

Shell Scripting

Jim Prewett

2014-03-28 Fri

Outline

Intro to Computing at CARC
Introduction to shell scripting

What are "Shell Scripts"?

- ▶ Shell scripts are simple programs that run shell commands for you.
 - ▶ Each line has a command(s) as if they were run on the command-line.
- ▶ They encode a process to perform some task.
 - ▶ You don't have to constantly re-type the same set of commands!
- ▶ Shell scripts can be run by the batch scheduler on our supercomputers.

Hello World

- ▶ Here is a very simple shell script that just simply prints "Hello World"

```
#!/bin/sh
# This line is a comment
echo Hello World # This is also a comment
```

Output Substitution

- ▶ You can use backquotes to specify that the output of a command be substituted into the command-line:
- ▶ Here we print a hello message including the current date/time:

```
echo Hello, it is now `date`
```

- ▶ Produces this:

```
Hello, it is now Mon Mar 31 11:28:12 MDT 2014
```

Pipelines

- ▶ Output from one command can be used as the input to another command using a "pipe".

- ▶ Count the number of files in the current directory:

```
ls | wc -l
```

- ▶ List PBS Output and Error files in the current directory:

```
ls | grep '\.[oe][0-9]'
```

- ▶ Count the PBS Output files in the current directory:

```
ls | grep '\.o[0-9] | wc -l'
```

Job Control

- ▶ Commands can be put into the "background".
 - ▶ This frees the shell to do other tasks.
- ▶ To put a job into the background when it is first run, simply add an & character at the end of the command:
 - ▶ eg.
- ▶ To return the job to the foreground, run `fg`.
- ▶ To wait for any backgrounded jobs to complete, run `wait`.
- ▶ To start 4 processes on a node you could run this:

```
myjob.exe &  
myjob.exe &  
myjob.exe &  
myjob.exe &  
wait
```