1.1 Getting Started

- Unixes
- Shells
- Your account on department and university systems
- date, who, echo, banner, cal, passwd
- stty
- the kernel
- metacharacters: erase, kill, escape
- ^-d, ^-c, ^-z, ^-s, ^-q
SENDING SOME MAIL

$ mail janet kal root<RET> ← To whom
I can't make the meeting.
I'm in a UNIX class.
How about next week. ← Message

Talk to you later,
Joe
.<RET> ← Send it!
$

Either a dot (.) or a <CTRL/d> on a line by itself will send your mail.
Figure 1.1. Architecture of UNIX Systems
FIGURE 2.1 Structure of a UNIX Command

```
Command Line
\
<table>
<thead>
<tr>
<th>Options</th>
</tr>
</thead>
<tbody>
<tr>
<td>ls -t</td>
</tr>
<tr>
<td>note1</td>
</tr>
<tr>
<td>note2</td>
</tr>
<tr>
<td>Arguments</td>
</tr>
</tbody>
</table>
```

<table>
<thead>
<tr>
<th>Section</th>
<th>Subject (SVR4)</th>
<th>Subject (Linux)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>User programs</td>
<td>User programs</td>
</tr>
<tr>
<td>2</td>
<td>Kernel's system calls</td>
<td>Kernel's system calls</td>
</tr>
<tr>
<td>3</td>
<td>Library functions</td>
<td>Library functions</td>
</tr>
<tr>
<td>4</td>
<td>Administrative file formats</td>
<td>Special files (/dev)</td>
</tr>
<tr>
<td>5</td>
<td>Miscellaneous</td>
<td>Administrative file formats</td>
</tr>
<tr>
<td>6</td>
<td>Games</td>
<td>Games</td>
</tr>
<tr>
<td>7</td>
<td>Special files (/dev)</td>
<td>Macro packages and conventions</td>
</tr>
<tr>
<td>8</td>
<td>Administration commands</td>
<td>Administration commands</td>
</tr>
</tbody>
</table>
Solaris 7 Reference Manual Collection

- man Pages(1): User Commands
- man Pages(1M): System Administration Commands
- man Pages(2): System Calls
- man Pages(3): Library Routines
- man Pages(4): File Formats
- man Pages(5): Headers, Tables and Macros
- man Pages(6): Demos
- man Pages(7): Device and Network Interfaces
- man Pages(9): DDI and DKI Overview
- man Pages(9E): DDI and DKI Driver Entry Points
- man Pages(9F): DDI and DKI Kernel Functions
- man Pages(9S): DDI and DKI Data Structures
1.2 Files and Other Common Commands

- Editors: ed, ex, vi, emacs, pico
- ls
- cat, pr, more
- mv, cp, rm
- file names
- miscellaneous useful commands: wc, grep, sort, tail, cmp, diff
<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>ls</strong></td>
<td>list names of all files in current directory</td>
</tr>
<tr>
<td><strong>ls filenames</strong></td>
<td>list only the named files</td>
</tr>
<tr>
<td><strong>ls -t</strong></td>
<td>list in time order, most recent first.</td>
</tr>
<tr>
<td><strong>ls -l</strong></td>
<td>list long: more information; also ls -lt</td>
</tr>
<tr>
<td><strong>ls -u</strong></td>
<td>list by time last used; also ls -lu, ls -lut</td>
</tr>
<tr>
<td><strong>ls -r</strong></td>
<td>list in reverse order; also -rt, -rlt, etc.</td>
</tr>
<tr>
<td><strong>ed filename</strong></td>
<td>edit named file</td>
</tr>
<tr>
<td><strong>cp file1 file2</strong></td>
<td>copy file1 to file2, overwrite old file2 if it exists</td>
</tr>
<tr>
<td><strong>mv file1 file2</strong></td>
<td>move file1 to file2, overwrite old file2 if it exists</td>
</tr>
<tr>
<td><strong>rm filenames</strong></td>
<td>remove named files, irrevocably</td>
</tr>
<tr>
<td><strong>cat filenames</strong></td>
<td>print contents of named files</td>
</tr>
<tr>
<td><strong>pr filenames</strong></td>
<td>print contents with header, 66 lines per page</td>
</tr>
<tr>
<td><strong>pr -n filenames</strong></td>
<td>print in n columns</td>
</tr>
<tr>
<td><strong>pr -m filenames</strong></td>
<td>print named files side by side (multiple columns)</td>
</tr>
<tr>
<td><strong>wc filenames</strong></td>
<td>count lines, words and characters for each file</td>
</tr>
<tr>
<td><strong>wc -l filenames</strong></td>
<td>count lines for each file</td>
</tr>
<tr>
<td><strong>grep pattern filenames</strong></td>
<td>print lines matching pattern</td>
</tr>
<tr>
<td><strong>grep -v pattern filenames</strong></td>
<td>print lines not matching pattern</td>
</tr>
<tr>
<td><strong>sort filenames</strong></td>
<td>sort files alphabetically by line</td>
</tr>
<tr>
<td><strong>tail filename</strong></td>
<td>print last 10 lines of file</td>
</tr>
<tr>
<td><strong>tail -n filename</strong></td>
<td>print last n lines of file</td>
</tr>
<tr>
<td><strong>tail +n filename</strong></td>
<td>start printing file at line n</td>
</tr>
<tr>
<td><strong>cmp file1 file2</strong></td>
<td>print location of first difference</td>
</tr>
<tr>
<td><strong>diff file1 file2</strong></td>
<td>print all differences between files</td>
</tr>
</tbody>
</table>
1.3 More about Files: Directories

- `pwd`
- pathnames: fully qualified vs relative
- changing and making directories: `cd`, `mkdir`, `rmdir`
1.4 The Shell

- Different shells; metacharacters
- filename shorthand
- input-output redirection $>$ $>$> $<$ $<$<
- the **time** command
- pipes $|$ processes: `ps`, `kill`, `pid`, `sleep`, `wait`, `nohup`, `nice`
- the environment: `PS1`, `PATH`, `MAIL`, `TERM`
- `stty`, `.profile`, `.login`, `export`
STANDARD INPUT, OUTPUT, AND ERROR
THE SHELL AND YOUR COMMANDS

$ banner hello world<RET>

#    ######   ####   ####
#    #      #      #      #
#    #      #      #      #
#    #      #      #      #
#    #      #      #      #
#    #      #      #      #
#    #      #      #      #
#    #######  #######  #######

#    #      #      #      #      #######
#    #      #      #      #      #      #
#    #      #      #      #      #      #
#    #      #      #      #      #      #
#    #      #      #      #      #      #
#    #      #      #      #      #      #
#    #      #      #      #      #      #
#    #      #      #      #      #      #

$ banner<RET>

Usage: banner "up to 10 char arg string"
REDIRECTING STANDARD OUTPUT

$ banner hi > bfile
$ banner there >> bfile
REDIRECTING STANDARD INPUT

$ mail sar < bfile
I/O REDIRECTION SUMMARY

command > file \hspace{1cm} \text{output of command}

\$ \text{banner hi > msg} \hspace{1cm} \text{will be stored in file}

command >> file \hspace{1cm} \text{output of command}

\$ \text{banner liz >> msg} \hspace{1cm} \text{will be appended to file}

command < file \hspace{1cm} \text{input to command}

\$ \text{mail liz < msg} \hspace{1cm} \text{will be contents of file}

- Redirection allows storage of output or use of stored data as input

- Redirection symbol always appears between a command and a file
command1 | command2

$ banner hi | mail carl<RET>

- UNIX System commands are the blocks
- Pipes are the connectors
- Output of command1 becomes input to command2
- Allows data to be processed by a sequence of commands
- Eliminates the need for temporary files
- Pipe symbol always appears between commands
FIGURE 8.2 The Three Destinations of Standard Output

FIGURE 8.3 The Three Sources of Standard Input
FIGURE 8.1  The Shell’s Interpretive Cycle

User: Command entered → $1 → Command completed

Shell: Sleeping → Waking → Waiting → Sleeping

Kernel: Scanning command → Command running
BACKGROUND EXECUTION (CONT)

$ ps<RET>
  PID  TTY      TIME  COMMAND
  222  tty35    0:04   sh
  5254  tty35   0:01   ps
  3765  tty35   0:10   find

← display process information

$ kill -9 3765<RET>

← terminate process

$ ps<RET>
  PID  TTY      TIME  COMMAND
  222  tty35    0:04   sh
  5256  tty35   0:01   ps

A process can only be terminated by the owner or super user.
```
$ ps
    PID   TTY       TIME   CMD
  4419 pts/3   0:00   ps
  4415 pts/3   0:00   sh
  4404 pts/3   0:02   csh
$ sleep 300 &
  4420
$ ps
    PID   TTY       TIME   CMD
  4420 pts/3   0:00   sleep
  4415 pts/3   0:00   sh
  4404 pts/3   0:02   csh
  4421 pts/3   0:00   ps
$ sleep 600 &
  4422
$ ps
    PID   TTY       TIME   CMD
  4420 pts/3   0:00   sleep
  4422 pts/3   0:00   sleep
  4415 pts/3   0:00   sh
  4404 pts/3   0:02   csh
  4423 pts/3   0:00   ps
$  ```