

# Operating System

---

**Introduction:** Operational View, Physical Overview of computer System, I/O Devices, Storage structure & Computer Communication N/w.

## Session 2

---

Tutor: Yeshi Wangchuk, Asstt. Lecturer, IT Dept.

# Operational View

- ★ A modern, general purpose computer system consists of a CPU and a number of device controllers connected through common bus providing access to the shared memory[*see Figure 1.3 in session 1*].
- ★ The current system is based on the **Von-Neumann Principle**.
- ★ The Principle states that a program is initially stored in memory & executed by fetching an instruction at a time. The basic cycle of operation is :
  - Fetch an instruction(**Fetch**)
  - Interpret the instruction(**Decode**)
  - Execute the instruction(**Execute**)

# The Von-Neumann Operational View

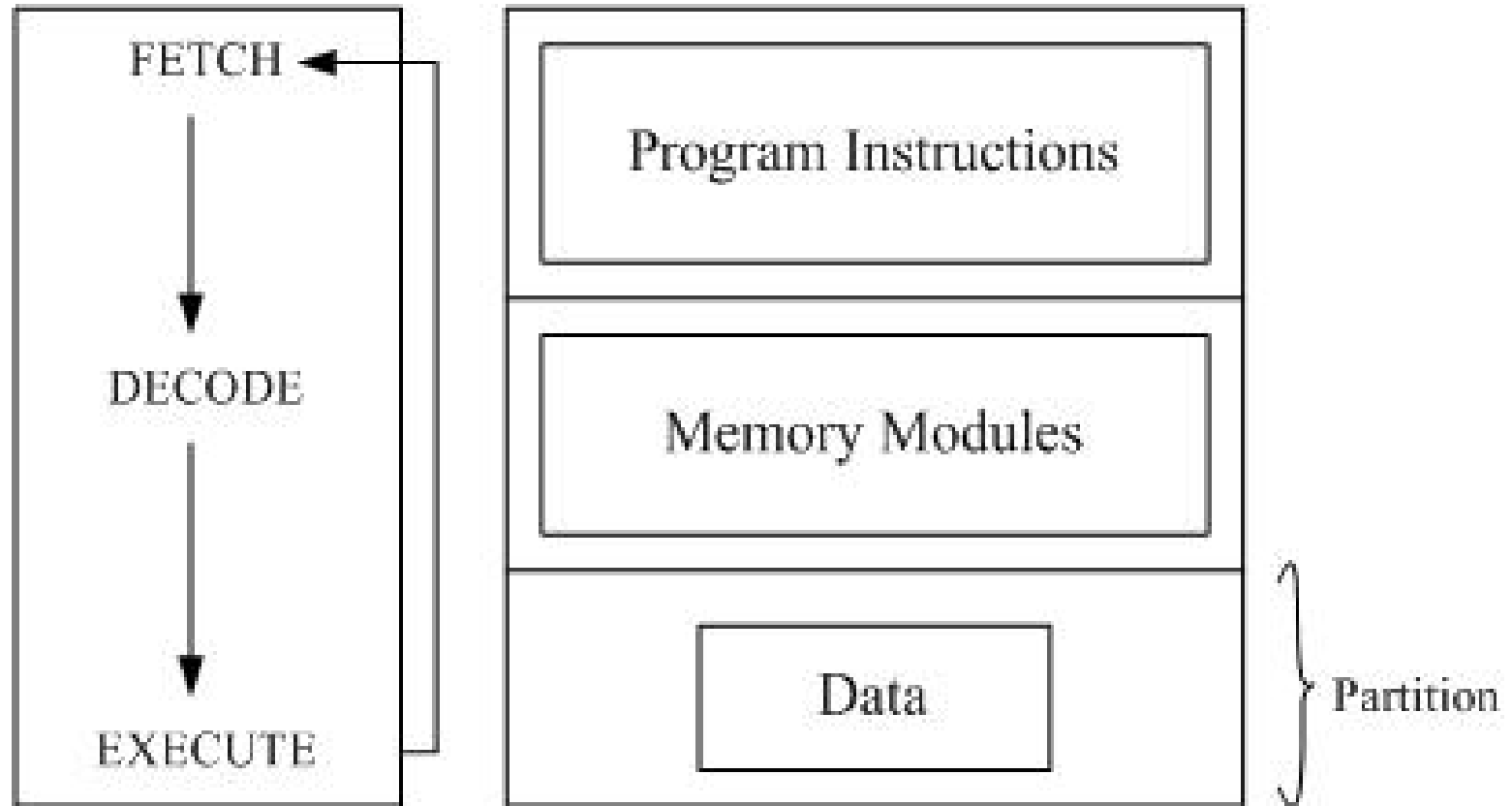


Figure 2.0

# Operational Overview

- ★ **Processor** — *executes programs, schedules and allocate processor time.*
- ★ **Memory** — *stores programs, and supports mechanisms to access data*
- ★ **Input output devices** — *supports all input and output operations*
- ★ **Communication mechanisms** - *with devices external to the system*
- ★ **Mutual exclusion** — *schedule the usage of shared device and fair access*
- ★ **Shell of an OS**
- ★ **Human computer interface (HCI)**

# The Physical Overview of a Computer System

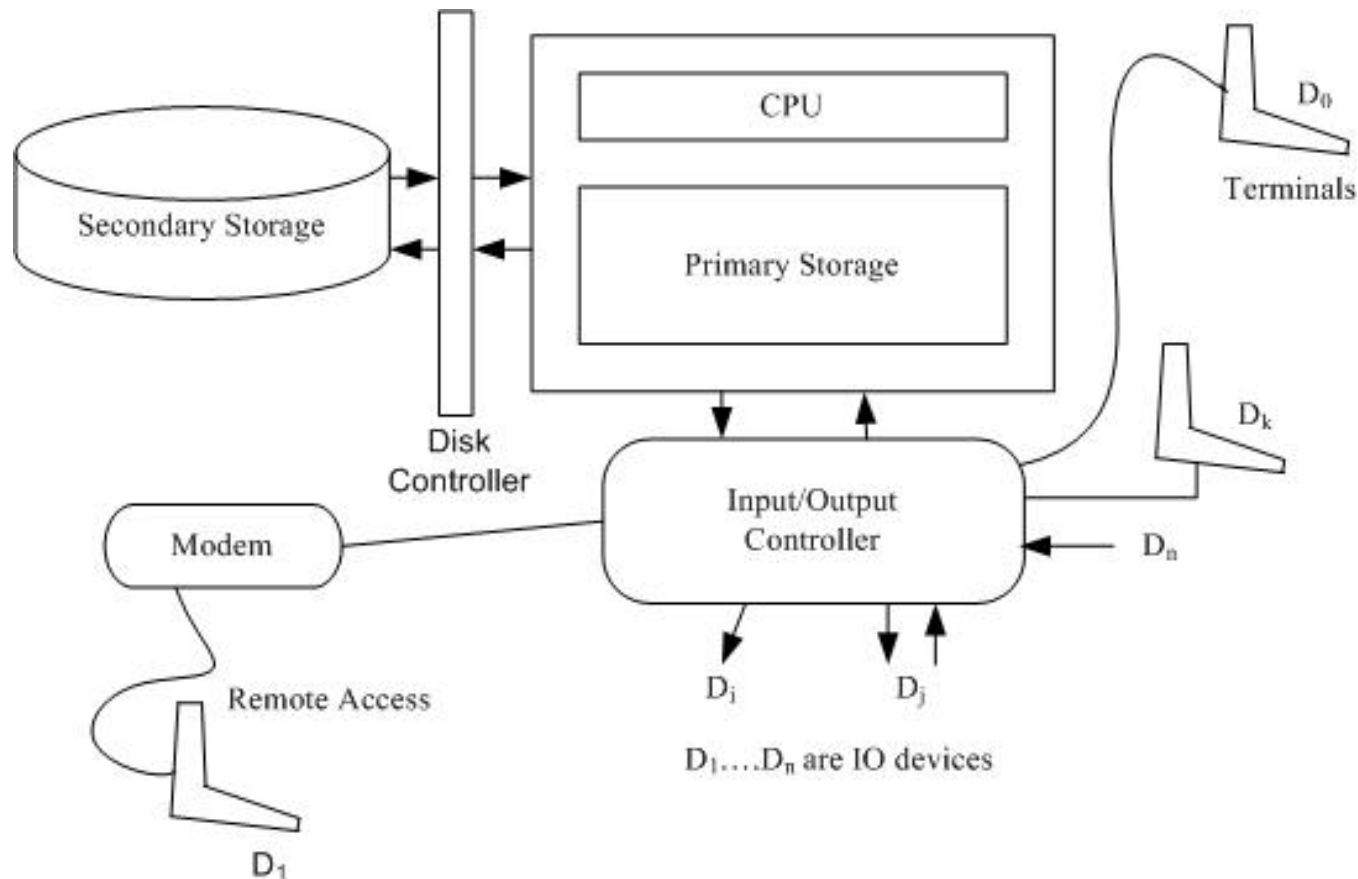


Figure 2.1

# I/O Interrupts

- ★ The peripheral devices communicate in a mode known as *Interrupt*.
- ★ The modern operation systems are *interrupt driven*.
- ★ Human input often used interrupt because human desire to guide the operation.
- ★ The OS knows which device is interrupting whenever I/O interrupts occurs.
- ★ **Eg.** Mouse click or keyboard input needs appropriate interrupt services

# Input Output Devices

The normal  
instruction cycle

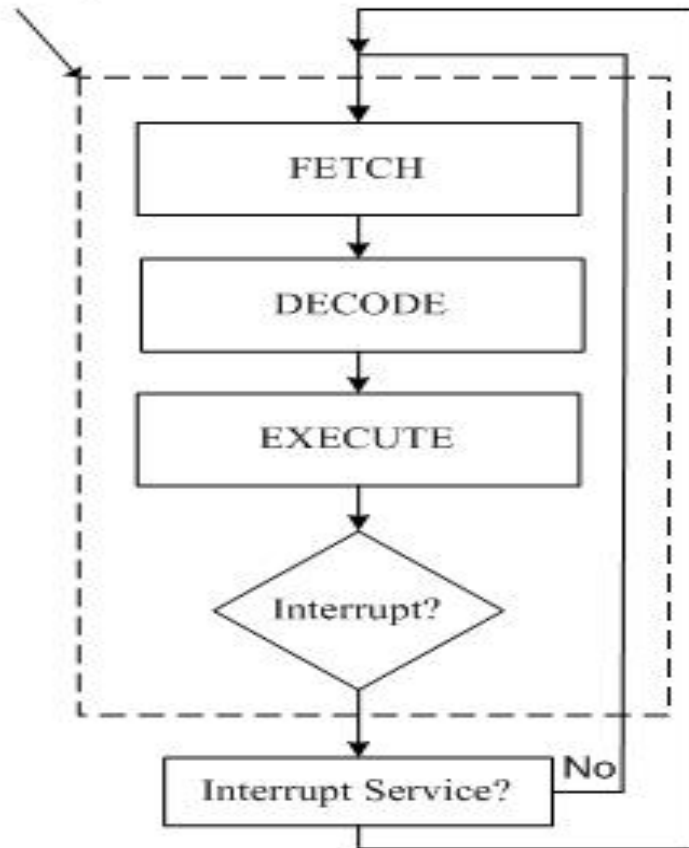


Figure 2.3

# Device-status table

- ★ OS uses table containing an entry for each I/O. This table is known as device-status table.

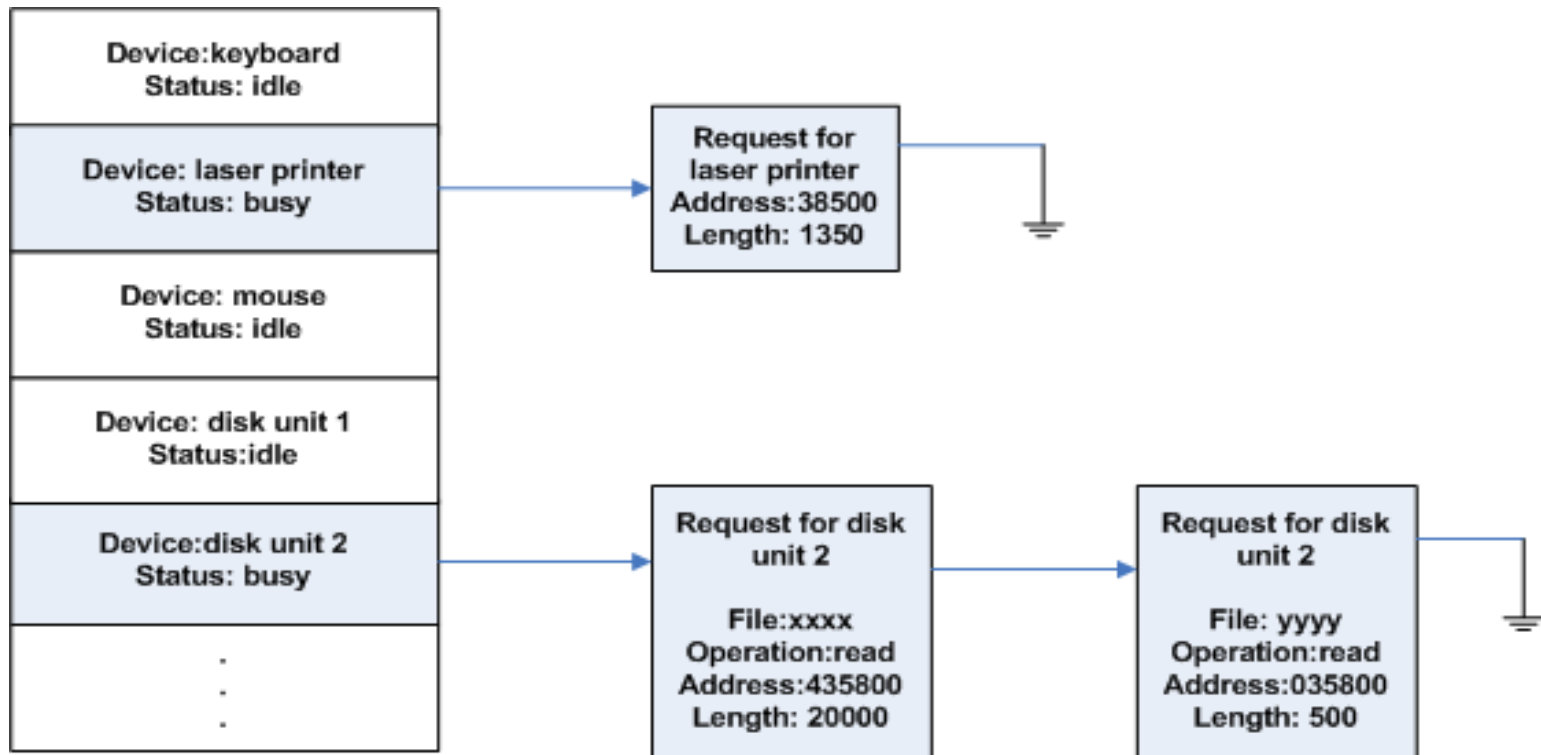


Figure 2.2



# Storage Structure

## ★ Main Memory:

- All computer program must be in main memory (also called as RAM-Random Access Memory) to be executed.
- The *load* and *store* instructions.
- The *load instruction* moves a word from main memory to register within the CPU whereas *store instruction* moves content from register to main memory.

## ★ Magnetic Disks

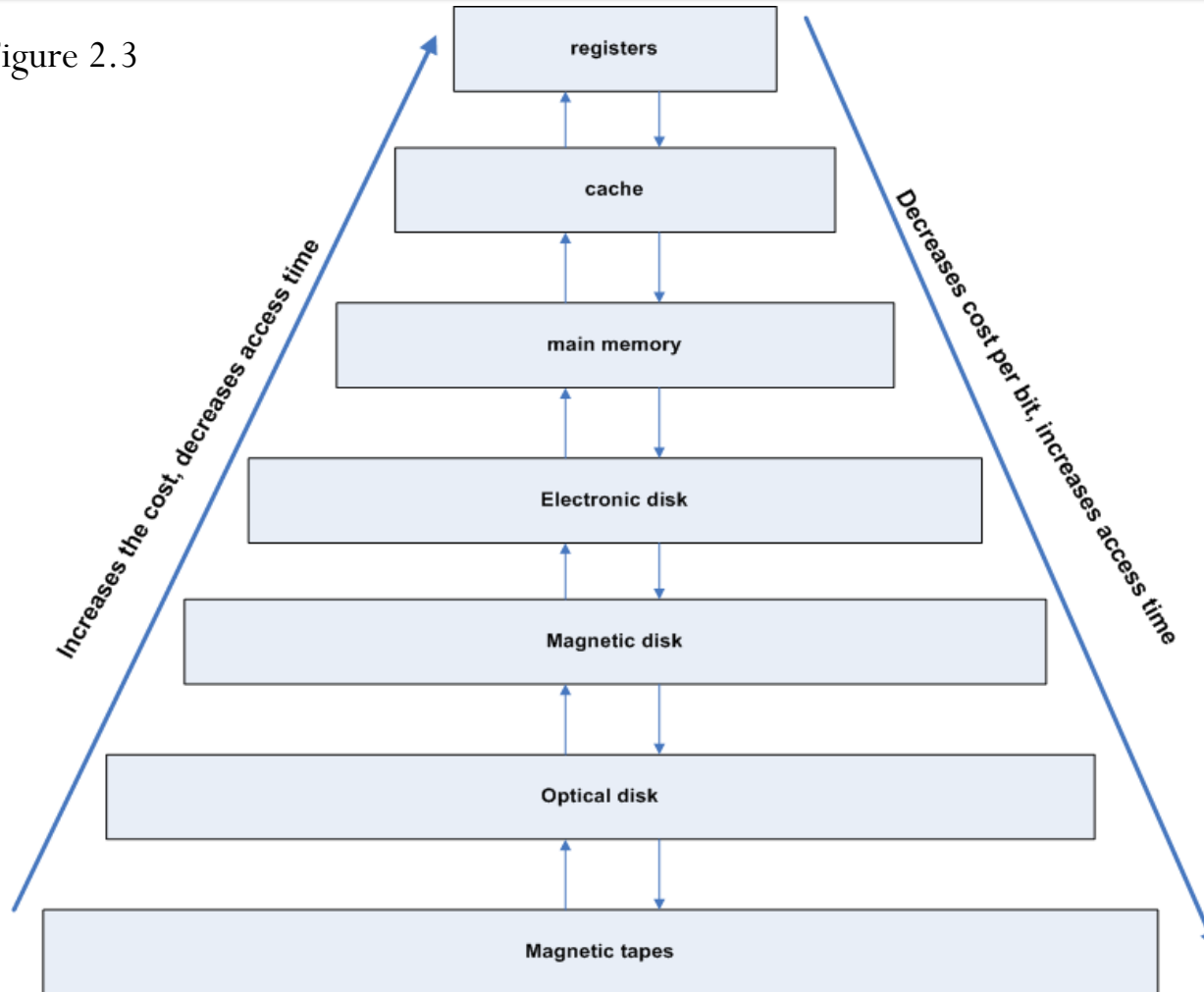
- It provides the bulk of secondary storage for modern computer systems.
- It is divided into *circular tracks* and each track is divided into many *sectors*.
- A disk drive is attached to a computer by a set of wires called an **I/O bus**.

## ★ Magnetic Tapes

- It was used as an early secondary-storage medium.
- Though it is relatively permanent and can hold large quantities of data, its access time is slower than main memory.
- Random access is about thousand times slower than the Magnetic Disks.
- The tapes are mainly used for backup, storage of infrequently used information and transferring information from one to another system.

# Storage Hierarchy

Figure 2.3



# Processes

---

- ★ Program in execution is called a *process*.
- ★ Program is a *static object* & process is a *dynamic object*.
- ★ Inter-Process Communication(IPC) forms the basis of *distributed computing*.
- ★ The machine seeking the service is *client* and the machine offering the service is *server*.

# Network Communication

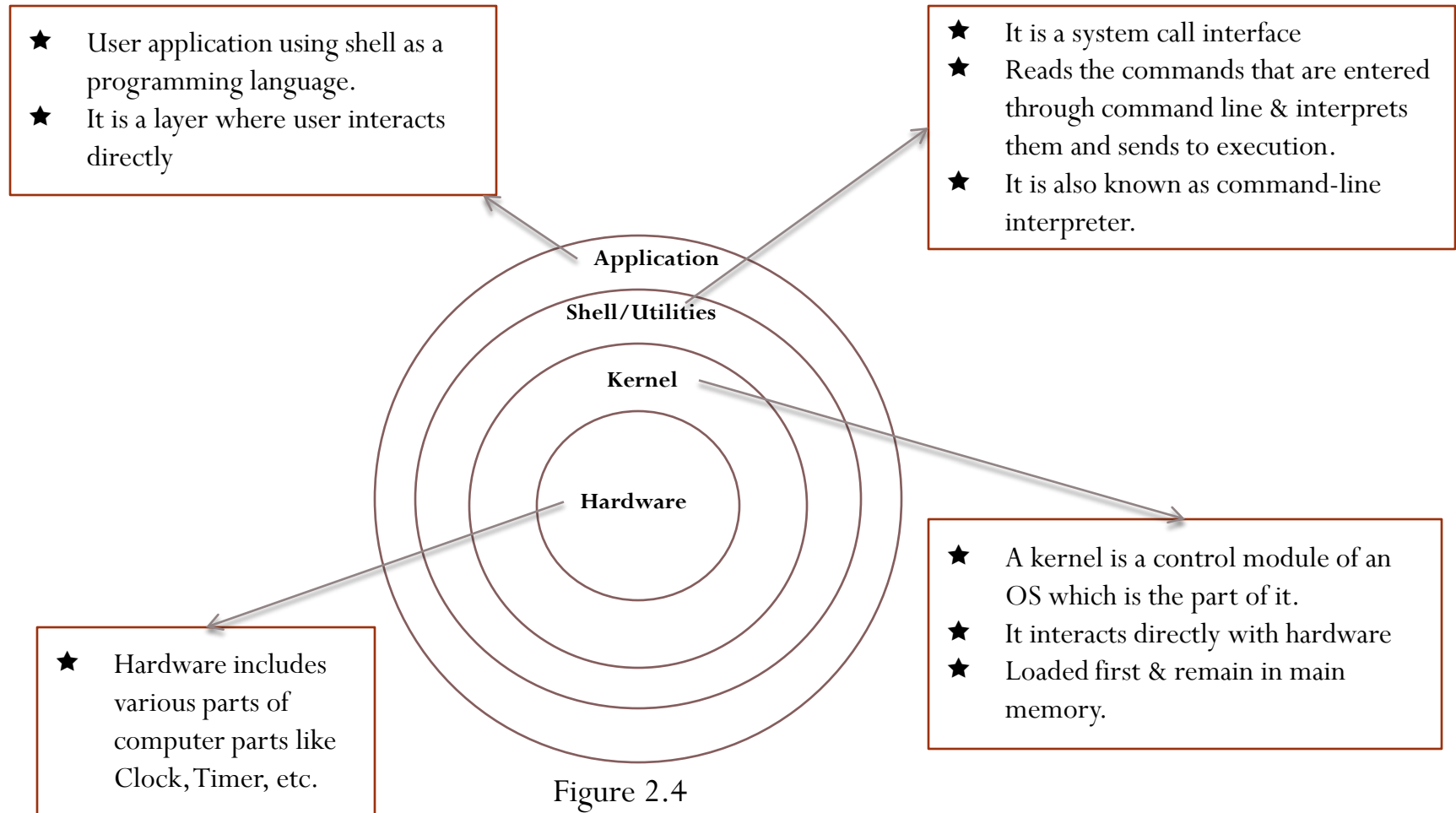
- ★ OS needed in **computers communication network** to enable the *remote resource sharing*.
- ★ *Program, files & data* could be moved around and *accessed*.
- ★ Leads to protocols like *FTP, telnet & RPC*.
- ★ The communication links are controlled by **communication processors(CPs)** in the communication subsystem.
- ★ CPs are responsible for *defining interface* through which the *sites communicate over the network* as well as *transferring information among the various sites*.

# Operating System Structure

---

- ★ Not all the systems have the same structure, however many modern OS supports the same system components.
- ★ System Components:
  - Process Management
  - Memory Management(includes both main memory & secondary-storage)
  - File Management
  - I/O System Management
  - Networking
  - Protection system
  - Command-Interpreter System

# UNIX OS Structure



# Any Questions? ....

---

Code snippet:

**if(Questions?)**

**ASK YOUR DOUBT;**

**else**

**THANK YOU & Meet You in Next Session;**