

# using rules

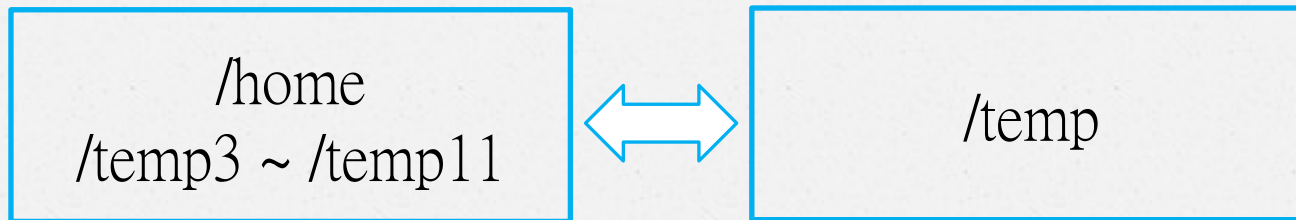
comet08

# architecture and using rules

- o 140.112.22.230

server

node3-1 ~ 3-40



- o quota:

- o node3-1,3-2,3-27: thermodynamic group only

- o notice:

- o make sure the characteristic of your program!

# Linux - Advanced

2012/01/31 蔡佩紋

# Outline

- o Input & Output
- o Tips
- o Script



# Input & Output

# Standard I/O



```
[virianty@comet08 ~/example]$ ./IOexample  
Please Enter a Number: 5  
positive  
Please Enter a Number: -3  
negative  
Please Enter a Number: 0  
zero
```

Input: 5, -3, 0

output: positive, negative, zero, Please Enter a Number:

# redirect

o < : standard input

o > : standard output

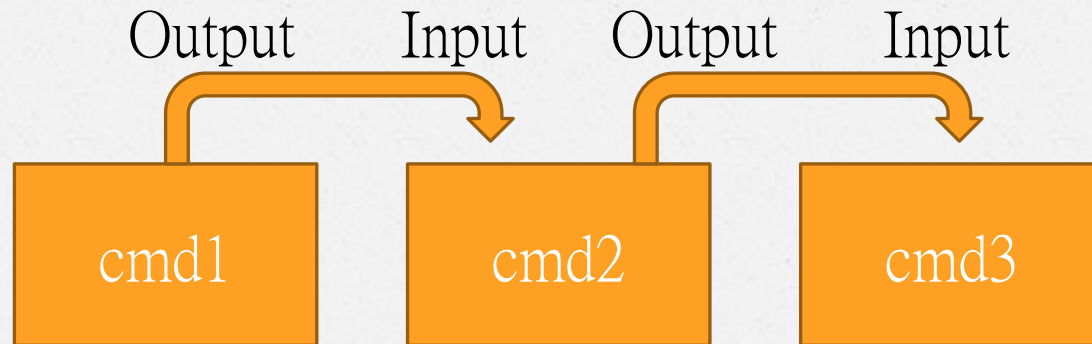
```
[virianty@comet08 ~/example]$ cat IOinput  
5  
-3  
0
```

```
[virianty@comet08 ~/example]$ ./IOexample < IOinput  
Please Enter a Number: positive  
Please Enter a Number: negative  
Please Enter a Number: zero
```

```
[virianty@comet08 ~/example]$ ./IOexample < IOinput > IOoutput  
[virianty@comet08 ~/example]$ cat IOoutput  
Please Enter a Number: positive  
Please Enter a Number: negative  
Please Enter a Number: zero
```

# pipe

- o take the standard output from last command to the standard input of this command
- o `cmd1 | cmd2 ...`
- o `|` (shift+\)







Tips

# Convenient Commands

- o `ctrl + a` - move to the start of command line
- o `ctrl + e` - move to the end of command line
- o `ctrl + d` / `tab`
  - check the content in directory
- o History
  - Review the commands you have used
- o `!`
  - Query the used commands

```
[virianty@comet08 ~]$ history
 1038  8:24  mkdir test
 1039  8:24  rmdir test
 1040  8:25  history
[virianty@comet08 ~]$ !m
mkdir test
[virianty@comet08 ~]$ !1039
rmdir test
```

# Command - sed

- o a: add new line after target line
- o i: insert new line before target line
- o c: substitution
- o d: delete
- o p: print
- o s: replacement

```
[virianty@comet08 ~/example]$ cat -n welcome
1  Woah, ohh
2
3  Welcome to the greatest show
4  Greatest show on earth
5  You've never seen before
6  Here the fairytale unfolds
7
8  What's behind the smoke and glass?
```

```
[virianty@comet08 ~/example]$ sed '4,5d' welcome
Woah, ohh
[virianty@comet08 ~/example]$ sed s/Welcome/welcome/g welcome
Woah, ohh
welcome to the greatest show
what's behind the smoke and glass?
Painted faces, everybody wears a mask
Are you selling them your soul?
```

# Command - cut

- o cut out selected portions of each line of a file
- o `cut -d '[symbol of serperation]' -f [columnum]`

```
[virianty@comet08 ~/example]$ ll
total 16
-rw-r--r-- 1 virianty student 234 2012-01-31 08:59 example.c
-rwxr-xr-x 1 virianty student 7026 2012-01-31 08:59 IOexample
-rw-r--r-- 1 virianty student 808 2012-01-31 08:40 welcome
[virianty@comet08 ~/example]$ ll | cut -d ' ' -f 3,4

virianty student
virianty student
virianty student
```

# Comment - awk

- o pattern-directed scanning and processing language
- o awk 'cond1{cmd1} cond2{cmd2}' [filename]

```
[virianty@comet08 ~/example]$ ll
total 16
-rw-r--r-- 1 virianty student 234 2012-01-31 08:59 example.c
-rwxr-xr-x 1 virianty student 7026 2012-01-31 08:59 IOexample
-rw-r--r-- 1 virianty student 808 2012-01-31 08:40 welcome
[virianty@comet08 ~/example]$ ll | awk '{print $3" is "$4}'
is
virianty is student
virianty is student
virianty is student
```

# Command - source

- o execute commands written in a file

```
[virianty@comet08 ~/example]$ cat cmd
mkdir test1
mkdir test2
mkdir test3
mkdir test4
mkdir test5
mkdir test6
mkdir test7
```

```
[virianty@comet08 ~/example]$ ls
cmd example.c IOexample welcome
[virianty@comet08 ~/example]$ source cmd
[virianty@comet08 ~/example]$ ls
cmd          IOexample  test2  test4  test6  welcome
example.c   test1     test3  test5  test7
```

# Symbolic link

- o a special type of file that contains a reference to another file or directory
- o `ln -s [src] [des]`
- o `unlink [des]`  
p.s. no `'/`

```
[virianty@comet08 ~/example]$ ln -s /temp6/virianty/MD_Class/final_project/ course
[virianty@comet08 ~/example]$ ls
cmd      example.c  test1  test3  test5  test7
course  IOexample  test2  test4  test6  welcome
```

# Personal Environment

- o About Shell
  - o tcsh – default in cluster comet08
  - o bash
  - o ~/.cshrc or ~/.bashrc
- o About vim
  - o ~/.vimrc



# shell setting

- o \$path - linux determines the executable search path with \$path
  - o ex: set path = ~/bin \$path
- o \$USER - determined by user
- o alias

```
5 set path = ( ./ ~/bin ~/bin/$HOSTTYPE ~/lib/$HOSTTYPE $path )
6
7 #my own settings
8 setenv C3_RSH 'ssh -x'
9 alias ssh 'ssh -X'
10 alias rm 'rm -i'
11 alias myjob 'qstat -nu $USER'
12 alias nmyjob 'showq -u $USER'
```

~/.cshrc

# vim setting

- o line number
- o Color
- o Tab

```
2
3 " line number
4 set number
5
13 " color setting
14 syntax on
15 set background=dark
```

~/vimrc

# search file

- o which
  - o executable program
- o locate
  - o general file

# script

tosh & bash

# Usage

o `chmod +x filename.sh`

`bash`

o `sh filename.sh`

`tcsh`

o `csh filename.sh`

# First Example

bash

tcsh

```
#!/bin/bash
```

```
#!/bin/tcsh
```

```
echo "Hello"
```

```
echo "Hello"
```

# variable and I/O

bash

```
name=test
```

```
echo $name
```

```
read name
```

```
echo $name
```

tcsh

```
set name = test
```

```
echo $name
```

```
set name = $<
```

```
echo $name
```

# calculate

bash

```
x=1
```

```
x=${x+1}
```

```
echo $x
```

tcsh

```
set x = 1
```

```
@ x = $x + 1
```

```
echo $x
```



# if

bash

```
if [ cond1 ]; then  
    cmd1  
elif [ cond2 ]; then  
    cmd2  
else  
    cmd3  
fi
```

tcsh

```
if ( cond1 ) then  
    cmd1  
else if ( cond2 ) then  
    cmd2  
else  
    cmd3  
endif
```

# condition

| bash | tcsh |
|------|------|
| -eq  | ==   |
| -ne  | !=   |
| -gt  | >    |
| -lt  | <    |
| -ge  | >=   |
| -le  | <=   |

# for loop

bash

```
array=(Ho1019 LEEBS afjl)
```

```
for member in ${array[*]}
```

```
do
```

```
    echo "$member is member of comet"
```

```
done
```

tosh

```
set array = (Ho1019 LEEBS afjl)
```

```
foreach member($array)
```

```
    echo "$member is member of comet"
```

```
end
```

# save standard output of executable file

bash

```
files=`ls`  
echo $files
```

tosh

```
set files = `ls`  
echo $files
```

p.s. ` (below ~)

# Q&A

鳥哥