Crash Course in High Performance Computing

Cyber-Infrastructure Days
October 24, 2013

Dirk Colbry
colbrydi@msu.edu
Research Specialist
Institute for Cyber-Enabled Research

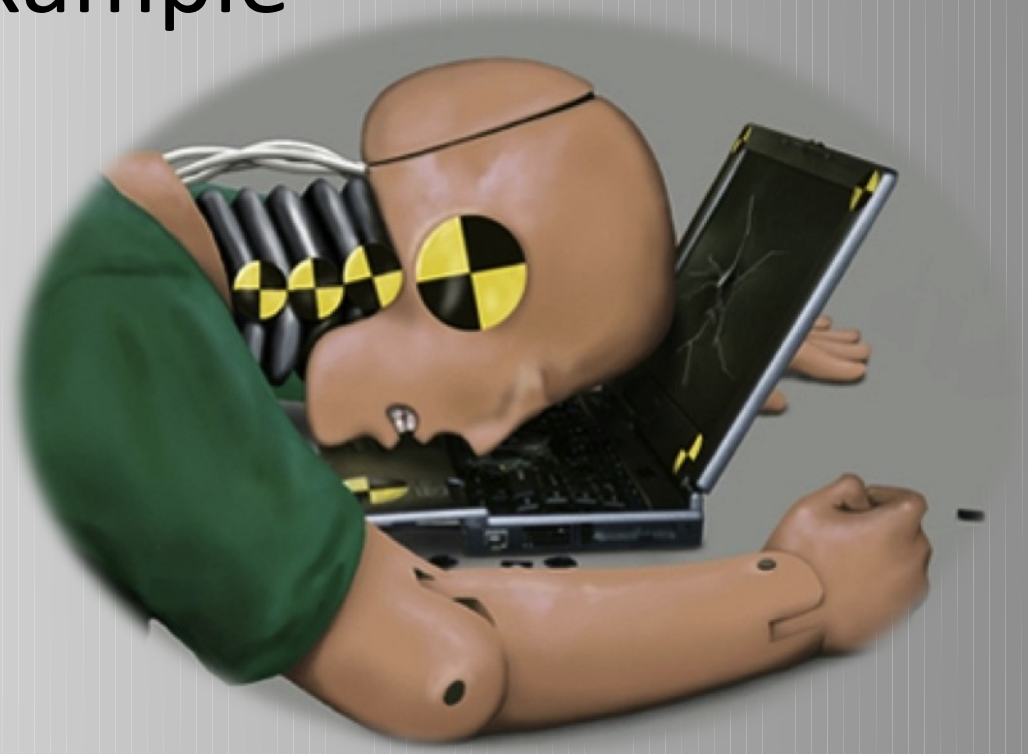
https://wiki.hpcc.msu.edu/x/QAMrAQ

MICHIGAN STATE UNIVERSITY

ICER

Agenda

- Introduction to the HPCC
- The Seven Steps to using the HPCC
- Hands on example



MICHIGAN STATE UNIVERSITY

ICER

How this workshop works

- We are going to cover some basics, lots of hands on examples, in a very short period of time.
- When you get tired of listening to me talk, skip ahead to an exercise and give it a try.
- Exercises are denoted by the following icon in your notes:

Red and Green Flags

- Use the provided sticky notes to help me help you.
  - **NO Sticky** = I am working
  - **Green** = I am done and ready to move on
  - **Red** = I am stuck and need more time and/or I could use some help

HPC Systems

FREE*

- Large Memory Nodes (up to 2TB!)
- GPU Accelerated cluster (K20, M1060)
- PHI Accelerated cluster (5110p)
- Over 540 nodes, 10000 computing cores
- Access to high throughput condor cluster
- 363TB high speed parallel scratch file space
- 50GB replicated file spaces
- Access to large open-source software stack and specialized bioinformatics VMs

Available Software

- Center Supported Development Software
  - Intel compilers, openmp, openmpi, mvapich, totalview, mkl, pathscale, gnu...
- Center Supported Research Software
  - MATLAB, R, fluent, abaqus, HEEDS, amber, blast, ls-dyna, starp...
- Customer Software
  - gromacs, cmake, cuda, imagemagick, java, openmm, siesta…
  - For a more up to date list, see the documentation wiki:
    - http://wiki.hpcc.msu.edu/

What if I need help?

- Ask us!
- Local Workshops
  - Software carpentry
  - Introduction to Linux and HPCC
  - Advanced HPCC
- Remote Training
  - VSCSE – Virtual School for Computer Science Education
  - XSEDE training Workshops

MSU Seminars in Research and Instructional Technology

Dec 17, 18, 2013

- Two days of no-cost seminars to faculty and graduate students on technology topics.
  - Morning sessions run from 8:30 to 11:30 am.
  - Afternoon sessions run from 1:30 to 4:30 pm.
  - Lunch is provided that will feature guest speakers on instructional technology.

  - Introduction to HPC
  - Advanced HPC

Seven Steps to using the HPCC (The Basics)

http://www.softwarecarpentry.org/

MICHIGAN STATE UNIVERSITY

ICER

Steps in Using the HPCC

1. **Get an account**
2. Install needed software (SSH, SCP, X11)
3. Transfer input files and source code
4. Compile/Test programs on a developer node
5. Write a submission script
6. Submit the job
7. Get your results and write a paper!!

Accounts

- PIs must request accounts for students:
  - http://www.hpcc.msu.edu/request
- All HPCC systems use MSU NetIDs and Passwords.
- We have temporary accounts for today.
- If you are Faculty/Staff and do not have an account, fill out the above request now. We can try to activate your account before the end of the workshop.

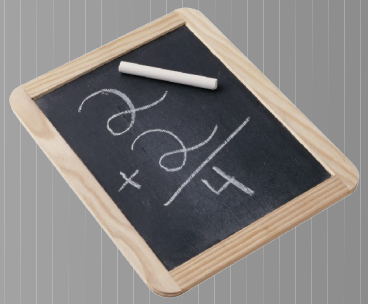
Steps in Using the HPCC

1. Get an account
2. **Install needed software (SSH, SCP, X11)**
3. Transfer input files and source code
4. Compile/Test programs on a developer node
5. Write a submission script
6. Submit the job
7. Get your results and write a paper!!

Required Software

- Secure Shell (ssh)
- File transfer
  - Secure Copy (scp)
  - Mapping home directories
- Graphical User Interface (x11)
  - Optional

Apple



- Run Terminal program
  - ssh – already installed

    ssh –X userid@hpcc.msu.edu
  - scp – already installed

    scp ./mylocalfile userid@hpcc.msu.edu:~/mylocalfile

- May need to install Xquarts (mac X11 Server)
  - Installer should be on USB drive

Windows Software

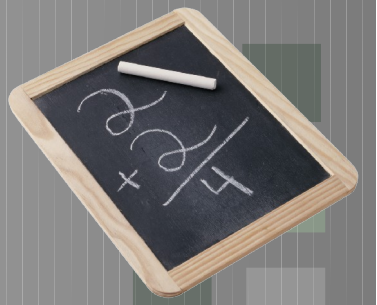
- PuTTY:
  - http://www.chiark.greenend.org.uk/~sgtatham/putty/
- Xming:
  - http://www.straightrunning.com/XmingNotes/
- Xming install:
  - https://wiki.hpcc.msu.edu/x/swAk
- WinSCP:
  - http://winscp.net

MobaXterm (windows)

- Complete toolbox for remote computing:
  - Multi-tab terminal
  - X11 server
  - SSH
  - File transfer
  - More

- Opensource
- http://mobaxterm.mobatek.net/

Exercise: Portable HPCC



- If you have Windows
- Plug in your USB thumb drive
- Open the thumb drive folder and select
  - PortableApps
- You should see a new menu in your system tray for navigating

Exercise: Connect to HPCC

- Step 1: Log into gateway.hpcc.msu.edu

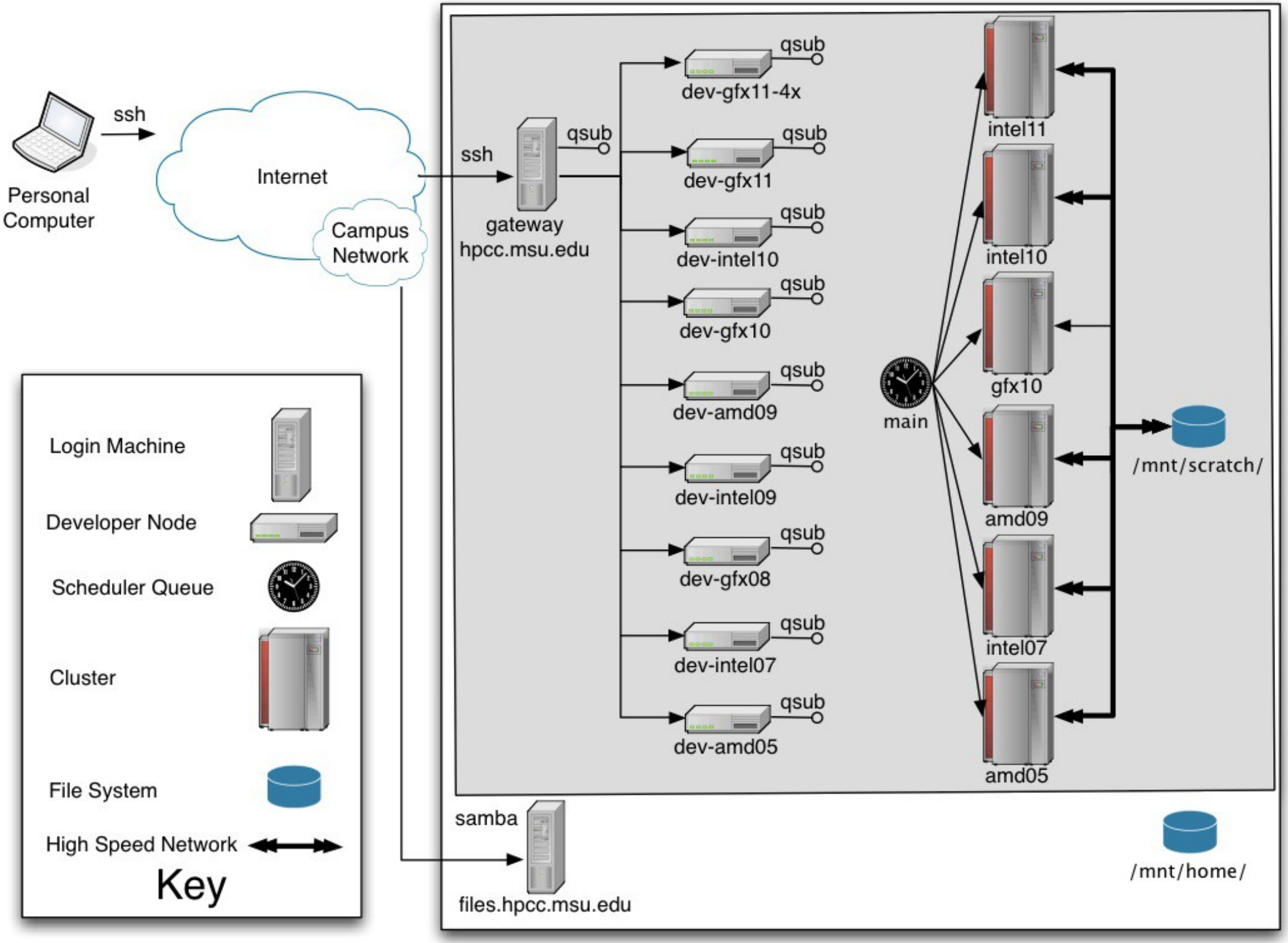
```
Welcome to Michigan State's High Performance Computing Center
** Unauthorized access is prohibited **

We recommend using dev-amd09 (or nodes with low usage).
For GPU development please use green nodes.
For MIC development please use underlined nodes

      Development Nodes (usage)                Filesystem Information
  --------------------------------            --------------------------
dev-intel07 (low)    dev-amd09 (low)            ${HOME} at 95% usage
dev-intel10 (high)   dev-gfx10 (low)            (used ~48G of 50G)
dev-gfx13 (low)      dev-phi13 (low)
                                              **WARNING - REACHING QUOTA**
                                           Request at: www.hpcc.msu.edu/quota

               Cluster Load (utilization)
            --------------------------------
     short jobs (< 4 hrs) (85%)          general jobs (73%)
     large memory jobs    (100%)          gpu jobs     (9%)
```

- Step 2: ssh into a dev node (developer node)

```
>ssh dev-intel10
```

- Step 3: execute a command

```
>echo "Hello world"
```


Personal Computer
ssh
Internet
Campus Network
ssh
gateway
hpcc.msu.edu
qsub
dev-gfx11-4x
dev-gfx11
dev-intel10
dev-gfx10
dev-amd09
dev-intel09
dev-gfx08
dev-intel07
dev-amd05
main
intel11
intel10
gfx10
amd09
intel07
amd05
/mnt/scratch/
/mnt/home/
samba
files.hpcc.msu.edu
Login Machine
Developer Node
Scheduler Queue
Cluster
File System
High Speed Network
Key

Command Line Interface

- Command Line Interface (CLI)
- Shell
  - Program to run Programs
- Bash (Bourne Again Shell)
- Use it because:
  - many tools only have command-line interfaces
  - allows you to combine tools in powerful new ways

Shell Navigation

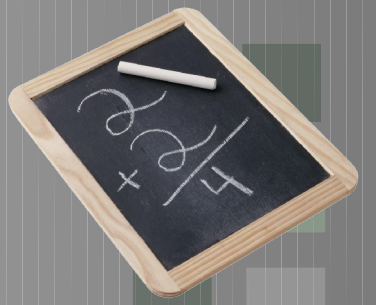
- Basic Navigation commands:

pwd	print working directory
cd	change working directory
ls	list directory

- Use the following symbols to indicate special directories:

.	current directory
..	parent directory
~	home directory
-	previous directory

Exercise – Shell Navigation



- Show the path to the current directory

```
>pwd
```

- Change to the scratch directory

```
>cd /mnt/scratch/
```

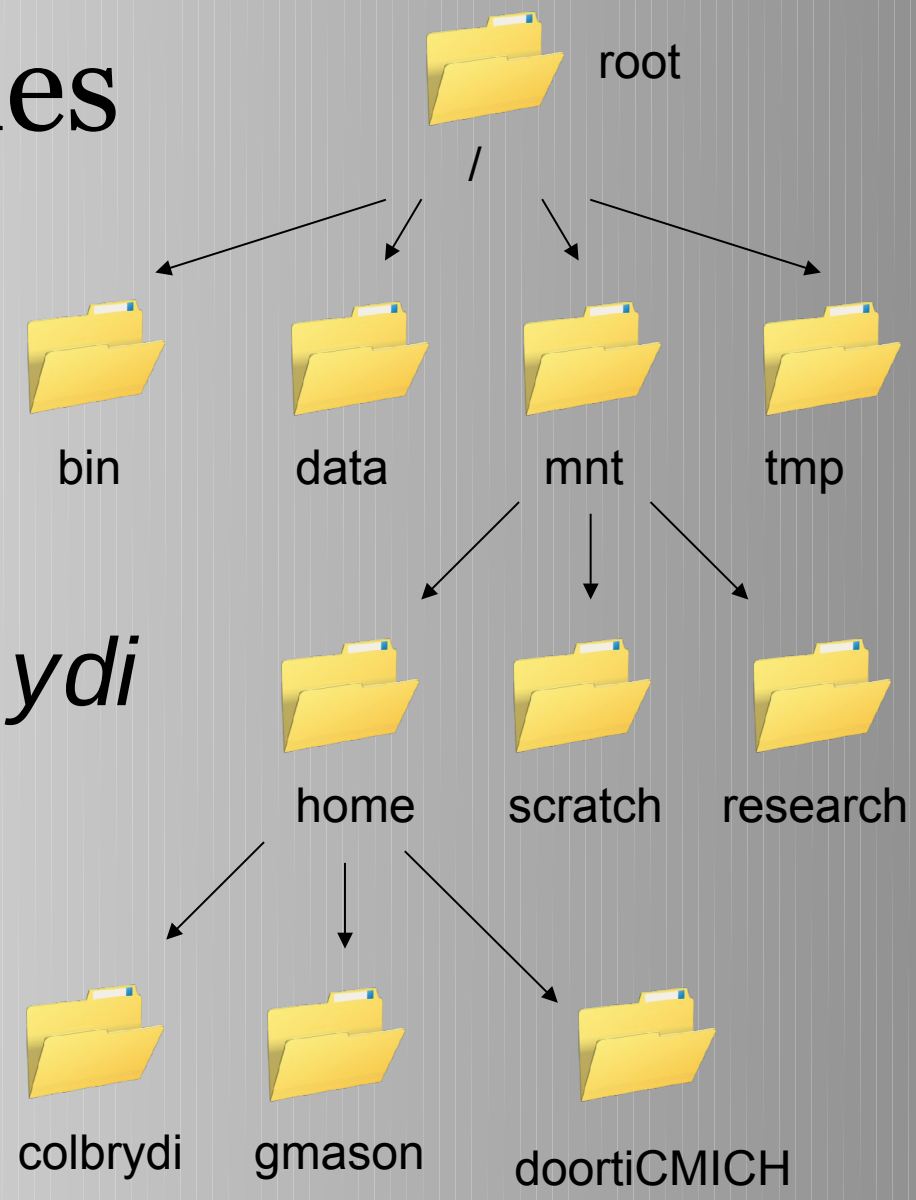
- List the contents of the current directory:

```
>ls
```

- Change back to home

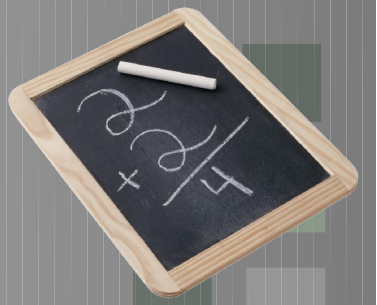
```
>cd ~
```

Directories



/ mnt / home/ colbrydi

Exercise – man Pages



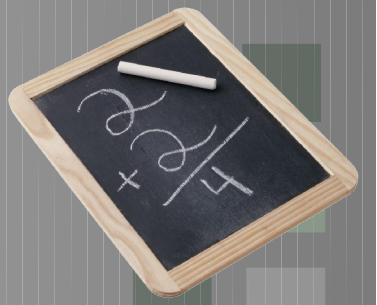
- Manual Pages (man pages)
- Built in documentation
- Very helpful if you know the command but do not know how to get it working

  >man pwd

- NOTE: Use ‘q’ to quit

- Not helpful if you do not know the command name… use google

Example: File Manipulation



- Try Commands

mkdir	make directory
cp	copy file
cat	display contents of text file
rm	remove file

- See the contents of your ".bashrc" file

```
> man cat
> cat .bashrc
```

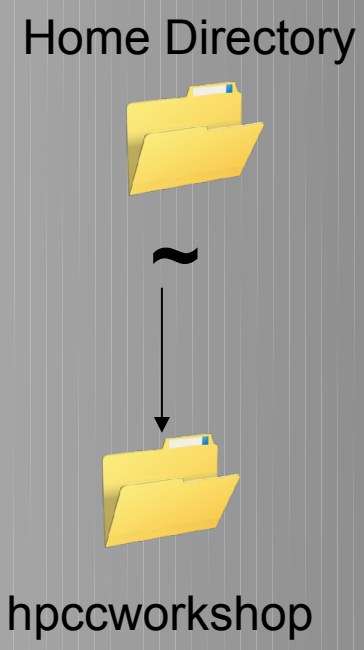
- Make a directory called "hpccworkshop", change to that directory and list the contents.

```
> mkdir hpccworkshop
> cd ./hpccworkshop
```

~ Home Shortcut

- */mnt/home/colbrydi*

```
>cd ~
>pwd
>cd ~/hpccworkshop
>pwd
```



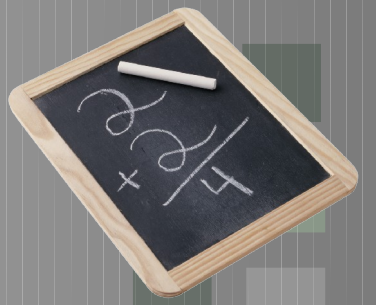
Available Software

- Center Supported Development Software
  - Intel compilers, openmp, openmpi, mvapich, totalview, mkl, pathscale, gnu...
- Center Supported Research Software
  - MATLAB, R, fluent, abaqus, HEEDS, amber, blast, ls-dyna, starp...
- Customer Software
  - gromacs, cmake, cuda, imagemagick, java, openmm, siesta…
  - For a more up to date list, see the documentation wiki:
    - http://wiki.hpcc.msu.edu/

Module System

- To maximize the different types of software and system configurations that are available to the users, HPCC uses a Module system
- Key Commands
  - **module avail** – show available modules
  - **module list** – list currently loaded modules
  - **module load** modulename – load a module
  - **module unload** modulename – unload a module
  - **module spider keyword** – Search modules for a keyword

Exercise – Module



- List loaded modules

```
>module list
```

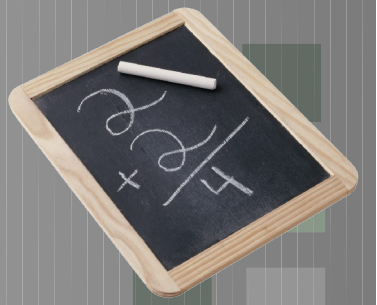
- Show available modules:

```
>module avail
```

- Try an example (Shouldn't work):

```
>powertools
```

Exercise: getexample



- Load a module:

```
> module load powertools
```

- Show powertools (should work now):

```
> powertools
```

- Run the “getexample” powertool

```
> getexample
```

- Download the helloworld example

```
> getexample helloworld
```

Steps in Using the HPCC

1. Get an account
2. Install needed software (SSH, SCP, X11)
3. **Transfer input files and source code**
4. Compile/Test programs on a developer node
5. Write a submission script
6. Submit the job
7. Get your results and write a paper!!

Current files

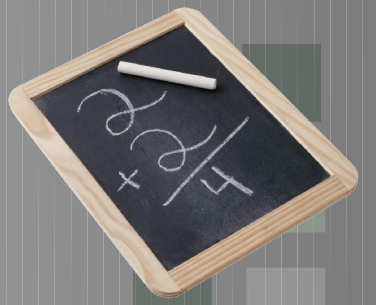
Home Directory

~

hpccworkshop

helloworld

Exercise: Make a Bash Script



- Make a file called **hello.sh** using notepad++ on your thumb drive or another text editor
- Put the following lines in your text file:

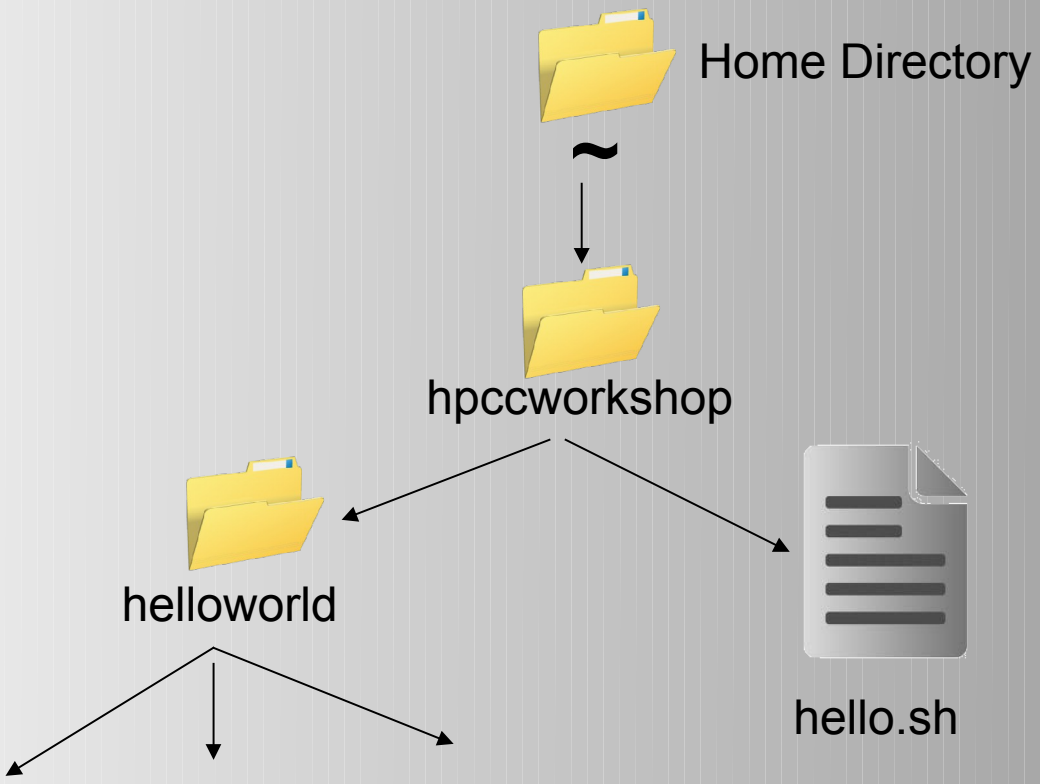
```
#!/bin/bash
echo "Hello ${USER} and Hello World"
```

SCP/SFTP – Secure File transfer

- WinSCP for Windows
- Command-line “scp” and “sftp”on Apple/Linux
- Many other scp and sftp clients out there as well
- Functions over SSHv2 protocol, very secure

Transfer File

- Copy the file **hello** to your home directory on the HPCC using scp (or Winscp)



scp ./hello colbrydi@hpcc.msu.edu:~/hpccworkshop

File Permissions

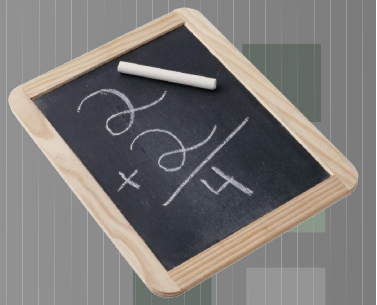
	 user	 group	 all
read	✔	✔	✗
write	✔	✗	✗
execute	✗	✗	✗

Permissions

- Common Commands

chmod	Change permissions (change mode)
ls -a -l	List all long (including permissions)

Example: permissions



- Try to run **hello.sh** command

```
> cd ./hello.sh
```

- Show current permissions

```
> ls -la
```

- Make the **hello.sh** file executable

```
> chmod u+x hello.sh
```

- Check permissions again

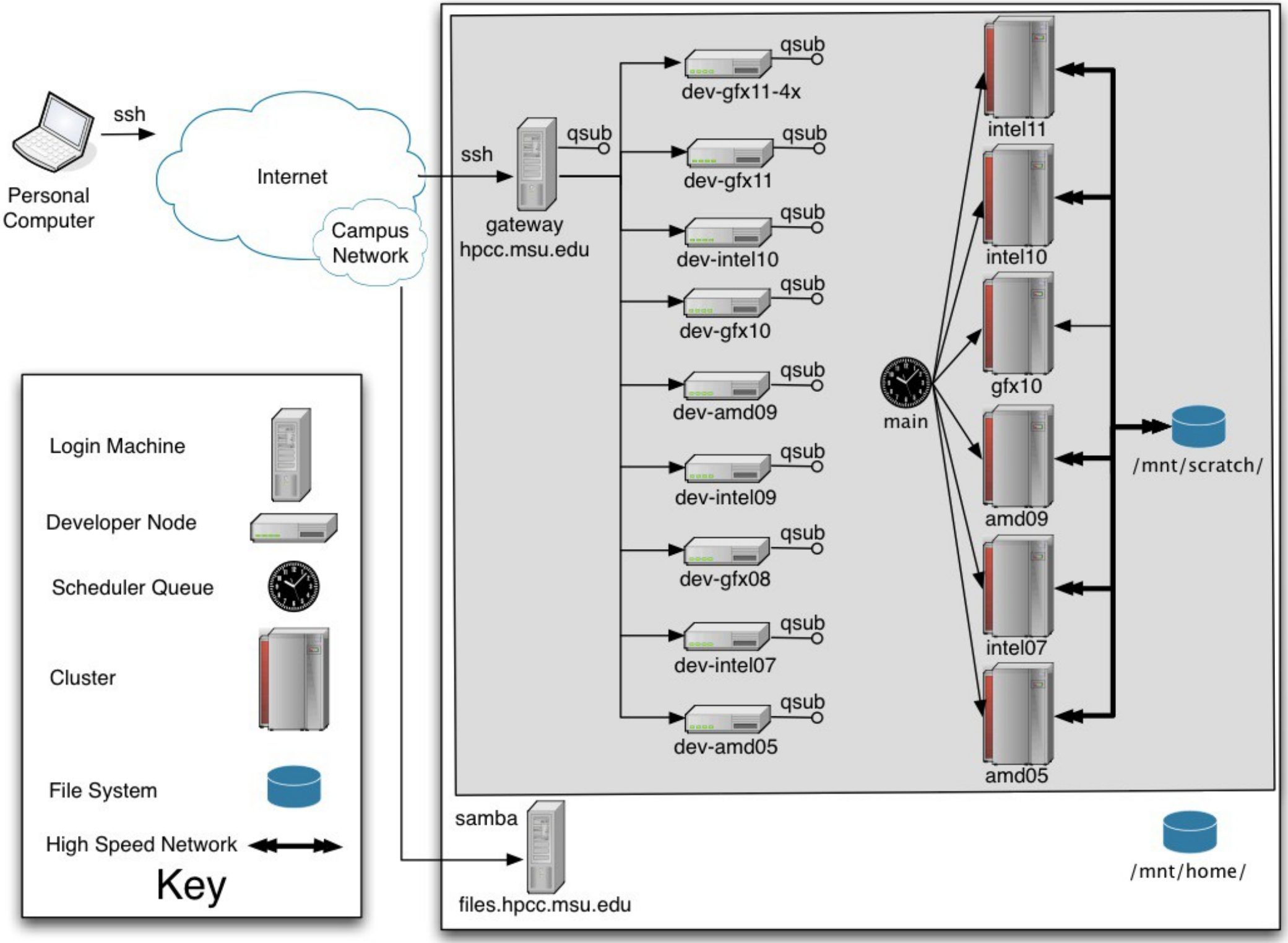
```
> ls -la
```

- Now you can run **hello.sh** as a command

```
> ./hello.sh
```

Steps in Using the HPCC

1. Get an account
2. Install needed software (SSH, SCP, X11)
3. Transfer input files and source code
4. **Compile/Test programs on a developer node**
5. Write a submission script
6. Submit the job
7. Get your results and write a paper!!


Personal Computer
ssh
Internet
Campus Network
ssh
gateway
hpcc.msu.edu
qsub
dev-gfx11-4x
dev-gfx11
dev-intel10
dev-gfx10
dev-amd09
dev-intel09
dev-gfx08
dev-intel07
dev-amd05
main
intel11
intel10
gfx10
amd09
intel07
amd05
/mnt/scratch/
/mnt/home/
samba
files.hpcc.msu.edu
Login Machine
Developer Node
Scheduler Queue
Cluster
File System
High Speed Network
Key

Running Jobs on the HPC

- Submission scripts are used to run jobs on the cluster
- The developer (dev) nodes are used to compile, test and debug programs

- However, the developer nodes are powerful systems too. **We don't want to waste their compute power.**

Advantages of running Interactively

- You do not need to write a submission script
- You do not need to wait in the queue
- You can provide input to and get feedback from your programs as they are running

Disadvantages of running Interactively

- All the resources on developer nodes are shared between all users.
- Any single process is limited to 2 hours of cpu time. If a process runs longer than 2 hours it will be killed.
- Programs that overutilize the resources on a developer node (preventing other to use the system) can be killed without warning.

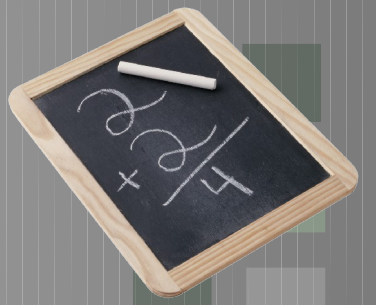
Developer Nodes

Name	Cores	Memory	GPUs	Notes
dev-amd05	4	8GB	-	
dev-intel07	8	8GB	-	
dev-gfx08	4	8GB	3	Graphics Node
dev-amd09	32	256GB	-	Fat Node
dev-intel09	8	32GB	-	Fastest Node
dev-gfx10	8	18GB	2	Graphics Node
dev-intel10	8	24GB	-	
dev-gfx11	4	8GB	2	Graphics Node
dev-gfx11-4x	8	18GB	4	Graphics Node

Compilers

- By default we use the gnu compilers. However, lots of other compilers are available including Intel and Portland compilers.
- The module system always sets environment variables such that you can easily test with other compilers.
  - ${CC}
  - ${FC}
  - Etc.

Exercise: Compile Code



- Make sure you are in the helloworld directory:

```
>pwd
```

- Run the gcc compilers:

```
>${CC} -O3 -o hello hello.c
```

- Run the program:

```
> ./hello
```

- Use “control-C” to cancel

Running in the background

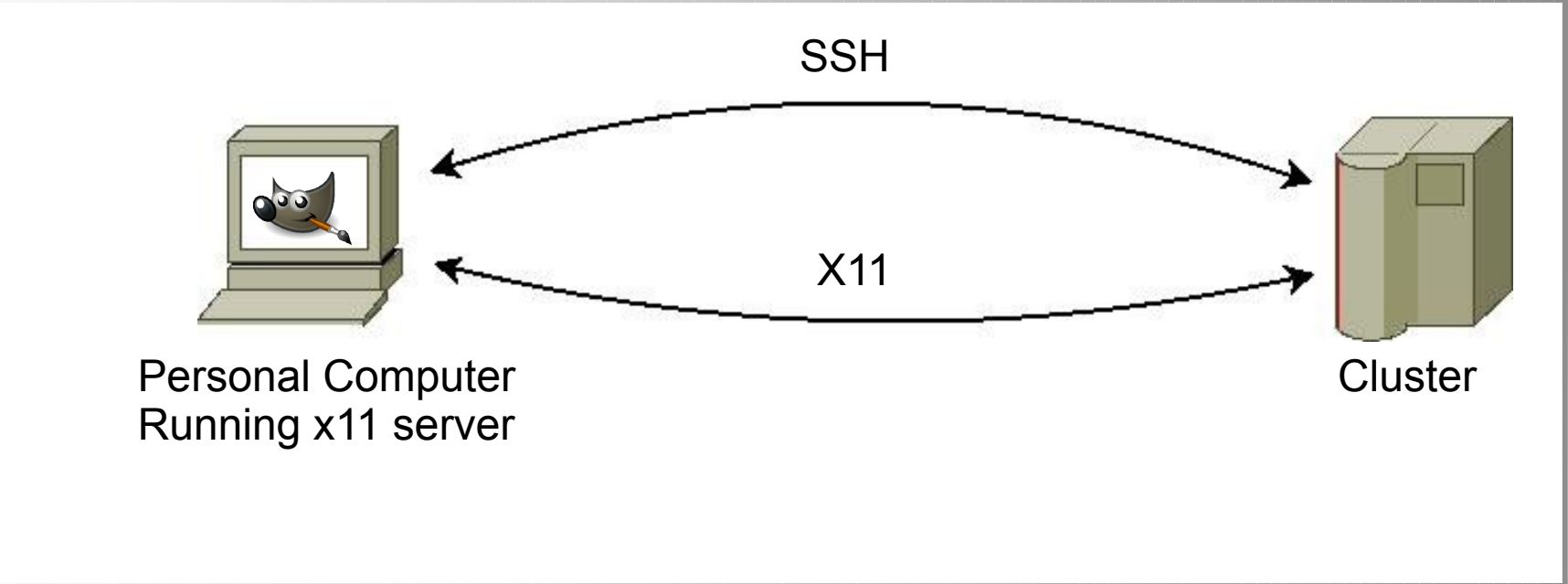
- You can run a program in the background by typing an "**&**" after the command.
- You can make a program keep running even after you log out of your ssh session by using "**nohup** command"
- You can run an entire session in the background even if you log in and out of your ssh session by using the "**screen**" or "**tmux**" commands
- All three of these options are common to linux and tutorials can be found online

CLI vs GUI

- CLI – Command Line Interface
- GUI – Graphical User Interface

What is X11?

- Method for running Graphical User Interface (GUI) across a network connection.



What is needed for X11

- X11 server running on your personal computer
- SSH connection with X11 enabled
- Fast network connection
  - Preferably on campus

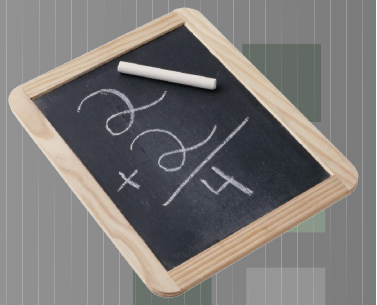
Graphical User Interface

- X11 Windows: Install Xming
  - Installation instructions at: https://wiki.hpcc.msu.edu/x/swAk
- ssh -X username@hpc.msu.edu
- Turn on x11 forwarding

- Note: Mac Lion Users should use XQuart

http://xquartz.macosforge.org/

Exercise: Transfer a file



- Try one of the following Commands

xeyes	Test X11
firefox	Web browser

> **xeyes**

> **firefox &**

> **ps <- Find the process ID #### for firefox**

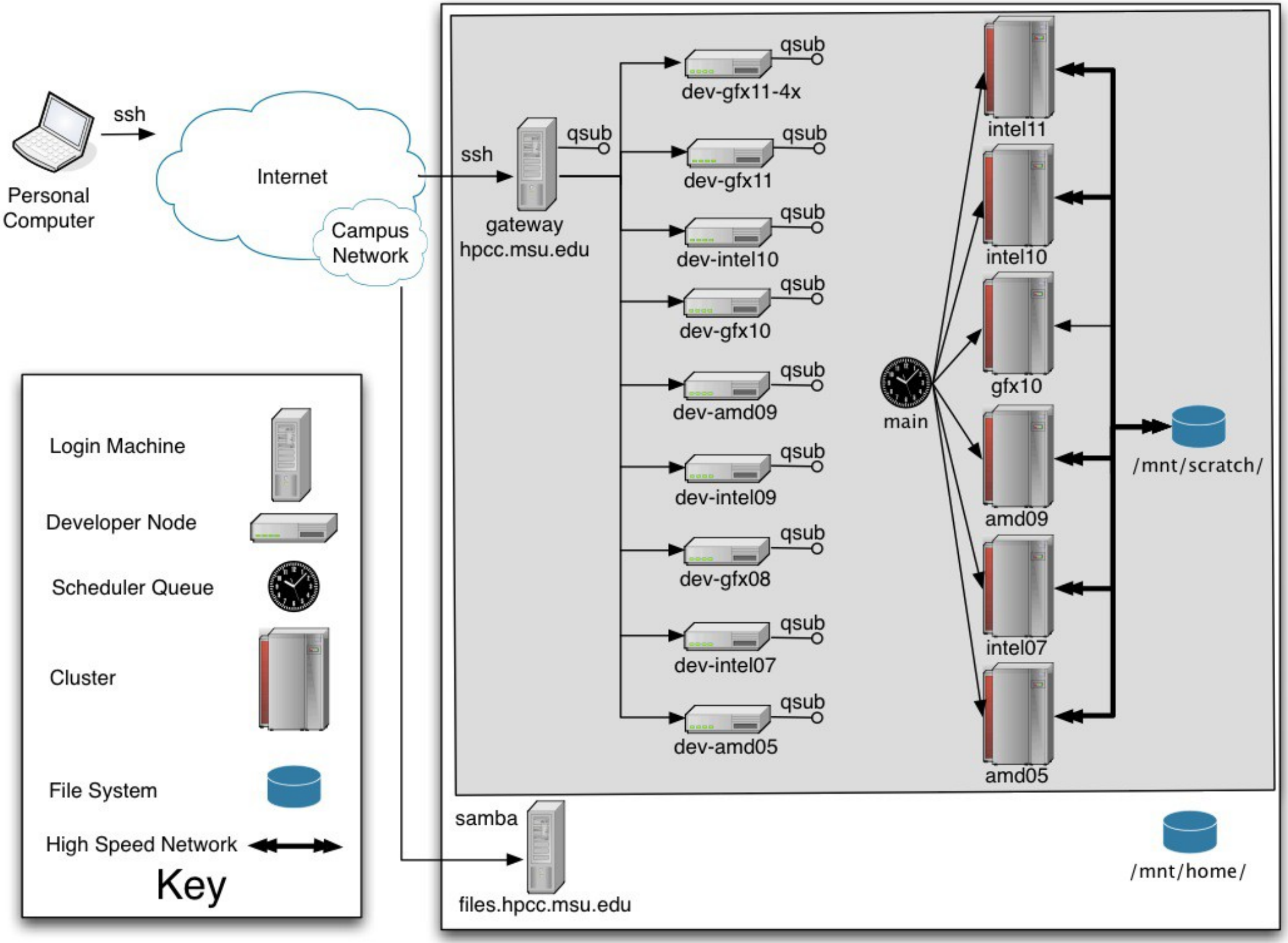
> **kill ####**

Programs that can use X11

- R - statistical computing and graphics
- matlab – Matrix Laboratory
- firefox – Web browser
- totalview – C/C++/fortran debugger
- gedit, gvim, emacs – Text editors
- And others…

Steps in Using the HPCC

1. Get an account
2. Install needed software (SSH, SCP, X11)
3. Transfer input files and source code
4. Compile/Test programs on a developer node
5. **Write a submission script**
6. Submit the job
7. Get your results and write a paper!!


Personal Computer
ssh
Internet
Campus Network
ssh
gateway
hpcc.msu.edu
qsub
dev-gfx11-4x
dev-gfx11
dev-intel10
dev-gfx10
dev-amd09
dev-intel09
dev-gfx08
dev-intel07
dev-amd05
main
intel11
intel10
gfx10
amd09
intel07
amd05
/mnt/scratch/
/mnt/home/
samba
files.hpcc.msu.edu
Login Machine
Developer Node
Scheduler Queue
Cluster
File System
High Speed Network
Key

Submission Script

1. List of required resources
2. All command line instructions needed to run the computation

Typical Submission Script

Shell Comment

Define Shell

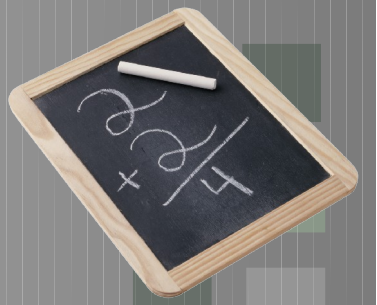
```
#!/bin/bash -login
#PBS -l walltime=10:00:00,mem=3Gb,nodes=10:ppn=1
#PBS -j oe

cd ${PBS_O_WORKDIR}

./myprogram -my input arguments

qstat -f ${PBS_JOBID}
```

Resource Requests

Shell Commands

Special Environment Variables

Example: Submit a job



- Go to the top helloworld directory

```
>cd ~/hpccworkshop/helloworld
```

- Look at the simple submission script

```
>nano hello.qsub
```

- Nano is a simple program you can use to edit text files on the HPCC
- See bottom line of nano for commands the "^" character indicates the "control" key

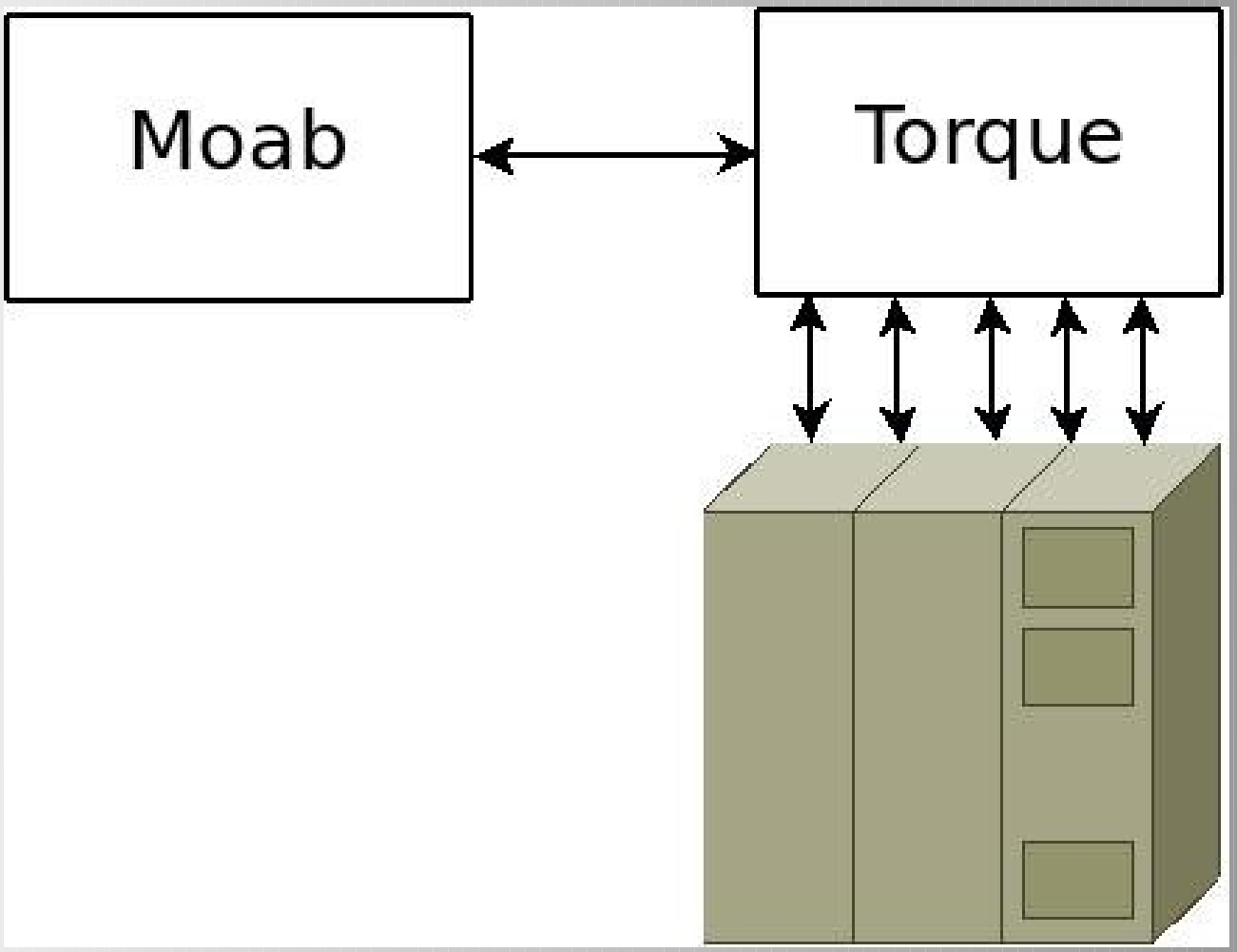
hello.qsub

```
#!/bin/bash -login
#PBS -l walltime=00:05:00
#PBS -l nodes=1:ppn=1,feature=gbe

cd ${PBS_O_WORKDIR}

./hello

qstat -f ${PBS_JOBID}
```

Steps in Using the HPCC

1. Get an account
2. Install needed software (SSH, SCP, X11)
3. Transfer input files and source code
4. Compile/Test programs on a developer node
5. Write a submission script
6. **Submit the job**
7. Get your results and write a paper!!

Resource Manager and scheduler



Not First In First Out!!

Common Commands

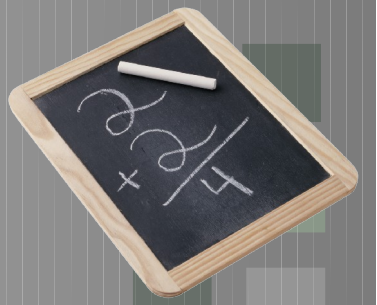
- **qsub** <Submission script>
  - Submit a job to the queue
- **qdel** <JOB ID>
  - Delete a job from the queue
- **showq** –u <USERNAME>
  - Show the current job queue
- **checkjob** <JOB ID>
  - Check the status of the current job
- **showstart** –e all <JOB ID>
  - Show the estimated start time of the job.

Submitting a job

- qsub –arguments <Submission Script>
  - Returns the job ID. Typically looks like the following:
    - 5945571.cmgr01

- Time to job completion

Example: Submit a job, cont.



- Submit the file to the queue

```
>qsub hello.qsub
```

- Record jobid number (######) and wait at most 30 seconds
- Check the status of the queue

```
>showq
```

Example: Monitor a job

- Get the status of the job:

```
>qstat -f ######
```

- When will a job start:

```
>showstart -e all ######
```

Scheduling Priorities

- Jobs that use more resources get higher priority (because these are hard to schedule)
- Smaller jobs are backfilled to fit in the holes created by the bigger jobs
- Eligible jobs acquire more priority as they sit in the queue
- Jobs can be in three basic states:
  - Blocked, eligible or running

Cluster Resources

Year	Name	Description	ppn	Memory	Nodes	Total Cores
2005	amd05	Dual-core 2.2GHz AMD Opteron 275	4	8GB	96	384
2007	intel07	Quad-core 2.3GHz Intel Xeons E5345	8	8GB	124	992
2009	amd09	Sun Fire X4600 (Fat Node)	16	128GB	1	16
			32	256GB	4	128
2010	gfx10	Nvidia Cuda Node (no IB)	8	18GB	41	256
2010	intel10	Intel Xeon E5620 (2.40 GHz)	8	24GB	192	1536
2011	intel11	Intel Xeon 2.66 GHz E7-8837	32	512GB	1	32
			32	1TB	1	32
			64	2TB	2	128

System Limitations

- Scheduling
  - 5 eligible jobs at a time
  - 144 running jobs
  - 256 submitted jobs (increasing soon)
- Resources
  - 1 week of walltime
  - 144 cores (nodes * ppn)
  - ppn=64
  - 1TB memory on a single core
  - ~200 GB Hard Drive

Job completion

- By default the job will automatically generate two files when it completes:
  - Standard Output:
    - Ex: jobname.o5945571
  - Standard Error:
    - Ex: jobname.e5945571
- You can combine these files if you add the join option in your submission script:
  - “#PBS -j oe”
- You can change the output file name
  - #PBS -o /mnt/home/netid/myoutputfile.txt

Other Job Properties

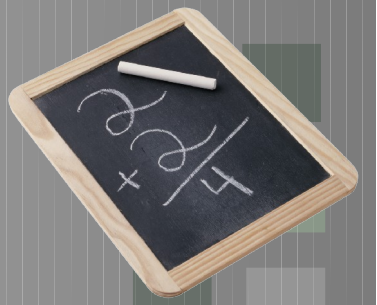
- resources (-l)
  - Walltime, memory, nodes, processor, network, etc.
- #PBS –l feature=gpgpu,gbe
- #PBS –l nodes=2:ppn=8:gpu=2
- #PBS –l mem=16gb
- Email address (-M)
  - Ex: #PBS –M colbrydi@msu.edu
- Email Options (-m)
  - Ex: #PBS –m abe

Many others, see the wiki: http://wiki.hpcc.msu.edu/

Steps in Using the HPCC

1. Get an account
2. Install needed software (SSH, SCP, X11)
3. Transfer input files and source code
4. Compile/Test programs on a developer node
5. Write a submission script
6. Submit the job
7. **Get your results and write a paper!!**

Example: getexample



- See list of examples

```
>getexample
```

- Pick one

```
>getexample examplename
```

- Change to the example directory

```
>cd examplename
```

- Run the example. Look for README file and/or submit qsub file

```
>qsub examplename.qsub
```

Feedback

https://docs.google.com/forms/d/1NVE7TYF310vSw0t

We are here to help

- www.hpcc.msu.edu/contact
  - Questions
  - Schedule Consultations
  - Code Reviews
  - Programming help
  - Hardware Purchasing
  - Help with Grants
  - Support for Grants