

Linux Systems and Open Source Software

Basis of Hardware and Software

Chia-Heng Tu
Dept. of Computer Science and Information Engineering
National Cheng Kung University
Fall 2022



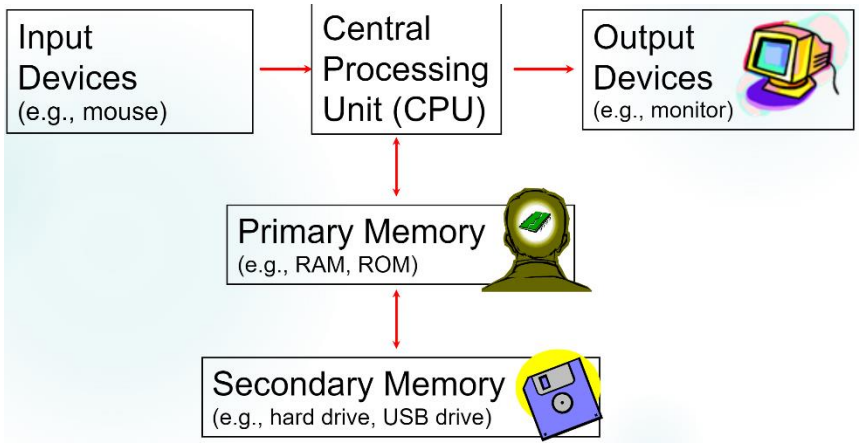
Outline

- Computer Hardware
- Boot Up
- Linux Systems



Computer Hardware

- **Essential hardware components**
- **Central Processing Unit (CPU)**
  - is a unit that reads and executes computer program instructions
- **Input Devices**
  - refer to equipment that puts data into a form a computer can process
  - Examples: keyboard, mouse, webcam, etc.
- **Output Devices**
  - refer to equipment that translates processed information from the CPU into a form that can be understood by a human.
- **Primary memory**
  - refers to memory where the data and programs that are in use at the time are stored
  - As programmers, this memory holds the values of our variables
- **Secondary memory**
  - refers to memory where a user stores data and programs for as long as desired
  - Examples: Hard drive, CDs/DVDs, USB sticks, etc.

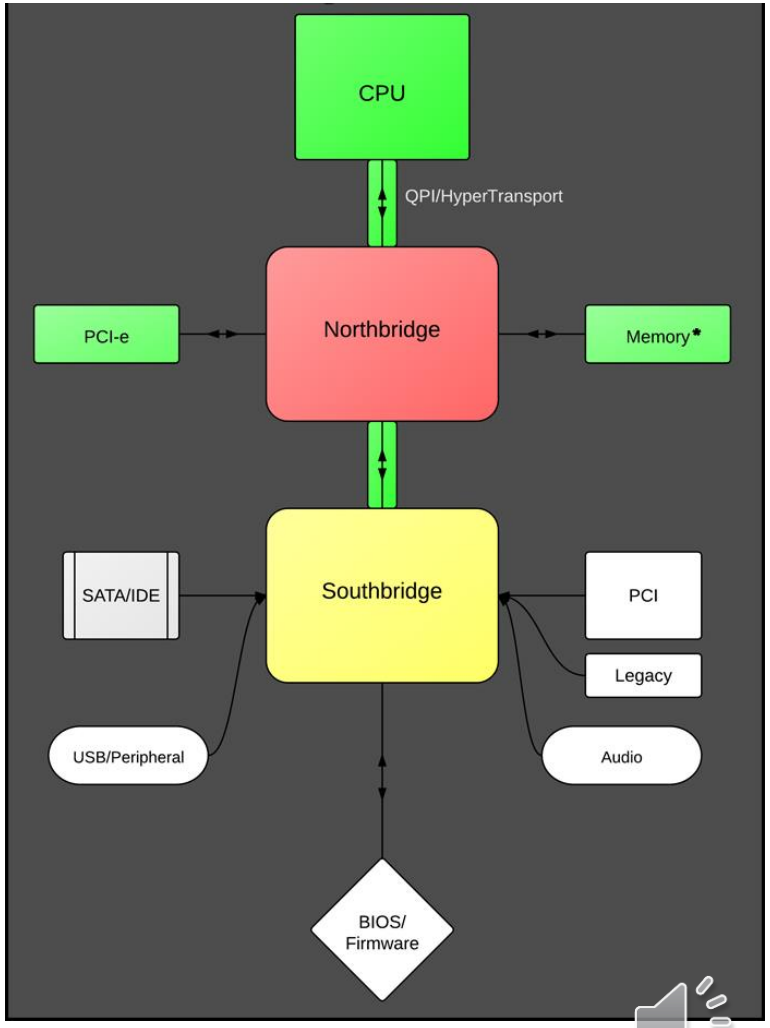


Courtesy of The Internal Components of a Personal Computer (PC), 2015

Hardware Organization of Modern x86 Computers



- Display the computer hardware in a different perspective, where the "*hubs*" of the components are highlighted
  - The **North Bridge** was previously responsible for communicating with PCI-e and memory, and
  - the **South Bridge** communicated with SATA and IDE, USB, firmware chips, PCI, legacy devices, and audio



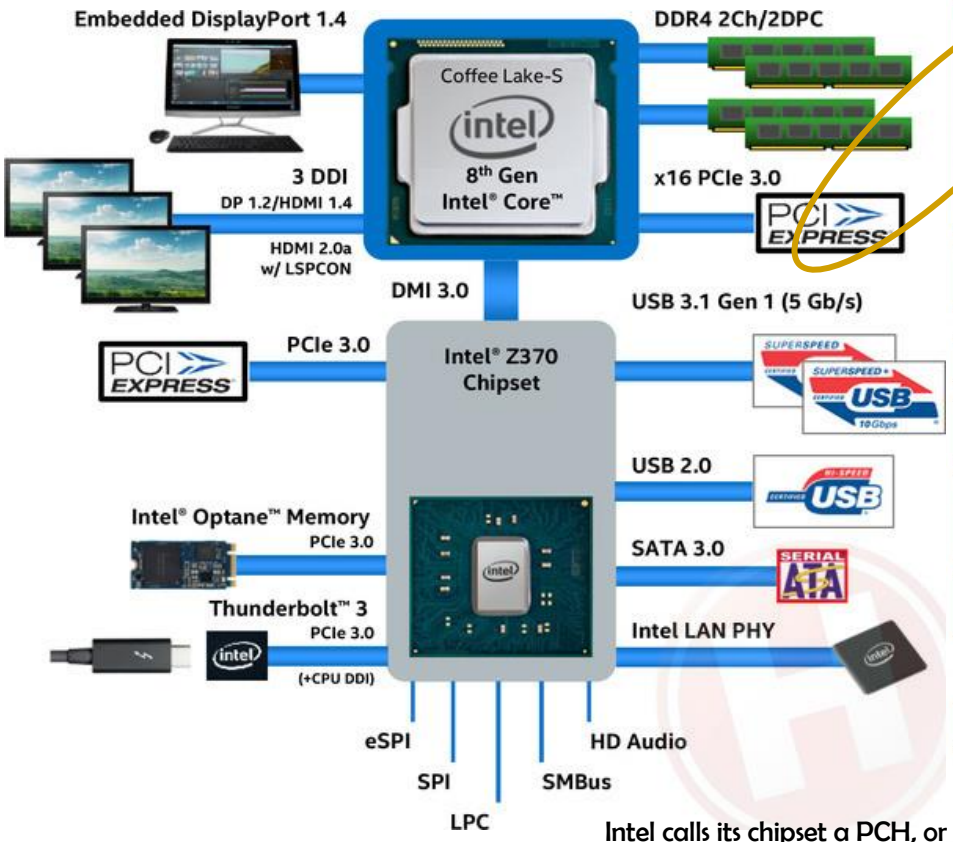
- Both AMD and Intel unified the old **North Bridge** and **South Bridge** into a single **chipset**
  - These days, all of these devices talk to either **the CPU** or **the unified chipset**
- **The high-speed controller are merged into the CPU**
  - **E.g., memory controller**
  - Intel's IMC and AMD's SOC determine whether memory slots can operate in dual-channel or quad-channel, control memory clocks, manage DRAM refreshes, writing, and reading, and have some security features related to memory
- **The low-speed I/O devices are controlled by the chipset**

Courtesy of De Intel Z370-chipset, 2017

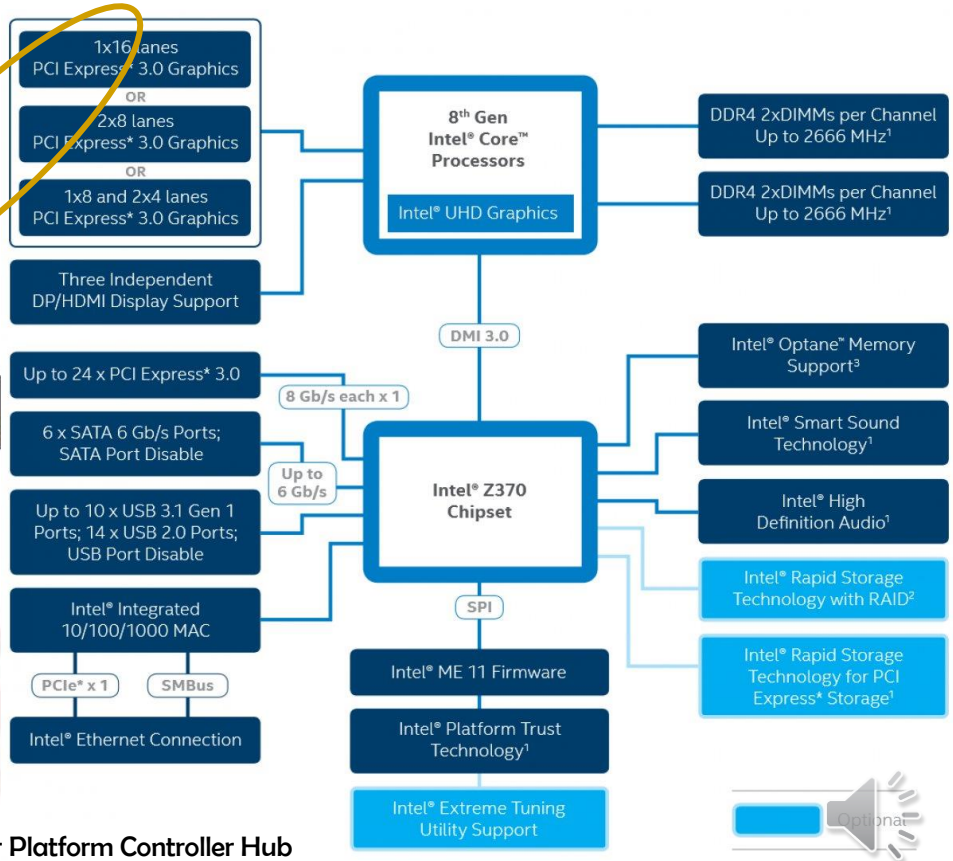


Architecture of Intel Chipset (Coffee Lake/Z370)

With technical specifications info



With technical details



Intel calls its chipset a PCH, or Platform Controller Hub

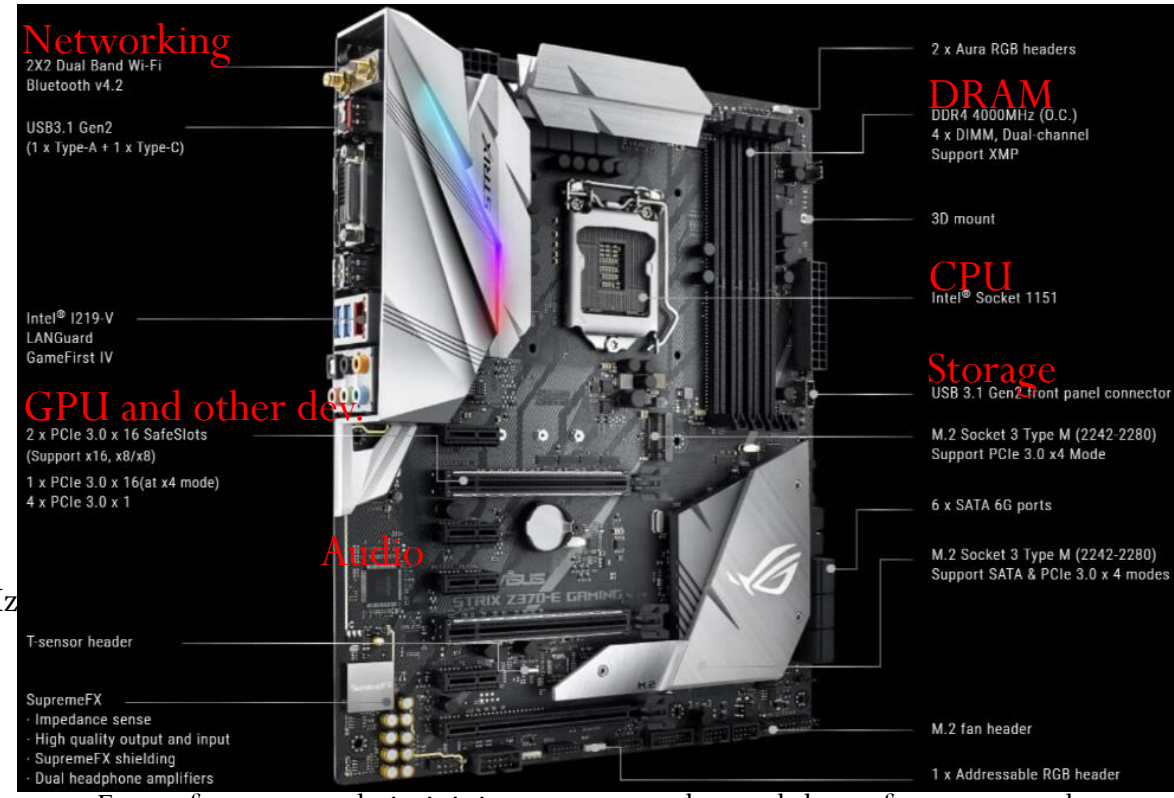
Courtesy of De Intel Z370-chipset, 2017
What is a Chipset? AMD vs. Intel (Z390 vs. Z490, etc.), 2018



Motherboard for the Intel Chipset (Coffee Lake/Z370)

ASUS ROG STRIX Z370-E GAMING

- CPU
  - Intel® Socket 1151 9th / 8th Gen Intel® Core™, Pentium® Gold and Celeron® Processors
    Supports Intel® 14 nm CPU
- GPU
  - **Multi-GPU Support**
  - Supports NVIDIA® 2-Way SLI™ Technology
  - Supports AMD 3-Way CrossFireX™ Technology
- Wireless data network
  - Wi-Fi 802.11 a/b/g/n/ac
  - Supports dual band frequency 2.4/5 GHz
  - Supports MU-MIMO
- Audio
  - ROG SupremeFX 8-Channel High Definition Audio CODEC S1220A
  - The Z370 chipset has the audio support too
- I/O ports and others ...



- For performance analysis, it is important to understand the performance numbers of the underlying hardware
- You can check this page to know more about the basis for CPU frequency, DRAM (data bandwidth), and I/O inferface (data bandwidth).
- **BIOS** is somewhere on the motherboard.

Courtesy of ASUS ROG STRIX Z370-E GAMING, ASUS, 2019

Spec of Intel® Core™ i7-8700 Processor

Essentials

Export specifications

Product Collection	8th Generation Intel® Core™ i7 Processors
Code Name	Products formerly Coffee Lake
Vertical Segment	Desktop
Processor Number	i7-8700
Status	Launched
Launch Date	Q4'17
Lithography	14 nm
Use Conditions	PC/Client/Tablet
Recommended Customer Price	$303.00 - $312.00

Performance

# of Cores	6
# of Threads	12
Processor Base Frequency	3.20 GHz
Max Turbo Frequency	4.60 GHz
Cache	12 MB SmartCache
Bus Speed	8 GT/s DMI3
TDP	65 W

Supplemental Information

Embedded Options Available	Yes
Datasheet	View now

Memory Specifications

Max Memory Size (dependent on memory type)	128 GB
Memory Types	DDR4-2666
Max # of Memory Channels	2
Max Memory Bandwidth	41.6 GB/s
ECC Memory Supported ‡	No

Processor Graphics

Processor Graphics ‡	Intel® UHD Graphics 630
Graphics Base Frequency	350 MHz
Graphics Max Dynamic Frequency	1.20 GHz
Graphics Video Max Memory	64 GB
4K Support	Yes, at 60Hz
Max Resolution (HDMI 1.4)‡	4096x2304@24Hz
Max Resolution (DP)‡	4096x2304@60Hz
Max Resolution (eDP - Integrated Flat Panel)‡	4096x2304@60Hz
DirectX* Support	12
OpenGL* Support	4.5

# of Displays Supported ‡	3
Device ID	0x3E92

Expansion Options

Scalability	1S Only
PCI Express Revision	3.0
PCI Express Configurations ‡	Up to 1x16, 2x8, 1x8+2x4
Max # of PCI Express Lanes	16

Package Specifications

Sockets Supported	FCLGA1151
Max CPU Configuration	1
Thermal Solution Specification	PCG 2015C (65W)
$T_{JUNCTION}$	100°C
Package Size	37.5mm x 37.5mm

Advanced Technologies

Intel® Optane™ Memory Supported ‡	Yes
Intel® Turbo Boost Technology ‡	2.0
Intel® vPro™ Platform Eligibility ‡	Yes
Intel® Hyper-Threading Technology ‡	Yes
Intel® Virtualization Technology (VT-x) ‡	Yes
Intel® Virtualization Technology for Directed I/O (VT-d) ‡	Yes
Intel® VT-x with Extended Page Tables (EPT) ‡	Yes
Intel® TSX-NI	Yes
Intel® 64 ‡	Yes
Instruction Set	64-bit
Instruction Set Extensions	Intel® SSE4.1, Intel® SSE4.2, Intel® AVX2
Idle States	Yes
Enhanced Intel SpeedStep® Technology	Yes
Thermal Monitoring Technologies	Yes
Intel® Identity Protection Technology ‡	Yes
Intel® Stable Image Platform Program (SIPP)	Yes

Security & Reliability

Intel® AES New Instructions	Yes
Secure Key	Yes
Intel® Software Guard Extensions (Intel® SGX)	Yes with Intel® ME
Intel® Memory Protection Extensions (Intel® MPX)	Yes
Intel® OS Guard	Yes
Intel® Trusted Execution Technology ‡	Yes

You can find detailed specifications & numbers for the CPU from the official website

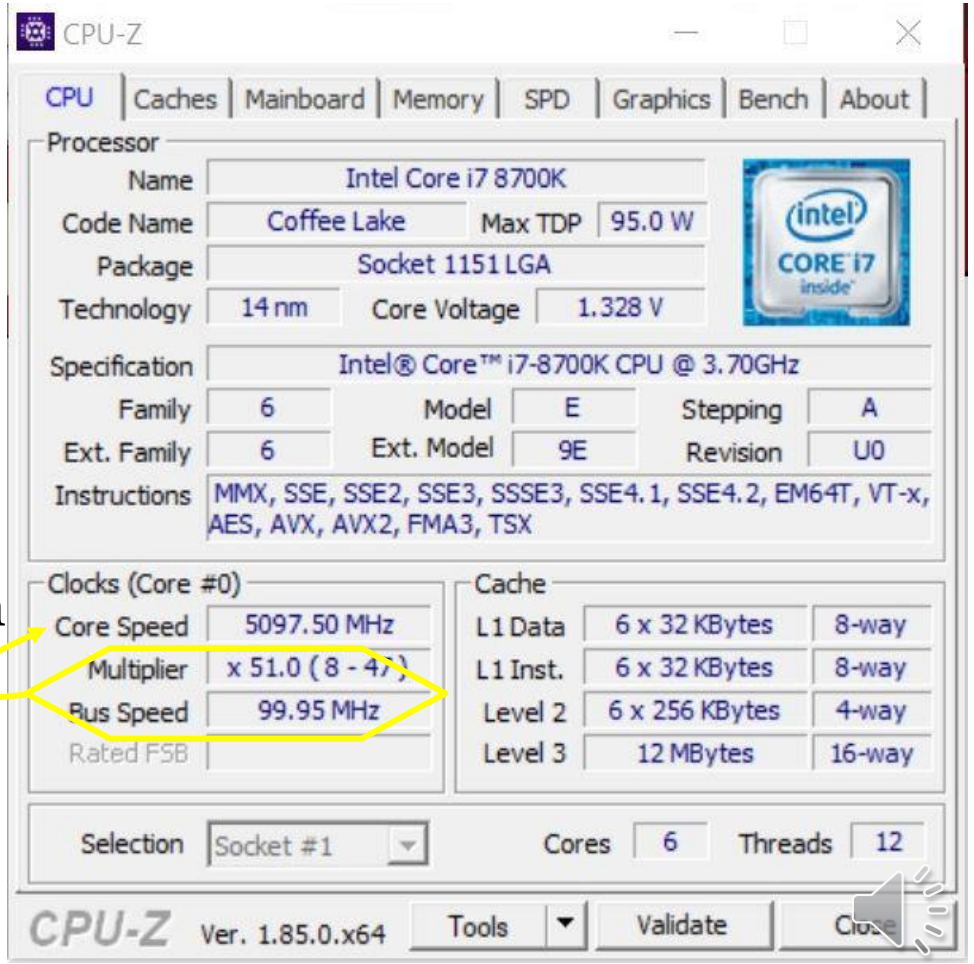
For more information, you can click the link to check the "question marks"

Courtesy of Intel® Core™ i7-8700 Processor, 2017



Get the Spec. Data of Intel® Core™ i7-8700 Processor

- **CPU-Z** is a popular software for us to get the hardware performance data
  - As shown in the right, it provides data for several hardware that is available on the underlying platform



Courtesy of The ASUS ROG Strix Z370-F Gaming Review: A $200 Motherboard at 5.1 GHz, 2018

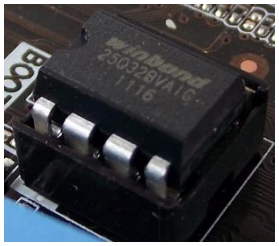


Initialize Computer HW w/ BIOS

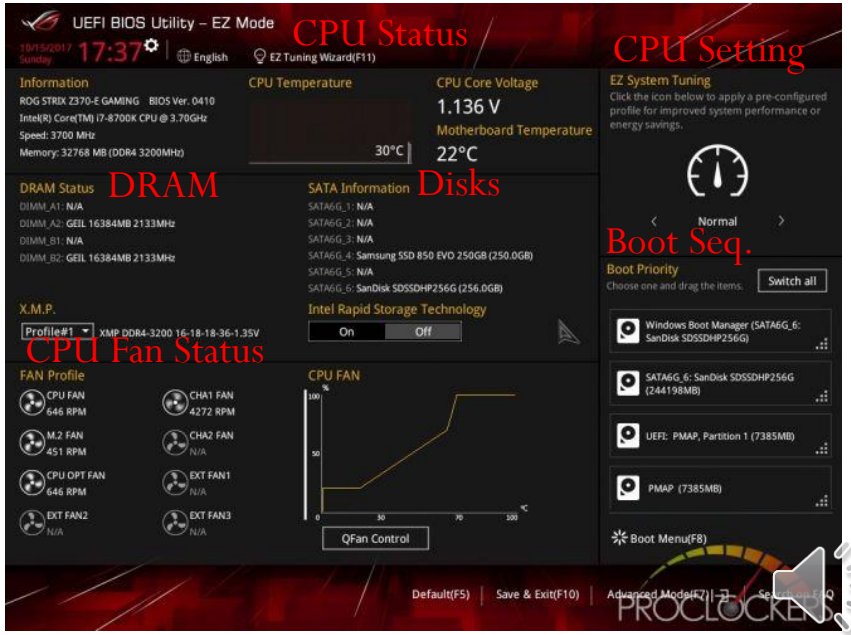
- **BIOS** (Basic Input/Output System)
- A non-volatile firmware used
  - to perform **hardware initialization** during the booting process (power-on startup), and
  - to provide runtime services for operating systems and programs
  - The BIOS firmware comes pre-installed on a personal computer's system board, and it is the **first software to run** when powered on
- BIOS controls the components on the motherboard
  - Operating frequency of CPU and memory
  - On/Off of the CPU cores
  - On/Off of the built-in GPU of the CPU
  - System time, etc.
- You can enter the BIOS menu (in the right figure) to do the configuration at the beginning of the power-on sequence by pressing **Del**, **F1**, or **F4**

AMD BIOS chips for a Dell310 (1980s)

Winbond BIOS from ASUS M5A97 EVO (2011)



BIOS Menu for the ASUS ROS STRIX Z370-E GAMING Board



Courtesy of BIOS, 2019
ASUS M5A97 EVO AM3+ Review, 2011
ASUS ROG STRIX Z370-E Gaming Motherboard Review, 2017



Boot Up Sequence

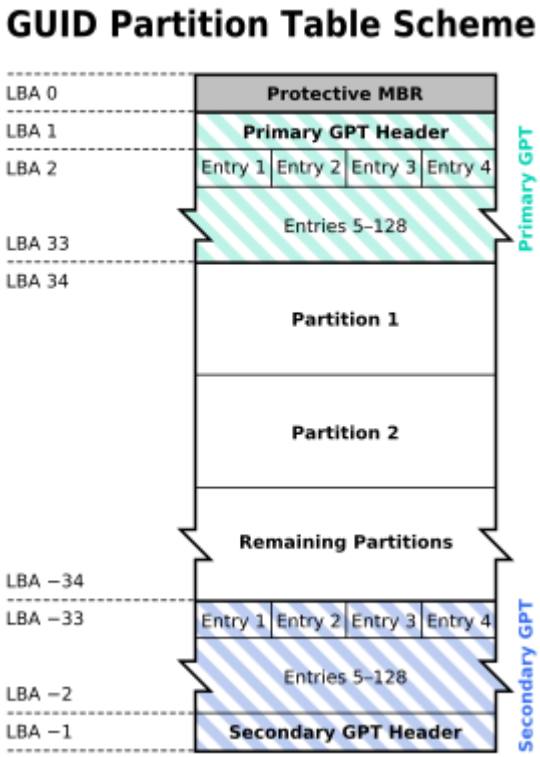
Power-on button is pressed

1. BIOS is used to scan, identify & initialize the hardware on the motherboard
2. BIOS loads the content on the **GPT** (or **MBR**) of the first Boot Device (listed in the Boot Priority)
   - Where the **GPT/MBR** contains the address of the Boot Loader
3. Boot Loader is loaded and starts to load the image of the Operating System
4. Operating System is loaded onto the system memory and starts the boot sequence …
   (Note: the first Linux process to run is **init**, some details kept in **/etc/init.d**)

GUID Partition Table (GPT)



- A standard for **the layout of partition tables** of a physical computer storage device
  - such as a hard disk drive or solid-state drive,
  - using universally unique identifiers, which are also known as globally unique identifiers (GUIDs)
- GPT is a part of the Unified Extensible Firmware Interface (UEFI) standard
  - Unified EFI Forum-proposed replacement for the PC BIOS
  - GPT uses **64 bits for logical block addresses (LBA)**, allowing a maximum disk size of 2^{64}sectors
  - For disks with 512-byte sectors, the maximum size is **9.4 ZB** (9.4×10^{21} bytes) or **8 ZiB** (2^{64} sectors × 2^9 bytes per sector)
- GPT is also used for some BIOS systems (without UEFI support) to support disks (> 2TB)
  - because of the limitations of master boot record (MBR) partition tables,
  - which **uses 32 bits for logical block addressing** (**LBA**) on traditional 512-byte disk sectors
  - For hard disks with 512-byte sectors, the MBR partition table entries allow a maximum size of **2 TiB** (2^{32} × 512 bytes)
  - MBR is originally used by Windows systems, e.g., MS-DOS



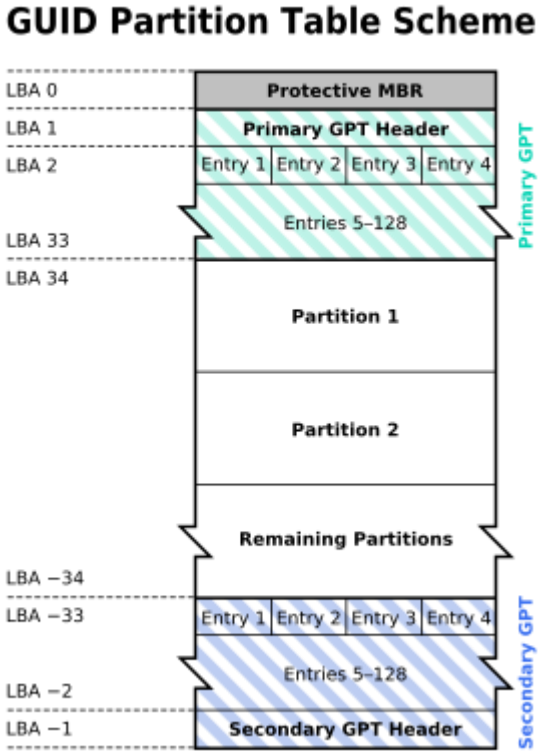
- Each logical block is 512 bytes in size and each entry has 128 bytes
- The corresponding partition entries are assumed to be located in LBA 2–33

Courtesy of Ch2, 鳥哥的Linux私房菜, 2016
GUID Partition Table, Wikipeida, 2019



GUID Partition Table (GPT) (Cont'd)

- LBA 0
  - The first logical block of the storage, which is partitioned into 512-byte blocks and the order of the blocks is used to uniquely identify the blocks
  - **This block is dedicated for MBR** to avoid MBR-based disk utilities from misrecognizing and possibly overwriting GPT disks
  - It contains *partition type* of EEh, where the entire GPT drive is indicated and identifies it as GPT
- LBA 1
  - Use ~100 bytes to record **the metadata for the table**
  - The position and size of this partition table
  - The CRC32 value for the indication of potential errors of the GPT data
  - The backup table (LBA -1~-34) is loaded whenever the CRC32 value checking is failed



- Each logical block is 512 bytes in size and each entry has 128 bytes
- The corresponding partition entries are assumed to be located in LBA 2–33

Courtesy of Ch2, 鳥哥的Linux私房菜, 2016
GUID Partition Table, Wikipeida, 2019

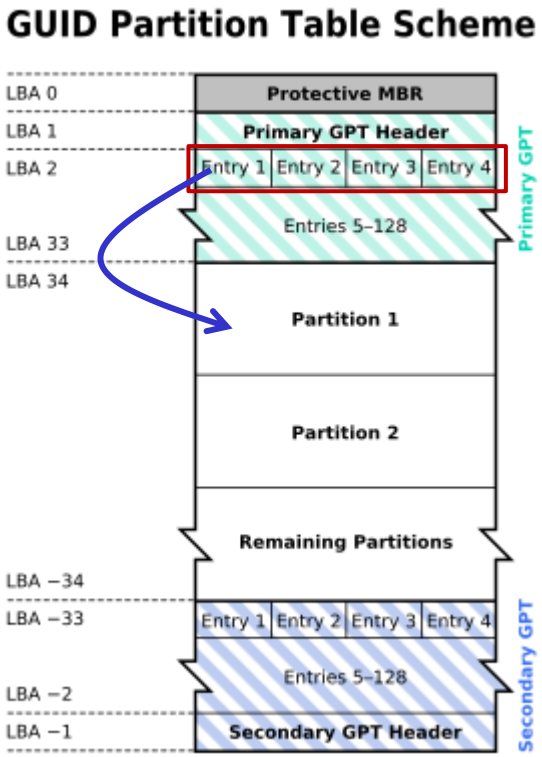


GUID Partition Table (GPT) (Cont'd)

- LBA 2-33
  - Each LBA contains **four** 128-byte partition entries
  - Each entry contains the metadata for **the partition**
  - (1) The first 16 bytes of each entry designate the partition type's globally unique identifier (GUID)
    - To show the OS/type of the partition; e.g., Windows/Basic data partition, HP-UX/Data partition, Linux/Root partition
    - To show the property of the partition: bootable, or data partition
  - (3) (4) The third and forth rows of the table record the starting and ending LBA for the partition represented
  - (6) The last row keeps the name of the partition

The layout of each 128-byte partition entry

	Offset	Length	Contents
1	0 (0x00)	16 bytes	Partition type GUID (mixed endian[6])
2	16 (0x10)	16 bytes	Unique partition GUID (mixed endian)
3	32 (0x20)	8 bytes	First LBA (little endian)
4	40 (0x28)	8 bytes	Last LBA (inclusive, usually odd)
5	48 (0x30)	8 bytes	Attribute flags (e.g. bit 60 denotes read-only)
6	56 (0x38)	72 bytes	Partition name (36 UTF-16LE code units)



- LBA 2-33 contains partition entries
- Maximum number of partitions supported by the table is 128 = 32 (LBA 2-33) * 4 (Each LBA contains 4 entries)

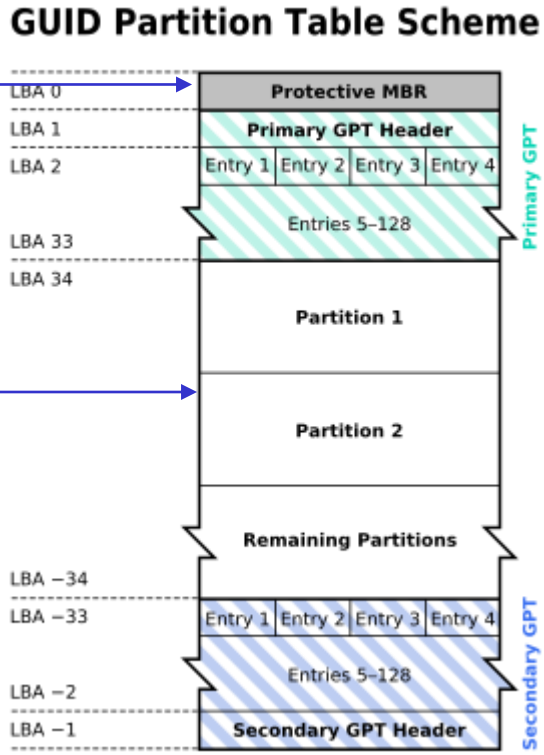
Courtesy of Ch2, 鳥哥的Linux私房菜, 2016
GUID Partition Table, Wikipeida, 2019



A Simplified Boot Up Sequence for BIOS-based PC

Power-on button is pressed

1. BIOS is used to scan, identify & initialize the hardware on the motherboard
2. BIOS loads **the first 440 bytes of the first sector (i.e., boot loader, GRUB)** of the first Boot Device (listed in the Boot Priority)
3. GRUB lists the bootable operating systems on the system
4. The loader of the selected OS to be booted will be loaded first to load the image of the selected OS …



Courtesy of Ch2, 鳥哥的Linux私房菜, 2016
Booting from GPT, 2012



Linux Distribution

- A Linux distribution (or distro)
  - is an operating system made from a software collection, which is based upon the **Linux kernel** and a **package management system**
- A typical Linux distribution
  - comprises a **Linux kernel**, GNU tools and libraries, a window system (the most common being the X Window System), a window manager, and a desktop environment

ubuntu

archlinux

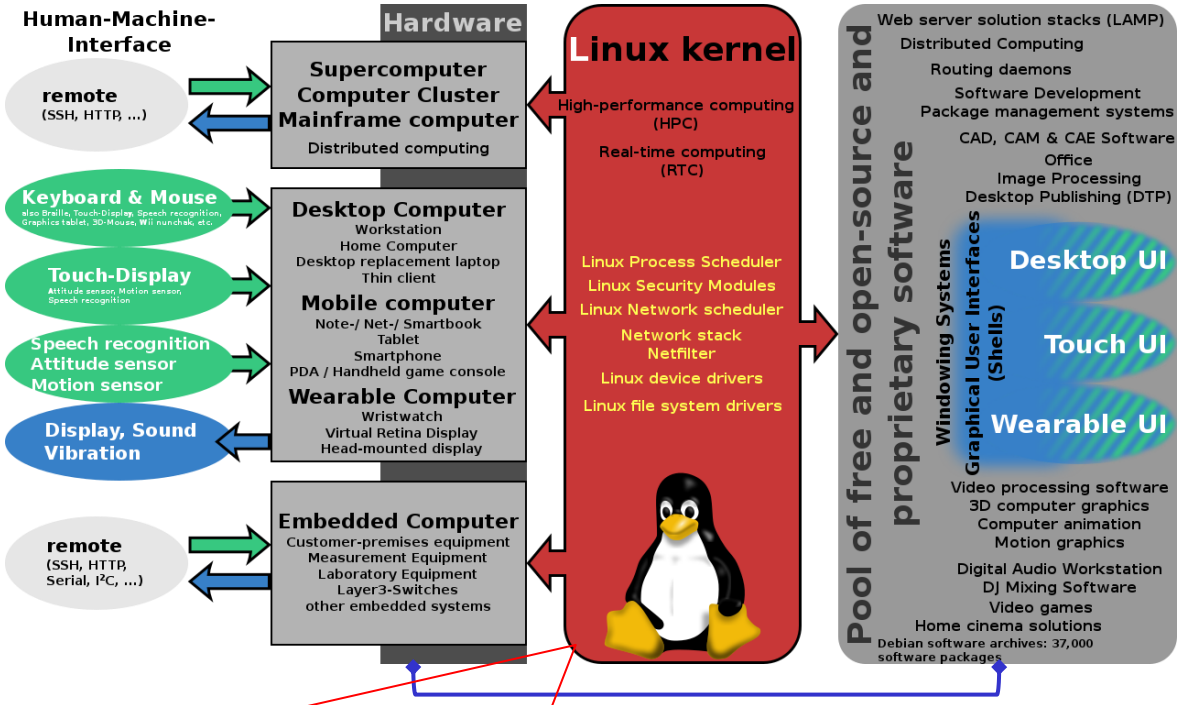
CentOS

chrome

Courtesy of Linux distribution, 2019
CentOS, Ubuntu, Arch Linux, Chrome OS

Linux Distribution (Cont'd)





Different distributions equipped with **different packages for different purposes**, but would use **the same version of Linux kernel**

- A Linux distribution is usually built around a package management system (PMS), which puts together the Linux kernel, free and open-source software, and occasionally some proprietary software
  - The package is typically provided as compiled code, with installation and removal of packages handled by a PMS rather than a simple file archiver
  - Each package intended for such a PMS contains meta-information such as a package description, version, and "dependencies"
  - The package management system can evaluate this meta-information to allow package searches, to perform an automatic upgrade to a newer version, to check that all dependencies of a package are fulfilled, and/or to fulfill them automatically
- Based on the above, **we can simply categorize the Linux distributions into groups according to their PMS**

Courtesy of Linux distribution, 2019
CentOS, Ubuntu, Arch Linux, Chrome OS



Linux Distributions (RPM-based)

- Red Hat Linux and SUSE Linux were the original major distributions that used the .rpm file format
- Both of these were later divided into **commercial** and **community-supported** distributions
  - Red Hat Linux was divided into a community-supported but Red Hat-sponsored distribution named **Fedora**, and a commercially supported distribution called **Red Hat Enterprise Linux**
  - SUSE was divided into **openSUSE** and **SUSE Linux Enterprise**

Distribution	Description
Red Hat Linux	Split into Fedora Core and Red Hat Enterprise Linux. The last official release of the unsplit distribution was Red Hat Linux 9 in March 2003
CentOS	Community-supported Linux distribution designed as an OpenSource version of RHEL and well suited for servers
Fedora	Community-supported Linux distribution sponsored by Red Hat
openSUSE	A community-developed Linux distribution, sponsored by SUSE. It maintains a strict policy of ensuring all code in the standard installs will be from FOSS solutions, including Linux kernel Modules. SUSE's enterprise Linux products are all based on the codebase that comes out of the openSUSE project.
Mandrake Linux	The first release was based on Red Hat Linux (version 5.1) and KDE 1 in July 1998. It had since moved away from Red Hat's distribution and became a completely separate distribution. The name was changed to Mandriva, which included a number of original tools, mostly to ease system configuration. Mandriva Linux was the brainchild of Gaël Duval, who wanted to focus on ease of use for new users.

Courtesy of List of Linux distributions, 2019



Linux Distributions (Debian-based)

- Debian is a distribution that emphasizes free software supports many hardware platforms
  - Debian and distributions based on it use the **.deb** package format and the **dpkg** package manager and its frontends, such as apt-get or synaptic
- There are other distributions:
  - Arch Linux from Pacman-based PCM
  - Chrome OS from Gentoo-based PCM

Debian-based Distribution	Description
BackTrack	Developed by Offensive Security and designed for penetration testing. In March 2013, the Offensive Security team rebuilt BackTrack around the Debian distribution and released it under the name Kali Linux
Kali Linux	Made to be a completely customizable OS, used for penetration testing. It is based on Debian GNU/Linux and is used mostly by security experts
Parsix	Optimized for personal computers and laptops. Built on top of Debian testing branch and comes with security support
PureOS	A GNU/Linux distribution based on Debian with a focus on privacy, security, and convenience
Ubuntu	A free and open-source operating system and Linux distribution based on Debian.

Courtesy of List of Linux distributions, 2019



Software Architecture of Linux Systems

- **User space** (or **userland; user mode**) refers to all code that runs outside the operating system's kernel, whereas
- **Kernel space** contains the software providing services for the userland programs, such as input/output, file system, networking
- The Linux kernel is a monolithic kernel
  - supporting true preemptive multitasking, virtual memory, shared libraries, demand loading, memory management, the Internet protocol suite, and threading

Various layers within Linux, also showing separation between the userland and kernel space

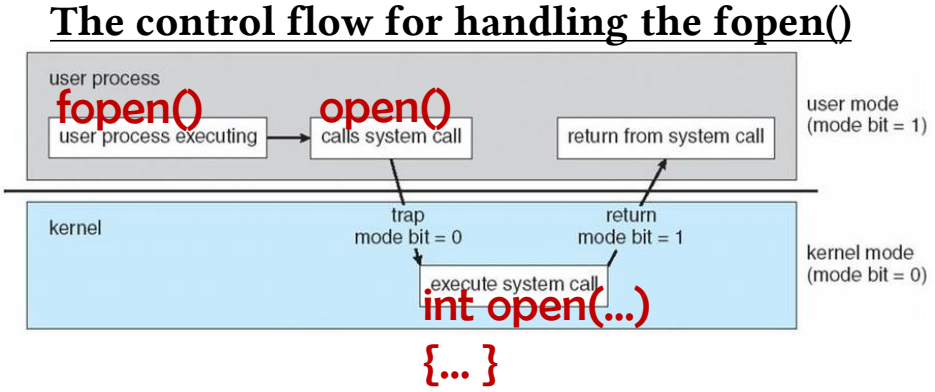
User mode	**User applications**	For example, bash, LibreOffice, GIMP, Blender, 0 A.D., Mozilla Firefox, etc.				
	Low-level system components:	**System daemons**: *systemd, runit, logind, networkd, PulseAudio, ...*	**Windowing system**: *X11, Wayland, SurfaceFlinger (Android)*	**Other libraries:** *GTK+, Qt, EFL, SDL, SFML, FLTK, GNUstep*, etc.		**Graphics**: *Mesa, AMD Catalyst, ...*
	C standard library	open(), exec(), sbrk(), socket(), fopen(), calloc(), ... (up to 2000 subroutines) *glibc* aims to be POSIX/SUS-compatible, *musl* and *uClibc* target embedded systems, *bionic* written for Android, etc.				
Kernel mode	**Linux kernel**	stat, splice, dup, read, open, ioctl, write, mmap, close, exit, etc. (about 380 system calls) The Linux kernel System Call Interface (SCI, aims to be POSIX/SUS-compatible)				
		Process scheduling subsystem	IPC subsystem	Memory management subsystem	Virtual files subsystem	Network subsystem
		Other components: ALSA, DRI, evdev, LVM, device mapper, Linux Network Scheduler, Netfilter Linux Security Modules: *SELinux*, *TOMOYO*, *AppArmor*, *Smack*				
Hardware (CPU, main memory, data storage devices, etc.)						

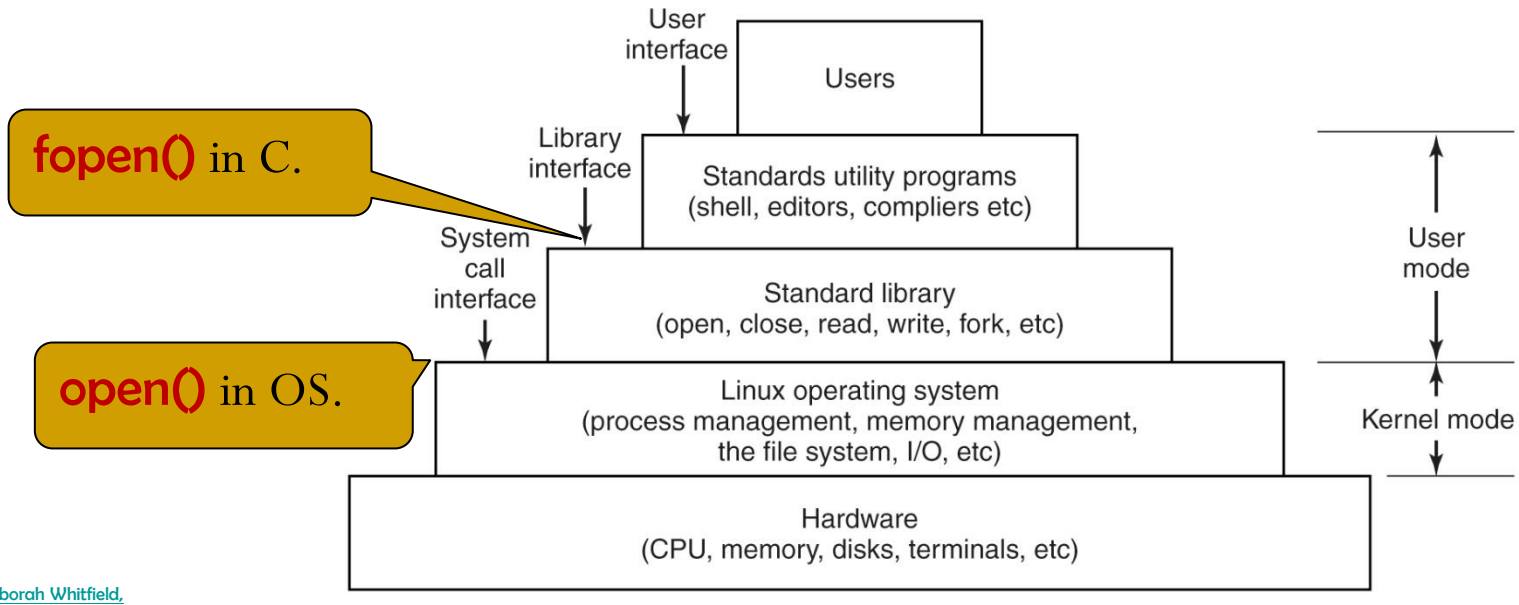
Courtesy of
ChO, 鳥哥的Linux私房菜, 2016
Linux kernel, 2019
User space, 2019
Kernel space, 2019



User and Kernel Spaces

- Pay attention to the hierarchical relationships
- Example: The function control flow for invoking the **fopen()** call in the C program
  - The call is served by the **open()** in Linux kernel







Linux Kernels

- The System Call Interfaces may be different across different versions of Linux Kernel
  - In fact, one kernel version would have very different features from another
    - You can check the system call interface from the example Linux source code
- Linux Kernels are often announced from kernel.org in different types:
  - **Mainline**
    - Mainline tree is maintained by **Linus Torvalds**
    - It's the tree where **all new features are introduced**
    - New mainline kernels are released every 2-3 months
  - **Stable**
    - After each mainline kernel is ***released***, it is considered "stable"
    - Any bug fixes for a stable kernel are backported from the mainline tree and applied by a designated stable kernel maintainer
    - There are usually only a few bugfix kernel releases until next mainline kernel becomes available -- unless it is designated a "longterm maintenance kernel"
    - Stable kernel updates are released on as-needed basis, usually 2-3 a month
  - **Longterm**
    - There are usually several "longterm maintenance" kernel releases provided for the purposes of ***backporting bugfixes*** **for older kernel trees** **(announced 5~10 years ago)**
    - Only important bugfixes are applied to such kernels and they don't usually see very frequent releases, especially for older trees



Linux Kernel Versioning

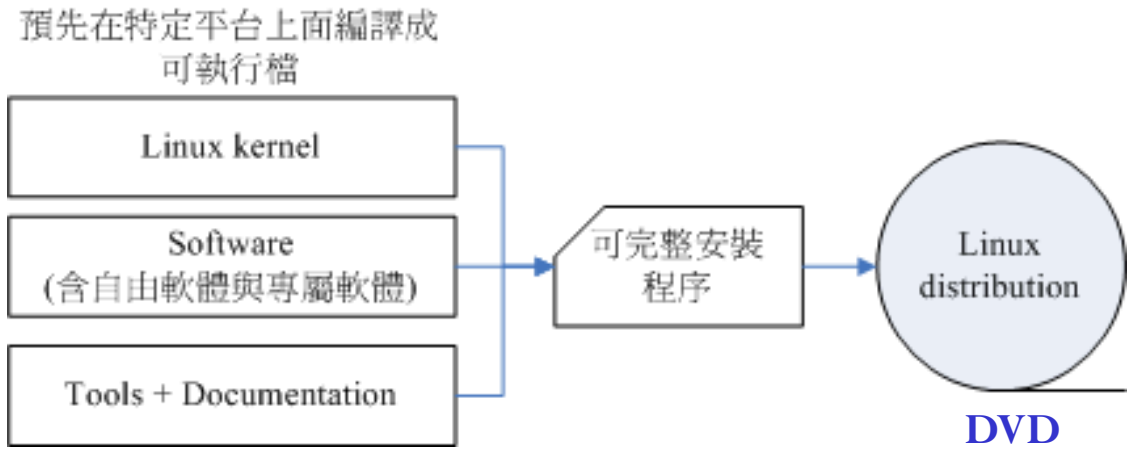
- The versioning system of Linux Kernels
  - E.g., The Linux image file **vmlinuz-4.3.0-2** in the boot folder
  - **w.xx.y-zzz**: the Linux Kernel version
  - **w**: kernel version = 4
  - **xx**: major version = 3
  - **y**: minor revision = 0
  - **zz**: bug fix number = 2
- The version of the Linux Distribution is different from that of its Linux Kernel
  - Ubuntu is version 5.4.0 and its Linux Kernel is 4.4.0
  - Check the versions on your computer via the commands

Courtesy of ChO, 鳥哥的Linux私房菜, 2016
3 Ways to Check Linux Kernel Version in Command Line, 2019



Linux Kernel and Distribution

- Linux Kernel is released in the **source format** (e.g., .c and .h files)
- For each Linux distribution, the **binaries** of the Linux Kernel and tools within are **pre-built** for you to facilitate the installation process





Access HW in Linux

Device	Filename in the Linux folder
SCSI/SATA/USB硬碟機	/dev/sd[a-p]
USB快閃碟	/dev/sd[a-p] (與SATA相同)
VirtI/O界面	/dev/vd[a-p] (用於虛擬機器內)
軟碟機	/dev/fd[0-1]
印表機	/dev/lp[0-2] (25針印表機) /dev/usb/lp[0-15] (USB 介面)
滑鼠	/dev/input/mouse[0-15] (通用) /dev/psaux (PS/2界面) /dev/mouse (當前滑鼠)
CDROM/DVDROM	/dev/scd[0-1] (通用) /dev/sr[0-1] (通用，CentOS 較常見) /dev/cdrom (當前 CDROM)
磁帶機	/dev/ht0 (IDE 界面) /dev/st0 (SATA/SCSI界面) /dev/tape (當前磁帶)
IDE硬碟機	/dev/hd[a-d] (舊式系統才有)

- During the Linux installation process, you will be asked to give the **location** (partition) to install the Linux Distribution in the DVD
  - Hardware can be accessed via the file system interface
  - i.e., the **/dev** folder
  - The first and second SATA hard drives are accessed via the files: **/dev/sda** and **/dev/sdb**
  - The first partition of the first SATA drive: **/dev/sda1**

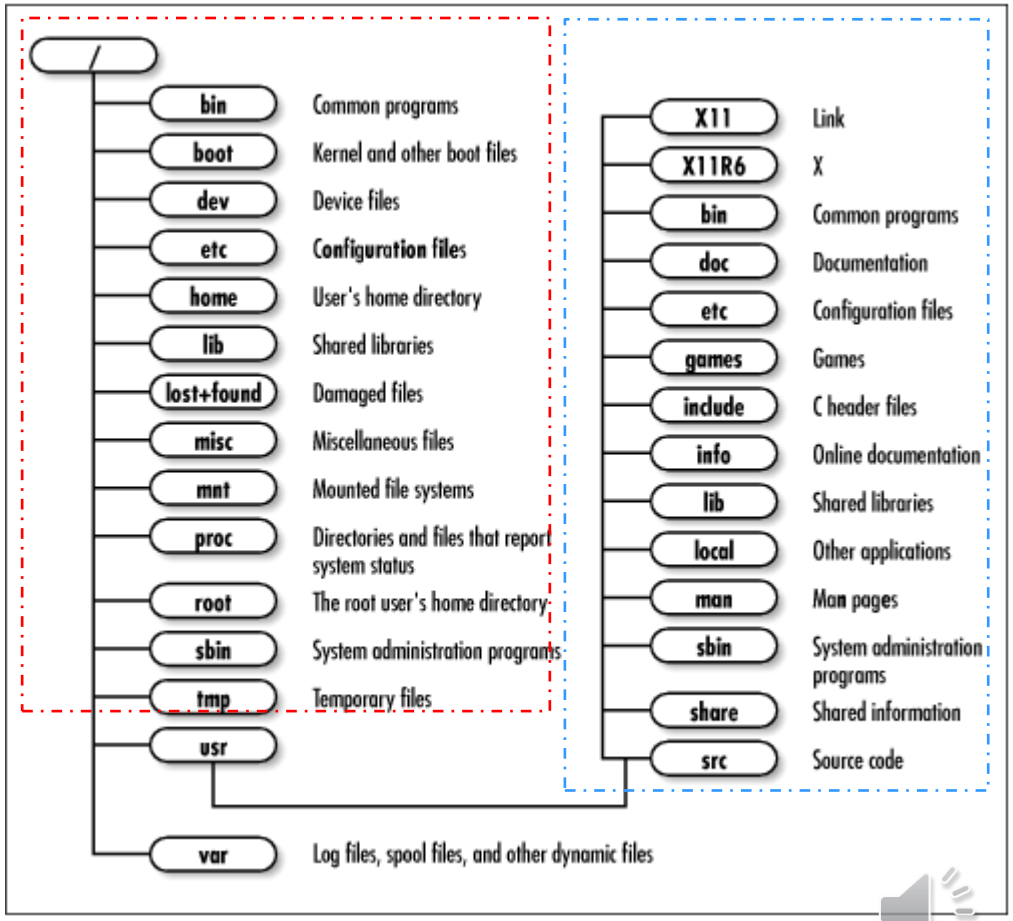
Courtesy of Ch2, 鳥哥的Linux私房菜, 2016
Linux allocated devices



Linux Directory Tree

- You could access to almost all of the resources in Linux via the *files*
- The **root directory /** is the basis for the files
  - **home** is the folder for the users on the system
  - **root** is the folder for the root user of the system
- In Linux, you can *mount* the disk partition to a folder
  - In the figure, **usr** is mounted on one partition, and the other folders are on **the other partition**
  - Hence, the data within **usr** is stored at the different place

The folders under the root directory.



Courtesy of Ch2, 鳥哥的Linux私房菜, 2016
Learning Debian GNU/Linux, 2001
Difference between dev, media and mnt? Mounting a device — role of /dev, /media and /mnt, and the mount command [closed], 2014



Advice for Installing Linux

- While you install the Linux on the system,
- you can customize the partitions on the disk(s), or
- you can also apply the default settings
  - Usually, you will have one partition for the root directory **/** and the other partition for the **swap** space
  - One downside for this setting is that some folders may consume a lot of disk space you have to do something to deal with it

Examples:

- You install too many programs and **/usr** may consume a significant amount of space to keep these files
- You download so many files from the web, and **/home/YOURNAME** will be occupied by the web files



Suggestions

- You may read through the Chapter 0~2 of the vbird website to know more about the Linux
- You can click the web links at the **left-bottom** corners if you want to know more about certain topics



QUESTIONS