Introduction to Linux

MAHMOOD NADERAN

SCU HIGH PERFORMANCE COMPUTING CENTER

1395/02/04

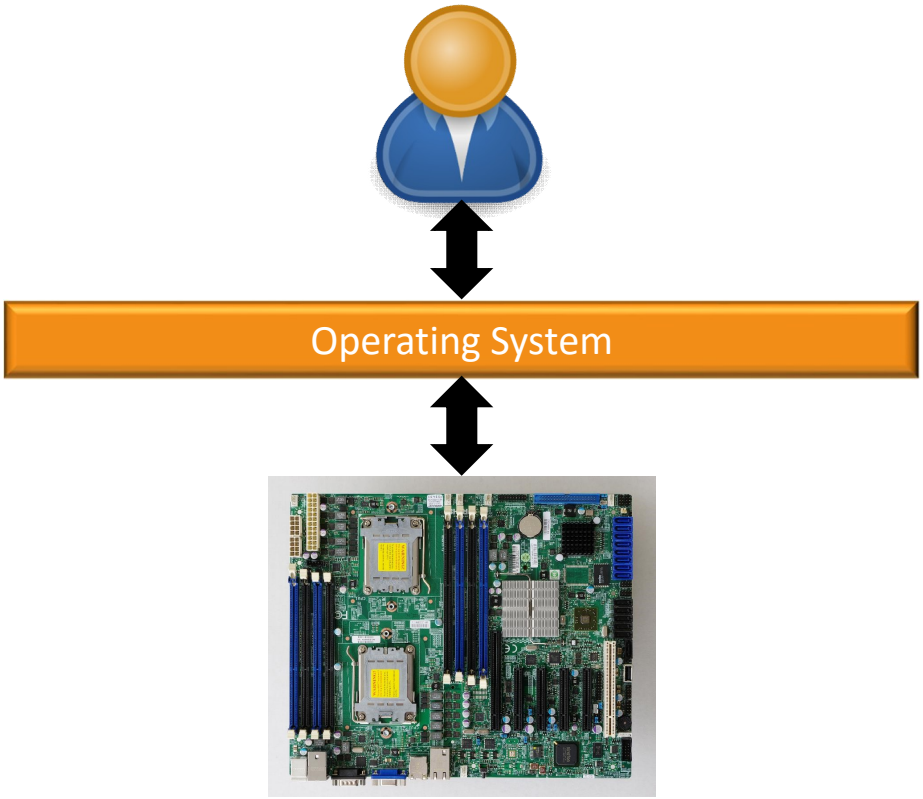
What is an operating system?

Operating system (OS) is a software which sits between human and hardware

Technically, OS is an abstraction layer between human and hardware

- Hides the details from lower to upper and vice versa

When computer is powered on, there are special instructions to load the OS

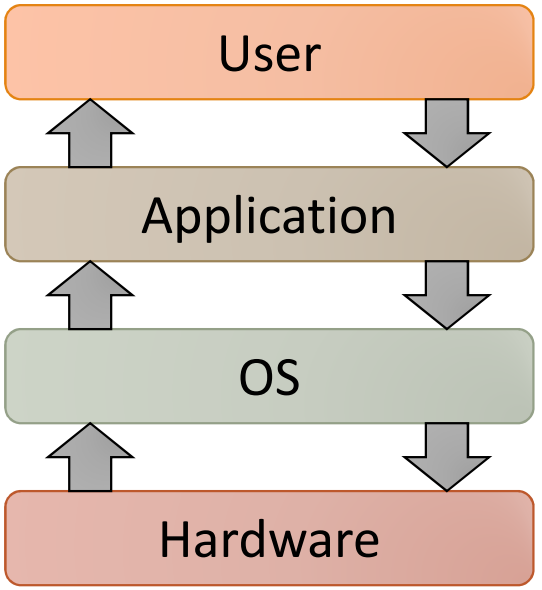


OS is a manager

- All hardware resources, e.g. CPU, memory, disk, GPU, network, printer, keyboard, touchpad, audio, display, …

Tasks of an OS

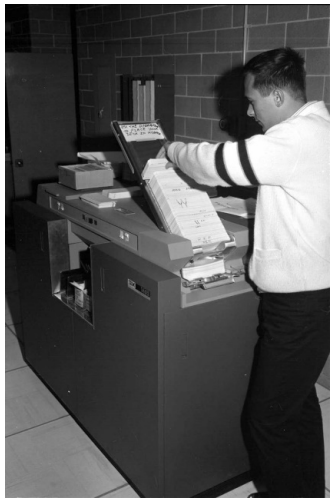
- Control hardware
- Run applications, e.g. web browser or engineering simulator
- Manage data and files to easily access them, e.g. create, copy, delete, …



History of OS

No OS before 60's

- Each user (expert) took over the control of the mainframe for a period of time and used punch cards to load a program into the memory and run



Punch card equipment [uwaterloo]

IBM 704 [wikipedia]

1969

Multics

- Bell Labs + MIT + General Electric started a project on time-shared systems for multiple users to access a mainframe simultaneously called Multiplexed Information and Computing Service

Bell Labs ultimately withdrew

Ken Thompson from Lab decided to write his own operating system

A team worked on PDP-7 to develop concepts such as file systems, processes, device files and command line interpreter

Ken Thompson Dennis Ritchie

A smaller Multics was created

The project later was coined as Unix
- written in PDP-11 assembly language

Bell Labs patented the applications
- The major application on the Unix running on PDP-11 was troff for typesetting

Unix became popular in the Bell Labs

1970

Thompson & Ritchie rewrote the Unix in C

1970

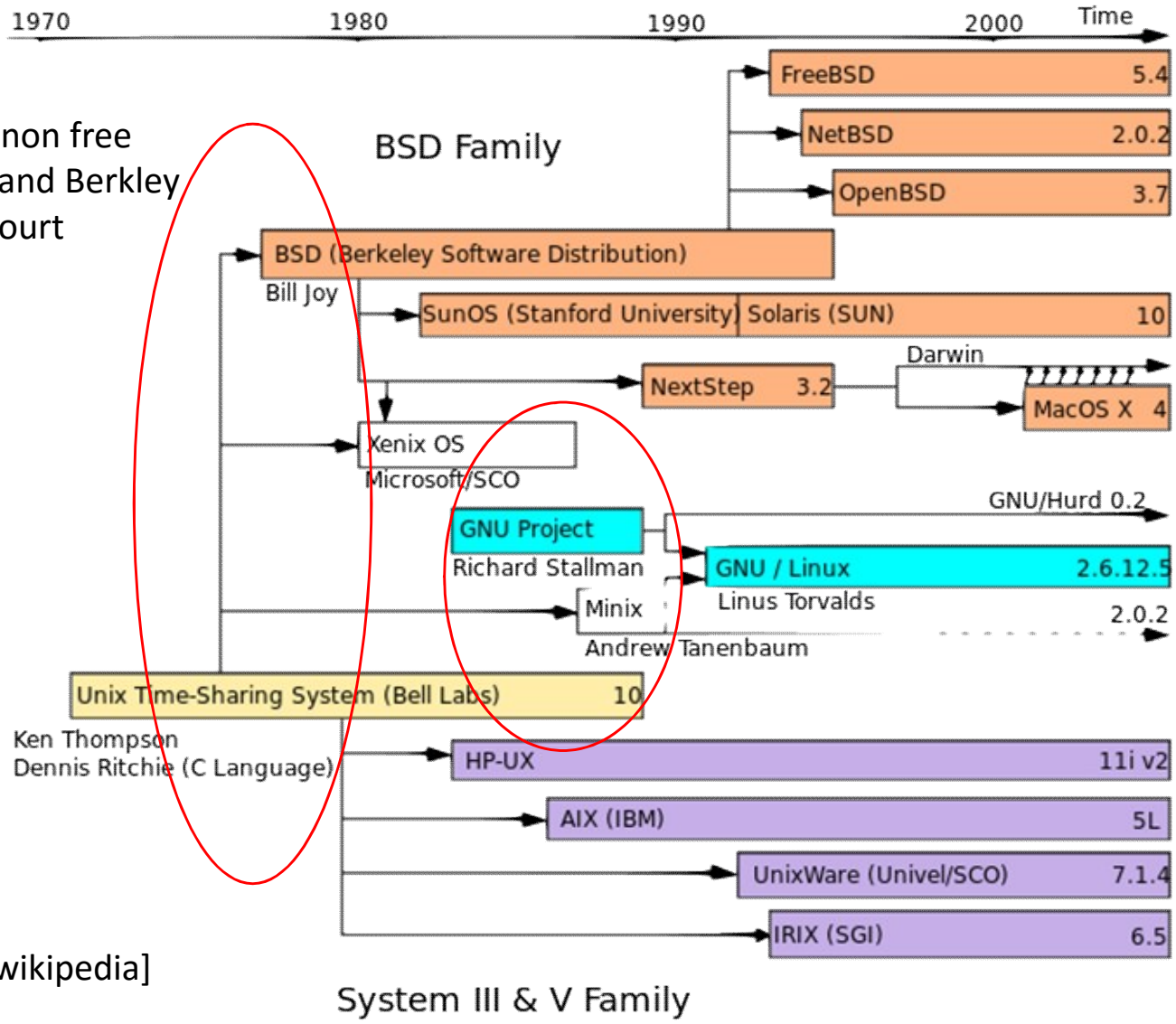
Ritchie and Thompson published the paper about Unix in SOSP

1973

The big problem

Costs and licenses

Unix was non free
Bell Labs and Berkley
went to court



[wikipedia]

- Richard Stallman started the GNU project to publish free software
- He wrote free C compiler and emacs editor

- Andrew Tanenbaum started Minix for educational purpose

25 August 1991

A message from Linus Torvalds to the newsgroup comp.os.minix

Hello everybody out there using minix
I'm doing a (free) operating system (just a hobby, won't be big and professional like gnu) for 386(486) AT clones. This has been brewing since april, and is starting to get ready. I'd like any feedback on things people like/dislike in minix, as my OS resembles it somewhat (same physical layout of the file-system (due to practical reasons) among other things).
I've currently ported bash(1.08) and gcc(1.40), and things seem to work. This implies that I'll get something practical within a few months, and I'd like to know what features most people would want. Any suggestions are welcome, but I won't promise I'll implement them :-)
Linus (torvalds@kruuna.helsinki.fi)

PS. Yes - it's free of any minix code, and it has a multi-threaded fs. It is NOT portable (uses 386 task switching etc), and it probably never will support anything other than AT-harddisks, as that's all I have :-(.
Linus Torvalds

Linus + Unix → Linux

Linux Distribution

A distro is an operating system made from a software collection, which is based upon the Linux kernel and often, a package management system

About 500 active distros

- Commercial vs. free
- Designed for enterprise users, power users or home users
- Designed for server, embedded or desktop devices
- Designed for security, portability and etc.
- ….

Famous distros

Debian: non-commercial distribution and one of the earliest

- Ubuntu

Fedora: sponsored by Red Hat to be a technology testbed for Red Hat's commercial Linux offering

- Centos

OpenSUSE: community distribution mainly sponsored by German company SUSE

Arch Linux: for experienced users

Linux structure

Kernel concept

Boot process

Run levels

File and folder structure

Shell

Partitioning

Linux structure

Kernel concept

Boot process

Run levels

File and folder structure

Shell

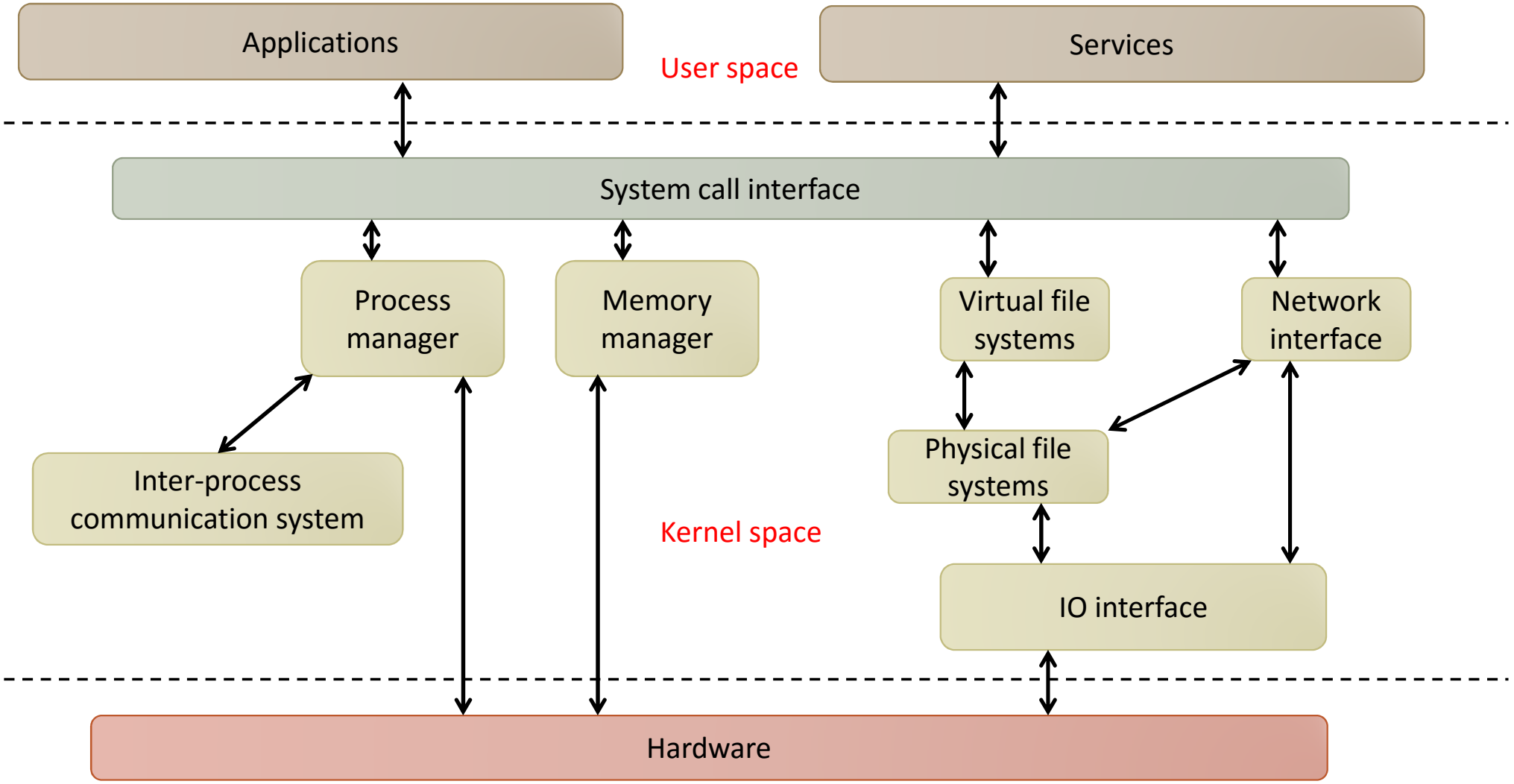
Partitioning

Kernel

Central part in most operating systems

- management of the system's resources
  - Memory management
  - File management
  - IO management
  - Networking
  - …
- communication between hardware and software components
  - Process communication
  - Device drivers
  - …

Kernel is always stored on memory until computer is tern off


Applications
Services
User space
System call interface
Process manager
Memory manager
Virtual file systems
Network interface
Inter-process communication system
Physical file systems
Kernel space
IO interface
Hardware

Processes

Processes carry out tasks within the operating system

- A program is a set of machine code instructions and data stored in an executable image on disk ➔ passive entity
- A process can be thought of as a computer program in action

A dynamic entity, constantly changing as the machine code instructions are executed by the processor

- Running
- Waiting
- Stopped
- Zombie

During the lifetime of a process it will use many system resources

- CPU, memory, ….

Inter-process communication (IPC)

Signals

- Sent by other processes or the kernel to a specific process (kill, stop, …)

Pipes

- Route output from one program to the input of another

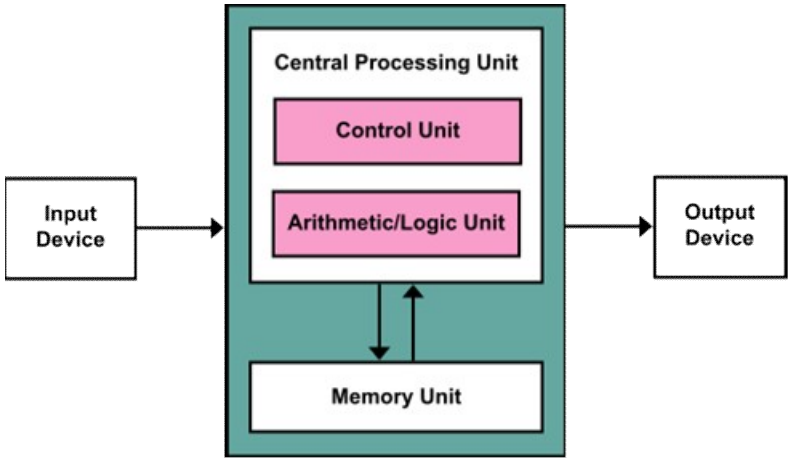
…

Memory management

Remembering Von-Neumann's architecture

Always need to have large memories

- The concept of virtual memory



Large address spaces

Protection

Memory mapping

Fair physical memory allocation

Shared virtual memory

….

File systems

To control how data is stored and retrieved

Different structure and logic, properties of speed, flexibility, security, size, …

- ext3, ext4, ISO9660
- SWAP: special file system when RAM is full

Operations on files

- Create, Open, Write/Read, Close, Delete

IO interface to directly communicate with hardware

- Queuing, bus grants, …

Networking

The client/server model

- Sockets for sending and receiving data similar to telephone line
- Address must be known

Operations

- Connect, Disconnect, Accept, Listening, Send, Receive, …

Different protocols (TCP/IP, UDP, RTSP, …)

Hardware modules (wired, wireless)

Linux structure

Kernel concept

Boot process

Run levels

File and folder structure

Shell

Partitioning

Linux boot procedure

After POST…

Master Boot Record (MBR) contains the primary boot loader

- 512-byte sector, located in the first sector on the disk
- Contains executable and error messages, partition table (4 primary) and a magic number for validation

After the MBR is loaded into RAM, the BIOS yields control to it

Boot loader calls kernel loader ➔GRand Unified Bootloader (GRUB)

- A multiboot software packet from GNU
- Download OS from network
- File system access
- …

After 3 stages, GRUB loads the kernel to the memory

Kernel executes the Init process (PID=1)

- Init is the root/parent of all processes executing on Linux
- The first processes that init starts is a script /etc/rc.d/rc.sysinit
- Based on the appropriate run level, scripts are executed to start various processes to run the system and make it functional

A run level is a software configuration which allows only a selected group of processes to exist

- 0: shutdown/halt
- 1: single user mode
- 2: multi-user mode W/O networking
- 3: only text/console
- 4: reserved
- 5: GUI mode
- 6: reboot

Locations of run levels

- /etc/rc.d/rc#.d/

rc#.d files are the scripts for a given run level that run during boot and shutdown

init.d is a directory that admin can start/stop individual daemons

Linux structure

Kernel concept

Boot process

Run levels

File and folder structure

Shell

Partitioning

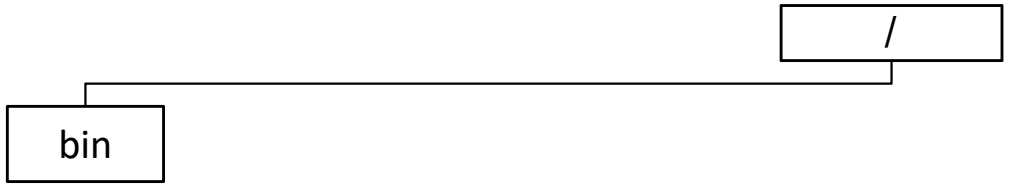
Linux file/folder structure

Any file system in UNIX must be mounted before it can be accessed

Every thing is Linux is a file, or process

Tree like structure

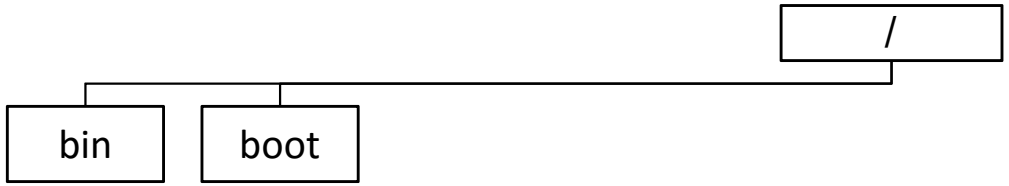
- Files are grouped in folders (directories)
- Directories are grouped under the root
- Directories are separated using '/'
- A file is addressed from the root or '/'
  - /folder1/folder2/file



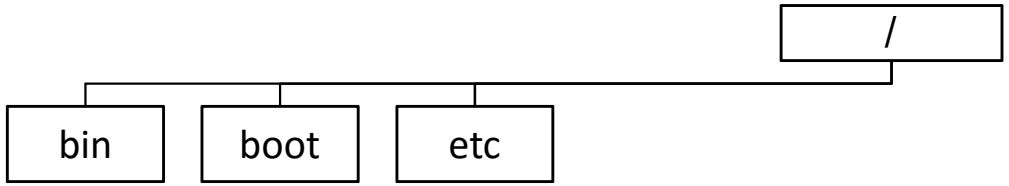
bin - Commands needed during booting up that might be needed by normal users

cd	Change directory
ls	List files in a directory
yes	Repeatedly print ‘y’ in the output

```
$ cd /bin
$ ls
alsaunmute  dash      find        ls          ntfscat       readlink      touch
arch        date      findmnt     lsblk       ntfsck        red           tracepath
chmod       ed        kill        nice        ping          sty
...
```

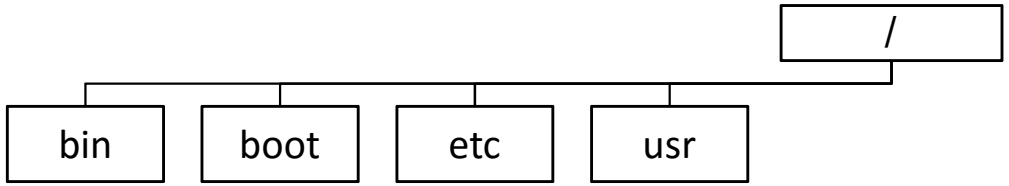


boot - Files used by the bootstrap loader, LILO. Kernel images are often kept here



boot - Files used by the boot, GRUB. Kernel images are often kept here

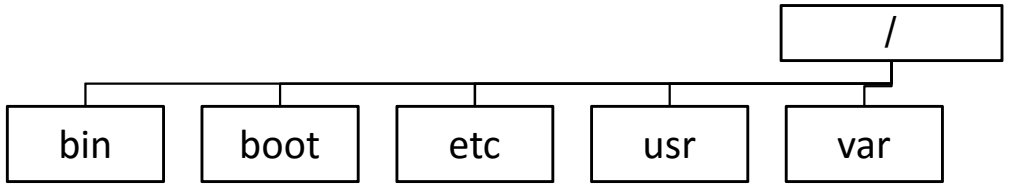
etc - Configuration files specific to the machine



boot - Files used by the boot, GRUB. Kernel images are often kept here

etc - Configuration files specific to the machine

usr - Contains all commands, libraries, man pages, games and static files for normal operation



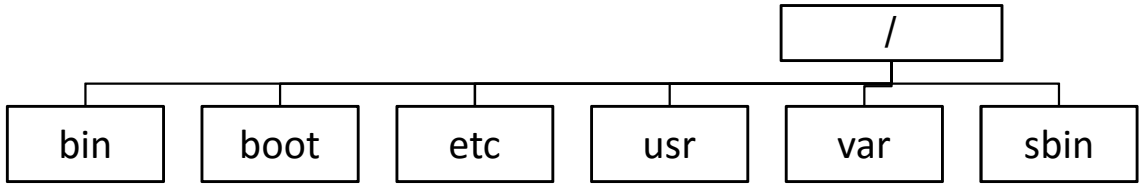
boot - Files used by the boot, GRUB. Kernel images are often kept here

etc - Configuration files specific to the machine

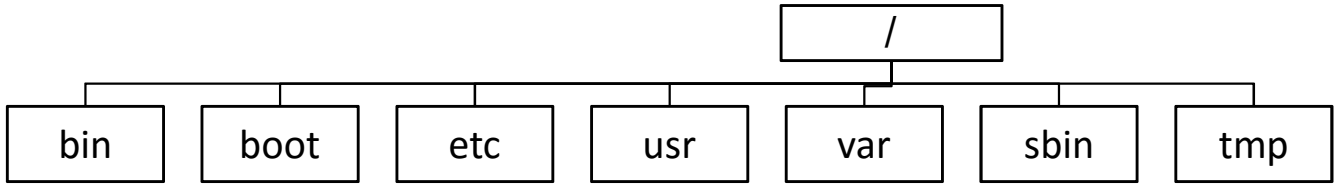
usr - Contains all commands, libraries, man pages, games and static files for normal operation

var - Contains files that change for mail, news, printers log files, man pages, temp files

- log
- lock

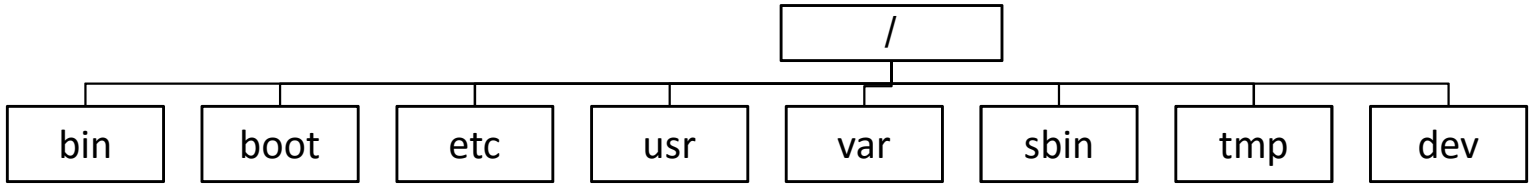


sbin - Like bin but commands are not intended for normal users. Commands run by LINUX.



sbin - Like bin but commands are not intended for normal users. Commands run by LINUX.

tmp - Temporary files. Programs running after bootup should use /var/tmp

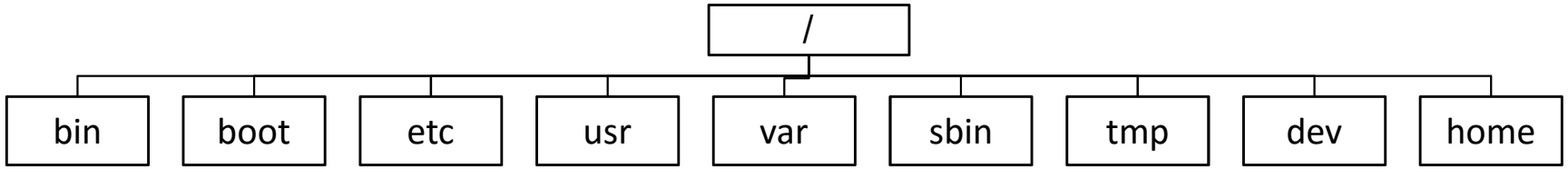


sbin - Like bin but commands are not intended for normal users. Commands run by LINUX.

tmp - Temporary files. Programs running after bootup should use /var/tmp

dev - Device files

- /dev/had or /dev/sda for hard disk
- /dev/null for disposing the output (the black hole!)



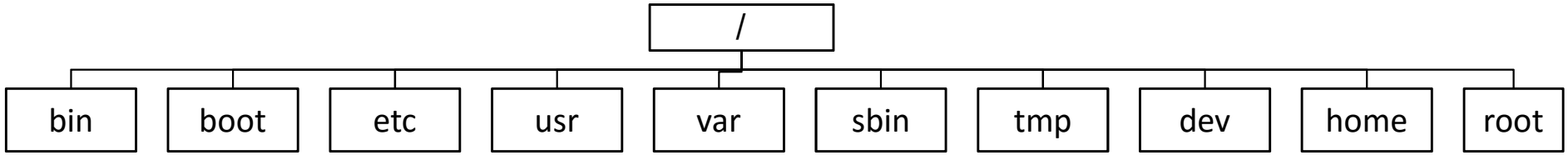
sbin - Like bin but commands are not intended for normal users. Commands run by LINUX.

tmp - Temporary files. Programs running after bootup should use /var/tmp

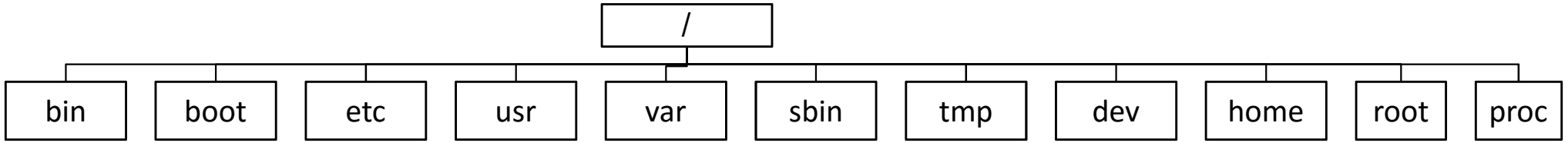
dev - Device files

- /dev/had or /dev/sda for hard disk
- /dev/null for disposing the output (the black hole!)

home - Contains the user's home directories along with directories for services



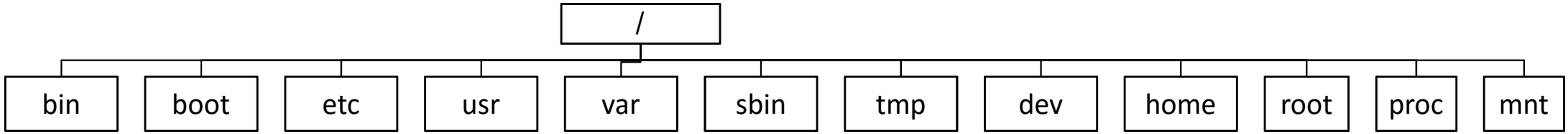
root - The home directory for the root user



root - The home directory for the root user

proc - This filesystem is not on a disk. It is a virtual filesystem that exists in the kernels imagination which is memory

- A directory with info about process number



root - The home directory for the root user

proc - This filesystem is not on a disk. It is a virtual filesystem that exists in the kernels imagination which is memory

- A directory with info about process number

mnt - Mount points for temporary mounts by the system administrator

Linux structure

Kernel concept

Boot process

Runlevels

File and folder structure

Shell

Partitioning

Linux Shell

A command line interpreter

Hides the underlying technical details

Users provides commands or scripts (a group of commands)

Things to provide: piping, variables, control structures

- PATH: a variable for finding executable, otherwise full path (date vs. /bin/date)

Examples: sh, bash, csh, tcsh, zsh

Linux structure

Kernel concept

Boot process

Runlevels

File and folder structure

Shell

Partitioning

Disk partitioning

Many reasons for partitioning!

- It is logical not physical means you can edit for various purposes
  - 250GB vs 750GB partitions
- Sharing partitions with others
- Multiple file systems
- Multiple OSes
- …

Four primary partitions and extended

IDE hard disks

- /dev/hda1 (hda is the first disk, 1 is the partition number)

SATA and SCSI hard disks

- /dev/sda1 (sda is the first disk, 1 is the partition number)

What we learnt today?

What is an operating system?

The history of Linux

Basic concepts

- Kernel
- Boot process
- Run levels
- File and folder structure
- Shell
- Partitioning

What is next?

Workshop on how to install Linux

- When: Tuesday 1395/02/14
- Where: will be announced
- Who: Mr. Hakimi

Don’t forget to bring your laptop!