CSGE602055 Operating Systems
CSF2600505 Sistem Operasi
Week 09: Storage, Firmware, Bootloader, & Systemd

Rahmat M. Samik-Ibrahim (ed.)

University of Indonesia

https://os.vlsm.org/
Always check for the latest revision!

REV220 07-Feb-2020

Operating Systems 2020-1

A [08-10, Rm 3114, Mo/We] — B/M [10:10-12, Rm 3114, Mo/We] — C [13-15, Rm 3114, Mo/We]

D [10-12, Rm 2307(Mo), Rm 3113(We)] — E [08-10, Rm 2307(Mo), Rm 3113(We)]

Week	Schedule	Topic	OSC10
Week 00	27 Jan - 02 Feb 2020	Overview 1, Virtualization & Scripting	Ch. 1, 2, 18.
Week 01	03 Feb - 09 Feb 2020	Overview 2, Virtualization & Scripting	Ch. 1, 2, 18.
Week 02	10 Feb - 16 Feb 2020	Security, Protection, Privacy,	Ch. 16, 17
		& C-language	
Week 03	17 Feb - 23 Feb 2020	File System & FUSE	Ch. 13, 14, 15
Week 04	24 Feb - 01 Mar 2020	Addressing, Shared Lib, & Pointer	Ch. 9
Week 05	02 Mar - 08 Mar 2020	Virtual Memory	Ch. 10
Reserved	09 Mar - 13 Mar 2020	Q & E	
MidTerm	14-21 Mar 2020 (TBA)	MidTerm (UTS)	Subject to change.
Week 06	23 Mar - 31 Mar 2020	Concurrency: Processes & Threads	Ch. 3, 4
Week 07	01 Apr - 07 Apr 2020	Synchronization & Deadlock	Ch. 6, 7, 8
Week 08	08 Apr - 14 Apr 2020	Scheduling + W06/W07	Ch. 5
Week 09a	15 Apr - 19 Apr 2020	Storage, Firmware, Bootldr, & Systemd	Ch. 11
OCL	20 Apr - 26 Apr 2020	OnLine & CoLearnIng	
Week 09b	27 Apr - 28 Apr 2020	Storage, Firmware, Bootldr, & Systemd	Ch. 11
Week 10	29 Apr - 05 May 2020	I/O & Programming	Ch. 12
Reserved	06 May - 10 May 2020	Q & A	
Final	11-18 May 2020 (TBA)	Final (UAS)	This schedule is
Extra	25 Jun 2020	Extra assignment confirmation	subject to change.

STARTING POINT — `https://os.vlsm.org/`

☐ **Text Book** — Any recent/decent OS book. Eg. (**OSC10**)
Silberschatz et. al.: **Operating System Concepts**, 10th Edition,
2018. See also `http://codex.cs.yale.edu/avi/os-book/OS10/`.

☐ **Resources**
  ☐ **All In One** — `BADAK.cs.ui.ac.id:///extra/` (**FASILKOM only!**).
  ☐ **Download Slides and Demos from GitHub.com**
    `https://github.com/UI-FASILKOM-OS/SistemOperasi/`
  ☐ **Problems** — `https://rms46.vlsm.org/2/`:
    195.pdf (W00), 196.pdf (W01), 197.pdf (W02), 198.pdf (W03),
    199.pdf (W04), 200.pdf (W05), 201.pdf (W06), 202.pdf (W07),
    203.pdf (W08), 204.pdf (W09), 205.pdf (W10).

☐ **Try Demos**
  ☐ Your own Ubuntu system.
  ☐ Ubuntu on VirtualBox, or VMWare, or . . .
  ☐ Windows Subsystem for Linux (**Windows 10 only!**).
  ☐ SSH to `BADAK.cs.ui.ac.id` (**FASILKOM only!**).

Agenda

Week 09 Storage, Firmware, Bootloader, & Systemd: Topics[1]

- Storage
- Storage Arrays
- BIOS
- Loader
- Systemd

---

[1]Source: ACM IEEE CS Curricula 2013

- Storage [Usage]
- Storage Arrays [Usage]
- BIOS [Usage]
- Loader [Usage]
- Systemd [Usage]

---

[1]Source: ACM IEEE CS Curricula 2013

Storage, Firmware, Bootloader, & Systemd

- Reference: (OSC10-ch11)
- Storage Capacity (2019)[1]
  - Legacy 3.5" Floppy Disk (1.4MB) – obsolete?
    - SuperDisk (up to 240 MB) –– never took off.
  - 4.7" Compact Disc (700MB) – obsolete?
    - 4.7" Digital Versatile Disc (up to 9GB) – ?
    - 4.7" Blu Ray (up to 128 GB) $\Rightarrow$ DVD++.
  - Tape Cartridge (up to 15TB)
    - Robotic System (up to 250 PB per unit)
    - NASA, Google, Microsoft are still using this!
    - Cheap but slow.
  - Hard Disk Drives (up to 16 TB).
    - From Perpendicular Magnetic Recording to
      Shingled Magnetic Recording technology (+25% – writing problems).
    - Mechanical Disk Arm Scheduling (Until When?).
  - Solid-State Disks (up to 16 TB).
    - SSD Price > HDD Price.
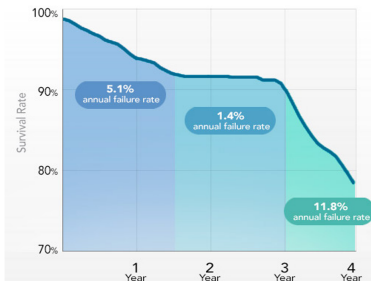   - Write Speed >> Read Speed.
    - (What is a) Flash Disk?

---

[1]Subject to change

Storage Failure Rates

- MTTDL: Mean Time To Data Loss
- MTTF: Mean Time To Failure
- BackBlaze (Cloud Backup Services)

https://www.extremetech.com/computing/
170748-how-long-do-hard-drives-actually-live-for

Figure: BackBlaze — Failure Rates of 25000 DISKS

Storage Management

- Attached-Storage.
  - Host-Attached Storage: via I/O.
  - Network-Attached Storage (NAS): via distributed FileSystem.
  - Storage Area Network (SAN): dedicated Network.
- Formating
  - Low Level (Physical)
  - High Level (FileSystem)
- Boot Block
- Disk Partition
  - "MBR"-scheme
    - upto 4 primary partition
    - upto 2 TB disk
  - "GPT"-scheme
    - "unlimited" partition
    - "unlimited" disk
    - redundancy
- Swap Space Management: On Partition or FileSystem?

RAID: Redundant Array of In* Disks

- RAID 0, 1, 5, 6, 10, 100
- Note (http://www.commodore.ca/windows/raid5/raid5.htm):
    - RAID was created to enhance data performance, reliability and availability.
    - Striping, parity checking and mirroring are three primary functions of RAID systems.
    - RAID performs its functions transparent to the operating system.
    - Systems are typically defined by ranks consisting of five disks each connected to one or two Disk Array Controllers.
    - Different RAID levels provide varying degrees of speed and data protection.
- Problems with RAID
- Stable-Storage Implementation

BIOS, Boot, & Systemd

- Firmware
  - BIOS: Basic Input Output System.
  - UEFI: Unified Extensible Firmware Interface.
  - ACPI: Advanced Configuration and Power Interface.
- Operating System (Boot) Loader
  - BOOTMGT: Windows Bootmanager / Bootloader.
  - LILO: Linux Loader.
  - GRUB: GRand Unified Bootloader.
- Operating System Initialization
  - Init (legacy)
  - UpStart
  - Systemd

Legacy BIOS

- Check Settings.
- Initialize CPU & RAM.
- POST: Power-On Self-Test.
- Initialize ports, LANS, etc.
- Load a Boot Loader.
- Handover to the Boot Loader.
- Provides "Native" (obsolete) Drivers only (not loadable).
- Provides "INT" services .
- Limitation.
  - Technology of 1970s.
  - 16 bits software.
  - 20 bits address space (1 MB).
  - 31 bits disk space (2 TB).

BIOS

Figure: BIOS

UEFI

- A Firmware Specification, not an Implementation!
- No (INT) service after boot.
- HII: Human Interface Infrastructure.
- Protected Mode.
- Flexible.
    - Technology of 2000s.
    - writen in C.
    - (third party) loadable drivers and tools.
    - Emulate Legacy BIOS transition (MBR block, INT service).
    - UEFI Shell: environment shell for diagnostic (no need for DOS).
- Problems
    - Who controls the Hardware?
    - Is "Secure Boot" a good thing?
    - How about a **NASTY/LOCKING/TROJAN** UEFI implementation?
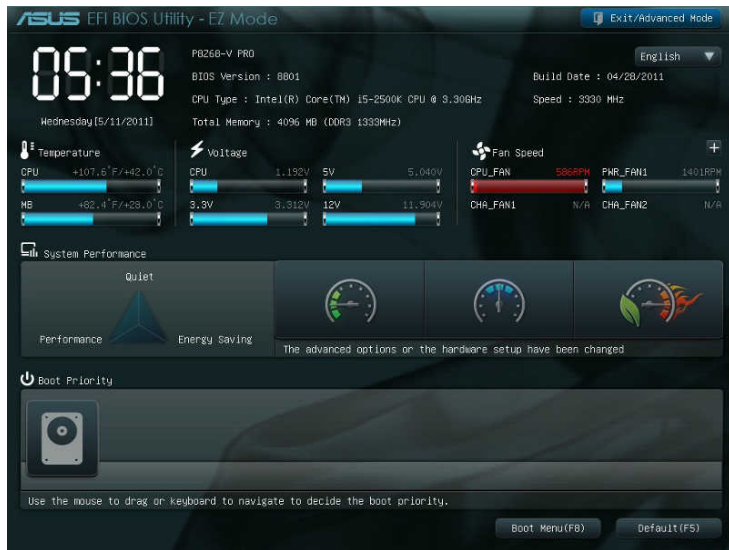   - Different **DRIVERS**.

UEFI

Figure: UEFI

Platform Initialization (PI) Boot Phases

Figure: UEFI Boot Process[1].

Operating System (Boot) Loader

- General
  - How/Where to start the operating system?
  - What to do?
  - How many ways to boot?
  - How many types of OS?
- Disk Partition
  - MBR: Master Boot Record (1983).
  - GPT: GUID (Globally Unique Identifiers) Partition Table (2010s).
- GRUB: GRand Unified Boot system
  - Stage 1: a small boot.img inside the MBR.
  - Stage 1.5 (core.img): FileSystem drivers after MBR.
  - Stage 2: Kernel Selection: Windows, Linux, BSD, etc.
- GRUB2
  - More flexible than GRUB legacy.
  - More automated than GRUB legacy.
  - Accept MBR and GPT.
  - Stage 1.5 (core.img): generated from diskboot.img.
  - No 1024 cylinder restriction.

Figure: GRUB[1].

---

[1]Source Shmuel Csaba Otto Traian 2013

init (SYSV legacy)

- File: /etc/inittab.
- Folders: /etc/rcX.d — X = runlevel.
    - Seven (7) different runlevels:
        - 0 (shutdown).
        - 1 (single-user/admin).
        - 2 (multi-user non net).
        - 3 (standard).
        - 4 (N/A).
        - 5 (3+GUI).
        - 6 (reboot).
    - SXX-YYY: Start
    - KXX-YYY: Kill.
- One script at a time in order.
- dependency is set manually.

UpStart - Ubuntu

- Developer: Ubuntu.
- Folder: /etc/init/.
- Control: initctl.
  - initctl list – listing all processes managed by upstart.
- better support for hotplug devices.
- cleaner service management.
- faster service management.
- asynchronous.

The All New "systemd"

- Replaces (SYSV) init and UpStart.
    - better concurrency handling: Faster!
    - better dependencies handling: No more "S(tarts)" and "K(ills)".
    - better crash handling: automatic restart option.
    - better security: group protection from anyone including superusers.
    - simpler config files: reliable and clean scripts.
    - hotplug: dynamic start/stop.
    - supports legacy systems (init).
    - overhead reducing.
    - unified management way for all distros.
    - bloated: doing more with more resources.
    - linux specific: NOT portable.

systemctl

```
for II in          \
   'systemctl list-unit-files | head -8; echo "(...)";
       systemctl list-unit-files| tail -8' \
   'systemd-analyze blame | wc -l; echo "===";
       systemd-analyze blame | head -15' \
   'systemctl --full  | wc -l; echo "===";
       systemctl --full  | head -10' \
   'systemctl list-units  | wc -l; echo "===";
       systemctl list-units  | head -10' \
   'systemctl list-units |grep .service|wc -l;echo "===";
       systemctl list-units|grep .service|head -10' \
   'systemctl list-units  | grep ssh.service' \
   'systemctl status ssh.service' \
   'systemctl is-enabled ssh' \
   'journalctl' \
   'journalctl -b' \
do
...
```

The End

- ☐ This is the end of the presentation.
- ☒ This is the end of the presentation.
- • This is the end of the presentation.