Introduction to Linux

Summer course 2003

Department of Computer Science and Engineering
The Chinese University of Hong Kong

Date: July 31, 2003 - Aug 1, 2003
Time: 10:00am - 12:30pm

Venue: Rm 924B Ho Sin-Hang Engineering Building, CUHK

`email: linux@cse.cuhk.edu.hk`

http://www.cse.cuhk.edu.hk/summercourse/linux

Table of Contents

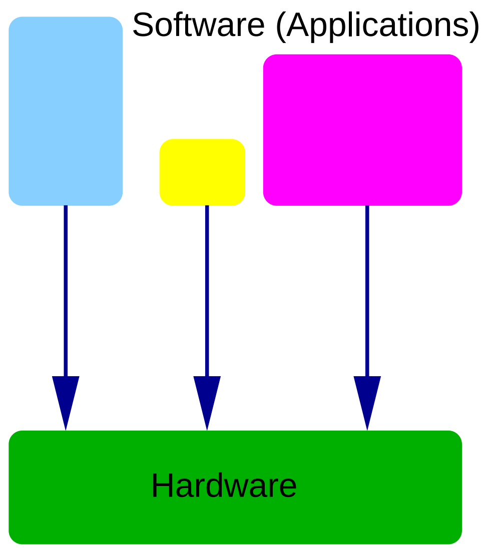
1. History
2. Philosophy
3. Unix/Linux principles
4. Basic concepts
5. Getting started
6. Working with Linux: Part I
7. Working with Linux: Part II
8. Installation highlights

Operating system

1. Every piece of software have to cater for the communication with the hardware.

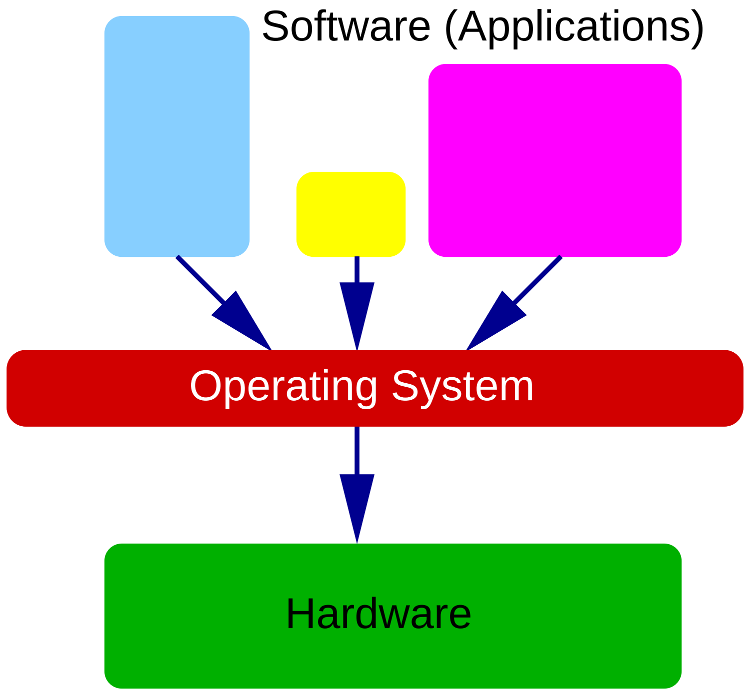
2. No one is going to relief the hardware contentions.

3. Inefficient.



Operating system

1. Housekeeper: managing hardware and serving software.



Operating system

1. Manages computing resources (*CPU, memory, I/O system, network devices, etc*)

   e.g. Scheduling of processes, swapping, paging, queuing/buffering network packets, etc.

2. To support programs that actually do the work you're interested in.

   e.g. Text editor - it needs OS support for interacting with the terminal (keyboard), filesystem, reserving memory, etc.

continue...

Operating system

continue...

1. Provide libraries for application programs to access resources managed by the OS.

2. Reference:
   - http://directory.google.com/Top/Computers/Software/Operating_Systems
   - http://www.wikipedia.org/wiki/

History

Table of Contents:

1. Unix
2. Linux
3. Free software, Open-source software

Unix

- An operating system - UNIX
- A collective term on different "flavors" of Unix.

Unix

- **Late 60s...**
- Multics project - an failed ambitious project on PDP-7
- Bell lab withdrew from Multics
- Dennis Ritchie, Ken Thompson and peoples in Bell Labs
- Reproduce a game "*space war* on PDP-7 with 4k memory for user programs
- UNICS (*UNiplexed Information and Computing Service*) - one man-month: whole OS, shell, editor, assembler, etc!

UNIX

- Dennis Ritche and Ken Thompson during 70s.

I’m Ken Thompson!

I’m Dennis Ritche!

UNIX

- PDP-11

Unix

- mid 70s...
- UC Berkeley licensed the Unix source code - improve and enhance Unix
- Develope TCP/IP
- Became BSD Unix
- Source code publicly available, redistribution without UCB approval.
- People dervied FreeBSD, NetBSD, OpenBSD, MacOS X...
- AT&T still owns the UNIX.

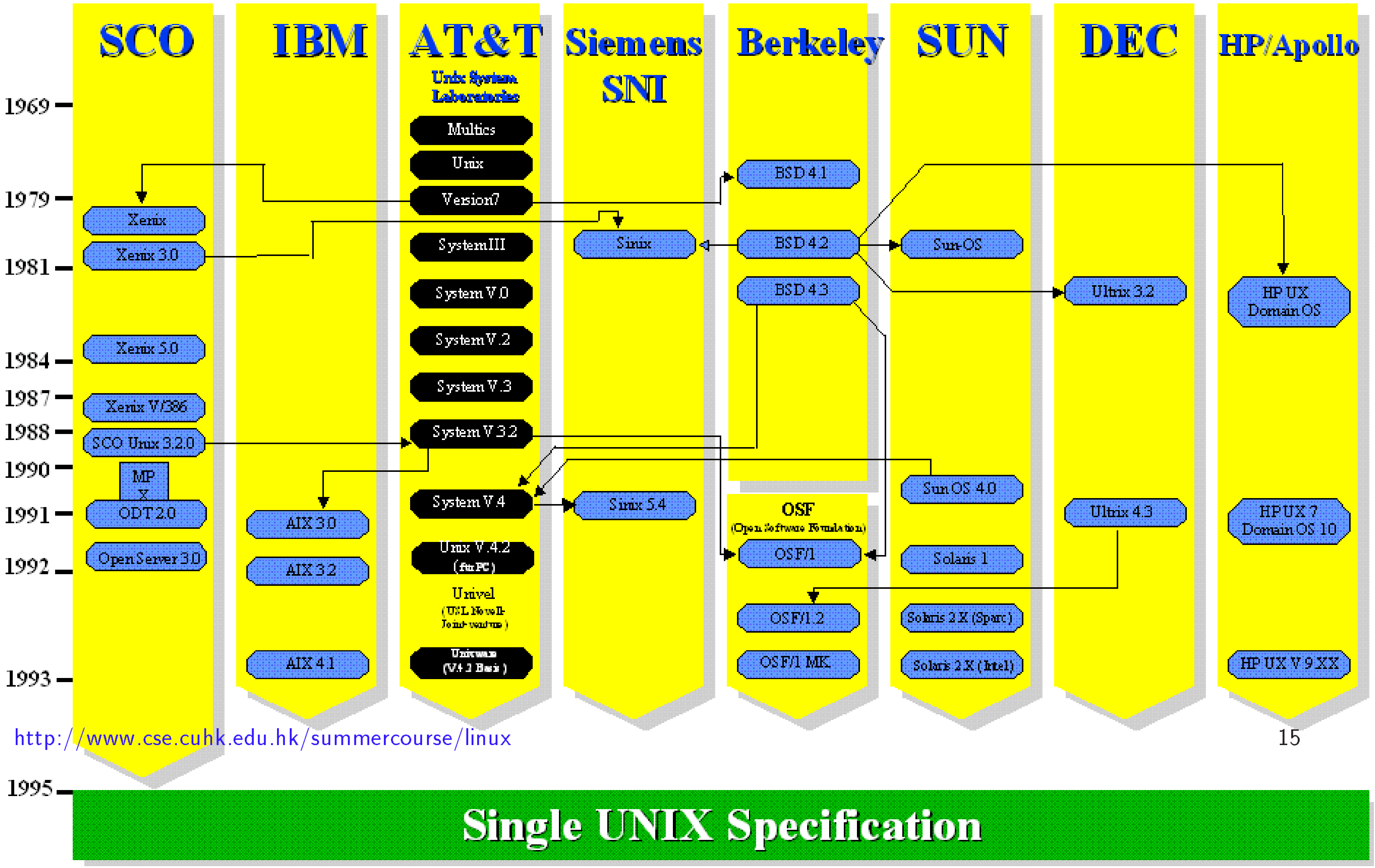
Unix

- 80s...
- Various vendors developed and marketed their own verion of Unix.
- AT&T and Sun joined to form "Unix International": AT&T System V
- Other venders joined together to form "OSF": OSF/1

UNIX

- 90s...
- Final released of BSD (4.4BSD)
- AT&T released System V release 4
- AT&T then sold UNIX to Novell
- Novell sold the Unix trademark and the specification to X/Open
- Novell sold the source code to SCO
- Late 90s, X/Open released various versions of Single Unix Specification.

Unix


SCO
IBM
AT&T
Unix System Laboratories
Siemens SNI
Berkeley
SUN
DEC
HP/Apollo
1969
1979
1981
1984
1987
1988
1990
1991
1992
1993
1995
Multics
Unix
Version7
System III
System V.0
System V.2
System V.3
System V 3.2
System V.4
Unix V.4.2 (für PC)
Univel
Unixware (V4.2 Basis)
Xenix
Xenix 3.0
Xenix 5.0
Xenix V/386
SCO Unix 3.2.0
MPX
ODT 2.0
Open Server 3.0
AIX 3.0
AIX 3.2
AIX 4.1
Sinix
Sinix 5.4
BSD 4.1
BSD 4.2
BSD 4.3
OSF
OSF/1
OSF/1.2
OSF/1 MK
Sun-OS
Sun OS 4.0
Solaris 1
Solaris 2.X (Sparc)
Solaris 2.X (Intel)
Ultrix 3.2
Ultrix 4.3
HP UX Domain OS
HP UX 7 Domain OS 10
HP UX V 9.XX
Single UNIX Specification

Unix

- Reference:
  - http://www.bell-labs.com/history/unix
  - http://internet-history.org/archives/early.history.of.unix.html
  - http://www.english.uga.edu/hc/unixhistoryrev.html
  - http://cm.bell-labs.com/cm/cs/who/dmr/hist.html
  - A Quarter Century of UNIX

    Peter H. Salus (Addison-Wesley 1994)

Unix

In http://www.unix.org/what_is_unix.html,

"UNIX - the worldwide Single UNIX Specification integrating X/Open Company's XPG4, IEEE's POSIX Standards and ISO C. Through continual evolution, the Single UNIX Specification is the defacto and dejure standard definition for the UNIX system application programming interfaces."

http://www.unix-systems.org/single_unix_specification/

http://www.unix-systems.org/version3/

UNIX

Now, UNIX is

- *not* just the original OS produced by AT&T.
- a specification defines what a "UNIX" system should be. *e.g. API (Library functions), behaviours, compiler, network protocol stack, UI (X), Internet services, etc*
- a specification to be used to develop portable applications that run on systems conforming to the Single UNIX Specification. (`http://www.unix-systems.org/version2/whitepaper.html`)
- Registered products are Solaris, AIX, Tru64 UNIX. (`http://www.opengroup.org/openbrand/register`)

Linux

- **Early 1990s**...
- Finnish CS student - Linus Torvalds
- Doing his OS assignment
- Hacking on Minix to improve its kernel.
- Developed from scratch.
- Not a Unix at all.

Linux

- **Late 1991...**
- Published a kernel of version 0.0.1 called "Linux" on newsgroup.
- Copyleft, GPL => Free and open => a community
- Continue to extend and improve
- http://www.oreilly.com/catalog/opensources/book/appa.html

Linux

- A community (hackers on Internet) effort.
- Based on industrial standard: POSIX.
- Highly modular.
- Key features: (http://www.linux.org.uk/Features.html):

  e.g. Multi-user and multi-tasking OS, virtual memory and proper memory management, advanced TCP/IP networking, SMP, dynamic linking, file ownership.

LINUX

- Owned by the community (GNU GPL)
- Source code of Linux is freely available to everyone.
- It is only the trademark "Linux" that is owned by Linus.
- `linux.com` and `linux.org` are NOT official websites.

  (*Probably* `http://www.kernel.org` *is*)

Linux

- First developed for 32-bit x86-based PCs (386 or higher).

- Now it also runs on (at least) Compaq Alpha AXP, Sun SPARC and UltraSPARC, Motorola 68000, PowerPC, PowerPC64, ARM, Hitachi SuperH, IBM S/390, MIPS, HP PA-RISC, Intel IA-64, DEC VAX, AMD x86-64 and CRIS architectures.

- Linux can't be called as UNIX, because UNIX is now *a specification*. The X/Open group charge expensively for carrying out the conformance tests.

LINUX

Working model

- Decentralized (Internet)
- Cooperation
- Various teams with team leaders

Linux: definition and history

Working model

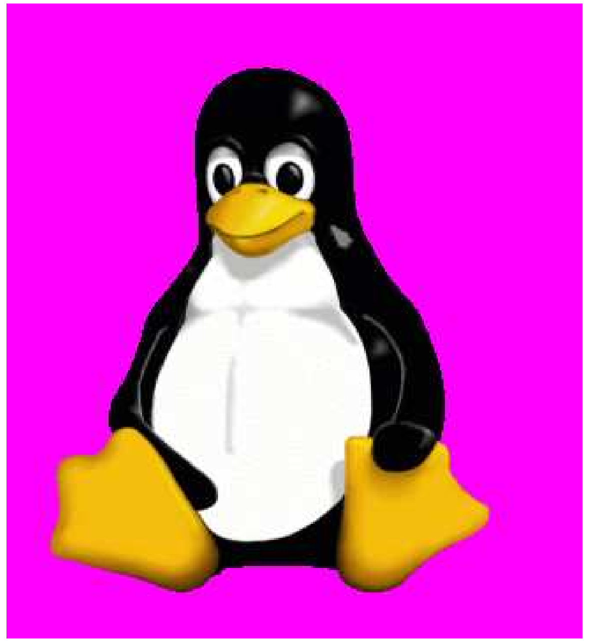
- Linus makes final decision
- Stable and development kernel
- Major patches (e.g. ac, aa, dj, mm patches)
- Vendor patches (e.g. RedHat)
- Unofficial patches

Linux: definition and history

Usage:

- Server (Internet servers, database servers)
- Multi-user workstation
- Desktop
- Embedded devices
- Network equipments

Linux: definition and history



Linux: definition and history

- Myths: http://www.geocities.com/SiliconValley/Hills/9267/fud2.html

Linux: definition and history

Timeline - http://www.cglug.org/presentation/intro_to_linux/gfx002.html

- 1984 - RMS started GNU and FSF.
- 1987 - Minix released.
- 1991 - 1) Linus posted news on comp.os.minix, 2) Linux kernel 0.01
- 1993 - Slackware: first commerical Linux distribution
- 1994 - 1) Linux kernel 1.0, 2) First release of RedHat Linux
- 1995 - RedHat Software founded
- 1996 - Linux kernel 2.0
- 1997 - RedHat Linux 4.2
- 1998 - Microsoft admited threats from Linux
- 1999 - Windows 2000 shipment, Linux kernel 2.2 series
- 2001 - Linux kernel 2.4
- 2002 - Developing Linux kernel 2.5 (3.0?)
- 2003 - RedHat 9

Proprietary software

- Early 80s, propiretary software was prevalent.
- AT&T commericalized UNIX, hence no source code is available.

Proprietary software

- No source code for the piece of software you purchase.
- You pay for the *usage* (execution) of a *single copy* of the software on a *single computer*.
- Non-tranferrable - if you don't need the software, you cannot give it away or sell it out.

Proprietary software

- You cannot (are not allowed) to modify the software to suit your needs. (What happens if you are not allowed to replace the stool in your house when it doesn't fit you?)

- Non-disclosure agreement - you have legal responsibility.

sounds ridiculous?

Proprietary software

- Over usage of software patents, *e.g. LZW algorithm (no free program to generate GIF so far!), RSA algorithm for public key, CSS (unfortunately a standard) on WWW.*
  - Patent period ususally about 20 years.
  - Patent holder may ask for loyality.
  - Many trivial and well-known features, algorithms, models and techniques can be patented - pitfalls for developers.
- Software corporates benefit from software patents in short-term:
  - Market share and monoply.
  - Loyalties.
- In long term, everyone will lose: *stagnation of innovation*

Free software, Open-source software

- A MIT (AI lab 1971) hacker called Richard Stallman (RMS) believes on a world of freedom.
- RMS was responsible for administering PDP-10 system.
- Freedom (or liberty) on studying, adapting, improving and sharing of software.
- Founded a project - GNU ("GNU"s Not Unix) project.
- GNU respects IP, all software are licensed in GPL (a concept of Copyleft).

Free software, Open-source software

- Free Software $\neq$ *freeware* $\neq$ *public-domain software.*
- It is about *freedom*, not the price.
- `http://www.gnu.org/philosophy/free-sw.html`

Free software, Open-source software

- Free software $\neq$ Open source software
- They are neither equivalent nor exclusive.
- "Open source software" is a marketing terms invented in 1998.
- GPL and Copyleft

Free software, Open-source software

- http://www.opensource.org
  http://www.gnu.org/philosophy/free-software-for-freedom.html
  http://opensource.oreilly.com
  http://tuxedo.org/~esr/writings/cathedral-bazzar
  http://www.opensource.org/advocacy/recommended.php
  http://www.oreilly.com/catalog/opensources/book/toc.html
  http://www.mitre.org/work/tech_papers/tech_papers_01/kenwood_software/index.html
  http://www.dwheeler.com/oss_fs_why.html

GPL and Copyleft

- Everyone was given the freedom to copy, distribute, and modify their copy of the software.
- No secrets.
- May charge for free software.
- No one is allowed to take away the freedeom (i.e., further restricting the license).
- The IP owner of the software has the copyright.
- http://www.gnu.org/licenses/licenses.html

GPL and Copyleft

Copyleft and the GNU GPL

- Copyleft is a copyright concept
- Copyleft uses copyright law
- Copyleft keeps software free

GPL and Copyleft

- GNU GPL (General Public License) is a specific implementation of copyleft.

- http://www.gnu.org/copyleft/gpl.html

- Controversial:

  "To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it..."

FS/OSS advantages

- Do you think Openess is important?
- Do you think Freedom is essential?

FS/OSS advantages

- Rapid bug fixes and mature releases.
- You have total control on the piece of software you owned (bought).
- You can customize it to suit your needs.
- You can hire someone to maintain and customize it for you without paying maintenance cost to the software vender.
- You will not get into trouble if the software company goes bankruptcy.

FS/OSS advantages

- You have more confidence of having a sane and backdoor-free software.
- You help in preventing wheel reinvention.
- More secure (or minimal impact due to security flaw).
- More reliable and robust software (peer-review), e.g. BIND, Sendmail, Apache HTTP, etc.
- Economical.
- Almost no legal risk.

GNU and RMS

- GNU is a project launched in 1994 by RMS.
- The purpose of this project is to create a Unix-like system composed of free software entirely.
- Almost all core utilities and programs of a Linux system are GNU software
- http://www.gnu.org
- http://www.stallman.org/rms.html
- http://www.gnu.org/gnu/thegnuproject.html
- http://www.oreilly.com/openbook/freedom/index.html

Unix/Linux core principles

- Simple
- Everthing is File
- Tools, tools, tools!
- "|" and "&"
- http://cm.bell-labs.com/cm/cs/upe
  http://www.cs.bell-labs.com/cm/cs/pearls

Distributions

- A Linux system usually comes in a "**distribution**".
- Distribution = the Linux OS kernel + tools + libraries + utilities + applications.
- Commercial (*usually with charges*) and non-commercial (*free of charge*).
- Pre-configured to specified purposes.
- Include configuration utilities and installers.
- Some of these are available for free over the internet, and some on CD-ROM.

What is Linux system anyway: Distributions

- Over 300+ different kinds of distributions.
- General-purpose distributions: e.g. RedHat Linux, Debian Linux, SuSE Linux, Mandrake, Slackware, ...
- Chinese Linux: e.g. CLE, Red-flag linux, ThizLinux, BluePoint, etc.
- Other distributions: handheld, floppy-based, network servers, Internet server, Application server, Database server, etc.
- `http://www.lwn.net/Distributions`

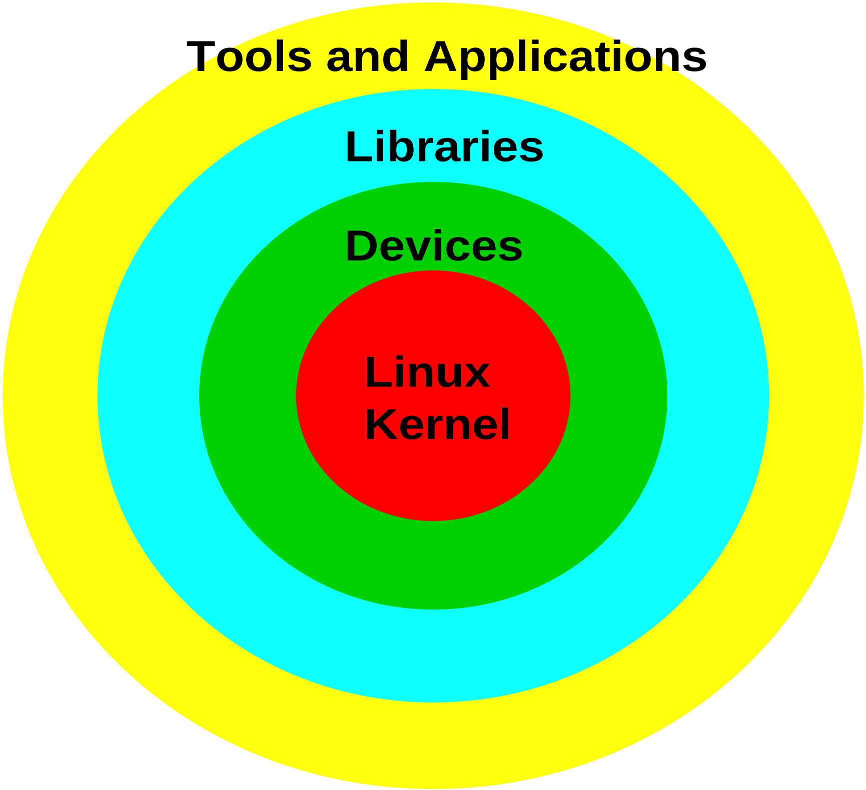
Basic concepts

- Linux is merely a kernel
- GNU/Linux system refers to an operating system built with Linux as kernel and GNU tools/utilies.
- A distribution is realization of GNU/Linux system.

  e.g. RedHat Linux, Debian Linux, SuSE Linux, etc.

  `http://old.lwn.net/Distributions`

Basic concepts



An Operating System

Basic concepts

- Multi-user system
- Can run on different hardware, x86, sparc, MIPS, etc.
- Shell
- Text-mode and Graphics mode
- Multiple Window Manager and Desktop systems

Getting started

- Booting and Shutdown
- Console and Window system
- Root and normal user
- Login and Logout

Booting

- Simple: Turn on your computer
- BIOS
- Boot loader (Lilo or GRUB)
- Load selected kernel (can be Windows)
- Various initialization (tons of messages!)
- Two possible results:
  1) Text-mode login prompt, 2) Graphical login prompt

Shutdown

- ***Never, never, never*** switch off the computer while it is running.
- Linux buffers unwritten data in memory => filesystem corruption.
- Applications may have opened files.
- Some users may be using the system.

Shutdown

1. Notifies all users and processes of the shutdown.
2. Block new logins.
3. All users logout.
4. All processes cleanly close the opened files.
5. All processes terminates in proper order.
6. Kernel flushes cached/unwritten data into disk.
7. System halts and possibly power down.

Shutdown

- Done by privileged user only, e.g. `root`
- Shutdown now:
  `shutdown -h now`
- Shutdown in 3 minutes:
  `shutdown -h -t +3 "==System halt for maintenance=="`
- Shutdown at 10:00pm: *(now is 8:30pm)*
  `shutdown -h -t 22:00`
- Reboot now:
  `shutdown -r`

Console and Window system

- Console: the terminal or monitor directly attached to the Linux system.
- Console acts as an interface between Linux and you.
- Accessing Linux through its console means that you've physical access to the system.
- Default installation of a Linux system: 63 virtual consoles.
- All virtual consoles are Console.
- Go to the virtual console 1: `Ctrl-Alt-F1`
  Go to the virtual console 2: `Ctrl-Alt-F2`, ...

Console and Window system

- Console is text-based (*in old days, there is no GUI*)

```
Debian GNU/Linux stable pc1234 tty1

pc1234 login: _
```

Console and Window system

- Graphical User Interface (GUI): Window system
- GUI for Linux: X *plus* Window manager *plus* Desktop system
- More than a GUI: a network-based client-server model.

X

- X is a network communication protocol.

  (http://www.x.org/X11_protocol.html)

- It lays a foundation for Window manager and Desktop system to work on.

- Latest release is X11R6.6

- http://www.x.org

X

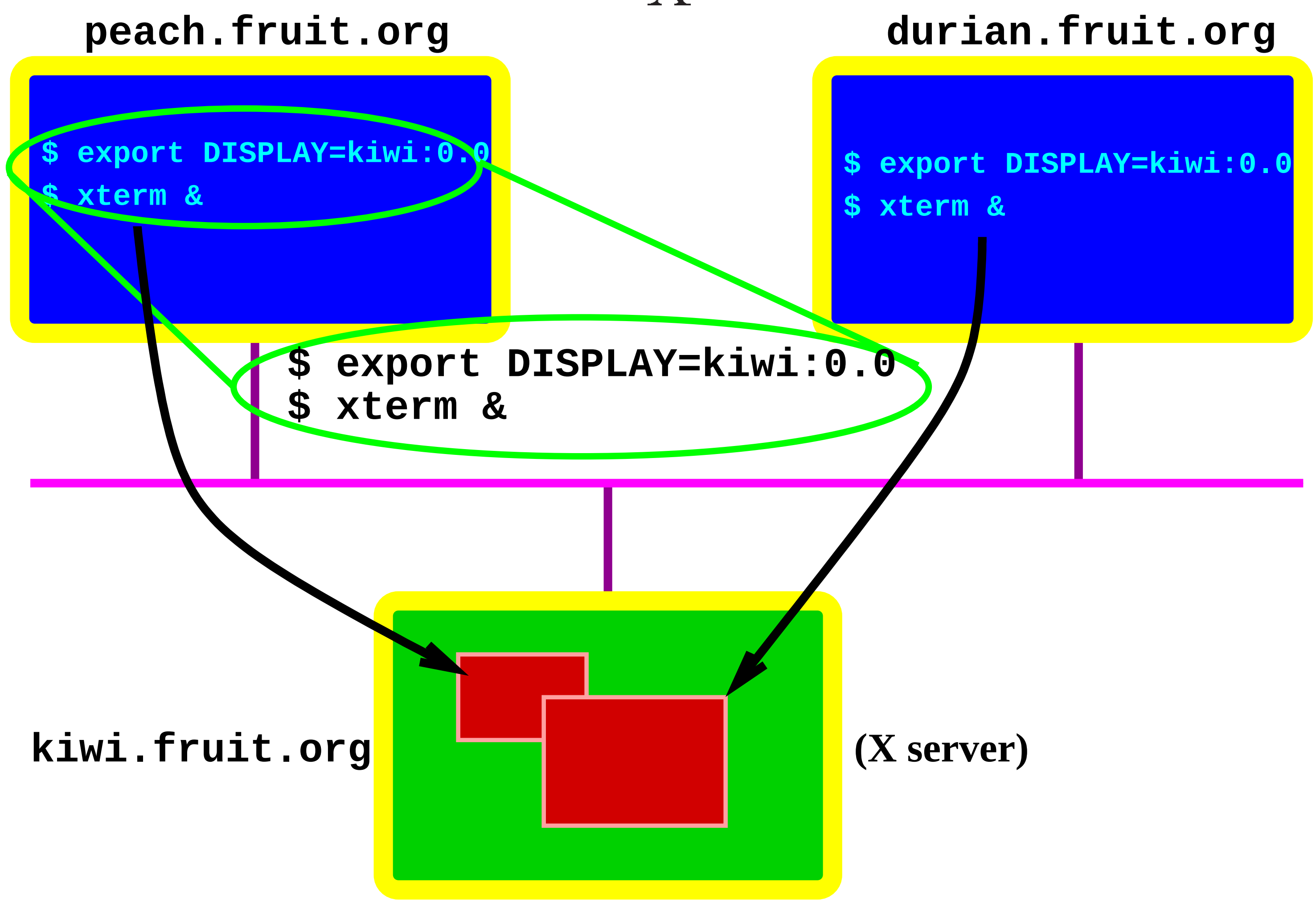
- Separate the application core computation from display processing.

  i.e., the computation for an animated 3D object can be done on one computer and the display can be shown on another computer.

X

- A PC running X is considered as a X server.
- An X application, e.g. Netscape, is a X client.
- X client connects to X server requesting for grapical displaying on the X server.
- The X server is responsible for displaying (rendering) the X client request.
- Note: X client and X server can be on <u>different PCs on the network</u>.

X


peach.fruit.org
$ export DISPLAY=kiwi:0.0
$ xterm &
durian.fruit.org
$ export DISPLAY=kiwi:0.0
$ xterm &
$ export DISPLAY=kiwi:0.0
$ xterm &
kiwi.fruit.org
(X server)

XFree86

- The X window system that runs on Linux is XFree86.
- URL: `http://www.xfree86.org`
- It provides a lot of drivers for different graphics chips for PC.

X

- ...show a naked X with a naked window.

Window Manager

- controls the way windows look and are displayed.
- provides some additional menu or program management capabilities.
- different window managers with different features and capabilities.
- allows customization.

Window Manager

- Afterstep, GNUstep, Window Maker, FVWM, Enlightenment, IceWM, MetaCity, Sawfish, etc
- `http://www.afterstep.org`
  `http://www.fvwm.org`
  `http://www.windowmaker.org`
  `http://www.enlightenment.org`
  `http://www.gnustep.org`
- `http://www.freshmeat.net` > browse > Desktop Environment > Window Manager
- ...show how to switch between window manager.

Themes

- http://freshmeat.net/articles/view/465
- http://www.themes.org
- ...show some nice themes.

Desktops

- Enrich the desktop experience of the users.
- Main contribution: Closely ties up X applications together.
- Now more than that: Their own widgets and X applications.
- Very sophisticated - X is merely a communication media for it, and Windows manager is merely a project (a element) of it.
- GNOME `http://www.gnome.org`, KDE `http://www.kde.org`
- ...show Gnome and KDE.

Login and Logout

- Multi-user system => username and password required
- Text mode login: 1) Virtual console, 2) Remote login

```
pc1234 login:  peterho
Password:

[peterho]$ _
[peterho]$ exit
```

- Logout: type "`exit`", "logout" or press "`Ctrl-D`".

Login and Logout

- GUI login: same as text mode login, username and password required.
- GUI logout: the desktop/window manager should have a "Logout" menu for this purpose.

WORKING WITH LINUX: PART I

- User activities and System activities
- Root and normal user
- Text editing
- Shell and command-line

What are we doing?

- Who am i?

```
$ whoami
peterho
```

- Who are logged in my system?

```
$ who


peterho  pts/2        Jun 17 17:25 (:0.0)
mary     pts/8        Jun  9 17:19 (:0.0)
tom      pts/16       Jul 18 10:57 (:0.0)
```

What are we doing?

- What are they doing?

```
$ w

 11:20:49 up 38 days, 21:43, 16 users,  load average: 0.02, 0.04, 0.07
USER     TTY      FROM              LOGIN@   IDLE   JCPU   PCPU WHAT
peterho  pts/2    :0.0             17Jun03  6:17   0.65s  0.51s w
mary     pts/8    :0.0             09Jun03 24:14m 14.14s 14.13s mozilla
tom      pts/16   :0.0             10Jun03  0.00s 10.93s  0.02s mutt
```

What are we doing?

- Who have been logged into my system?

  $ last

```
peterho  pts/16       :0.0              Fri Jul 18 10:57   still logged in
peterho  pts/16       :0.0              Fri Jul 18 10:57 - 10:57  (00:00)
mary     tty1                           Fri Jul 18 10:41 - 10:41  (00:00)
tom      tty2                           Fri Jul 18 10:40 - 10:43  (00:02)
root     tty6                           Fri Jul 12 23:40 - 23:43  (00:02)
```

What are we doing?

- All processes currently running in the system:

  $ ps aux

```
USER        PID %CPU %MEM   VSZ  RSS TTY      STAT START   TIME COMMAND
root          1  0.0  0.0  1212  444 ?        S    Jun09   0:00 init [2]
root          2  0.0  0.0     0    0 ?        SW   Jun09   0:26 [keventd]
root          3  0.0  0.0     0    0 ?        SWN  Jun09   0:00 [ksoftirqd_CPU0]
root          4  0.0  0.0     0    0 ?        SW   Jun09   0:51 [kswapd]
root          5  0.0  0.0     0    0 ?        SW   Jun09   0:00 [bdflush]
root          6  0.0  0.0     0    0 ?        SW   Jun09   0:00 [kupdated]
root         31  0.0  0.0     0    0 ?        SW   Jun09   0:00 [kjournald]
root         70  0.0  0.0     0    0 ?        SW   Jun09   0:00 [khubd]
root        110  0.0  0.0     0    0 ?        SW   Jun09   0:00 [kapmd]
root        140  0.0  0.0     0    0 ?        SW   Jun09   0:05 [kjournald]
root        141  0.0  0.0     0    0 ?        SW   Jun09   0:00 [kjournald]
root        142  0.0  0.0     0    0 ?        SW   Jun09   0:00 [kjournald]
root        143  0.0  0.0     0    0 ?        SW   Jun09   0:03 [kjournald]
daemon      166  0.0  0.0  1320  436 ?        S    Jun09   0:00 [portmap]
root        358  0.0  0.1  1984  744 ?        S    Jun09   0:43 /sbin/syslogd
root        361  0.0  0.0  1900  476 ?        S    Jun09   0:30 /sbin/klogd
```

What is he doing?

- All processes currently running in the system by a user:

  $ ps U peterho

```
11670 ?        S      0:00 /bin/sh /usr/local/mozilla/run-mozilla.sh /usr/local/mozilla/mozilla-bin
11691 ?        S      9:48 /usr/local/mozilla/mozilla-bin
11697 ?        S      0:06 /usr/local/mozilla/mozilla-bin
19696 ?        S      0:00 /usr/local/mozilla/mozilla-bin
26046 pts/2    S      0:00 mutt
26621 pts/9    S      0:00 [crxvt-big5]
26622 pts/16   S      0:00 bash
26627 pts/16   S      0:07 ssh emerald -X
26821 pts/9    T      0:00 man ps
```

What are running in the system?

- top
- Show list of processes running in the system.
- Refresh periodically.

Online help

- man
- info
- whatis
- apropos
- The help / usage of the command
- README files

Man

- Readm online manual

  e.g. `man ls`

Man

```
LS(1)                          User Commands                          LS(1)


NAME
       ls - list directory contents

SYNOPSIS
       ls [OPTION]... [FILE]...

DESCRIPTION
       List  information  about  the FILEs (the current directory by default).
       Sort entries alphabetically if none of -cftuSUX nor --sort.

       Mandatory arguments to long options are  mandatory  for  short  options
       too.

       -a, --all
              do not hide entries starting with .
```

Info

- Read Info documents
- A text-based navigation system
- Seemingly only GNU programs provide info documents.
- Command:

```
$ info
```

WHATIS

- Short single-line description of a command
- Command:

```
$ whatis ls

ls(1) - list directory contents
```

–HELP/-H

- Short (usage) description of a command
- Command:

```
$ ls --help
Usage: ls [OPTION]... [FILE]...
List information about the FILEs (the current directory by default).
Sort entries alphabetically if none of -cftuSUX nor --sort.

Mandatory arguments to long options are mandatory for short options too.
  -a, --all                  do not hide entries starting with .
```

Root and normal user

- Linux is a multi-user system - different users can use the system simultaneously.
- Accounts for each users.
- Associate with each account are *username*, *password*, *home directory*.

  e.g.,

  `peterho` is your username.
  `dFu2w9Ra` is your passord.
  `/home/peterho` is your home directory.

Root and normal user

- Every Linux system has a special account called "**root**".
- It is often refered as *superuser*.
- It is no different from an ordinary user account *except* that it has the highest privilege on the system.
- It can read all files in the system including those belong to ordinary users.
- It can earse all files.
- It can format harddisk.

Root and normal user

- The purpose of the "root" account is for system administration only,

  e.g. adding or deleting new users, creating new harddisk partitions, running system servers, etc.

- **Never** login as "root" unless absolutely required, otherwise you risk crashing your system!

Root and normal user

- To login as root, enter "`root`" in login prompt and then the root password.
- Alternatively, you may use "`su -`" command at the shell prompt of your account and then enter the root password.
- To identify if you have logged in as root, at shell prompt, enter `whoami` command.

Text editing

- Indisplensable
- No mark-up added to the file by the text editor.
- Essential for normal processing of many components of a Linux system.
- Used in writing documents, composing emails, making web pages, posting Usenet news, coding your C/Java programs.
- Choice of text editor: **vi**, **emacs**, **pico**, **joe**, **jed**, etc...

I use vi to write this document.

Emacs

- Developed by RMS (founder of GNU project).
- A GNU project.
- Invoke emacs:

  `$ emacs`

- Manual on web: http://www.gnu.org/manual/emacs
- Online-help: press Ctrl-h.
- It is extensible: add new functions using Lisp programs.

- International character set support.
- Many different editing modes: C program, Perl program, etc.
- Read emails, post Usenet news, compiling and debugging programs, etc.
- Calendar, diary, etc.

Vi

- Vim - Vi IMproved
- http://www.vim.org
- Online-help: type `:help` in vim.

Getting started with Vi

- Type "`vi` *filename*" at shell prompt.

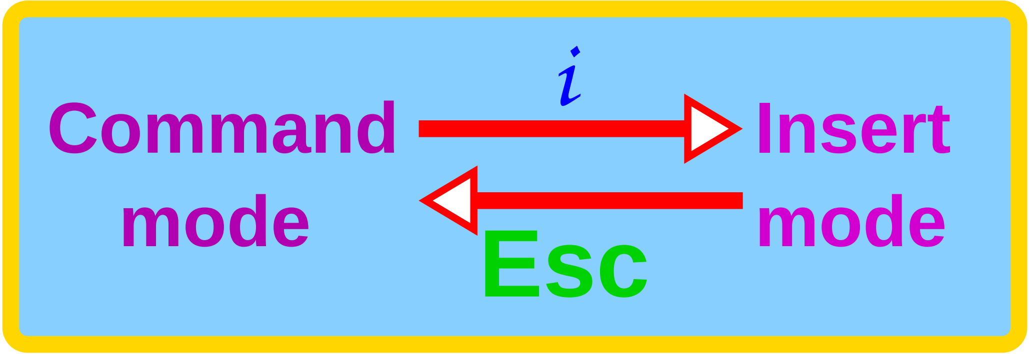
```
~
~
~
~
~
~                            VIM - Vi IMproved
~
~                             version 6.1.320
~                         by Bram Moolenaar et al.
~                 Vim is open source and freely distributable
~
~                        Help poor children in Uganda!
~               type  :help iccf<Enter>       for information
~
~               type  :q<Enter>               to exit
~               type  :help<Enter>  or  <F1>  for on-line help
~               type  :help version6<Enter>   for version info
~
~
```

Getting started with Vi

- Type ":q" to quit.
- Type ":wq" to save and then quit.
- Type ":q!" to abort the changes made to the file and then quit.
- Type ":w!" to force save the file.

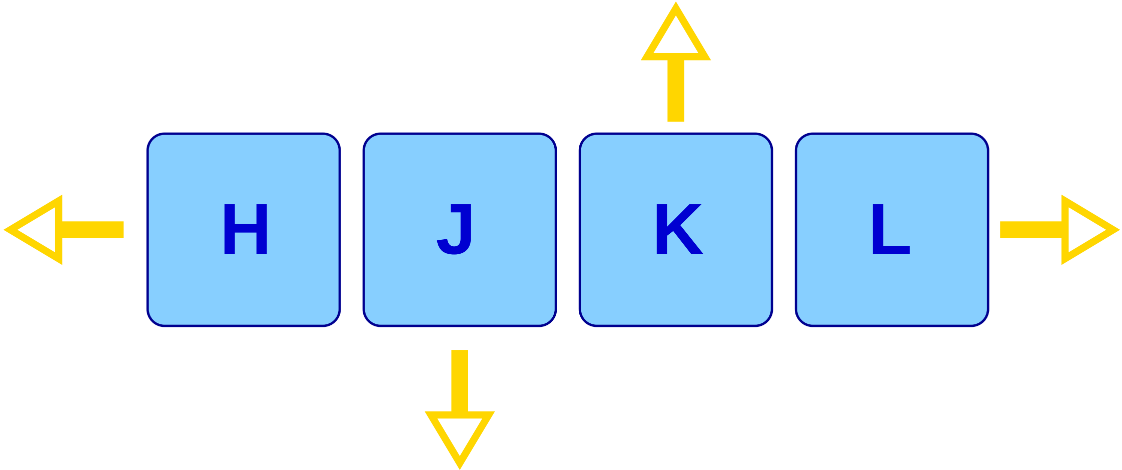
GETTING STARTED WITH VI

- Type "`i`" to begin editing (entering **Insert mode**).
- Press "*Esc*" key to stop editing for a while (entering **Command mode**).



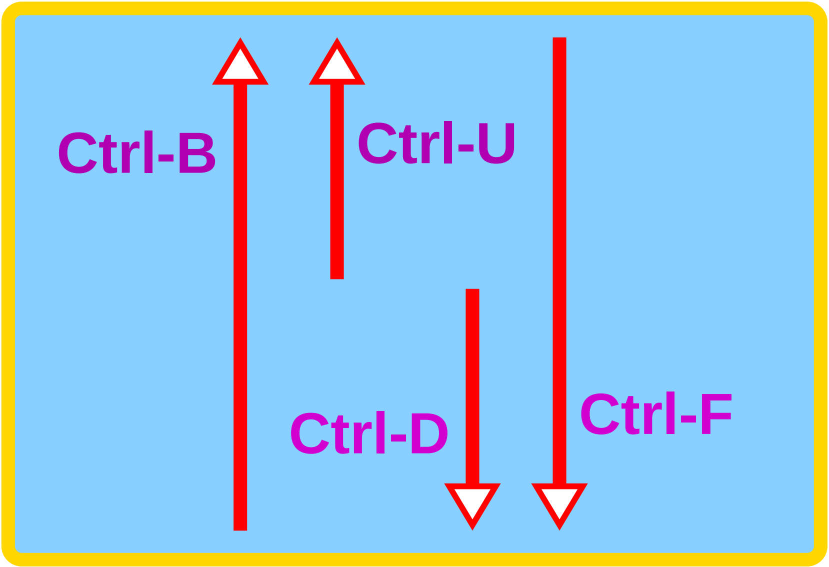
Getting started with Vi

- Use "`h`" to move cursor to the **left** for one character.
  Use "`k`" to move cursor **up** one line.
  Use "`j`" to move cursor **down** one line..
  Use "`l`" to move cursor to the **right** for one character.



Getting started with Vi

- `Ctrl-B` - Up one page.
  `Ctrl-U` - Up half page.
  `Ctrl-D` - Down half page.
  `Ctrl-F` - Down one page.



Getting started with Vi

- Online help:

  ```
  :help
  ```

- vimtutor - a **very useful** Vim-based learning document

  ```
  $ vimtutor
  ```

  if you spent 2 hours with vimtutor patiently, you'll be able to master Vi with well!

Shell

- A program acting as an interface between you and the computer.
- Accepts and interpret your command.
- Send it to the internal system for processing.
- Once you are logged into the system through consoles, you got the *shell*.
- All in all, it's a command language interpreter.

Shell

- Shell prompt of normal user:

  $

  or

  >

- Shell prompt of Root:

  #

- It is customizable.

- Shell prompt indicates where you enter command.

Shell

- Microsoft Windows has one and only one *not so good* shell:

  `command.com`

- In Linux/Unix, there are many. Commonly used are:

Bourne shell	`/bin/sh`
GNU Bourne-again shell	`/bin/bash`
C shell	`/bin/csh`
Tcsh shell	`/bin/tcsh`
Korn shell	`/bin/ksh`

- Default shell for all Linux system is **bash**.

Highlights of shell features

Navigators:

Keystroke	Action
`Ctrl-a`	Move the cursor to the **beginning** of the command line
`Ctrl-e`	Move the cursor to the **end** of the command line
`Ctrl-u`	Delete the entire command line
`Ctrl-k`	Delete all the way from the cursor to the end of the command line
`Ctrl-l`	Clear screen
`Ctrl-_`	Undo the last change
`Up/Down arrow`	Cycle through the command history

Highlights of shell features

- Filename completion using `TAB` key.
- List of commands separated by ";" can be issued together.
- Wild-card characters "*", "?".
- Regular expression.
- Shell programming.
- Input/Output redirection.

Highlights of shell features

- Job control.
- Command history.
- Shell customization.

Highlights of shell features

- ~ means $HOME, e.g. /home/peterho.
- !! means repeat the last command.
- The command `history` show the command history.

Pipelines

- A pipeline is a sequence of simple commands separated by '`|`'. (*invented at 1972 by Ken Thompson*)

- The output of each command in the pipeline is connected via a pipe to the input of the next command.

  i.e., each command reads the previous command's output.

e.g. Find the number of processes owned by root:

```
$ ps aux | grep '^root' | wc -l
```

Redirection

- Input redirection:

  e.g.

```
$ cat < file
```

- Output redirection:

  e.g.

```
$ ps aux >> file
```

Shell script

- Analogous to batch file of Microsoft Windows.
- Text file
- First line: "`#!/bin/bash`".
- Followed by commands and control flows.

Shell script

```
#!/bin/bash

PATH=/bin:/usr/bin:/usr/local/bin

cd $HOME

oldfiles=`find ./Mail \( -name '*.bak' -o -name '*.o' \) -mtime +7`
echo $oldfiles | mail -s "Old files deleted" root
rm $oldfiles

d=`date '+%Y%m%d'`
for f in ./Mail/*; do
    mv $f $f.$d
done

exit 0;
```

Shell script

- Execution of a shell script *script.sh*:

```
$ /bin/bash script.sh
```

- If *script.sh* is executable (i.e., mode = u+x), execute directly:

```
$ ./script.sh
```

Job control

- You can run more than one job in the shell simultaneously.

```
$ ls -l | grep '.bak$'; tar cpf /tmp/a.tar ./*.bak'; date
```

- Only one can be foreground. The rest must either be runing in the background or suspended.

- All jobs will be run in foreground by default.

  e.g.

```
$ ls -lR /
```

Job control

- To run job in background: append "&" to the command line.

  e.g.

  ```
  $ ls -lR / &
  ```

- To send the foreground job to backgroud, you need to:
  1. Type Ctrl-Z to suspend it first.
  2. Type "bg" to send it background.

Job control

- To show how many jobs are running:

  `$ jobs`

```
[1]-  Running                gv -seascape linux-intro.ps &
[2]+  Stopped                vim linux-intro.tex
[3]+  Running                mozilla &
```

Shell customization

- Customizing your shell prompt:

  e.g.

```
$ PS1='\u\h $ '
peterho@pc1234 $ PS1='\u\h:\w $ '
peterho@pc1234:/home/peterho $ PS1='\u\h \t $ '
peterho@pc1234 17:54:32 $ _
```

.BASHRC & .BASH_PROFILE

- Customization files.
- /etc/bachrc - system-wide customization file.
- ~/.bashrc - bash resource file.
- ~/.bash_profile - resource file used on login.

Bash reference manual

- Bash reference manual

  http://www.gnu.org/manual/bash

Working with Linux: Part II

1. File system: hierarchy, ownership and permission, journaling, commands
2. Shell programming
3. Job scheduling using cron and at
4. Managing processes and disks
5. Installing software (rpm)

Filesystem

- A systematic way to organize files and directories.
- The underlying techonology facilitates speed of access and safety of storage.
- The most popular filesystem technology used on Linux is **ext2** and **ext3**.

FILESYSTEM

- A filesystem consists of *files* and *directories*.
- A filesystem hierarchy starts from root - "**/**"
- "**/**" is a directory.
- There are two special directories: "**.**" and "**..**".
- "**.**" is **current directory**".
- "**..**" is **parent directory**".
- They exist in every directory.

Filesystem

- /home is an object named "**home**" under "/" directory.
- /bin/cp is an object named "**cp**" under "/bin" directory.

Filesystem

There is a convention in filesystem organization.

e.g. `ls -l /`

```
root
  bin  sbin  home  opt  lib  dev  etc  boot  var  usr  mnt  proc  tmp  lost+found
             |               |         |               :
             |-peterho       |-hda     |-vmlinuz       :.. /exports/d1
             |-mary          |-hda1                 /mnt/exports/d1
                             |-hda2
                             |-hdc
                                                     (NFS mount
                                                      a filesystem on
                                                      another machine.)
```

http://www.pathname.com/fhs/

Filesystem

Directory	Description
/bin	Essential system tools and command
/dev	Devices
/etc	System configuration files and startup files
/home	Home directories
/lib	System libraries
/proc	Images fo running processes
/root	Home directory of the superuser "root"
/sbin	Essential system admin tools and command
/tmp	Temporary files
/usr	User tools, libraries and binaries
/var	Run-time files, spool directories

Filesystem

- Case-sensitive.
- Filename extension non-essential, but useful (for users).
- Filename not limited to 8.3 convention as in Microsoft systems.

Filesystem commands

Commands	Description
`cd`	Change directory
`cp`	Copy a file
`rm`	Remove (del) a file **permanently**
`mv`	Relocate / rename a file
`ls`	List directory content
`mkdir`	Create a directory
`rmdir`	Remove a directory
`chmod`	Change file/directory permission
`pwd`	Show current directory name

Remark: read man-page for more detail. e.g. man ls

Filename specification

- `ls -l *.htm*`
- `ls -l ?.html`
- `ls -l [d-h]*.html`
- `ls -l  /*.jpg`
- `ls -l  /a[!b]c.tif`

File ownership

- Each file/directory is **owned** by a user and group.

- e.g.

  `ls -l` *index.html*

```
-rw-------   1 peterho  peterho    16868 Jul 10 11:07 /home/peterho/index.html
```

- Another user, e.g. *mary*, cannot access the file unless peterho grant an access to her.

File ownership

Granting a read permission of */home/peterho/index.html* to *mary*:

peterho

```
$ whoami
peterho
$ chmod o+r ĩndex.html
$ ls -l ĩndex.html
-rw----r-- 1 peterho peterho 16868 Jul 10 11:07 /home/peterho/index.html
```

File ownership

mary

```
$ whoami
mary
$ cat  peterho/index.html
<html>
<body>testing</body>
</html>
```

Symbolic link & Hard link

- ln -s *target* *symbolic-link*

  e.g. `ln -s index.3.html index.html`

```
$ ls -l
-rw-------   1 peterho  peterho      345 Jul 30 19:34 index.3.html
lrwxrwxrwx   1 peterho  peterho       12 Jul 30 19:35 index.html -> index.3.html
```

- ln *target* *hard-link*

  e.g. `ln index.4.html index.html`

```
$ ls -l
-rw-------   2 peterho  peterho      345 Jul 30 19:34 index.4.html
-rw-------   2 peterho  peterho      345 Jul 30 19:34 index.html
```

Deleting the target of the symbolic link renders the symbolic link dangling.

Mounting filesystem

- Bringing a filesystem under another filesystem hierarchy
- Accessing the foreign filesystem as if it is part of the host filesystem.

e.g.

```
$ mount -t iso9660 /dev/hdc /mnt/cdrom

$ ls /mnt/cdrom/2003/school_camp/img1.jpg
```

(Try: exporting a directory for mounting. After mounting, shutdown the NFS server to see the effect and then reboot it to see the stateless effect.)

Unmounting filesystem

- Detach a previously mounted foreign filesystem.

e.g.

```
$ umount /mnt/cdrom
```

Accessing floppy disk

1. mtools - mdir, mcopy, mdel, etc

2. mounting filesystem

   e.g.

   ```
   $ mount -t vfat /dev/fd0 /mnt/floppy
   ```

   N.B. do not eject your floppy without umounting it first.

Cron & At

- Scheduling jobs at specific time.
- **Cron** daemon looks for crontab files every minute.
- All entries with time specification match the current time will be executed.
- Cron is used to schedule *periodic* tasks.
- **At** job is similar to cron job, but it is used to do the task once at specified time.

  e.g.

  ```
  at 4am tomorrow
  at 10:30pm + 3 days
  ```

Working with Linux: Part III

1. Office suite
2. GIMP
3. Play CD, MP3 and MIDI
4. Web browsing and email

Working with Linux: Part III

Word-processing on Linux http://www.cbbrowne.com/info/wp.html

1. Openoffice - http://www.openoffice.org
2. Abiword - http://www.abisource.org/
3. Wordperfect
4. enscript
5. LaTeX

GIMP

- GNU Image Manipulation Program
- Photoshop-like
- Some nice found in GIMP, but not in Photoshop
- Can read Photoshop PSD file
- http://www.gimp.org

Others

- ImageMagick - An image manipulation tools.
- xfig and tgif - Vector-drawaing tools.
- gnuplot/dia - Graphic/chart plotting.

Postscript and PDF

- gv/ghostscript - View postscript file
- acroread/xpdf - View PDF file

Sound

- mpg123 - MPEG player for layer 1, 2, and 3
- xmms - Audio player
- Grip - CD player and CD rippler `http://sourceforge.net/projects/grip/`

Sound

- cdrecord - command-line tool for burning CD
- mkisofs - command-line tool for creating ISO9660 format image

Web browser

- Mozilla `http://home.mozilla.org`
- Netscape Communciator `http://home.netscape.com`
- Konqueuer `http://www.kde.org`
- Galeon `http://www.gnome.org`
- Opera `http://www.opera.com`
- lynx `http://lynx.browser.org`

Email clients

- Mozilla http://www.mozilla.org
- Netscape Messager http://home.netscape.com
- Mutt http://www.mutt.org
- Pine http://www.washington.edu/pine

Installation on PC with Windows pre-installed

- Installing Linux on a PC requires at least one spare harddisk partition.
- Usually, all PC harddisk space is claimed by Windows.
- However, not all space are used.

Installation on PC with Windows pre-installed

Our tactic: *squeeze the Windows partition for space to create extra partitions for Linux*

1. Turn off virutal memory of Windows.
2. Defragment the Windows partition:

   `C:WINDOWS\SYSTEM32\DEFRAG.EXE`

3. Load FIPS into `C:\`
4. Run `C:\FIPS`
5. Download partboot.img and partroot.img

6. RAWRITE partboot.img and partroot.img into 2 floopy disks respectively.

7. Reboot using the partboot.img floopy.

8. Insert the partroot.img floopy as told.

9. Enter "parted".

10. Enter "resize 1 start end".

11. Reboot using RedHat Linux CD.

Software installation highlights

- Package
- Tarball
- Source

Package

- For RedHat (-like) system, **RPM** is used.
- For Debian system, **DEB** is used.

RPM

- RPM - Redhat Package Manager
- Used to build, install, query, verify, update, and erase individual software packages.
- A package consists of an archive of files and meta-data used to install and erase the archive files.
- The meta-data includes helper scripts, file attributes, and descriptive information about the package.
- It maintains a primary level of dependency checking.

RPM

- Install

  e.g.

  ```
  $ rpm  -i zsh-4.0.6-5.i386.rpm
  ```

RPM

- Upgrade

  e.g.

  `$ rpm -U bash-2.05b-20.1.i386.rpm`

  or

  `$ rpm -F bash-2.05b-20.1.i386.rpm`

RPM

- Erase

  e.g.

  ```
  $ rpm  -e zsh-4.0.6-5.i386.rpm
  ```

RPM

- Query

  e.g.

```
$ rpm -q kernel
$ rpm -ql kernel
$ rpm -qi kernel
$ rpm -qip zsh-4.0.6-5.i386.rpm
```

REFERENCE

- The Linux Documentation Project `http://tldp.org`
- HOWTOs `http://tldp.org/docs.html#howto`
- Guides `http://tldp.org/guides.html`
- The Linux Cookbook `http://dsl.org/cookbook`
- RedHat `http://www.redhat.com`
- Debian `http://www.debian.org`
- Linux Kernel `http://www.kernel.org`

- Oreilly - Linux/Unix http://linux.oreilly.net

- RedHat Linux Getting Started Guide http://www.redhat.com/docs/manuals/linux/RHL-9-Manual/getting-started-guide/index.html