

Lab 1

Programming Environment, File
Systems, Modules, Batch System



Initial Login

- Login to the Sun Constellation system (ranger) using your training account login:
- Linux/OSX:

```
% ssh train##@ranger.tacc.utexas.edu
```
- Windows:
Use the provided SSH client (TACC laptops)
or download PuTTY.



Useful Commands

- Try out a few of the following common commands:

- `env`
- `df -h`
- `ls -la`
- `uname -a`
- `cdw`
- `echo $WORK`
- `cd`



3



File Systems – scp vs. bbcp

- Let's compare transfer speeds for encrypted (scp) and unencrypted (bbcp) transfer to *ranch.tacc.utexas.edu*
- Create a 512 MB file using the `dd` command

```
% dd if=/dev/zero of=zero_512 bs=1M count=512
```
- Transfer this file to ranch twice, noting the transfer speeds reported by each method. (You will need to type your password each time.)

```
% bbcp -V zero_512 ${ARCHIVER}:${ARCHIVE}
% scp zero_512 ${ARCHIVER}:${ARCHIVE}
```
- After completing the transfers, remove the data files!

```
% rm zero_512
% ssh ranch rm zero_512
```



4



Modules

- List the arguments available in the module command
% **module**
- List the modules that are presently loaded
% **module list**
- List the modules that are available
% **module avail**
- Determine which mpicc is being used and switch compiler/MPI stacks
% **which mpicc**
% **module unload mvapich**
% **module swap pgi intel**
% **module load mvapich-devel**
% **which mpicc**



5



Get the Lab 1 Datafile

- Untar the file lab1.tar file (in ~train00) into your directory:
% **tar xvf ~train00/lab1.tar**
- Change into the lab1 directory:
% **cd lab1**
% **ls**



6



SGE Batch (Ranger)

- `cd` to `$HOME/lab1/batch`.
- Compile the simple “hello world” fortran or C MPI code

```
% mpif90 mpihello.f90 -o mpihello
OR
% mpicc mpihello.c -o mpihello
```
- Look over the “job” script, and submit the program to SGE

```
% qsub job
```
- Watch the status of your job (you may have to be quick to see anything!)

```
% watch showq -u -l (Ctrl-C to quit watching)
```
- Now, put a “sleep 60” statement in the jobs script, resubmit it, & `qdel` the job. (Edit the job file with `emacs/vi`, see provided reference if needed.)

```
% qsub job (observe the returned jobid or determine it from bjobs)
% qdel jobid
```



7



Precision

- Look over the `precision.f` program in the **precision** directory.

```
% cd $HOME/lab1/precision
```
- The program computes prints $\sin(\pi)$. The π constant uses “E” (single precision) format in one case and “D” (double) in the other.

```
% pgf90 precision.f90
(or)
% ifort precision.f90
% ./a.out
```

- By default, Fortran compilers treat “.f” files as F77 fixed format programs.
- The `-FR` option specifies that the file is free format.



8



Makefiles

- `cd` to the `using_makefiles` directory.
- Read over the Makefile.
- Compile the program (this generates `a.out`)
 `% make`
- Simulate a change to one of the source files.
 `% touch suba.f`
- Execute `make` again. Only `suba.f` should be recompiled!



9



Library Generation

- Generate a library (a collection of object files) with the `ar` command.
 `% ar rv libsub.a suba.o subb.o`
- Now use the library when building a new `a.out`.
 `% ifort -FR prog.f libsub.a`
- All system software uses (`.a` or `.so`) libraries.
- You can link system libraries into your personal code in the same manner.



10

