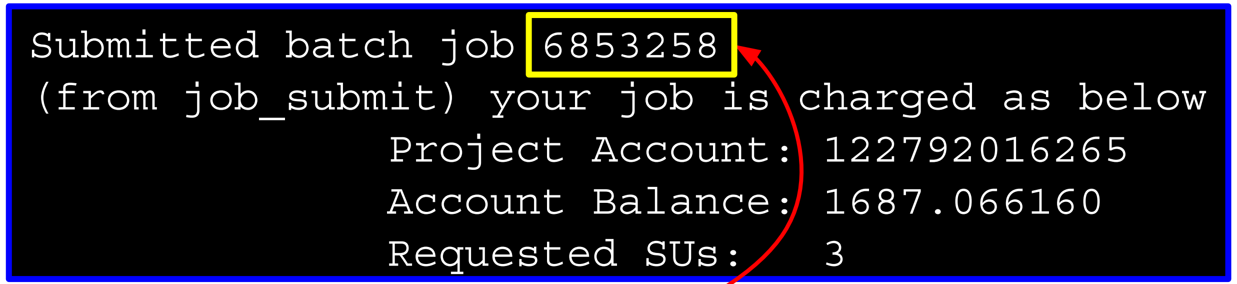
Whatever commands or scripts you want to run. Here, we set up the modules we need for our environment and run a python program.

(We will practice with job files in a few slides)!

Submit a 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
```

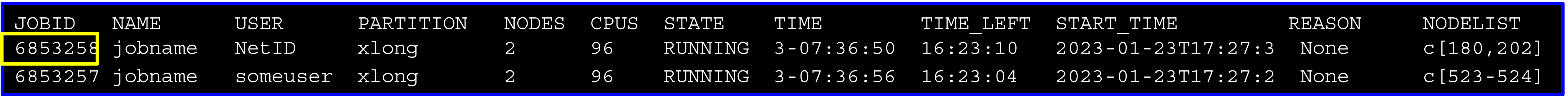
Check status

matching JOBID

```
squeue -u netID
```

or

```
squeue --me
```



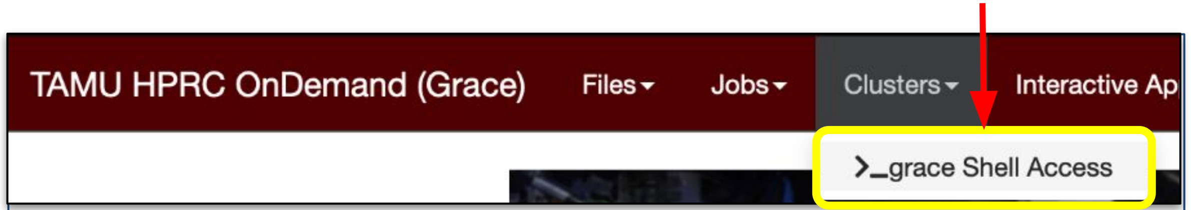
```
JOBID   NAME     USER     PARTITION  NODES  CPUS  STATE    TIME        TIME_LEFT  START_TIME          REASON   NODELIST
6853258 jobname  NetID    xlong      2      96    RUNNING  3-07:36:50  16:23:10   2023-01-23T17:27:3  None     c[180,202]
6853257 jobname  someuser xlong      2      96    RUNNING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

Hands-On Activity

1. Navigate to `/scratch/training/Intro-to-Grace`
2. Copy `hello_world.slurm` and `hello_world.py` to your home directory
3. Return to your home directory and submit the job file using `sbatch`.
4. Check that the job is running in a Slurm queue with `squeue`.
5. When your job completes, check the contents of the output file.

Hands-On Activity

Linux command line tools



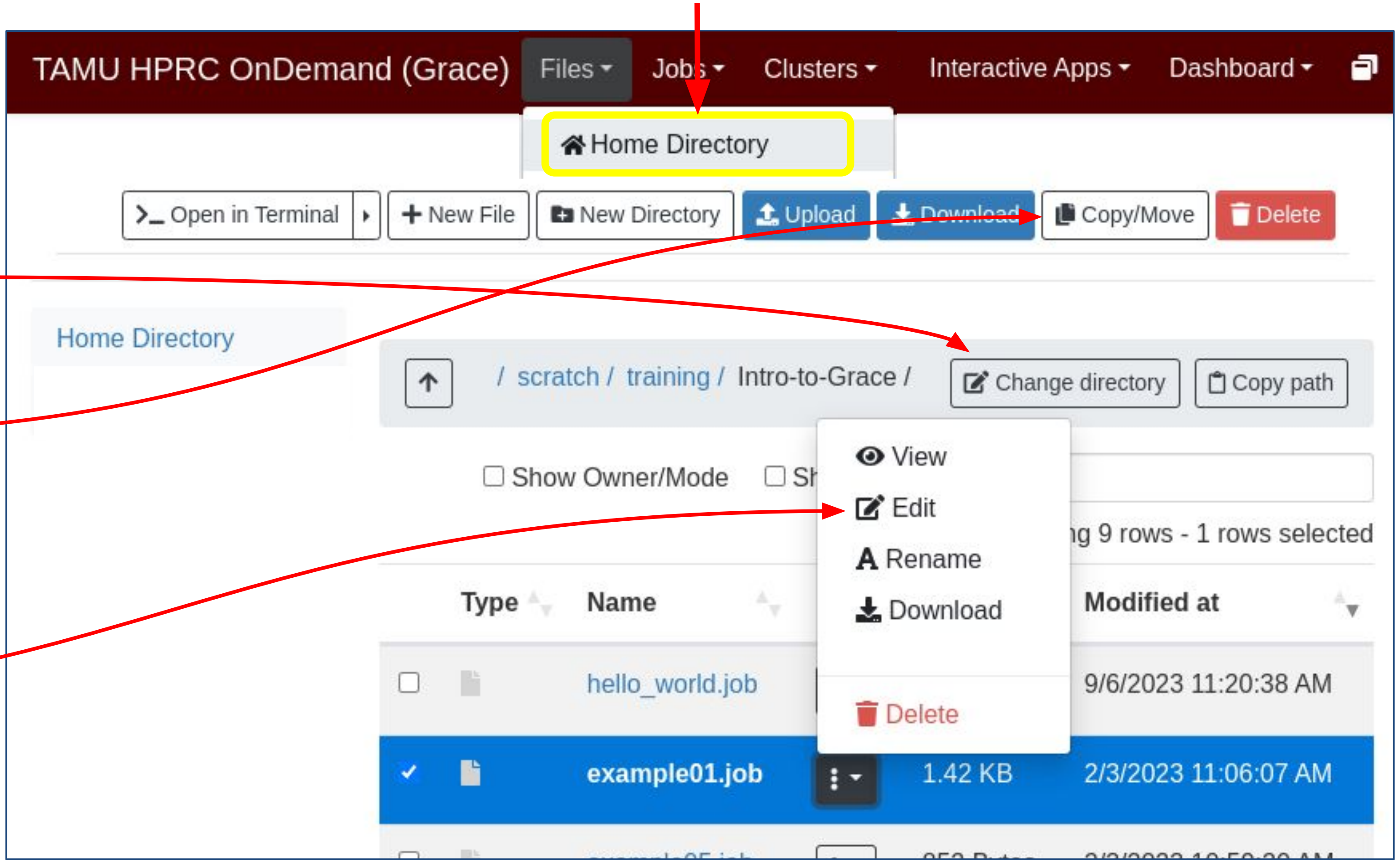
```
cd /scratch/training/Intro-to-Grace
```

```
cp example01.job $HOME
```

```
cd ~
```

```
vi example01.job
```

Portal file navigator

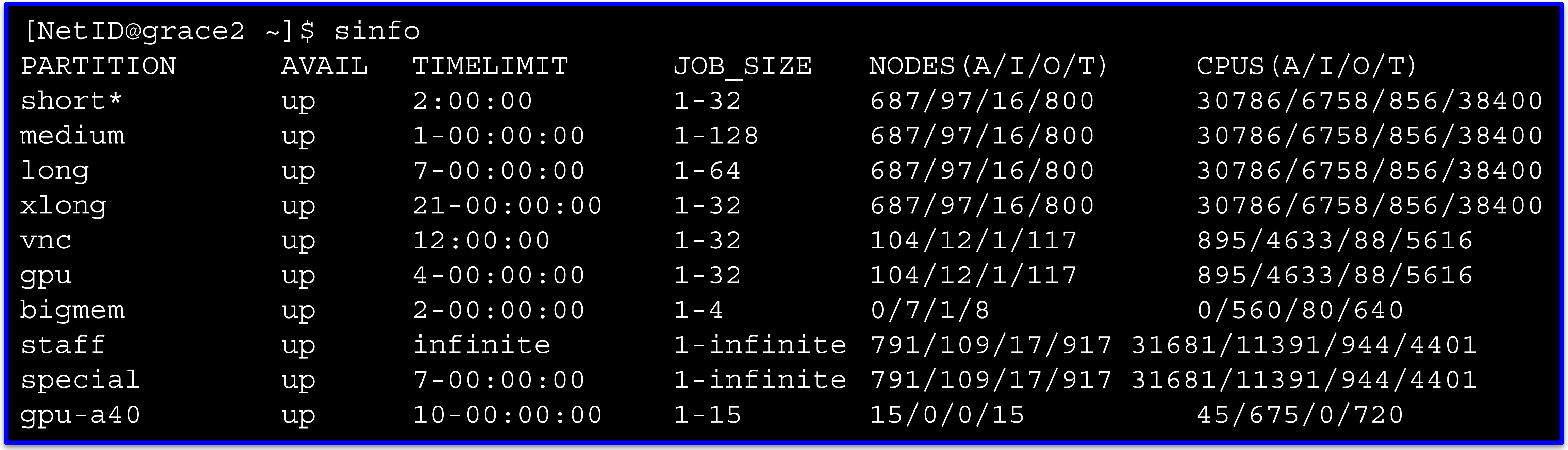


Type	Name	Size	Modified at
☐	hello_world.job		9/6/2023 11:20:38 AM
✓	example01.job	1.42 KB	2/3/2023 11:06:07 AM

Then submit and check on the command line with **`sbatch`** and **`squeue`**.

Batch Queues

- Job submissions are auto-assigned to batch queues based on the resources requested (e.g. number of cores/nodes and walltime limit)
- Use **`sinfo`** to check their status:



```
[NetID@grace2 ~]$ sinfo
PARTITION      AVAIL  TIMELIMIT      JOB_SIZE    NODES(A/I/O/T)       CPUS(A/I/O/T)
short*         up     2:00:00        1-32        687/97/16/800        30786/6758/856/38400
medium         up     1-00:00:00     1-128       687/97/16/800        30786/6758/856/38400
long           up     7-00:00:00     1-64        687/97/16/800        30786/6758/856/38400
xlong          up     21-00:00:00    1-32        687/97/16/800        30786/6758/856/38400
vnc            up     12:00:00       1-32        104/12/1/117         895/4633/88/5616
gpu            up     4-00:00:00     1-32        104/12/1/117         895/4633/88/5616
bigmem         up     2-00:00:00     1-4         0/7/1/8              0/560/80/640
staff          up     infinite       1-infinite  791/109/17/917   31681/11391/944/4401
special        up     7-00:00:00     1-infinite  791/109/17/917   31681/11391/944/4401
gpu-a40        up     10-00:00:00    1-15        15/0/0/15            45/675/0/720
```

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**

https://hprc.tamu.edu/kb/User-Guides/Grace/Batch/#batch-queues

Job Submission and Tracking

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

https://hprc.tamu.edu/kb/Helpful-Pages/Batch-Translation/

Need Help?

First check the FAQ

- Grace User Guide
- Email your questions to help@hprc.tamu.edu

Help us help you -- when you contact us, tell us:

- Which Cluster you're using
- Your 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 Knowledge Base

High Performance
Research Computing
DIVISION OF RESEARCH

Thank you.

Any questions?

Please let us know what you thought of this course by filling out this survey:
https://u.tamu.edu/hprc_shortcourse_survey

High Performance
Research Computing
DIVISION OF RESEARCH

Thank you.

Any questions?

Accessing Grace: via SSH

- SSH command is required for accessing Grace / Terra:
  - If off-campus:
    Set up and start VPN (Virtual Private Network): u.tamu.edu/VPnetwork
  - If on-campus or you've done the above:

```
ssh NetID@grace.tamu.edu
```

  - *Two-Factor Authentication* enabled for CAS, VPN, SSH
- Mac and Linux users can use their "Terminal" applications. For Windows:
  - MobaXTerm (preferred, includes SSH and X11)
  - PuTTY SSH
  - Windows Subsystem for Linux
- Grace has 5 login nodes (check your bash prompt to see which you're on).
  - 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.

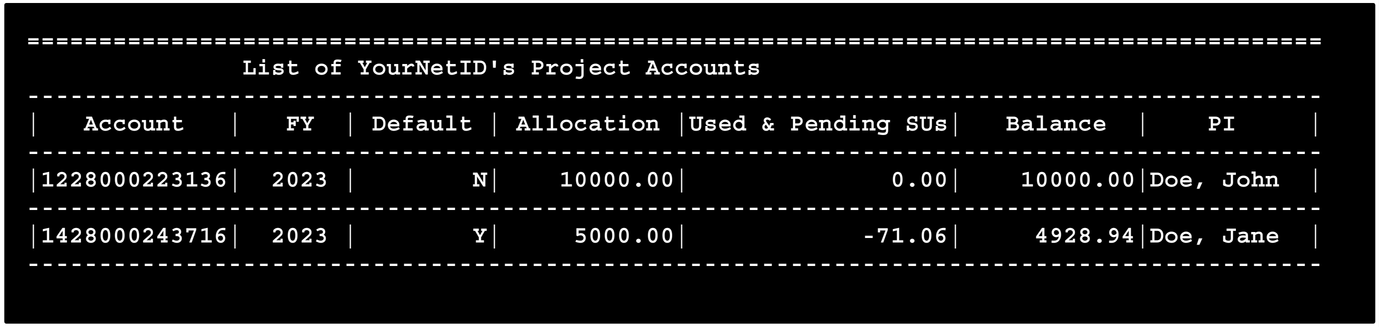
https://hprc.tamu.edu/kb/User-Guides/Grace/Access/

Consumable Computing Resources

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

https://hprc.tamu.edu/kb/User-Guides/AMS/Service_Unit/

```
myproject
```

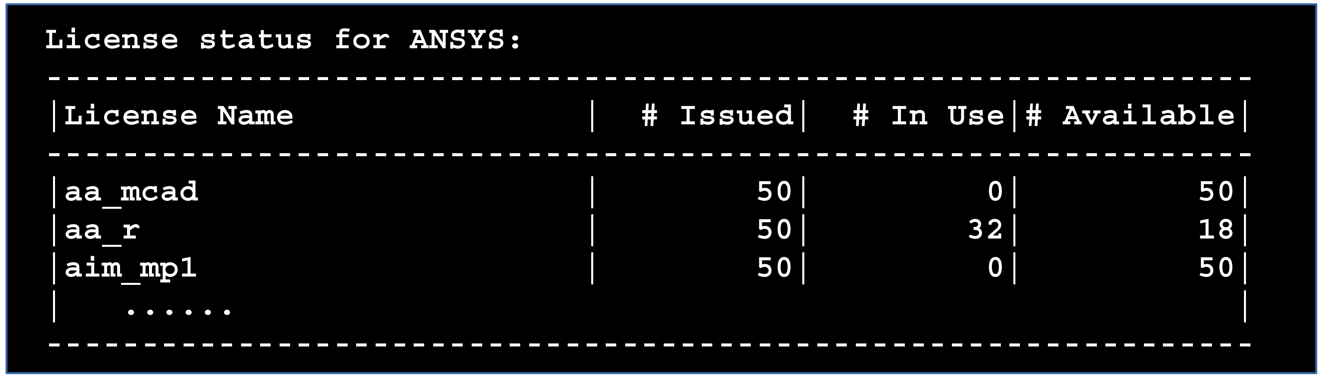


```
=====================================================================================
                 List of YourNetID's Project Accounts
-------------------------------------------------------------------------------------
|    Account    |  FY  | Default | Allocation |Used & Pending SUs|   Balance   |   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
    - https://hprc.tamu.edu/kb/Software/useful-tools/License_Checker/

Find available license for "ansys":

```
license_status -s ansys
```



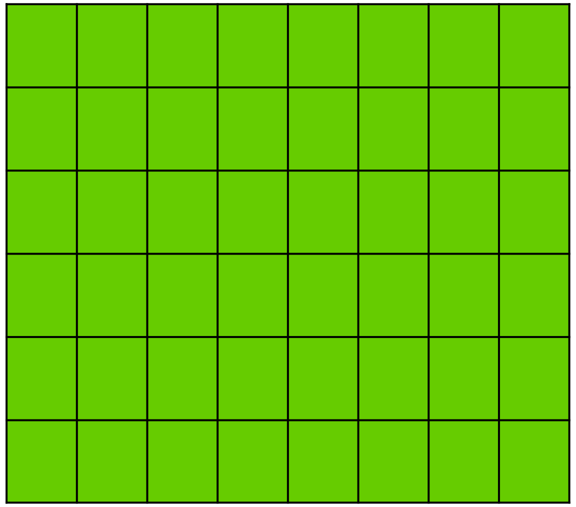
```
License status for ANSYS:
-------------------------------------------------------------------
|License Name                      |  # Issued|  # In Use|# Available|
-------------------------------------------------------------------
|aa_mcad                           |        50|         0|     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

For more information about this command:

```
license_status -h
```

Mapping Jobs to Cores per Node on Grace

A.

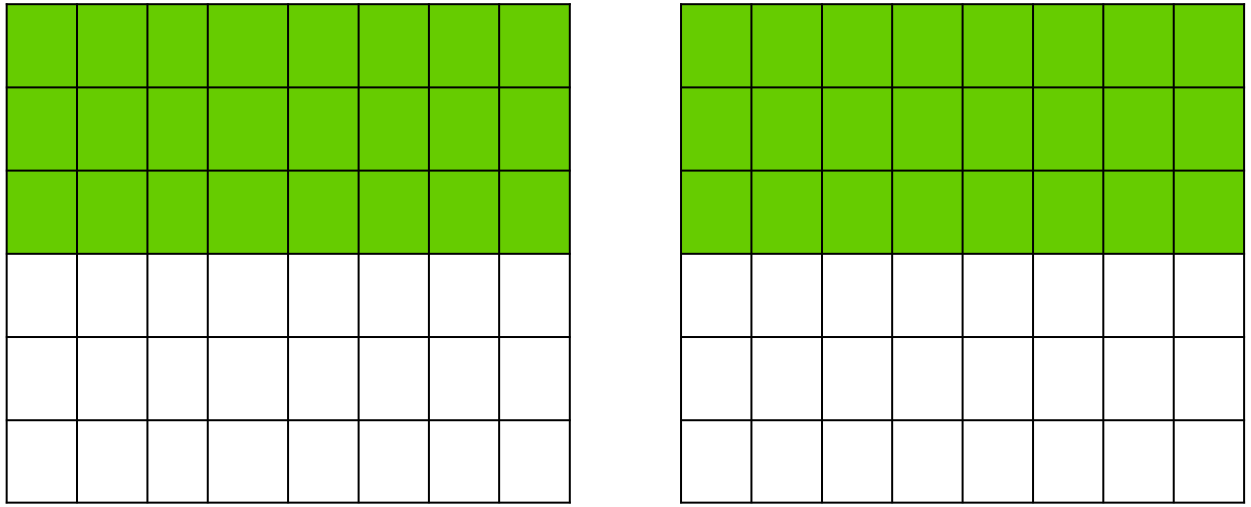


48 cores on
1 compute node

```
#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

`#SBATCH --export=NONE`
`#SBATCH --get-user-env=L`
	Initialize job environment.

`#SBATCH --time=HH:MM:SS`	Specifies the time limit for the job.
`#SBATCH --ntasks=NNN`	Total number of tasks (cores) for the job.
`#SBATCH --ntasks-per-node=XX`	Specifies the maximum number of tasks (cores) to allocate per node

`#SBATCH --mem=nnnnM`
`or`
`#SBATCH --mem=nG`
`(memory per NODE)`
	Sets the maximum amount of memory (MB).
	G for GB is supported on Grace

https://hprc.tamu.edu/kb/Helpful-Pages/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)

SUs = 1.5

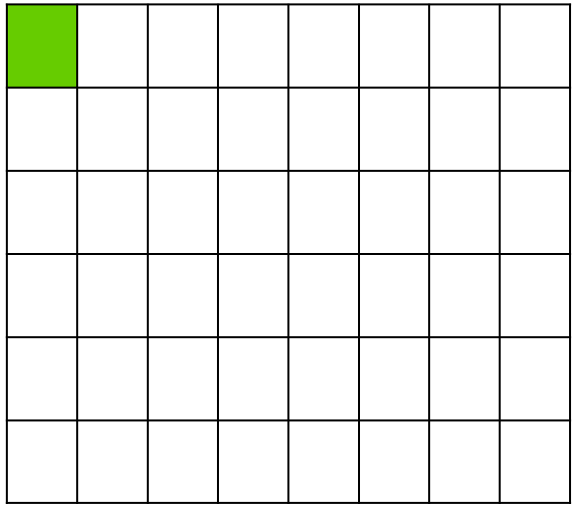
```
#!/bin/bash
#

##NECESSARY JOB SPECIFICATIONS
#SBATCH --job-name=JobExample1          # Set the job name to "JobExample1"
#SBATCH --time=01:30:00                 # Set the wall clock limit to 1hr and 30min
#SBATCH --ntasks=1                      # Request 1 task (core)
#SBATCH --mem=4G                        # Request 4GB per node
#SBATCH --output=stdout.%j              # Send stdout and stderr to "stdout.[jobID]"

##OPTIONAL JOB SPECIFICATIONS
#SBATCH --account=123456                # Set billing account to 123456
#SBATCH --mail-type=ALL                 # Send email on all job events
#SBATCH --mail-user=email_address       # 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)

SUs = 156

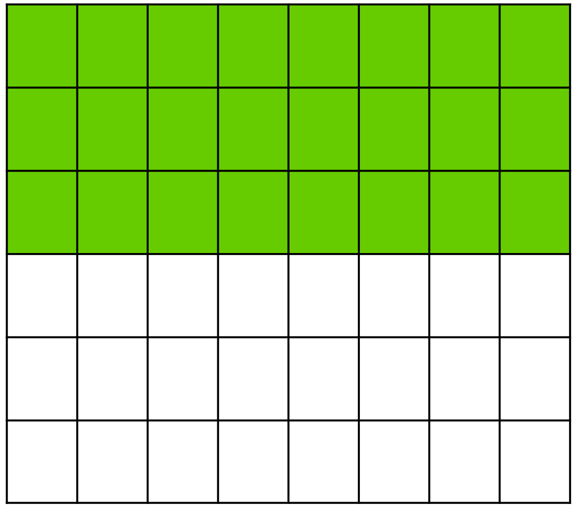
```
#!/bin/bash
#

##NECESSARY JOB SPECIFICATIONS
#SBATCH --job-name=JobExample2         # Set the job name to "JobExample2"
#SBATCH --time=6:30:00                 # Set the wall clock limit to 6hr and 30min
#SBATCH --nodes=1                      # Request 1 node
#SBATCH --ntasks-per-node=24           # Request 24 tasks(cores) per node
#SBATCH --mem=48G                      # Request 48GB per node
#SBATCH --output=stdout.%j             # Send stdout to "stdout.[jobID]"
#SBATCH --error=stderr.%j              # Send stderr to "stderr.[jobID]"

##OPTIONAL JOB SPECIFICATIONS
#SBATCH --account=123456               # Set billing account to 123456
#SBATCH --mail-type=ALL                # Send email on all job events
#SBATCH --mail-user=email_address      # 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.

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
```

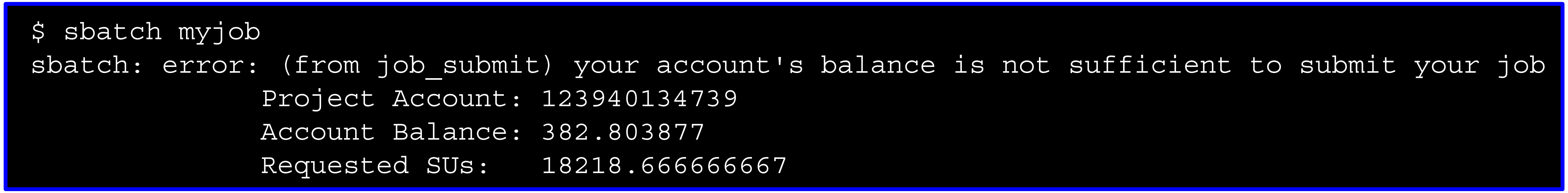
# use `cat` command to see CRLF characters

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

# use `dos2unix` command to correct

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

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)

https://hprc.tamu.edu/kb/FAQ/Accounts/
https://hprc.tamu.edu/kb/User-Guides/AMS/Service_Unit/
https://hprc.tamu.edu/kb/User-Guides/AMS/UI/

HIGH PERFORMANCE RESEARCH COMPUTING

HPRC Primer Grace

August 30, 2024

NOTICE: This course was already taught at 10:00 AM this morning. You may view a past recording of this primer at link in the chat. We apologize for the inconvenience.

High Performance
Research Computing
DIVISION OF RESEARCH