



Introductory Note 2

UNIX Shell Commands Reference Card

Robert Evans

10th November, 2001

Copyright ©1998,2000,2001 R.Evans.

Email: Robert.Evans@cs.cf.ac.uk

Abstract

The *Shell* is the command interpreter on UNIX systems. This Note introduces some of the basic features of the Shell and lists many of the commands or programs available on the UNIX systems in the Department.

Contents

1 The Shell	1
2 Command Summary	5
2.1 Logging out	5
2.2 Files and Directories	5
2.2.1 Commands for accessing floppy disks	5
2.3 File Editors	5
2.4 Manipulating data	5
2.5 Manipulating data (cont'd)	6
2.6 Compressed files	6
2.7 Information	6
2.8 Status	6
2.9 Printing	7
2.10 Messages between Users	7
2.11 Network News	7
2.12 Networking	7
2.13 Programming	8
2.13.1 General	8
2.13.2 C	8
2.13.3 C++	8

2.13.4 JAVA	8
2.13.5 FORTRAN	8
2.13.6 Prolog	8
2.13.7 Other Languages	8
2.14 Text Processing	8
2.14.1 General Commands	8
2.14.2 Troff	9
2.14.3 TeX	9
2.15 Word Processing	9
2.16 Database Management	9

1 The Shell

The UNIX command interpreter or *shell* is the program you interact with when you log into a Sun workstation and start a terminal window, or when you log in to a multi-access Sun UNIX system over the Internet via a terminal emulator such as `telnet` or `putty`,

The default login shell for users in the Computer Science Department is the T-Cshell (`tcsh`). It prompts you with a percent symbol (%) preceded by an identification string. There are other shells available. They all have similar characteristics but each has its own particular features. The T-Cshell is an extended version of the (`csh`) C-Shell interpreter. This Note assumes you are using the T-Cshell.

The T-Cshell has the following features:

- commands are invoked by naming them. Most UNIX commands are simply programs which are run by the shell. For example,

```
scmabc-% ls
```

runs the program `ls` which reads your directory and lists the name of your files.

- When you type a command name, the shell will search a set of directories until it finds the program. This set is known as the *search path*.

The search path includes your current directory and one or two directories in your home directory. You can write your own programs and invoke them automatically (by naming them) from your current directory, or from subdirectories "bin" or "solarisbin" in your home directory. The T-CShell keeps a hash table of its path to speed-up access. You need to type the command **rehash** to update the table if you write a new command and put it in "bin" or "solarisbin".

- commands often have *argument* strings which may, for instance, represent filenames. E.g.

```
scmabc-% cp fileA fileB
```

is the copy command `cp` with two filename arguments; "fileA" is copied to a new file, "fileB".

Some commands have *flag* argument strings usually beginning with a "-". The flags modify the behaviour of the program being invoked:

```
scmabc-% ls -lt
```

makes `ls` give a long listing of the files sorted by time of creation.

- the shell will expand *wildcards* to match filenames in the current directory. For example,

```
scmabc-% ls -l *.c
```

will give a directory listing of the files with names “*something.c*” (conventionally C program source files).

- most UNIX commands and programs adhere to a concept of *standard input* and *standard output*. The standard input is a stream of data which the program reads and the standard output is a stream of output written by the program. Often these are both attached to the terminal so that input comes from your keyboard and output goes to your screen. The shell lets you *redirect* the standard input and output.

The symbol “<” is used to redirect standard input from a file and the symbol “>” is used to redirect standard output to a file. For example:

```
scmabc-% cat < fileA
```

makes `cat` read from file “fileA”. It sends its standard output to the terminal or screen.

```
scmabc-% cat < fileA > fileB
```

reads from “fileA” and writes to “fileB”.

- the Shell has the facility to *pipe* the output of one program to the input of another. The pipe symbol is “|”. For example:

```
scmabc-% ls | wc -w
```

pipes the output of `ls` (viz., your filenames) into the word count program `wc`. The “-w” flag tells `wc` to count the number of words in its input. So the above command counts the number of files in your directory.

- You may assign *aliases* for commands or groups of commands:

```
scmabc-% alias xx exit
```

sets up an alias “xx” to stand for the command `exit`.

- the shell has string and numeric valued variables.

```
scmabc-% set x = "hello world"
scmabc-% echo $x
```

prints “hello world” on the screen. Some variables are pre-set, e.g. `$home` is your home directory. Type `set` to see a list of assigned variables. The symbol “~” can also be used to refer to your home directory.

- the T-Cshell is an interpretive programming language with *while* loops, *foreach* loops, *if-then-else* statements, etc. See the Sun workstation on-line documentation for more details.

- scripts of shell commands can be written. These can be invoked in the same way as compiled programs (i.e. just by naming them). For example:

```
scmabc-% cat > ~/bin/countfiles
#!/bin/csh
ls | wc -w
^D
scmabc-% chmod +x ~/bin/countfiles
```

creates a C-Shell script file in your "bin" directory. The `chmod` command changes its protection mode to make it *executable*.

The first line of the script tells UNIX that the script is written in the C-Shell language (UNIX scripts can be written in any language), while the second line tells the system to run the directory listing command, `ls`, and pipe its output to the word count program, `wc`.

```
scmabc-% rehash
```

tells the shell to make a new table of the files on its search path and now

```
scmabc-% countfiles
```

will execute the script and output the number of files in your directory,

- the shell has "job control". Programs which don't require any terminal interaction can be run in the background.

```
scmabc-% sort bigfile > sortedfile &
scmabc-%
```

The "&" puts the `sort` program into the background and the Shell is available immediately for other commands.

The special character "Z" can be used to suspend a program which is running in the foreground:

```
scmabc-% sort bigfile > sortedfile
scmabc-% ^Z
Stopped
scmabc-%
```

You may now use `bg` to put the program into the background or `fg` to continue it in the foreground. The command

```
scmabc-% jobs
```

lists the status of all stopped or background jobs along with a reference number (1,2,3...). Use this number preceded by a "%" to make `bg` or `fg` act on a particular job. If a background job needs terminal input, it will stop until you bring it into the foreground.

- the shell has a history mechanism - it remembers the last few commands.

```
scmabc-% history
```

lists the remembered commands along with a reference number. On a workstation, you can cut and paste from the history to rerun a command. You can also use the symbol "!" to rerun any command from the history:

```
scmabc-% !23
```

reruns command number "23" from the history.

```
scmabc-% !so
```

reruns the last command starting "so...".

```
scmabc-% !!
```

reruns the last command.

See the manual page on the C-shell for more details (type `man csh`). The T-Cshell has an additional mechanism which allows you to recall and edit previous commands using the keyboard cursor keys. See the manual page on the T-Cshell (`man tcsh`) for instructions.

2 Command Summary

Here is a summary of some of the commands available. For more details refer to the manual page in Section 1 of the UNIX Reference Manual. You can see these online by using the `man` command. Just type `man` followed by the name of the command you want to see.

2.1 Logging out

`logout` - log off UNIX

Note, on the Sun SPARCStations or from a PC CDE session you will need to exit the Common Desktop Environment instead, see Introductory Note 3.

2.2 Files and Directories

These commands allow you to create directories and handle files.

`cat` - concatenate and print data
`cd` - change current directory
`chgrp` - change file group
`chmod` - change file mode
`cp` - copy file data
`file` - determine file type
`find` - find files
`grep` - search file for regular expression
`head` - give first few lines
`just` - text justification program
`lpq` - spool queue examination program
`lpr` - spool file for line printing
`lprm,`
`cancel` - remove jobs from line printer queue
`ls` - list and generate statistics for files
`mkdir` - make a new directory
`more,`
`page` - display file data at your terminal
`mv` - move or rename files
`pwd` - print working directory
`rm,`
`rmdir` - remove (unlink) files or directories
`tail` - print last lines from file
`touch` - update access and modification times of a file

2.2.1 Commands for accessing floppy disks

The `mtools` commands are for accessing MSDOS disks.

`mcopy` - copy to/from floppy disk
`mmdir` - list directory of floppy disk
`mcd` - change MSDOS directory
`mdel` - delete an MSDOS file

2.3 File Editors

Editors are used to create and amend files.

`emacs` - GNU project Emacs
`ex, edit` - line editor
`nedit` - easy-to-use GUI text editor
`xemacs` - emacs with mouse action
`dtpad` - Workstation CDE text editor
`pico` - easy text editor for `vdus`
`vi` - standard text editor

`Vi`, `pico` and `emacs` are screen-based editors which run on a `vdu` or in a workstations terminal emulation window; `dtpad`, `nedit` and `xemacs` are graphical user interface (GUI) -based editors with cut and paste, mouse-controlled cursor positioning etc.

2.4 Manipulating data

The contents of files can be compared and altered with the following commands.

`awk` - pattern scanning and processing language
`cmp` - compare the contents of two files
`comm` - compare sorted data
`cut` - cut out selected fields of each line of a file
`diff` - differential file comparator
`expand,`
`unexpand` - expand tabs to spaces, and vice versa
`gawk` - pattern scanning and processing language

2.5 Manipulating data (cont'd)

join	- join files on some common field
look	- find lines in sorted data
perl	- data manipulation language
paste	- merge file data
sed	- stream text editor
sort	- sort file data
split	- split file into smaller files
tr	- translate characters
uniq	- report repeated lines in a file
wc	- count words, lines, and characters

2.6 Compressed files

Files may be compressed to save space. Compressed files can be created and examined.

compress	- compress files
uncompress	- uncompress files
zcat	- cat a compressed file
zcmp,	- compare compressed files
zdiff	
zmore	- file perusal filter for crt viewing of compressed text
gzip	- GNU alternative compression method
gunzip	- uncompress gzipped files

2.7 Information

Manuals and documentation are available on-line. The following commands give information.

answerbook2	- invoke netscape to view for Sun documentation
applications_doe	invoke netscape to view applications pages
apropos	- locate commands by keyword lookup
dthelpview	- CDE Workstation help viewer
intro_doc	- display Intro Notes in netscape
man	- displays manual pages online

manpage	- displays manual pages in netscape
whatis	- describe a command
netscape	- World Wide Web information viewer for workstations

2.8 Status

These commands list or alter information about the system.

chfn	- change your finger entry
date	- print the date
determin	- automatically find terminal type
du	- print amount of disk usage
finger	- print information about logged-in users
groups	- show group memberships
homequota	- show quota and file usage
iostat	- report I/O statistics
kill	- send a signal to a process
last	- show last logins of users
lun	- list user names or login ID
netstat	- show network status
passwd	- change your login password
printenv	- display value of a shell variable
ps	- print process status statistics
quota -v	- display disk usage and limits
reset	- reset terminal mode
script	- keep script of terminal session
stty	- set terminal options
time	- time a command
tset	- set terminal mode
tty	- print current terminal name
uptime	- display system status
users	- print names of logged in users
vmstat	- report virtual memory statistics
w	- show what logged in users are doing
who	- list logged in users

2.9 Printing

Files can be printed using shell commands, using the CDE print manager, or direct from some applications.

You must specify a printer by name. Printers are called

tl3	Teaching Lab 3 (C/2.08) dot matrix printer
tl3_lw	Teaching Lab 3 laser printer
tl2_lw	Teaching Lab 2 (C/2.05) laser printer
tl1_lw	Teaching Lab 1 (C/2.04) laser printer

Most commands which can be used to print files, expect the printer name to be given following a `-P` argument.

Files may be sent to the printers as simple text files or they may be processed in various ways for the laser printers.

- `lp -d printer` - send a file to a printer
- `lpr -Pprinter` - send a file to a printer
- `a2ps -Pprinter` - format text file in PostScript and print on laser printer
- `dvips -Pprinter` postprocess TeX file into Postscript and print on laser printer

2.10 Messages between Users

The UNIX systems support on-screen messages to other users and world-wide electronic mail.

- `pine` - vdu-based mail utility
- `elm` - alternative vdu-based mail utility
- `frm,`
`from`
`mail` - identifies sender of mail
- `mail` - simple send or read mail program
- `dtmail` - CDE mail handling tool on SPARCStations
- `mesg` - permit or deny messages
- `parcel` - send files to another user
- `talk` - talk to another user
- `write` - write message to another user

2.11 Network News

The University host `netnews.cf.ac.uk` receives the "Network News" - a bulletin board of information from users in the US, Europe and elsewhere. These commands enable you to read and subscribe to the news. The command `tin` is the recommended news handling program. It works on vdu-based systems or in a shell window on a workstation. It is a *threaded* newsreader which presents you with articles related to a particular topic one after another.

- `tin` - threaded news reader and poster
- `netscape` - web browser and news reader and poster

2.12 Networking

The Computer Science Department is connected to the JANET Internet Protocol Service (JIPS), the UK Universities' network.

These commands are used to send and receive files from Campus UNIX hosts and from other hosts on JIPS and the Internet around the world.

- `ftp` - file transfer program
- `rmp` - remote file copy
- `rlogin` - remote login to a UNIX host
- `rsh` - remote shell
- `tftp` - trivial file transfer program
- `telnet` - make terminal connection to another host
- `ssh` - secure shell terminal or command connection
- `scp` - secure shell remote file copy
- `sftp` - secure shell file transfer program
- `netscape` - web browser

(Some of these commands may be restricted for security reasons).

2.13 Programming

The following programming tools and languages are available.

2.13.1 General

dbx - Sun debugger
 workshop - integrated program development environment on SPARCStations
 runide - Java integrated development environment on SPARCStations
 make - maintain program groups
 nm - print program's name list
 size - print program's sizes
 strip - remove symbol table and relocation bits

2.13.2 C

cb - C program beautifier
 cc - ANSI C compiler for Suns SPARC systems
 ctrace - C program debugger
 cxref - generate C program cross reference
 workshop - SPARCStation development environment
 gcc GNU ANSI C Compiler
 indent - indent and format C program source

2.13.3 C++

CC - C++ compiler for Suns SPARC systems
 workshop - SPARCStation development environment
 g++ GNU C++ Compiler

2.13.4 JAVA

javac - JAVA compiler
 appletviewer JAVA applet viewer
 runide - Java integrated development environment on SPARCStations

2.13.5 FORTRAN

f77 - Fortran 77 compiler
 f90 - Fortran 90 compiler
 f95 - Fortran 95 compiler
 fsplit - split a multi-routine Fortran file
 workshop - SPARCStation development environment

2.13.6 Prolog

sicstus - Sicstus Prolog

2.13.7 Other Languages

(Not available on all systems).

bc - interactive arithmetic language processor
 gcl - GNU Common Lisp
 squeak - smalltalk
 mathematica symbolic maths package
 matlab - maths package
 perl - general purpose language
 php - web page embedded language
 asp - web page embedded language

2.14 Text Processing

Troff is the standard UNIX text formatter. TeX is also available for documents intended for a LaserPrinter.

2.14.1 General Commands

fmt - simple text formatter
 spell - check text for spelling error
 ispell - check text for spelling error
 gv - gnu PostScript previewer for workstations
 sdtimage - PostScript previewer for SPARCStations

2.14.2 Troff

eqn	- mathematical preprocessor for troff
grap	- pic preprocessor for drawing graphs
nroff	- text formatting language
pic	- troff preprocessor for drawing pictures
tbl	- prepare tables for nroff or troff
troff	- text formatting and typesetting language
groff	- GNU troff interface for laserprinting

groff can be used to invoke all the preprocessors as necessary.

2.14.3 TeX

dvips	- convert a DVI file to POSTSCRIPT
tex	- text formatting and typesetting
latex	- latex formatter
xdvi	- dvi previewer for DECStations and SPARCStations

2.15 Word Processing

StarOffice is available on the Sun workstations. This is an Office package which attempts compatibility with MS Office.

2.16 Database Management

Mysql, Oracle and informix are available on the SPARCStations.

setoracle	- set up oracle environment and path
sqlplus	- run the Oracle SQL interpreter
sqlldr	- run the Oracle SQL data loader
dtsql	- run the Oracle worksheet interface
mysql	- run the mysql SQL interpreter

Other database systems are available for research use. See your supervisor for information.