Shell environments and Scripting

Leonie Chevalier and Manodeep Sinha

Swinburne University

What is a SHELL?

shell is what interprets the commands at the command-line prompt. The operating system actually executes the command

What does SHELL do for you?

- "name" resolution

- Is "cd" a command for the OS or something else

- If OS command, multiple versions might exist — which one to actually run

Setting up the SHELL Environment

- Different kinds of SHELLs, bash, csh, tcsh, zsh…

- Initial setup via .bashrc/.cshrc/.tcshrc/.zshrc

- But wait - gotchas

- .bash_profile executes on login shells (where you type a login/passwd and interact)

- .bashrc executes on new terminals and non-interactive sessions (e.g., scp)

Use one file to rule them all (e.g., .bashrc here)

```
# contents of .bash_profile

if [ -f ~/.bashrc ]; then

        .  ~/.bashrc

fi
```

Contents of .bashrc

```
# Source global definitions

if [ -f /etc/bashrc ]; then

        . /etc/bashrc

fi

# your settings follow
```

Loading modules on supercomputers

```
if [[ $- == *i* ]];  Only in interactive shells

then

    module load intel/2018.1.163-gcc-6.4.0

    module load  gcc/7.3.0 git/2.16.0 gsl/2.4

    module load python/3.6.4 numpy/1.14.2-python-3.6.4

endif
```

Setting aliases

- Frequent, long commands (or mis-typed)

- alias c="/usr/bin/clear"

- alias cdd="cd"

- alias cdc="cd"

Useful aliases

- alias pm="ps aux | grep $USER" (all your running processes)

- alias back='cd "$OLDPWD"' (go back to the last directory you were in)

- alias lh="ls -lth -G|head" (show the top 10 most recently modified files)

Setting the command prompt

- PS1='[\w @\h]' (current dir, name of computer)

Setting Paths

- export PATH=$HOME/bin: $PATH

- export TEXINPUTS=.:$HOME/latex:$TEXINPUTS

- export IDL_PATH=.:+$HOME/codes/utils/idl:$IDL_PATH

- etc..

- export EDITOR = "emacs"

Other tips?

- Your tip here

bash scripting

```bash
#!/usr/bin/bash
filecounter=$(ls N4526_dist_*.1.in | wc -l)
iterations=$(ls N4526_dist_1.*.in | wc -l)

for i in `seq 1 $filecounter`
do
    if [ ! -d prob_N4526_$i ]; then
        mkdir -p  prob_N4526_$i
    fi
    for j in `seq 1 $iterations` # number of iterations
    do
        ./gmm N4526_dist_$i.$j.in 0 0.75 1.1 > temp.text
        cp temp.text prob_N4526_new/gmm.out.$i_$j
        tail -1 temp.text >> gmm.N4526_$i.results

        mv peakprob.out peakprob.out.$i.$j

        rm temp.text
    done
    mv peakprob.out.$i* prob_N4526_$i
done
```