

INTRODUCTION TO QUEST

KYLE YAKAL-KREMSKI

4/30/2012

HOW TO ACCESS QUEST

- › **Set up an account with a testing allocation**

- › <http://bit.ly/QuestAccnt>

- › Allocation gives 2,500 processor hours

- › Little information needed, hear back in a business day

- › **On PCs – install a SSH client, like PuTTY**

- › **Access by**

```
ssh you123@quest.northwestern.edu
```

- › **100GB of storage for data**

- › **Accessible off-network**

UNIX 101

- › **Text-based, command prompt**
- › **Commands follow the general form**
`command -option(s) argument(s)`
- › **Useful commands**
 - › `pwd` : show current location
 - › `man` : open manual
 - › `cd` : change directory
 - › `mkdir` : make directory
 - › `cp` : copy
 - › `mv` : move
 - › `rm` : remove
 - › `ls` : list files
- › **Use vi or emacs as text editors**
- › **Wolverton Guide: <http://bit.ly/WolvUnix>**

THINGS TO AVOID

- › **Using too much memory. Memory is shared by node (8 or 16 processors)**
 - › Esp. recursive routines
- › **Excessive read/write - this slows file accessing for all users**
- › **Running jobs on the head (log-in) nodes**
 - › Submit jobs using the `mSub` command, using a `.q` file
 - › Check the queue with `qstat`
 - › Kill jobs with `canceljob <jobnumber>`

USING AVAILABLE PROGRAMS

- › `module load <programname>`
- › **Graphics require X11 server (Xming)**
- › **Matlab, Mathematica, TeX, Paraview, GnuPlot, Eclipse, and others**
- › **Native support for Python, Perl, Java**
- › **Load existing code using `SCP` command on local command prompt**