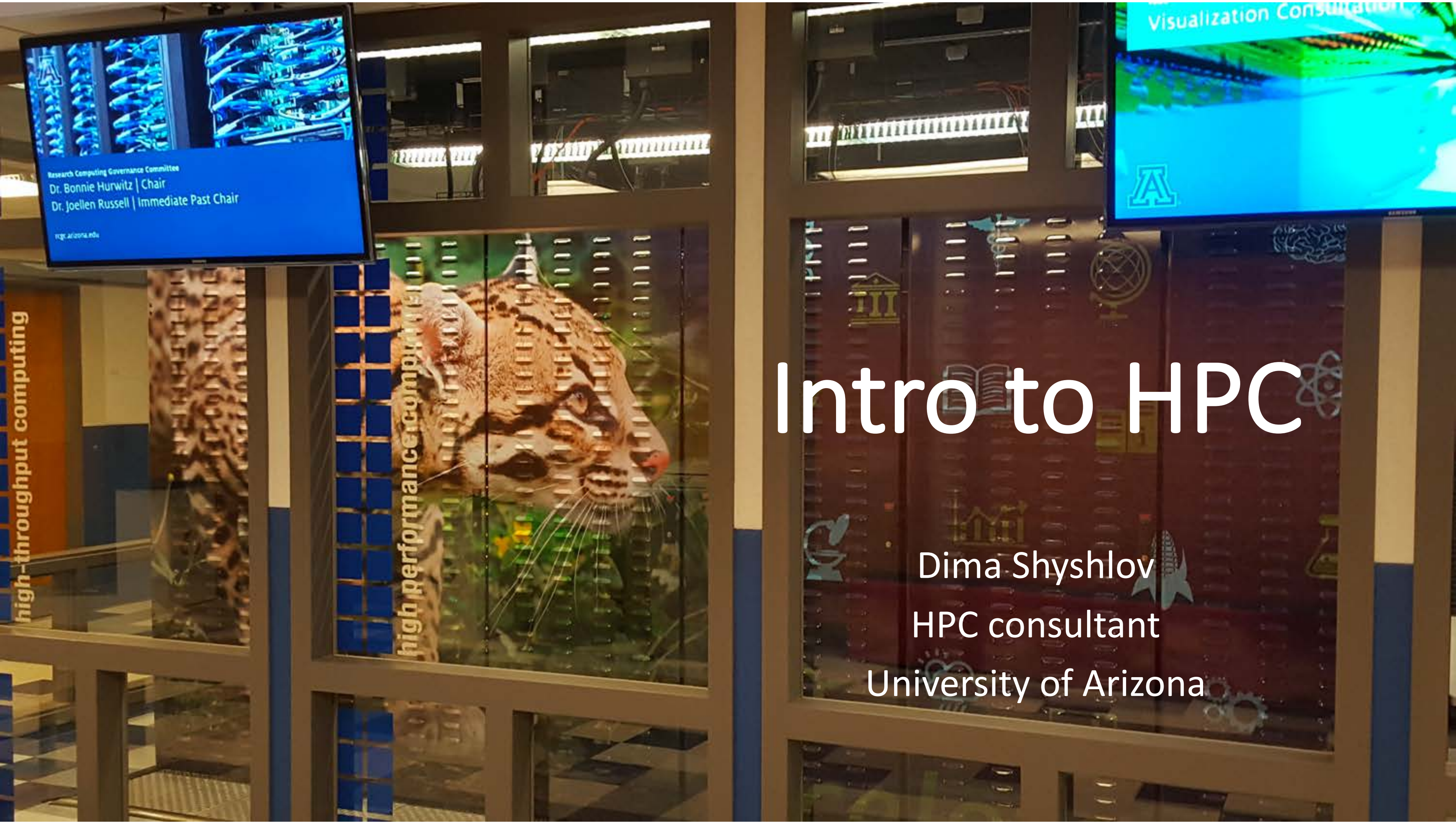

Research Computing Governance Committee
Dr. Bonnie Hurwitz | Chair
Dr. Joellen Russell | Immediate Past Chair
rcgc.arizona.edu
Visualization Consultation
high-throughput computing
high performance computing

Intro to HPC

Dima Shyshlov

HPC consultant

University of Arizona

Intro to HPC*

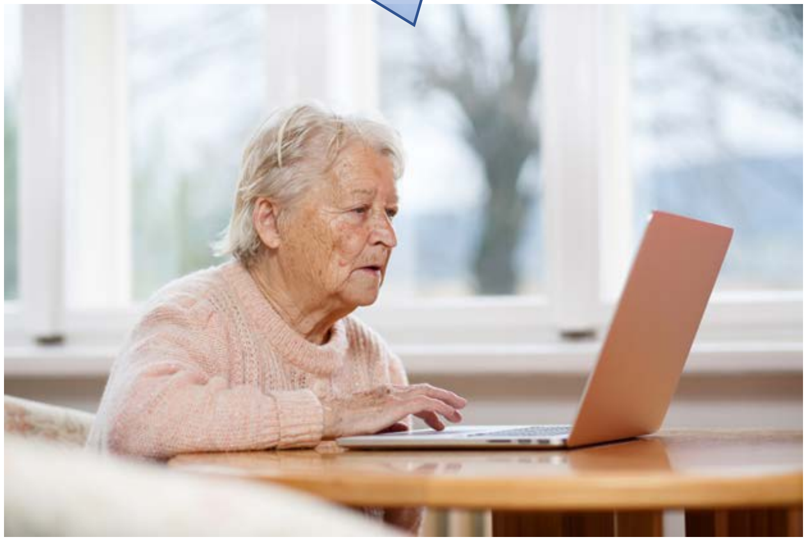
*HPC – high-performance computing

- Why use HPC?
- Anatomy of the HPC cluster
- Basics of working with the Linux shell
- Submitting jobs
- Being a good HPC user

Why HPC?

Research is easy!

It’s still running…



Why HPC?

Problem

- Computation takes too long

- Computation is too big

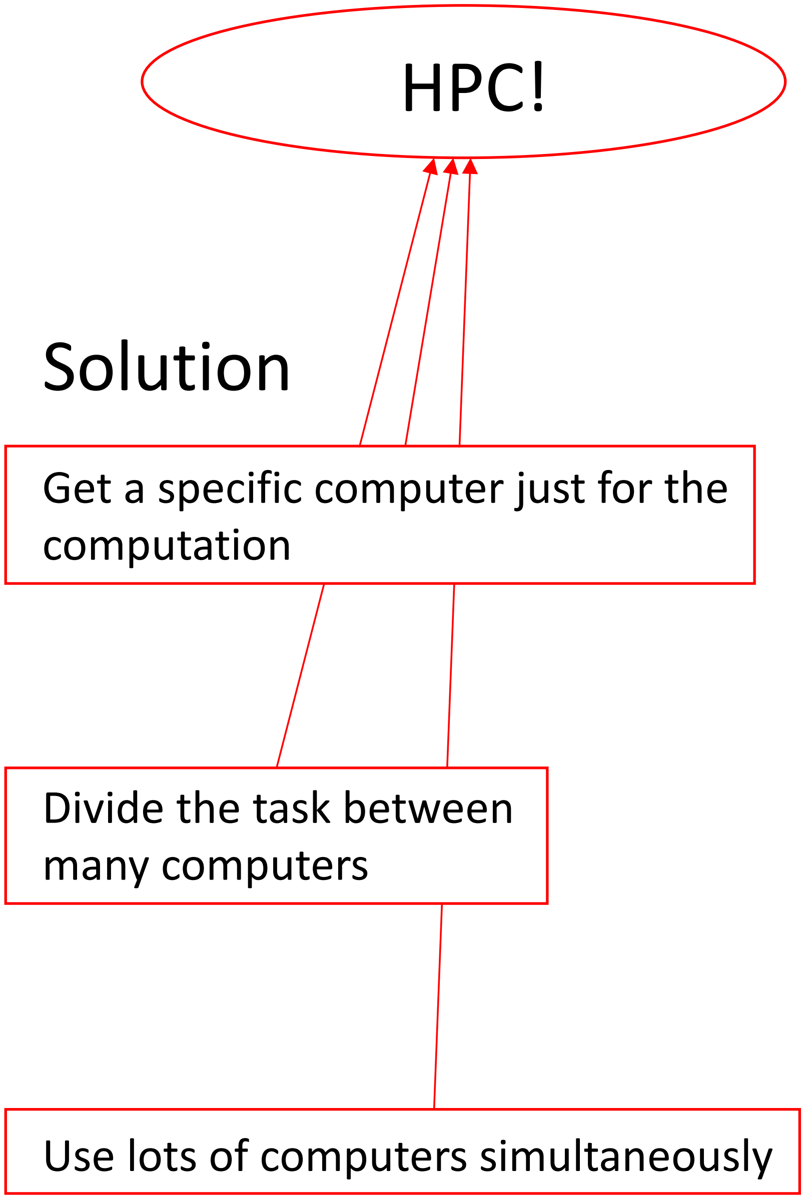
- Too many computations

Why HPC?

Problem

- Computation takes too long
- Computation is too big
- Too many calculations

Solution



HPC!

Get a specific computer just for the computation

Divide the task between many computers

Use lots of computers simultaneously

Why HPC?

Modern instrument for High-Performance Computing is a **cluster**, consisting of lots of connected individual computers (nodes).
Supercomputer is a commonly used nickname.

Why HPC?

Laptop

Supercomputer

Why HPC?

Laptop

Supercomputer

Why HPC?

Laptop

Personal

Supercomputer

Shared

Why HPC?

Laptop

Local

Supercomputer

Remote

Why HPC?

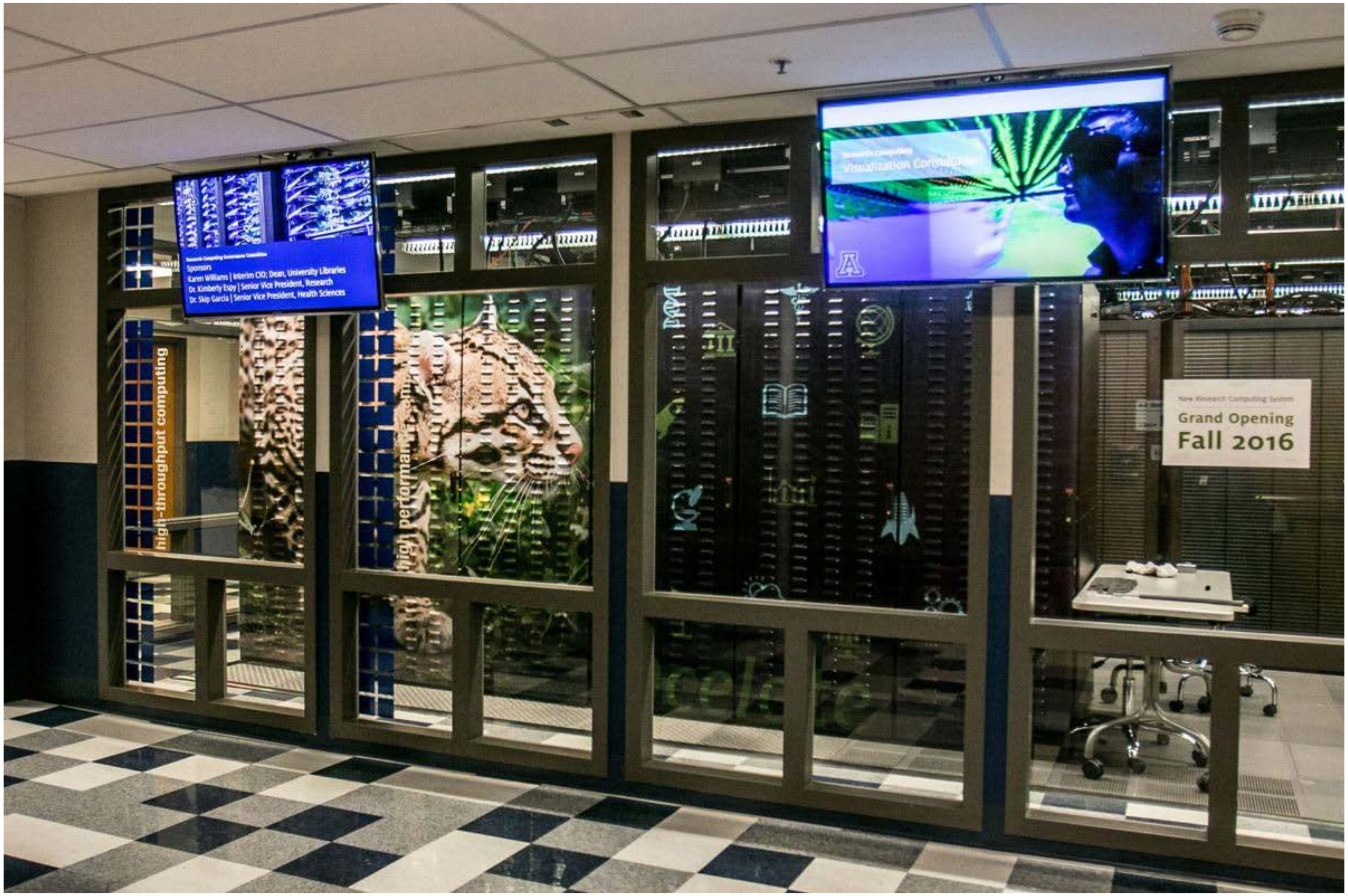
Laptop

Interactive

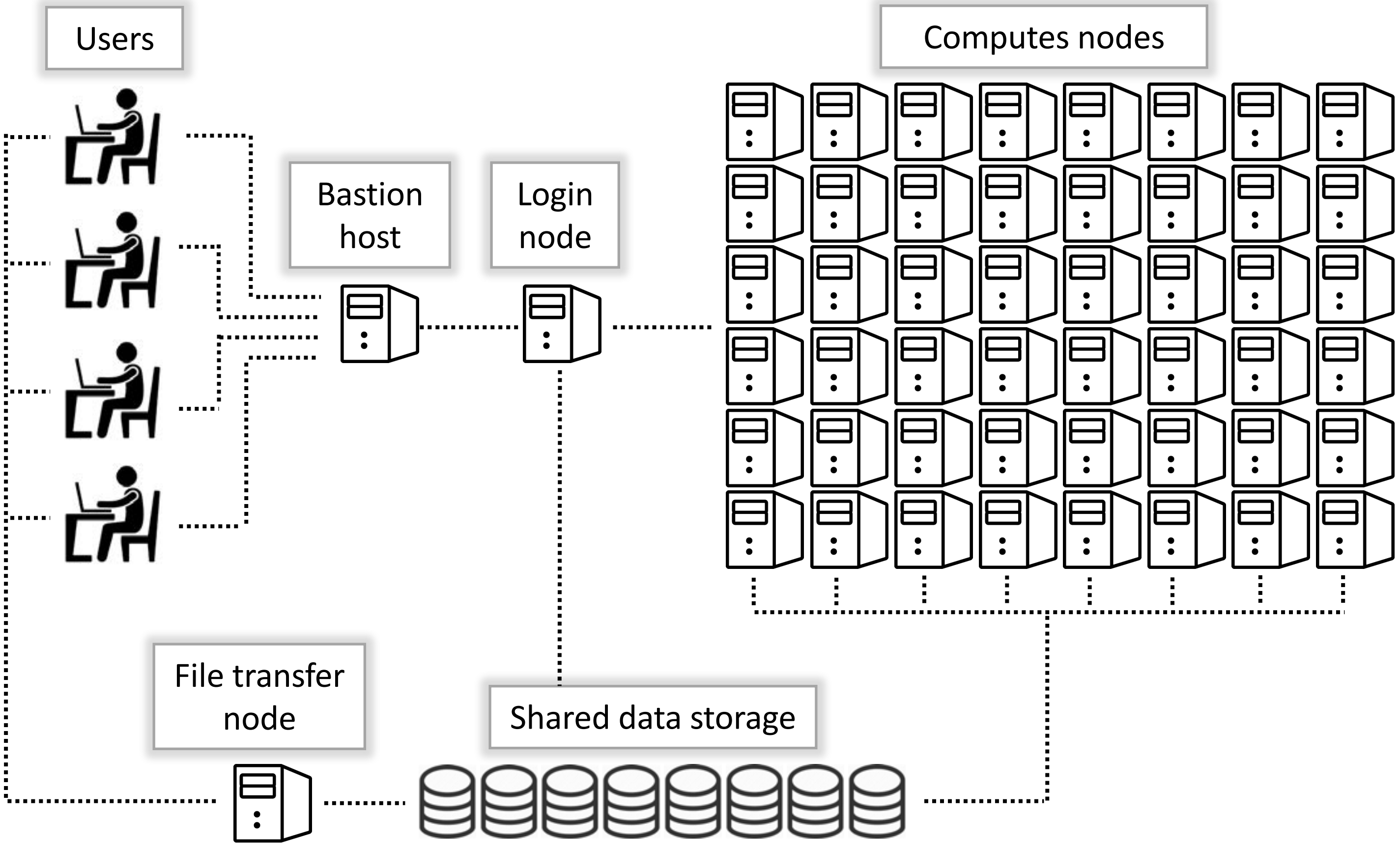
Supercomputer

Batch

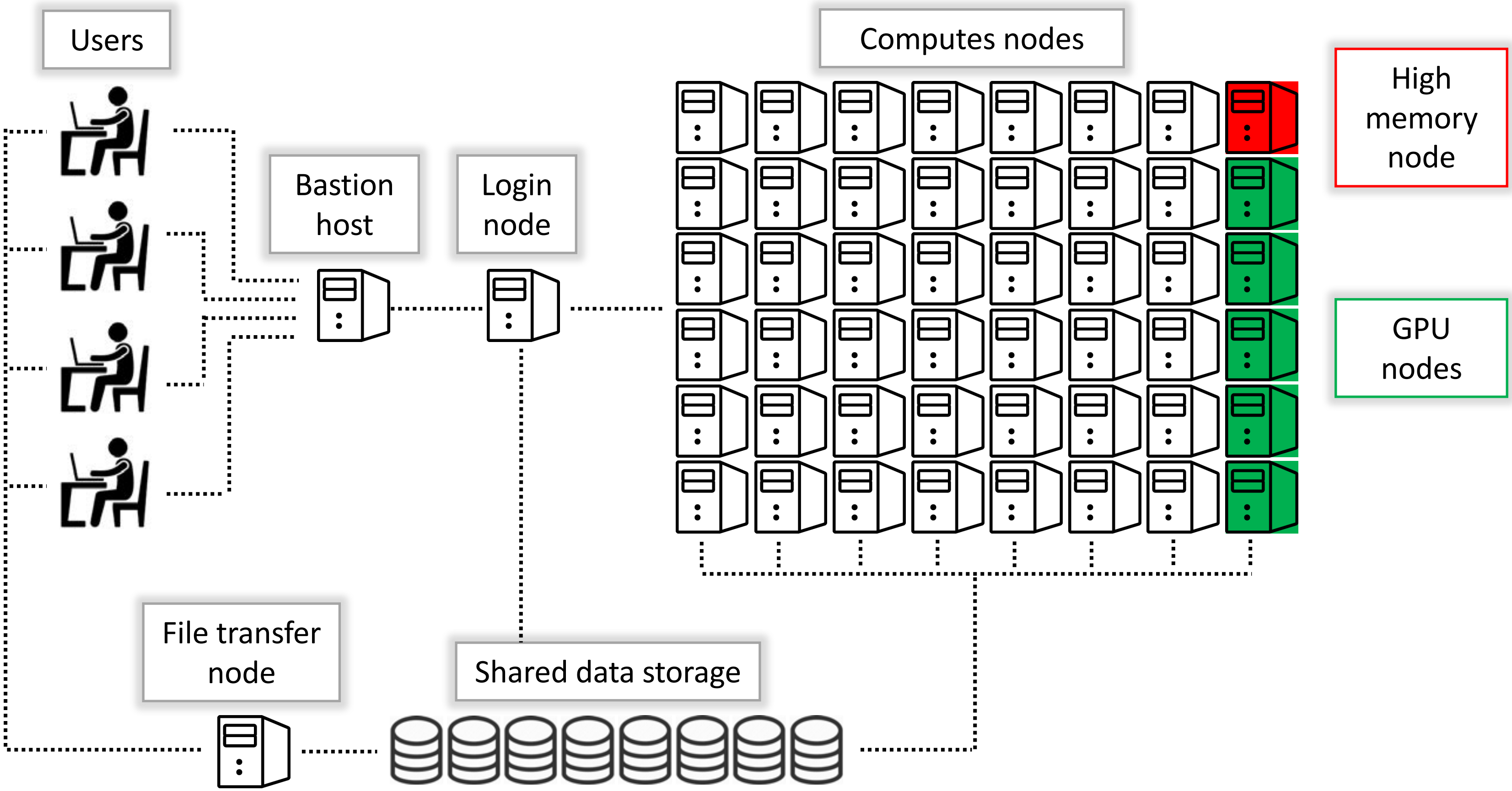
Ocelote


Research Computing Governance Committee
Sponsors
Karen Williams | Interim CIO; Dean, University Libraries
Dr. Kimberly Espy | Senior Vice President, Research
Dr. Skip Garcia | Senior Vice President, Health Sciences
Research Computing
Visualization Consultation
high-throughput computing
high performance
ocelote
New Research Computing System
Grand Opening
Fall 2016

The diagram of the UA HPC cluster

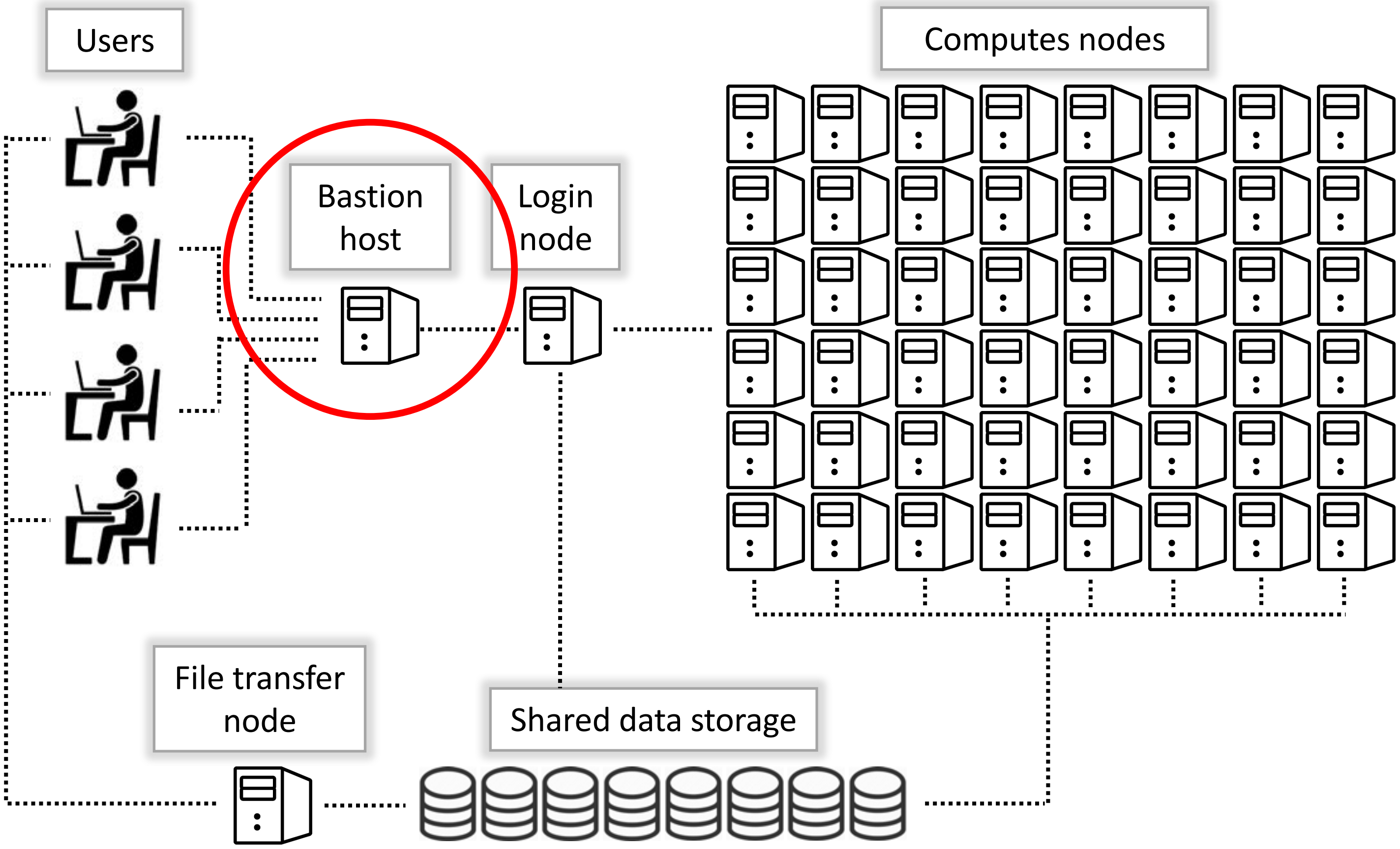


The diagram of the UA HPC cluster



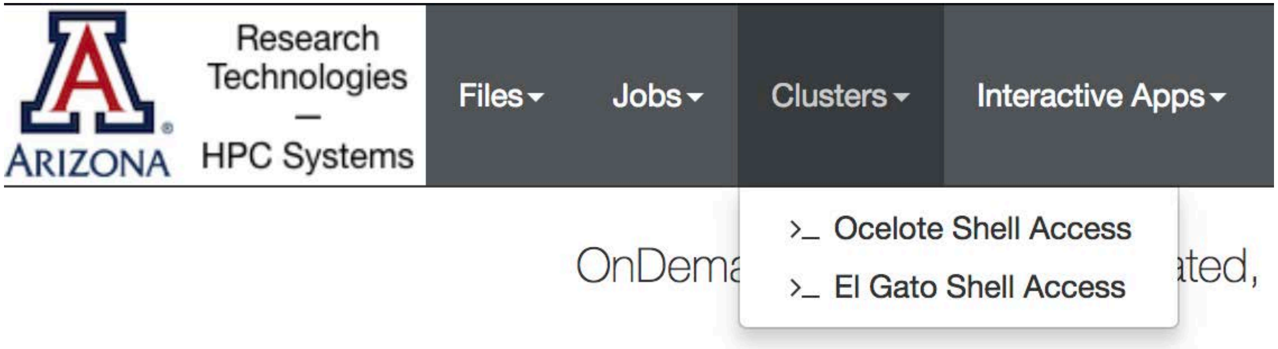
Connecting to Ocelote

```
ssh hpc.arizona.edu
```



Connecting to Ocelote (web browser)

- Open **ood.hpc.arizona.edu** in your web browser and login with your NetID and password.
- From the “Clusters” drop-down menu choose which HPC cluster you would like to access:

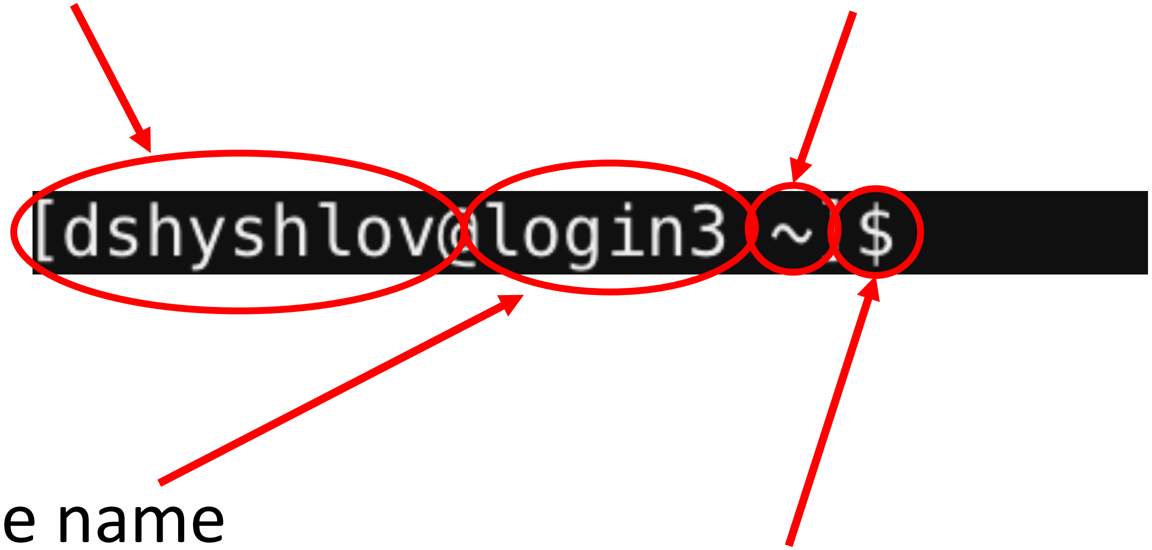


- Exercise - connect to Ocelote.

Command line

Your NetID
(who are you)

Name of the current
directory

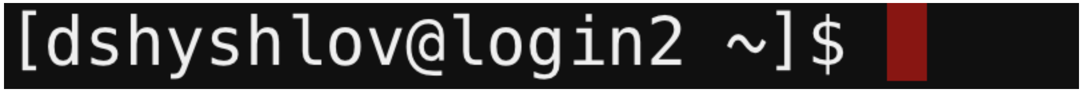


```
[dshyshlov@login3 ~]$
```

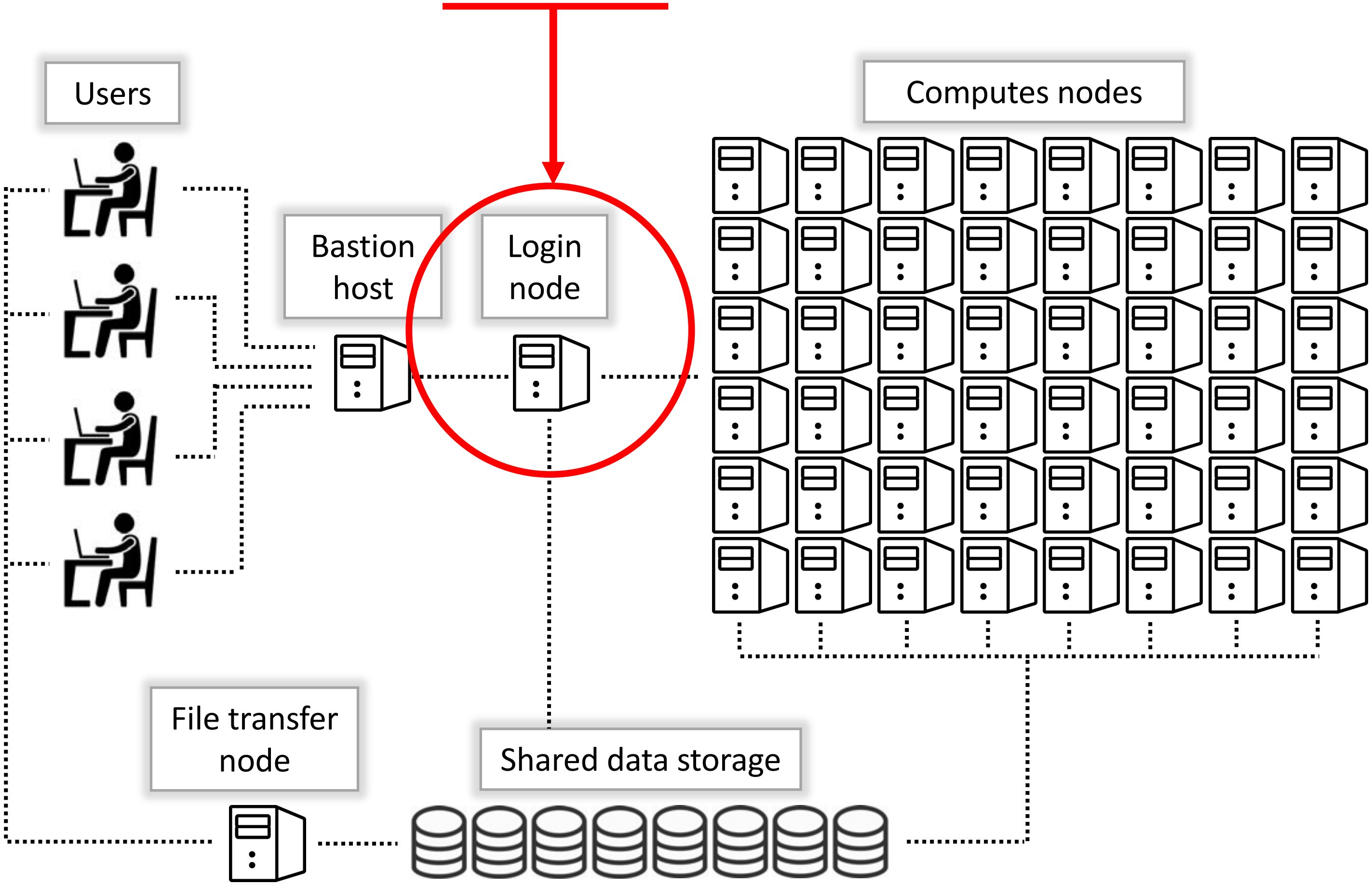
Node name
(where are you)

Prompt
(what are you going to do)

Login node



```
[dshyshlov@login2 ~]$
```



Login node

Login node

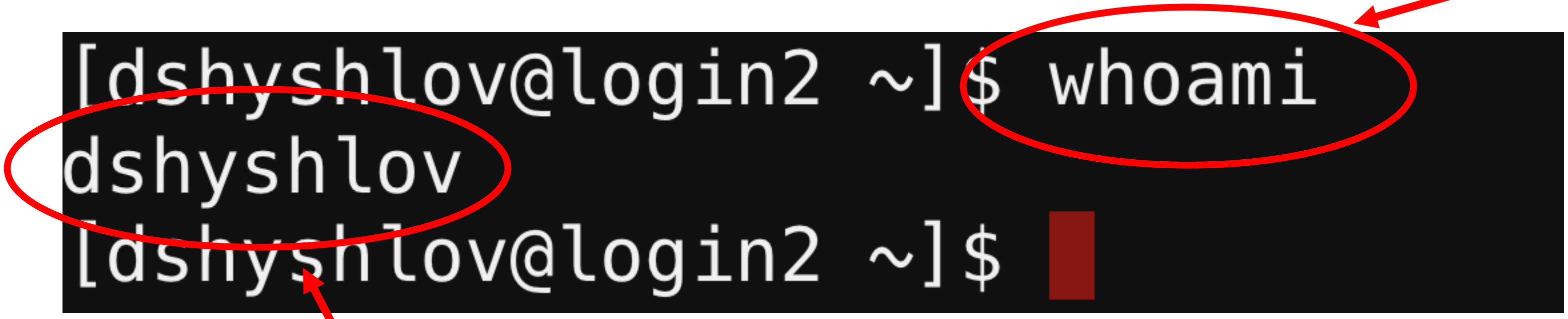
Compute nodes

Login node

- Login node is a computer intended for users to prepare and manage computations:
  - submit jobs
  - edit files
  - compile codes
  - manage files
  - small-scale testing

- **DO NOT** run any calculations on the login node

Working with a Linux shell

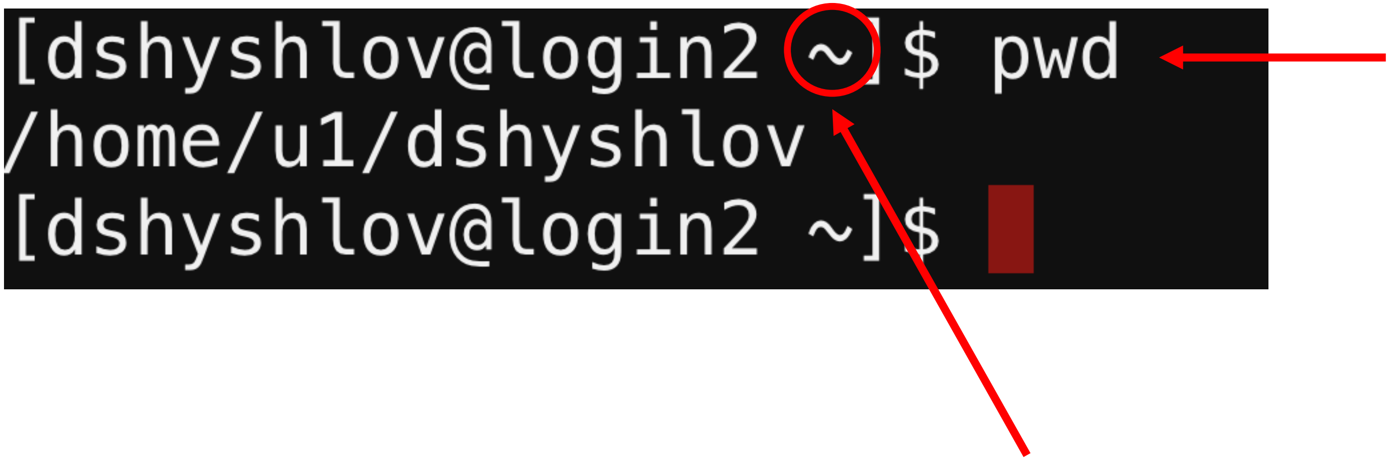


```
[dshyshlov@login2 ~]$ whoami
dshyshlov
[dshyshlov@login2 ~]$ 
```

Command

Output

Working with a Linux shell



```
[dshyshlov@login2 ~]$ pwd
/home/u1/dshyshlov
[dshyshlov@login2 ~]$ 
```

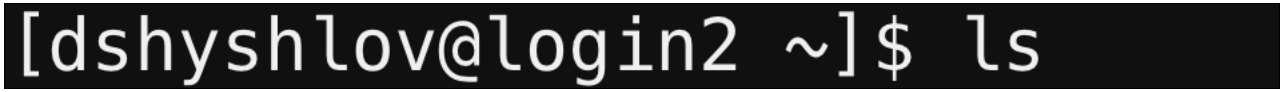
Path to Working Directory

Name of the current directory

- ~ is a shortcut for your /home directory

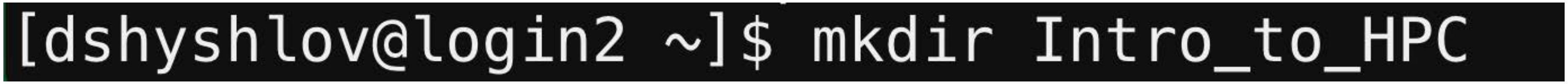
Working with a Linux shell

- List all the files and directories



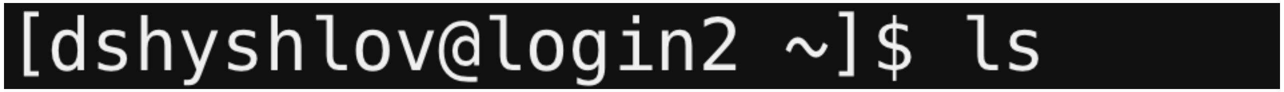
```
[dshyshlov@login2 ~]$ ls
```

- Make a directory



```
[dshyshlov@login2 ~]$ mkdir Intro_to_HPC
```

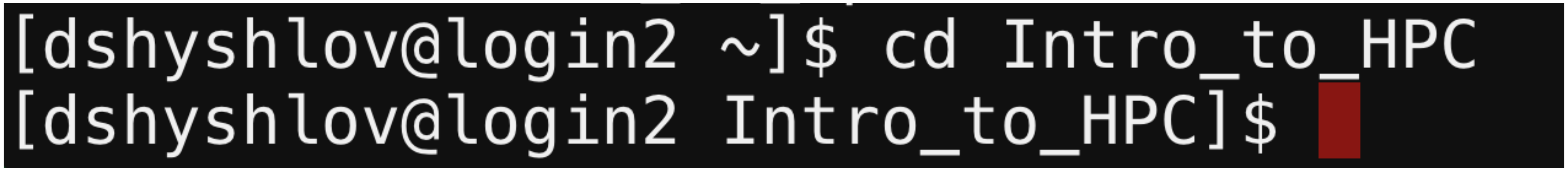
- List all the files and directories again



```
[dshyshlov@login2 ~]$ ls
```

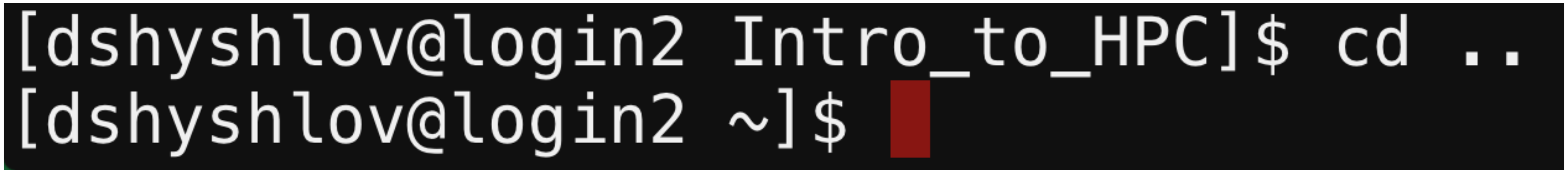
Working with a Linux shell

- Change directory



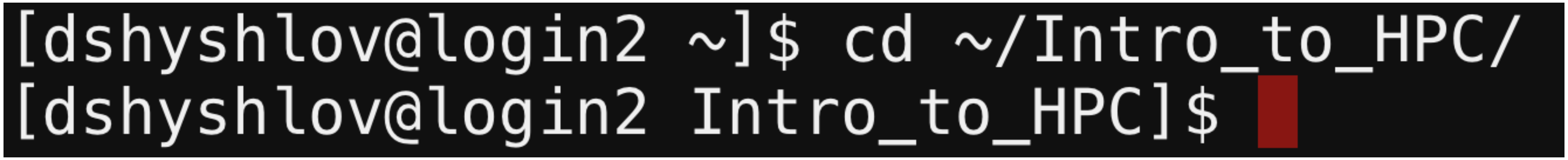
```
[dshyshlov@login2 ~]$ cd Intro_to_HPC
[dshyshlov@login2 Intro_to_HPC]$ 
```

- Go back a level



```
[dshyshlov@login2 Intro_to_HPC]$ cd ..
[dshyshlov@login2 ~]$ 
```

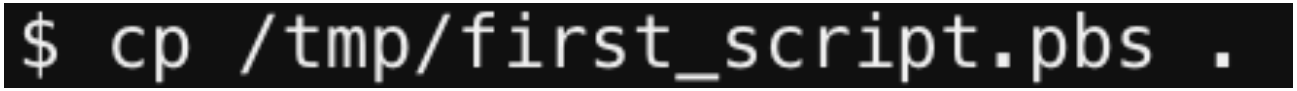
- Change directory using absolute path



```
[dshyshlov@login2 ~]$ cd ~/Intro_to_HPC/
[dshyshlov@login2 Intro_to_HPC]$ 
```

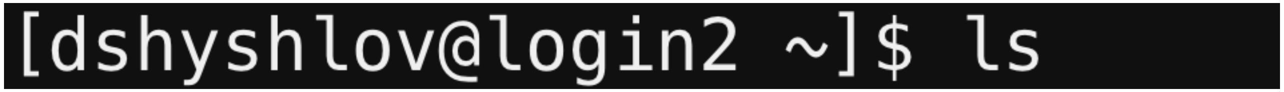
Working with a Linux shell

- Copy a file



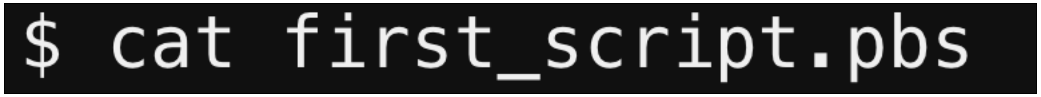
```
$ cp /tmp/first_script.pbs .
```

- List all the files and directories again



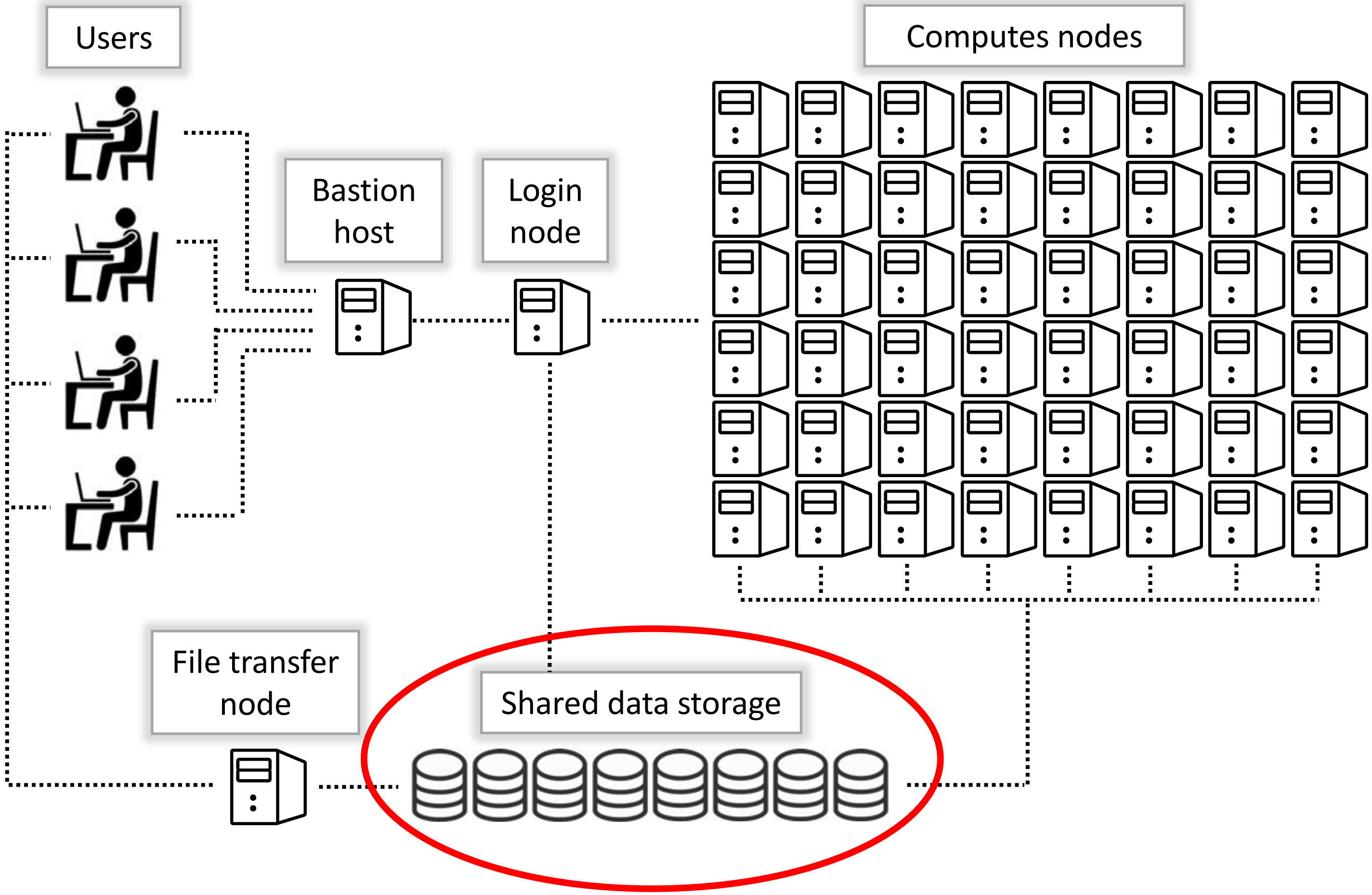
```
[dshyshlov@login2 ~]$ ls
```

- View contents of the file on the screen



```
$ cat first_script.pbs
```

Storage



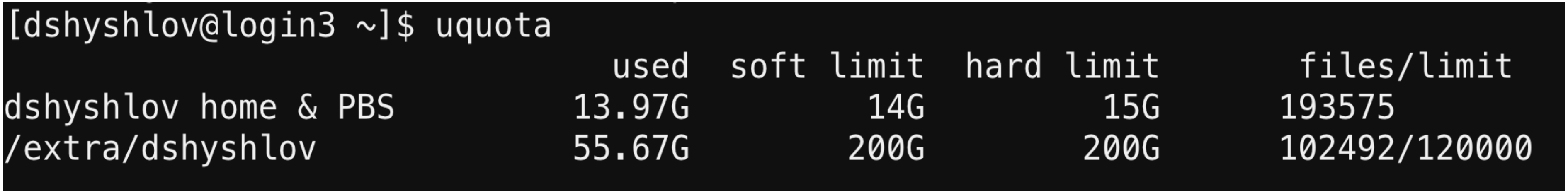
/home /extra

Storage

- Every users gets two default storage locations :
  - /home
    - the default home directory
    - 15GB
    - the only backed up storage on UA HPC/extra
  - /extra
    - full path: /extra/NetID
    - 200GB
    - not backed up
    - has file count limit 600 files/GB

Storage

- Command to list all the available storage options – uquota

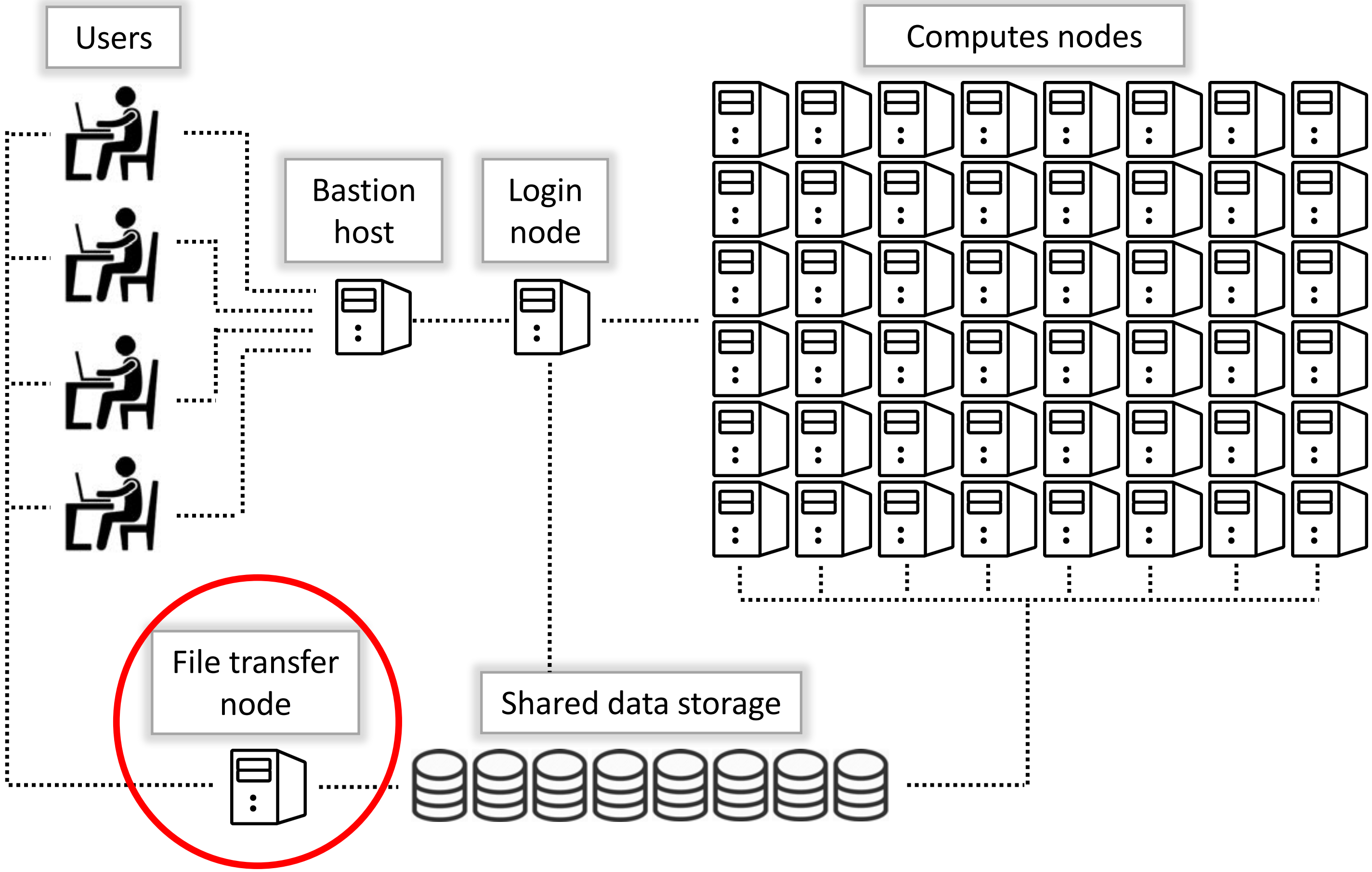


```
[dshyshlov@login3 ~]$ uquota
                               used  soft limit  hard limit        files/limit
dshyshlov home & PBS         13.97G         14G         15G       193575
/extra/dshyshlov             55.67G        200G        200G       102492/120000
```

Storage

	Storage	Back-up	File limits	Speed
/home	15 GB	Nightly	None	
/extra	200 GB	None	600 files / GB	
/xdisk	200 – 1000 GB (45 day limit)	None	None	
/rsgrps	Rented space	None	600 files / GB	
/tmp	Varies ~ 800 GB (Ocelote)	None	None	Fastest (on node)

Transferring Files



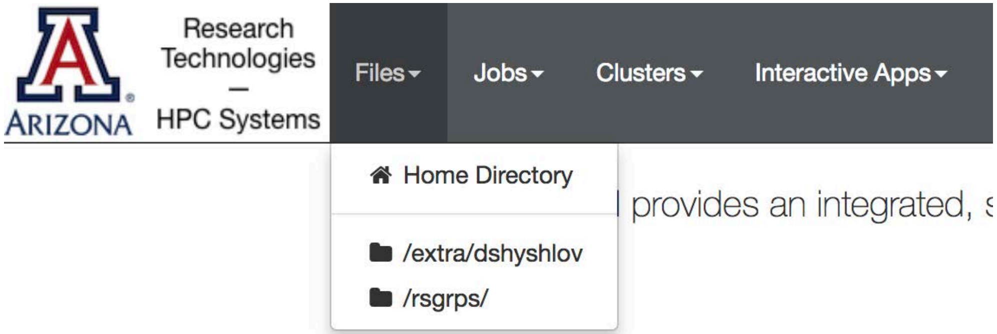
filexfer.hpc.arizona.edu

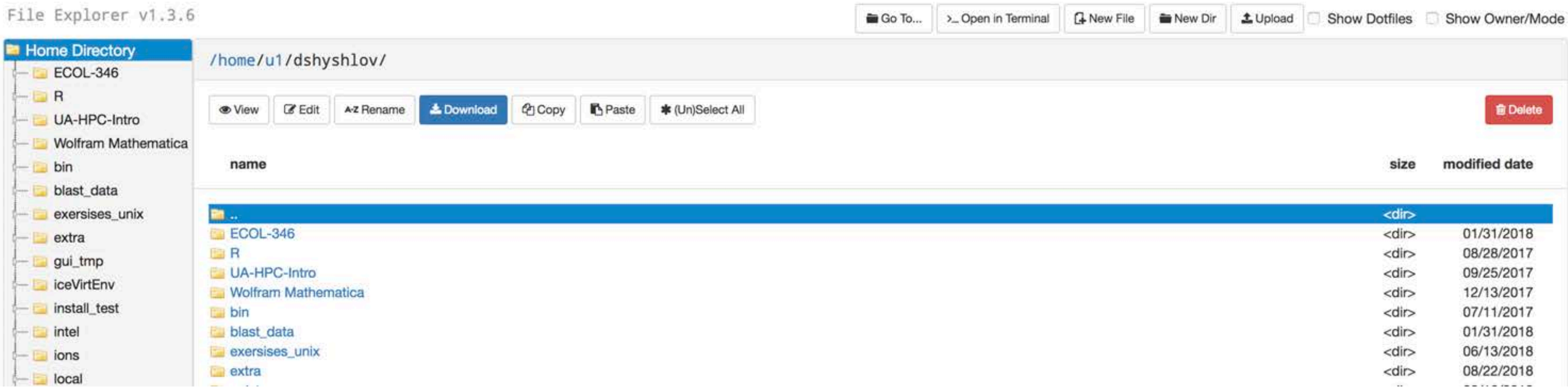
Transferring Files

- Ocelote has two specific nodes for file transfer
  - hostname – filexfer.hpc.arizona.edu
- Command line options:
  - scp, sftp, rsync, irods
- GUI options
  - Windows based: WinSCP
  - Cross-platform: Cyberduck
- Parallel data transfer
  - Globus
  - best option for large files

Transferring Files (web browser)

- Display and manage your files
- Edit text files
- Drag and drop files to/from the file explorer

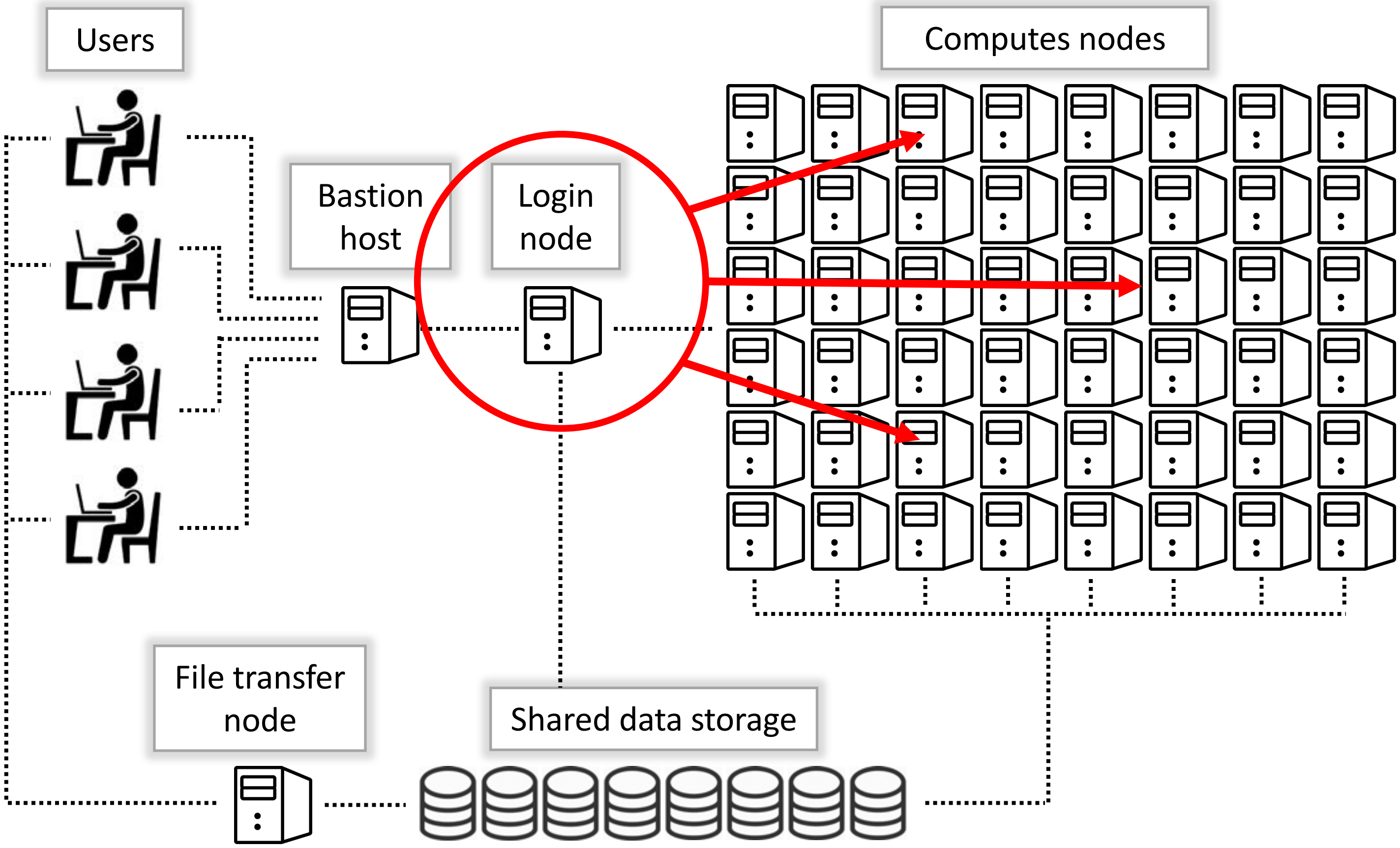




/home/u1/dshyshlov/

name	size	modified date
..	<dir>	
ECOL-346	<dir>	01/31/2018
R	<dir>	08/28/2017
UA-HPC-Intro	<dir>	09/25/2017
Wolfram Mathematica	<dir>	12/13/2017
bin	<dir>	07/11/2017
blast_data	<dir>	01/31/2018
exersises_unix	<dir>	06/13/2018
extra	<dir>	08/22/2018

From the login node to compute



From the login node to compute

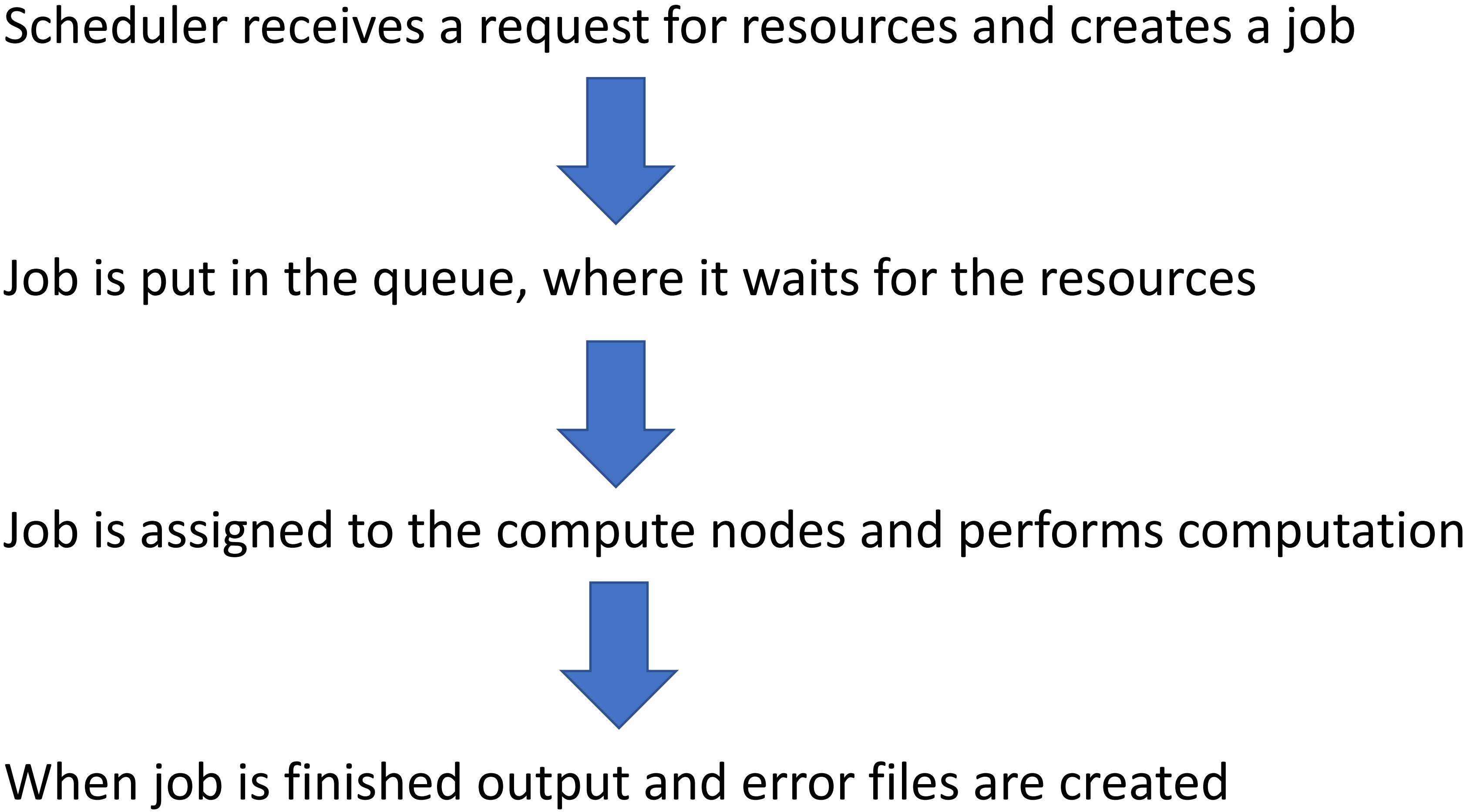
- How do we know if there are any available nodes?
- How do we decide who gets what and when?
- How do we ensure that a task gets the resources it needs?

- Scheduler!
- Software that manages the HPC resources and decides which computation runs where and when.

Scheduler

- Ocelote uses the scheduler PBS Pro.
- Every computation that requests resources from the scheduler is called a *job*.
- *Submitting a job* means requesting resources from the scheduler and giving it a list of commands to run.

Scheduler



Scheduler receives a request for resources and creates a job

Job is put in the queue, where it waits for the resources

Job is assigned to the compute nodes and performs computation

When job is finished output and error files are created

Queues

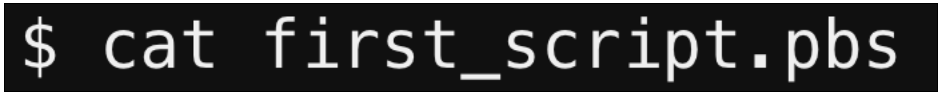
- Standard queue
  - 36,000 CPU-hours/month per group
  - higher priority

- Windfall queue
  - No time limit
  - Preemption

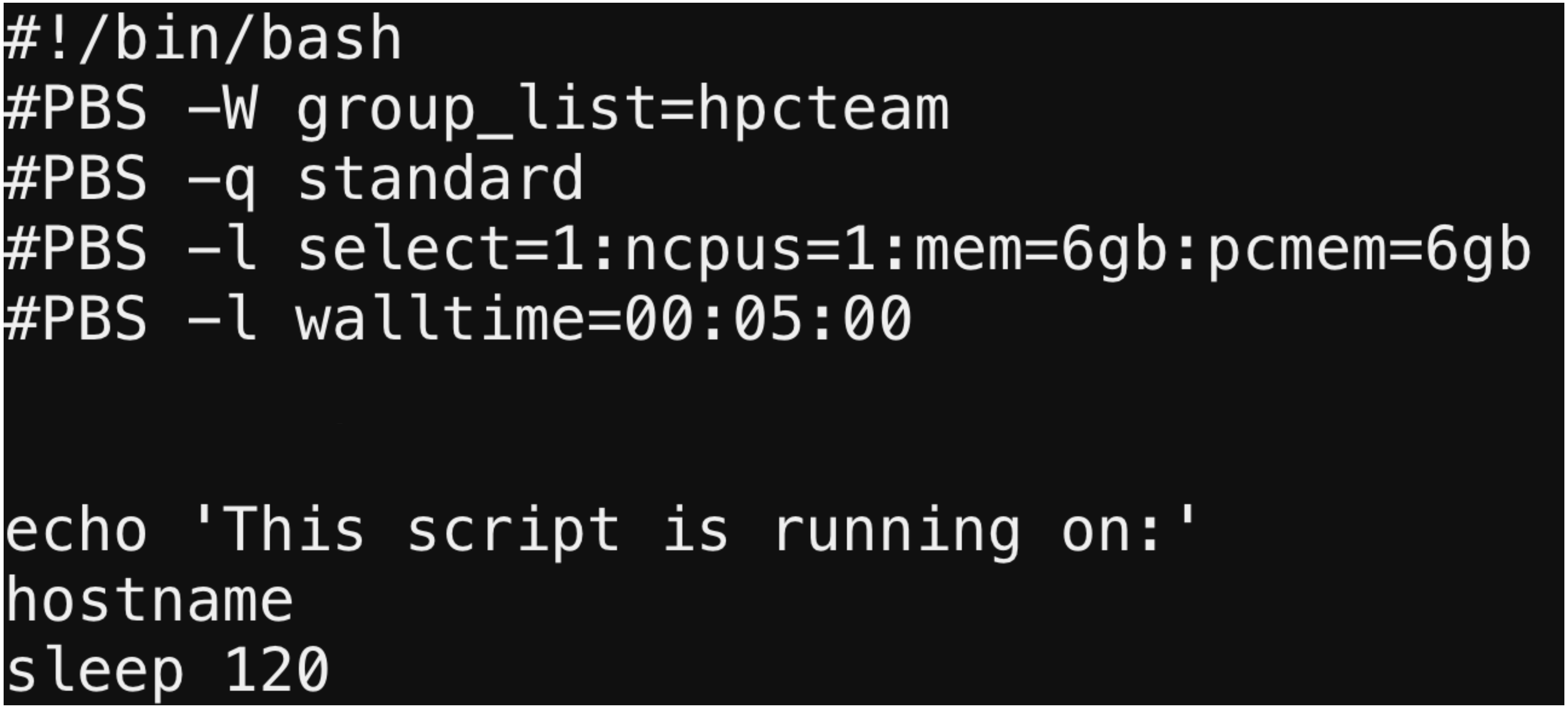
- Debug queue
  - 10 min limit

PBS script

- View the contents of a file first_script.pbs with a command



```
$ cat first_script.pbs
```

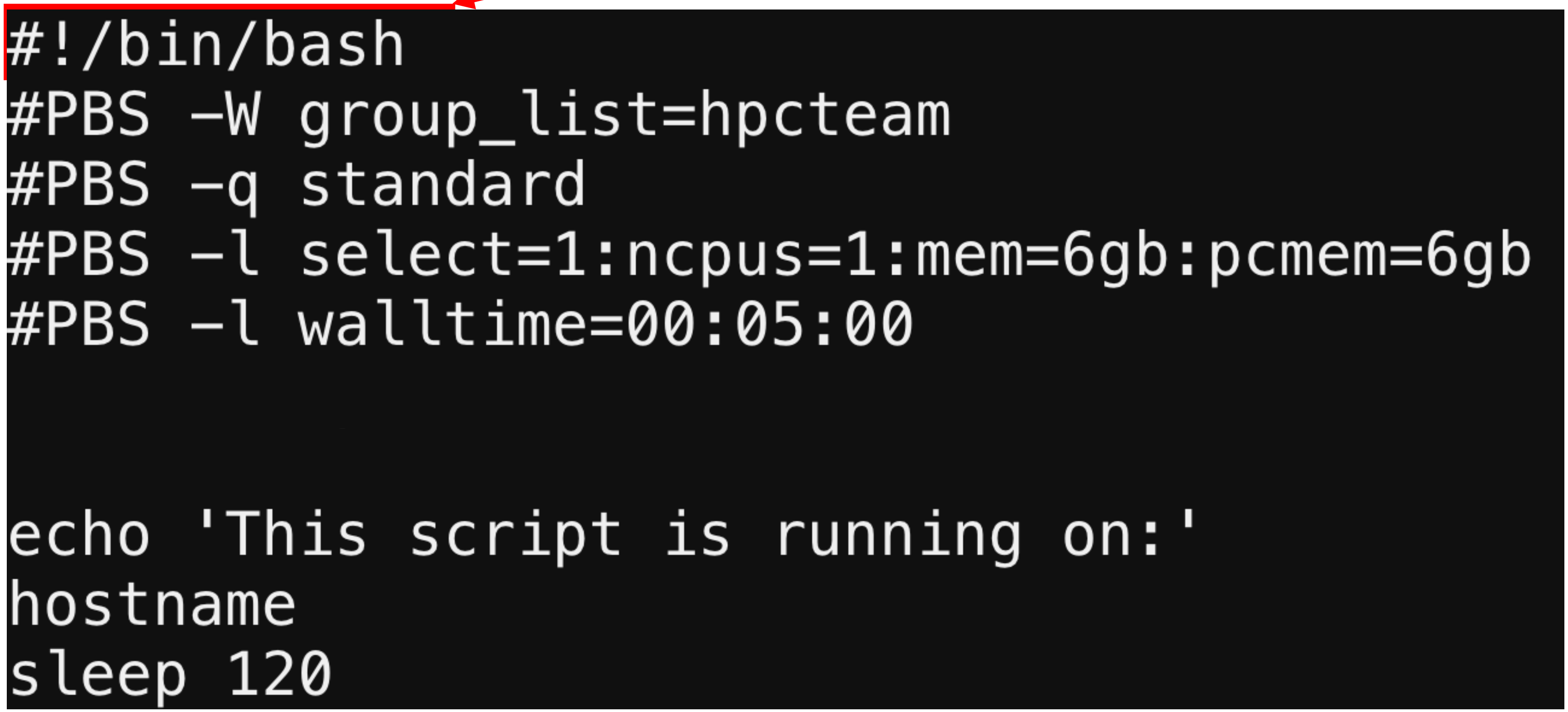


```
#!/bin/bash
#PBS -W group_list=hpcteam
#PBS -q standard
#PBS -l select=1:ncpus=1:mem=6gb:pcmem=6gb
#PBS -l walltime=00:05:00


echo 'This script is running on:'
hostname
sleep 120
```

PBS script

It’s a bash shell script

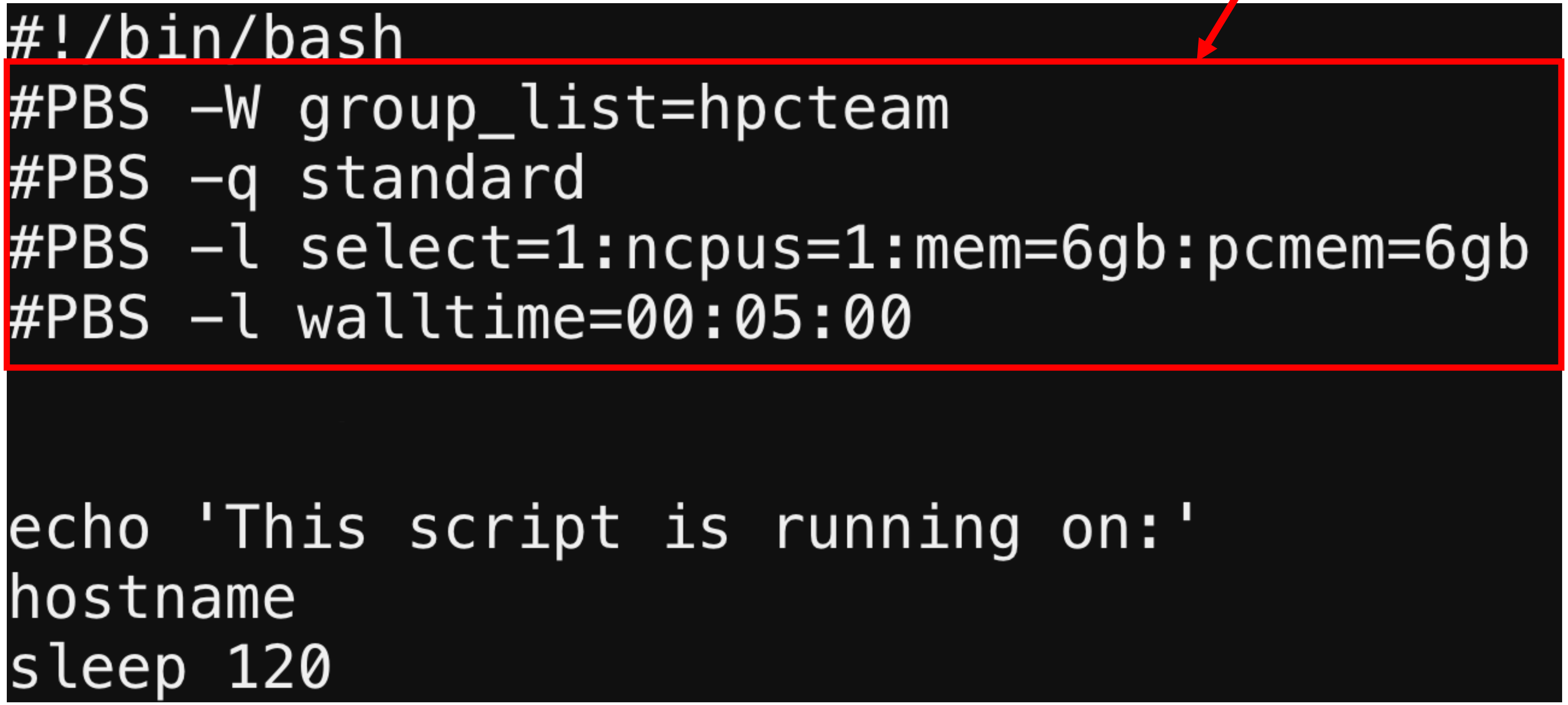


```
#!/bin/bash
#PBS -W group_list=hpcteam
#PBS -q standard
#PBS -l select=1:ncpus=1:mem=6gb:pcmem=6gb
#PBS -l walltime=00:05:00


echo 'This script is running on:'
hostname
sleep 120
```

PBS script

PBS directives

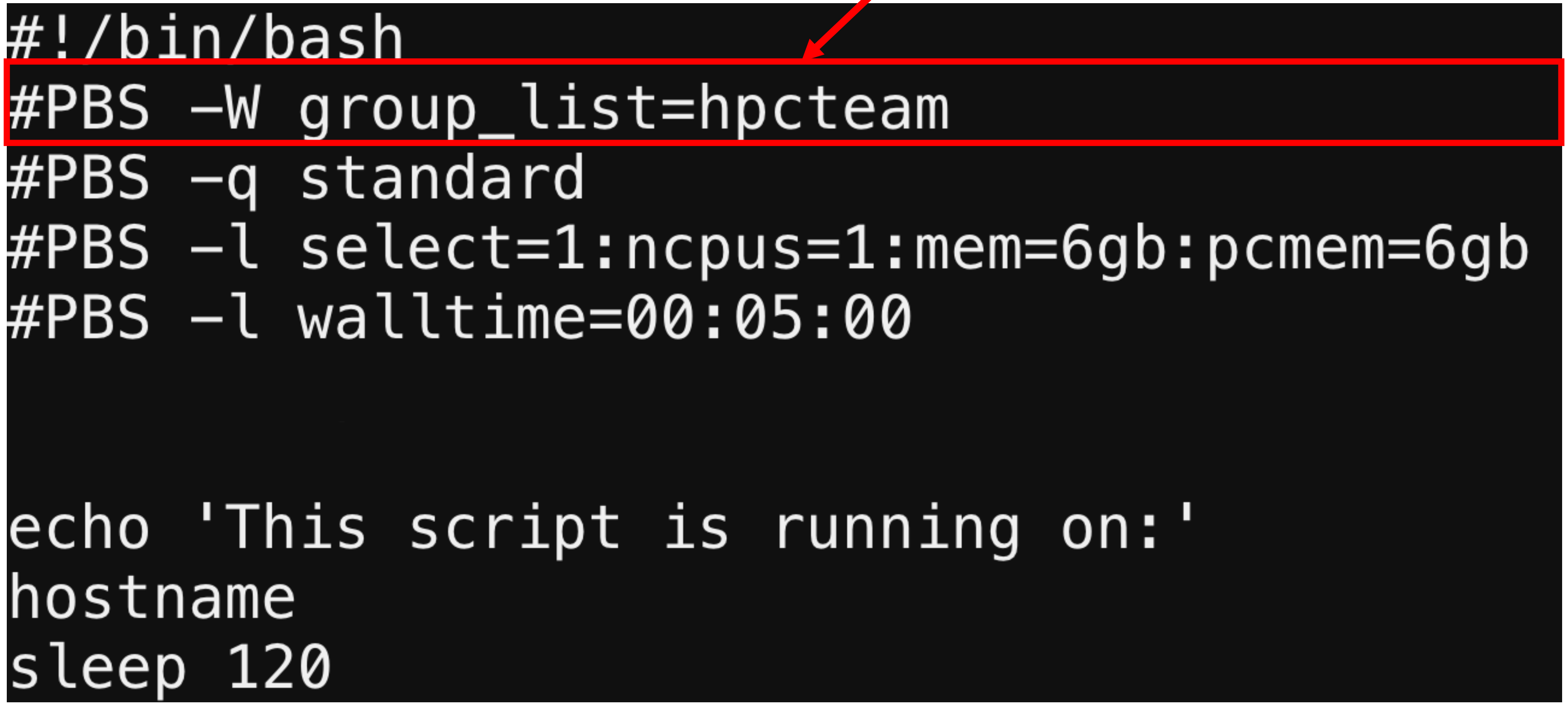


```
#!/bin/bash
#PBS -W group_list=hpcteam
#PBS -q standard
#PBS -l select=1:ncpus=1:mem=6gb:pcmem=6gb
#PBS -l walltime=00:05:00


echo 'This script is running on:'
hostname
sleep 120
```

PBS script

Name of your HPC group (usually it is your sponsor’s NetID). You can find your groups name with a `va` command.

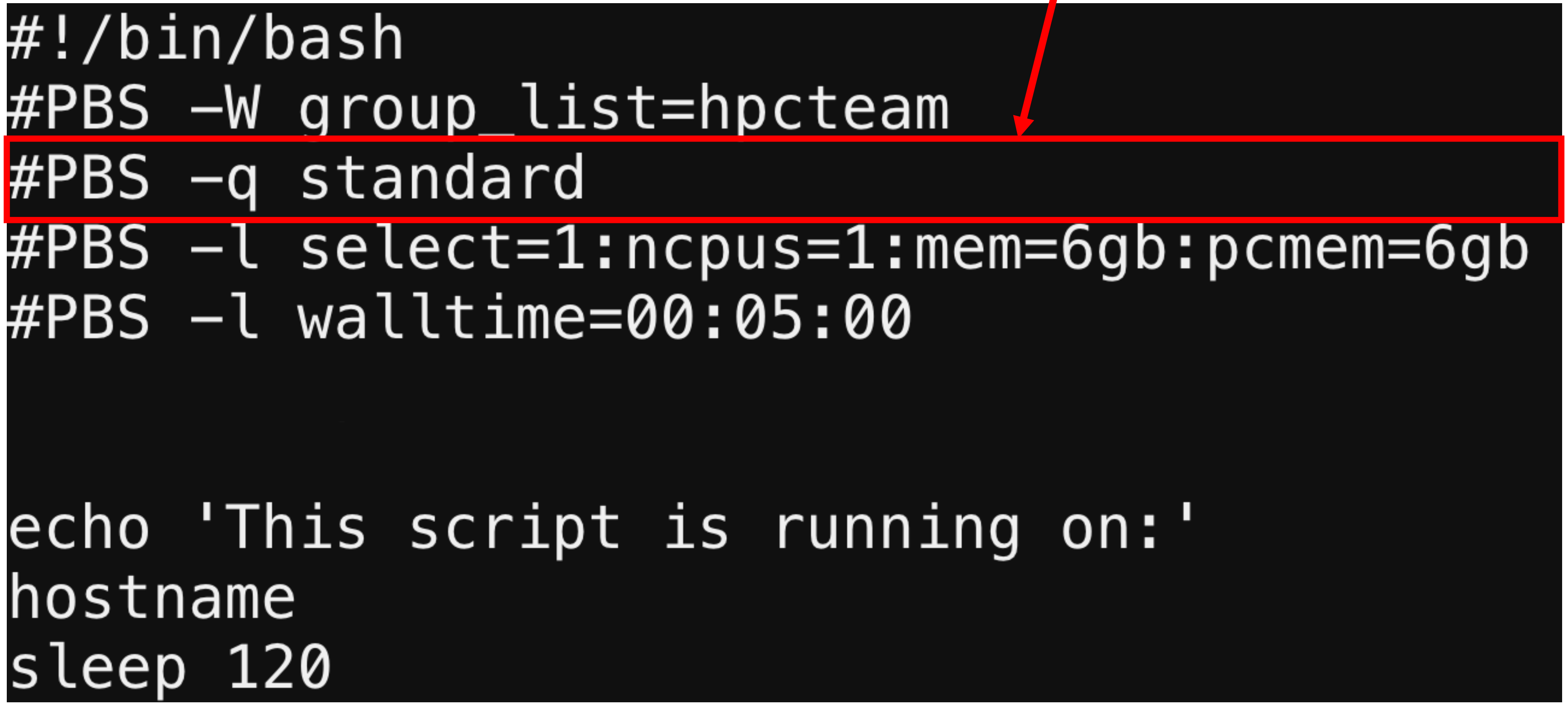


```
#!/bin/bash
#PBS -W group_list=hpcteam
#PBS -q standard
#PBS -l select=1:ncpus=1:mem=6gb:pcmem=6gb
#PBS -l walltime=00:05:00


echo 'This script is running on:'
hostname
sleep 120
```

PBS script

Type of queue

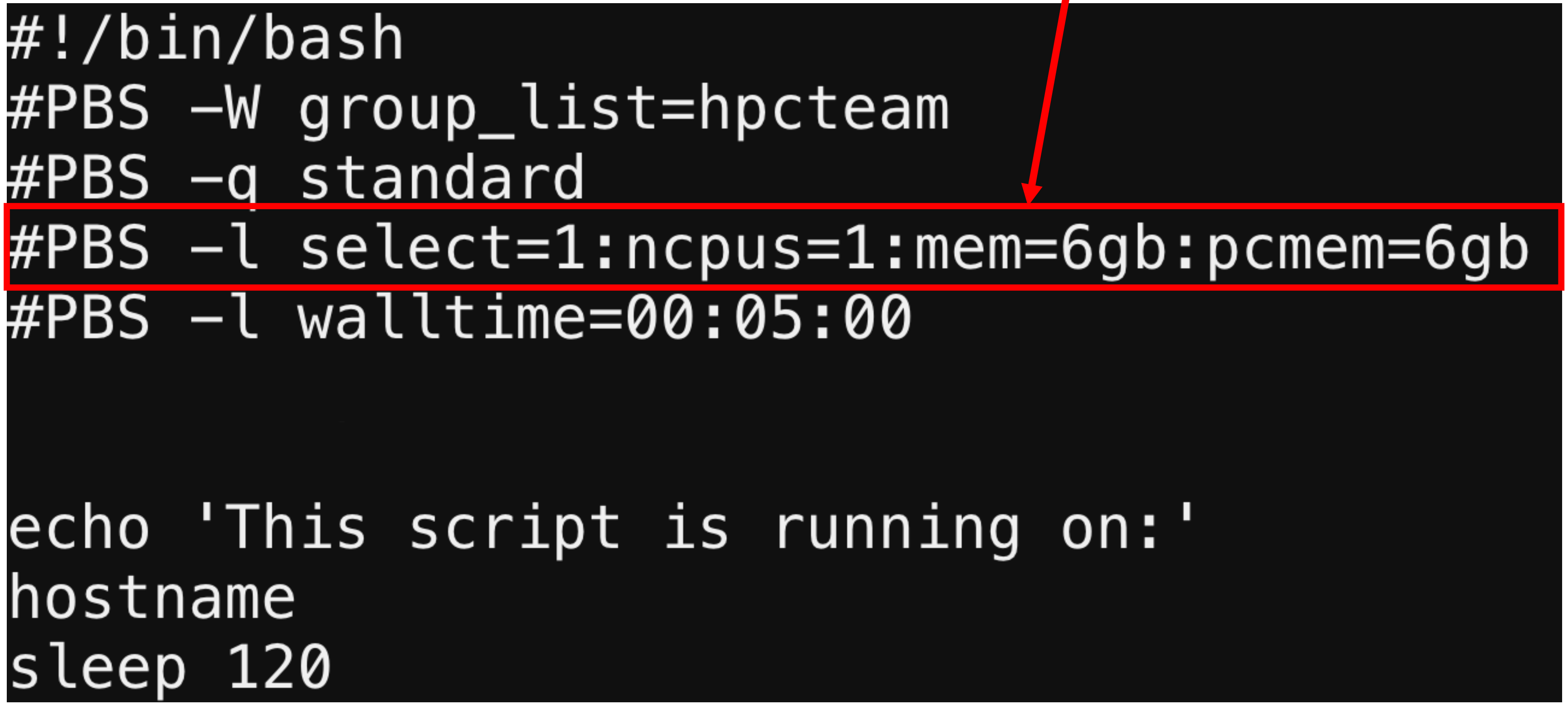


```
#!/bin/bash
#PBS -W group_list=hpcteam
#PBS -q standard
#PBS -l select=1:ncpus=1:mem=6gb:pcmem=6gb
#PBS -l walltime=00:05:00


echo 'This script is running on:'
hostname
sleep 120
```

PBS script

“Select statement”.
Create a custom computer from Ocelote’s resources for your job.

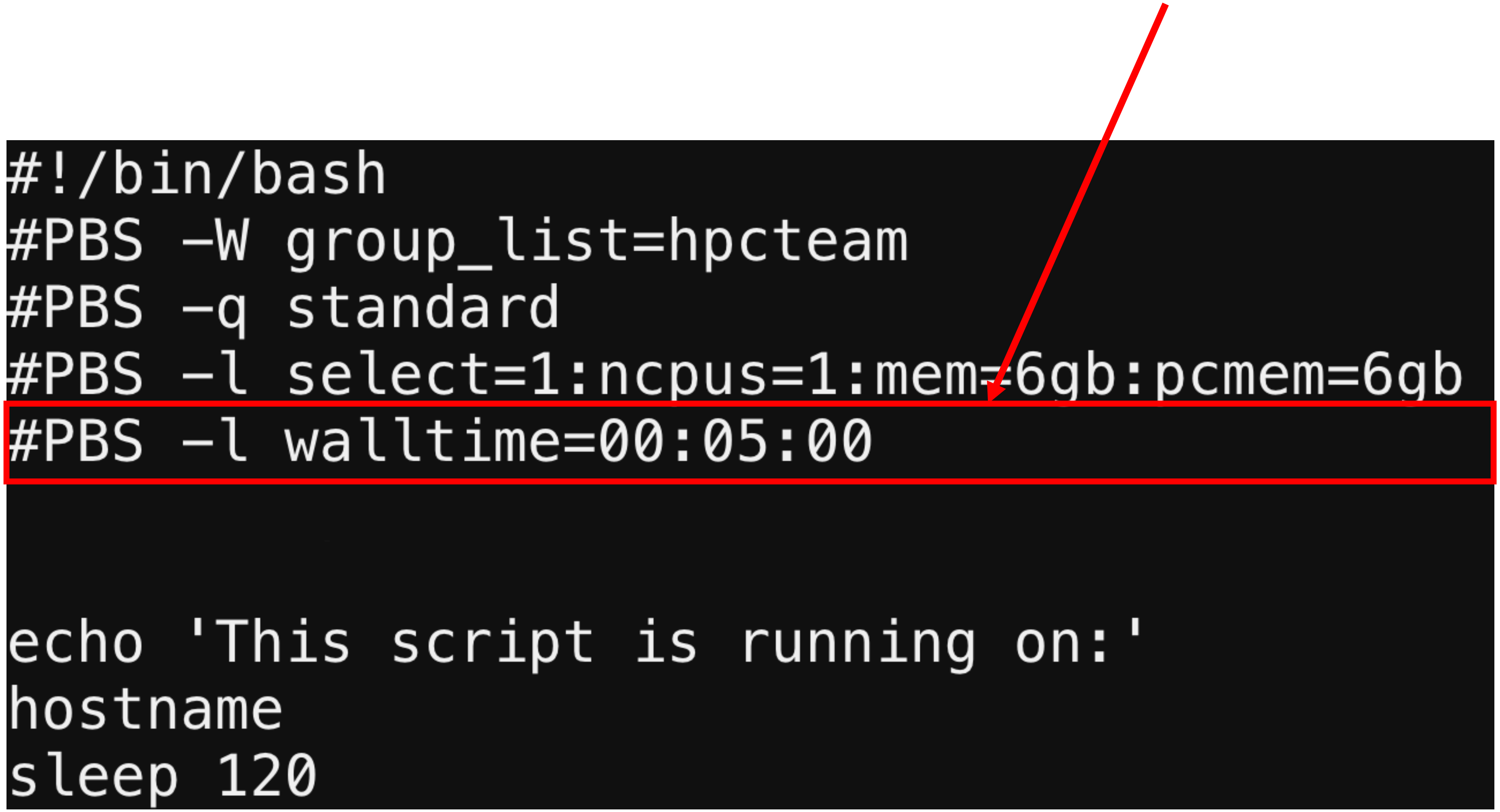


```
#!/bin/bash
#PBS -W group_list=hpcteam
#PBS -q standard
#PBS -l select=1:ncpus=1:mem=6gb:pcmem=6gb
#PBS -l walltime=00:05:00


echo 'This script is running on:'
hostname
sleep 120
```

PBS script

How long your custom computer will exist.

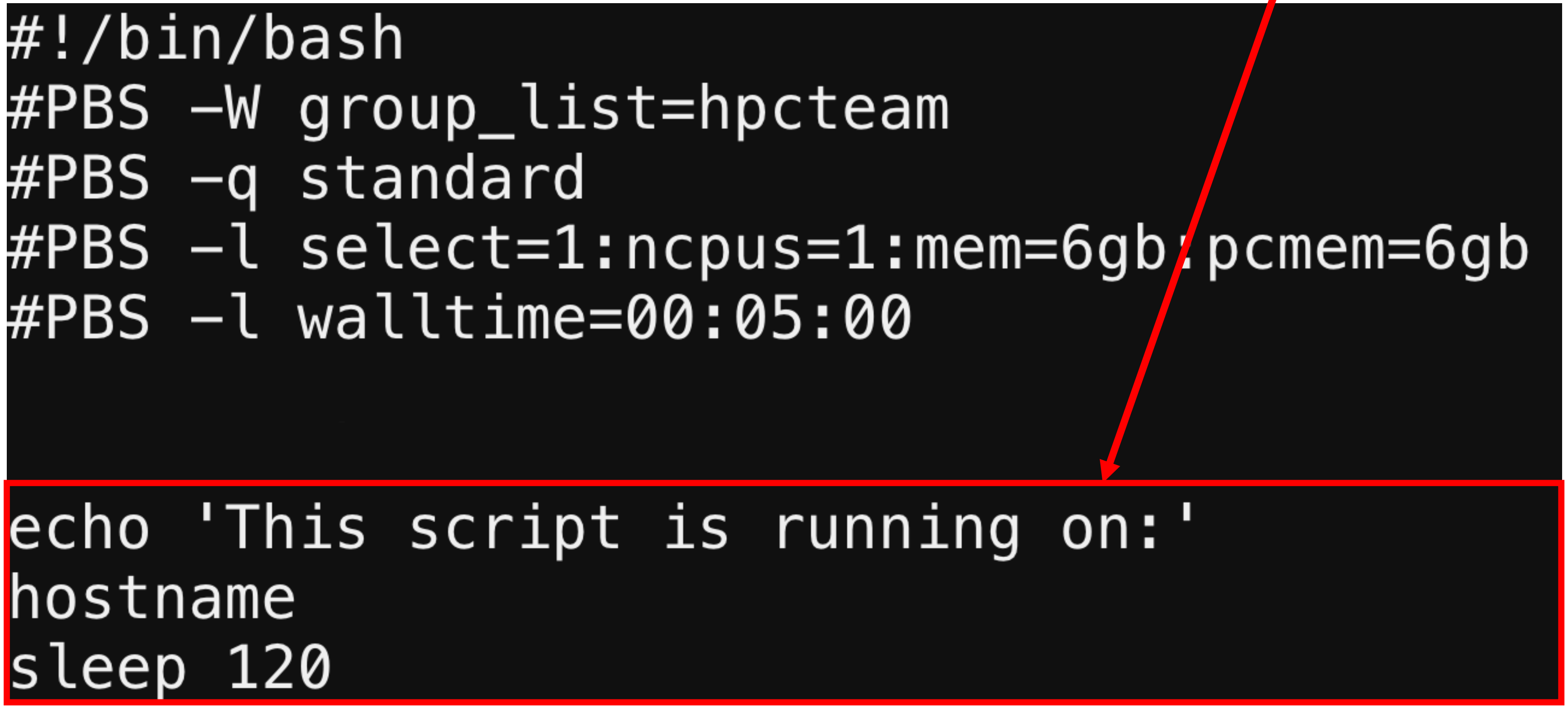


```
#!/bin/bash
#PBS -W group_list=hpcteam
#PBS -q standard
#PBS -l select=1:ncpus=1:mem=6gb:pcmem=6gb
#PBS -l walltime=00:05:00


echo 'This script is running on:'
hostname
sleep 120
```

PBS script

List of commands to run on your custom computer

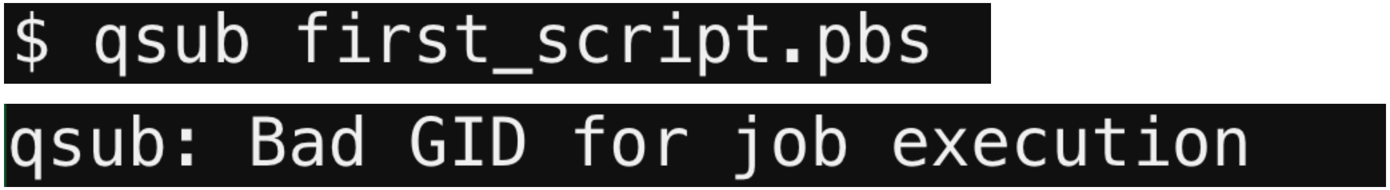


```
#!/bin/bash
#PBS -W group_list=hpcteam
#PBS -q standard
#PBS -l select=1:ncpus=1:mem=6gb:pcmem=6gb
#PBS -l walltime=00:05:00


echo 'This script is running on:'
hostname
sleep 120
```

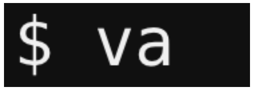
Exercise – running a batch job

- Submit your first job on Ocelote.



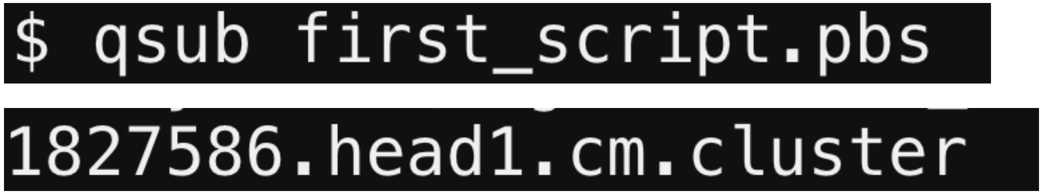
```
$ qsub first_script.pbs
qsub: Bad GID for job execution
```

- Edit first_script.pbs to correct the group name.



```
$ va
```

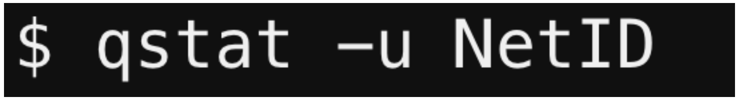
find the group name



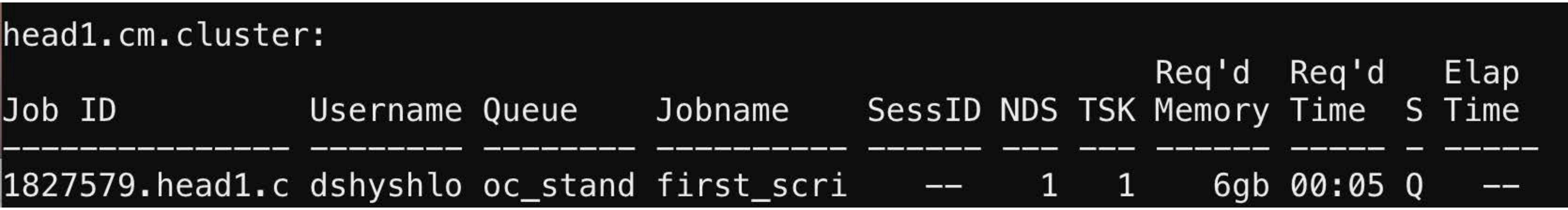
```
$ qsub first_script.pbs
1827586.head1.cm.cluster
```

Exercise – running a batch job

- Check on the status of the job.



```
$ qstat -u NetID
```



```
head1.cm.cluster:
                                                                   Req'd  Req'd   Elap
Job ID          Username Queue    Jobname    SessID NDS TSK Memory Time  S Time
--------------- -------- -------- ---------- ------ --- --- ------ ----- - -----
1827579.head1.c dshyshlo oc_stand first_scri    --    1   1    6gb 00:05 Q   --
```

- Look in the standard output and error files (use **cat** command).

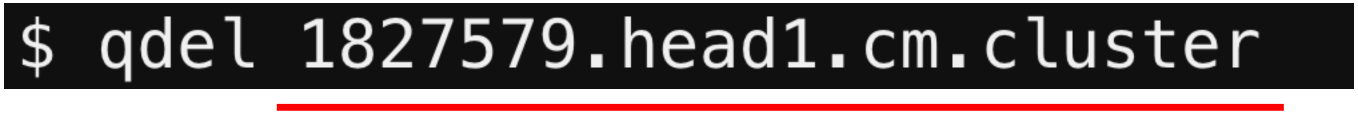
Exercise – customizing a job

- Submit the job that uses 4 cores from 1 node.

- What parameters do you need to change in the PBS script?

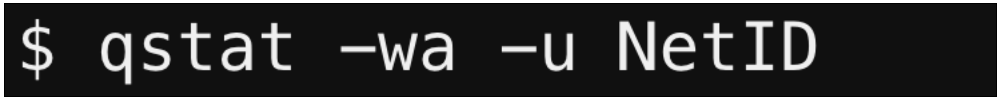
Exercise – deleting a job

- You can delete the job from the queue or when it’s running with a **qdel** command.



```
$ qdel 1827579.head1.cm.cluster
```

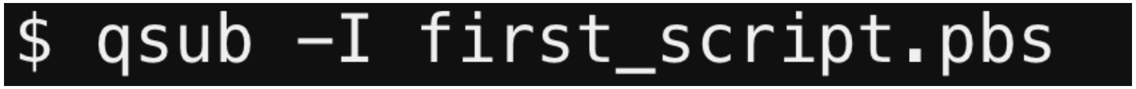
- You can find a full job ID with a command



```
$ qstat -wa -u NetID
```

Interactive job

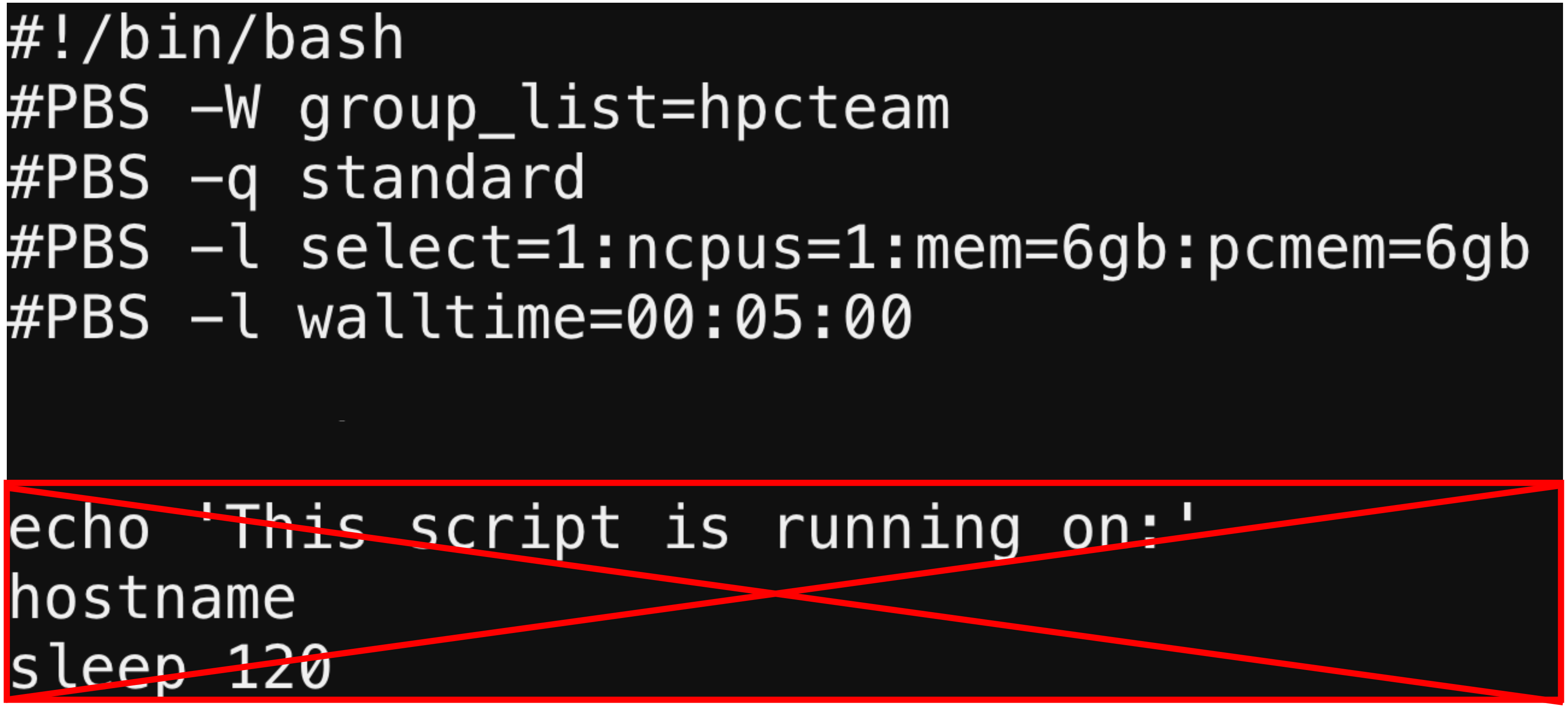
- Request compute node resources for interactive work
  - interactive data analysis or compute
  - long-running debugging
  - copying large files



```
$ qsub –I first_script.pbs
```

- Close the interactive session with the command ***exit*** or ***logout***.

Interactive job



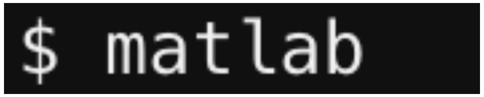
```
#!/bin/bash
#PBS -W group_list=hpcteam
#PBS -q standard
#PBS -l select=1:ncpus=1:mem=6gb:pcmem=6gb
#PBS -l walltime=00:05:00
```

~~echo 'This script is running on:'~~
~~hostname~~
~~sleep 120~~

Accessing Software

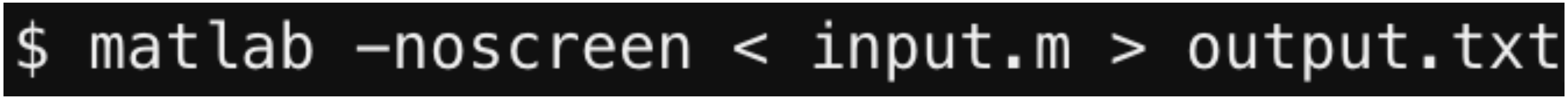
GUI

Command line



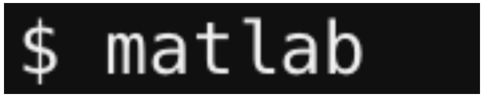
```
$ matlab
```

Or more often…

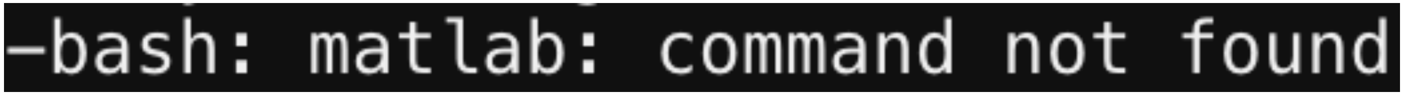


```
$ matlab -noscreen < input.m > output.txt
```

Accessing Software

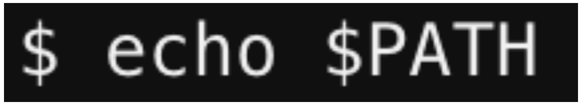


```
$ matlab
```



```
-bash: matlab: command not found
```

- To run any software in Linux the system must know where to look for appropriate binaries and libraries.
- Check your environment:

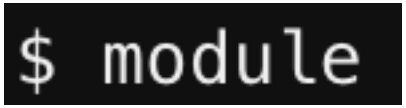


```
$ echo $PATH
```

Accessing Software

- Environment modules are a convenient way to customize your environment to use software

- Manage your modules with a command

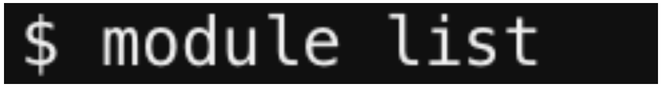


```
$ module
```

- Running *module* command without any options will open a help page.

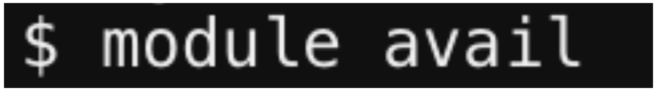
Accessing Software: hand-on activity

- List your current modules



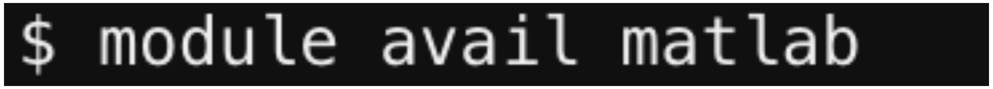
```
$ module list
```

- List all the available modules



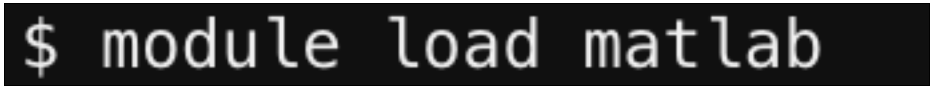
```
$ module avail
```

- List all the available MATLAB modules



```
$ module avail matlab
```

- Load a MATLAB module

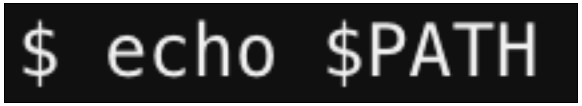


```
$ module load matlab
```

- List your current modules

Accessing Software: hands-on activity

- Check your environment again:



```
$ echo $PATH
```

- Anything new?

- When you run a batch job *module load* commands must be in the PBS script.

Being a good HPC user

- Things that can “break” the system:
    - heavy use of the login node
    - too many jobs
    - too many files
    - heavy I/O jobs
    - copying GB of data
    - *for* loops in PBS scripts

Getting help

- HPC documentation – docs.hpc.arizona.edu
- HPC consulting – hpc-consult@list.arizona.edu
- Visualization consulting – vislab-consult@list.arizona.edu
- Statistics consulting – stat-consult@list.arizona.edu