The picture can't be displayed.

Chapter 1: Introduction

Objectives

- To provide a grand tour of the **major operating systems components**
- To provide coverage of **basic computer system** organization

What is an Operating System?

- A program that acts as an **intermediary** between a user of a computer and the computer hardware.
- Operating system goals:
  - Execute user programs and make solving user problems **easier**.
  - Make the computer system **convenient** to use.
- Use the computer hardware in an **efficient** manner.

Software

The two most common types of software are :

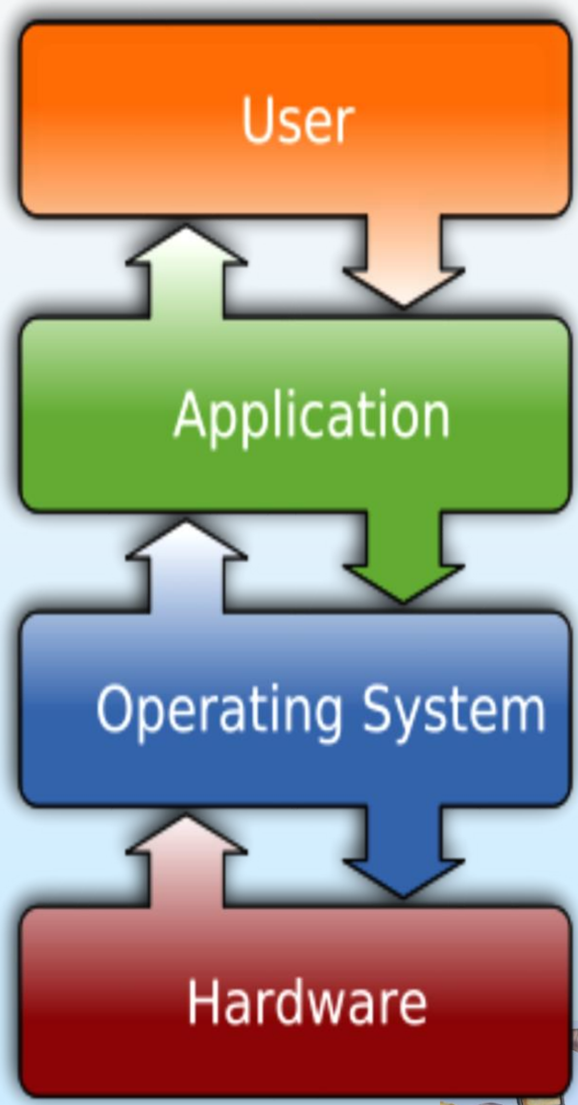
- System software
- Application software.

What is System Software?

System Software refers to the operating system and all utility programs that manage computer resources at a low level. Systems software includes compilers, loaders, linkers, and debuggers.

What is Application Software?

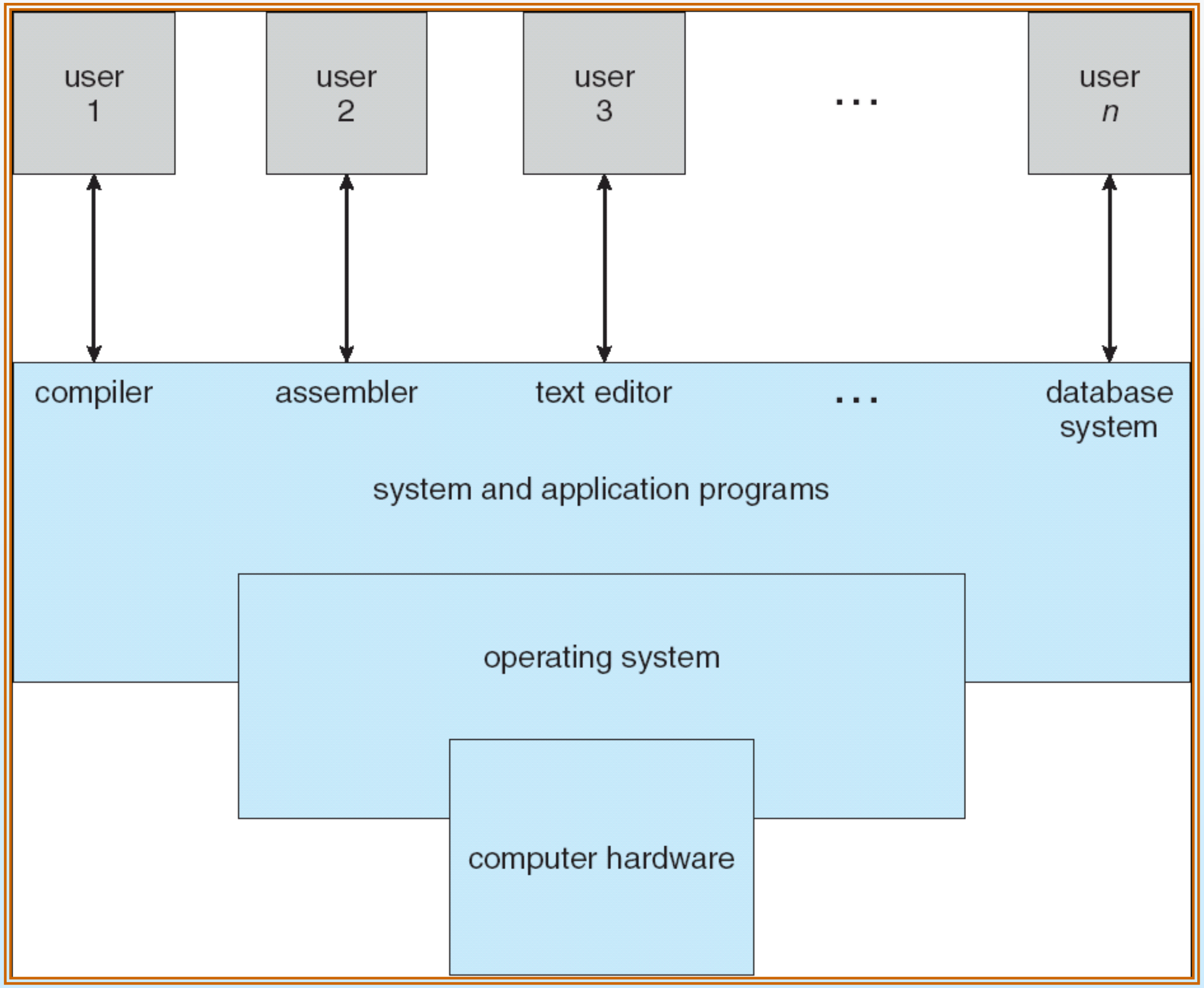
Applications software comprises programs designed for an end user, such as word processors, database systems, and spreadsheet programs.



Computer System Structure

- Computer system can be divided into four components
  - **Hardware** – provides basic computing resources
    - CPU, memory, I/O devices
  - **Operating system**
    - Controls and coordinates use of hardware among various applications and users
  - **Application programs** – define the ways in which the system resources are used to solve the computing problems of the users
    - Word processors, compilers, web browsers, database systems, video games
  - **Users**
    - People, machines, other computers

Four Components of a Computer System



Operating System Definition

- OS is a **resource allocator**
  - Manages all resources
  - Decides between conflicting requests for efficient and fair resource use

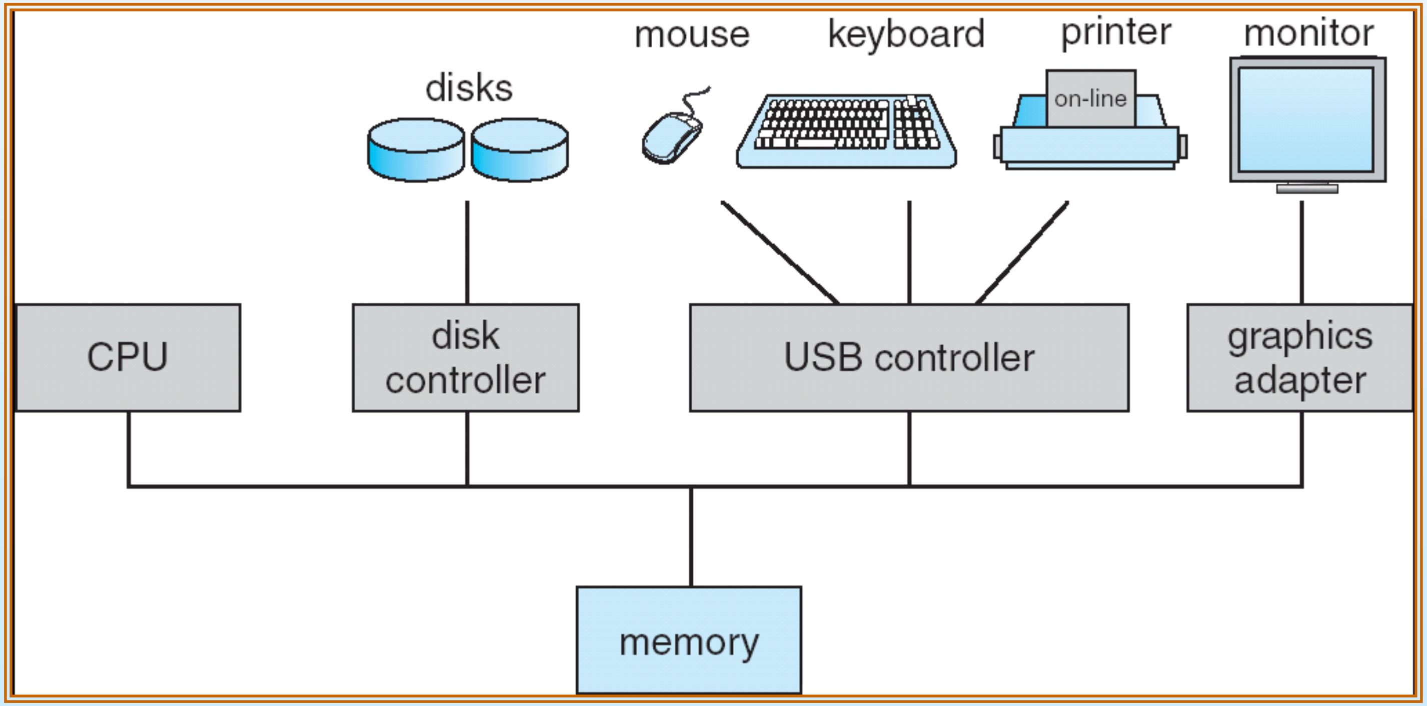
- OS is a **control program**
  - Controls execution of programs to prevent errors and improper use of the computer

Operating System Definition (Cont.)

- No universally accepted definition
- "Everything a vendor ships when you order an operating system" is good approximation
  - But varies wildly
- "The one program running at all times on the computer" is the **kernel.** Everything else is either a system program (ships with the operating system) or an application program

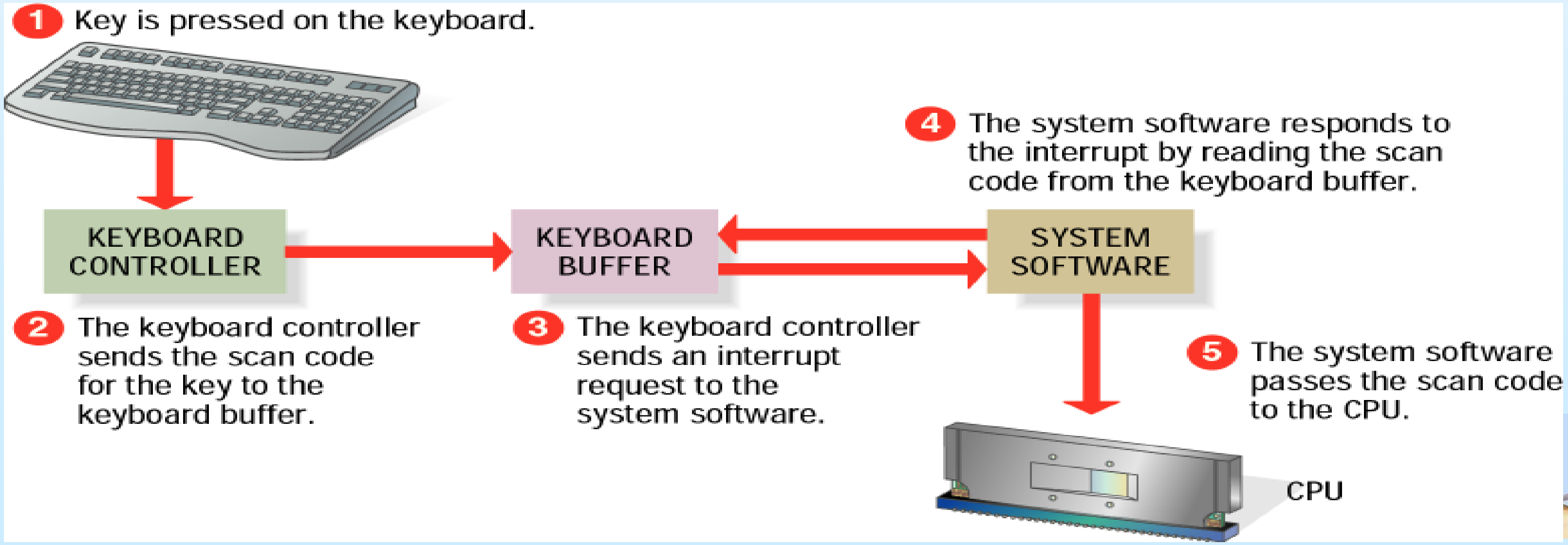
Computer System Organization

- Computer-system operation
  - One or more CPUs, Memory, device controllers connect through a common bus which is called system bus.



Computer-System Operation

- I/O devices and the CPU can execute **concurrently**.
- Each **device controller** is in charge of a particular device type.
- Each device controller has a local buffer.
- CPU moves data from/to main memory to/from local buffers
- Device controller informs CPU that it has finished its operation by causing an *interrupt*.



1 Key is pressed on the keyboard.

KEYBOARD CONTROLLER

KEYBOARD BUFFER

SYSTEM SOFTWARE

CPU

2 The keyboard controller sends the scan code for the key to the keyboard buffer.

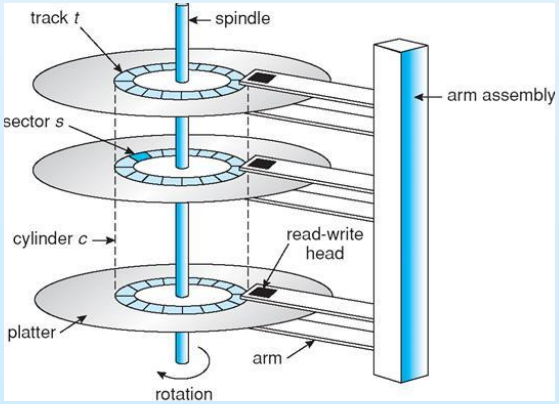
3 The keyboard controller sends an interrupt request to the system software.

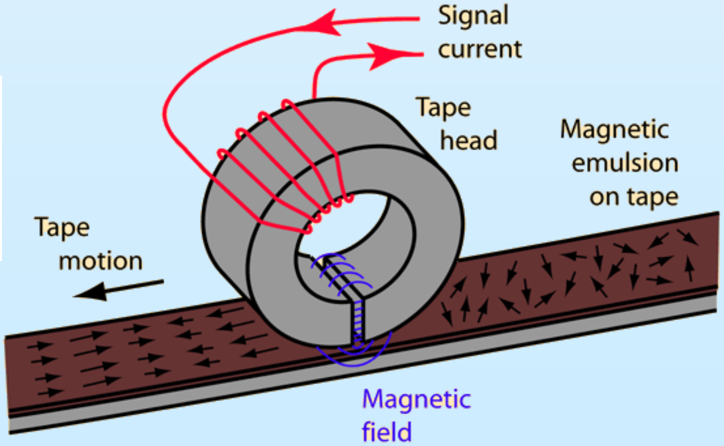
4 The system software responds to the interrupt by reading the scan code from the keyboard buffer.

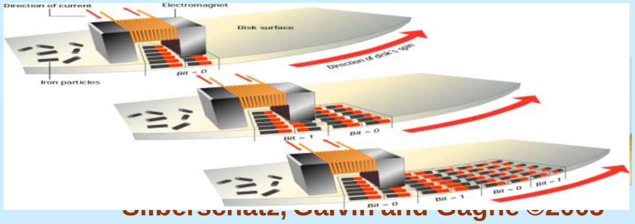
5 The system software passes the scan code to the CPU.

Storage Structure

- **Main memory** – only large storage media that the CPU can access directly.
- **Secondary storage** – extension of main memory that provides large nonvolatile storage capacity.
  - **Magnetic disks** – rigid metal or glass platters covered with magnetic recording material
    - Disk surface is logically divided into *tracks*, which are subdivided into *sectors*.
    - The *disk controller* determines the logical interaction between the device and the computer.
  - Magnetic Tapes

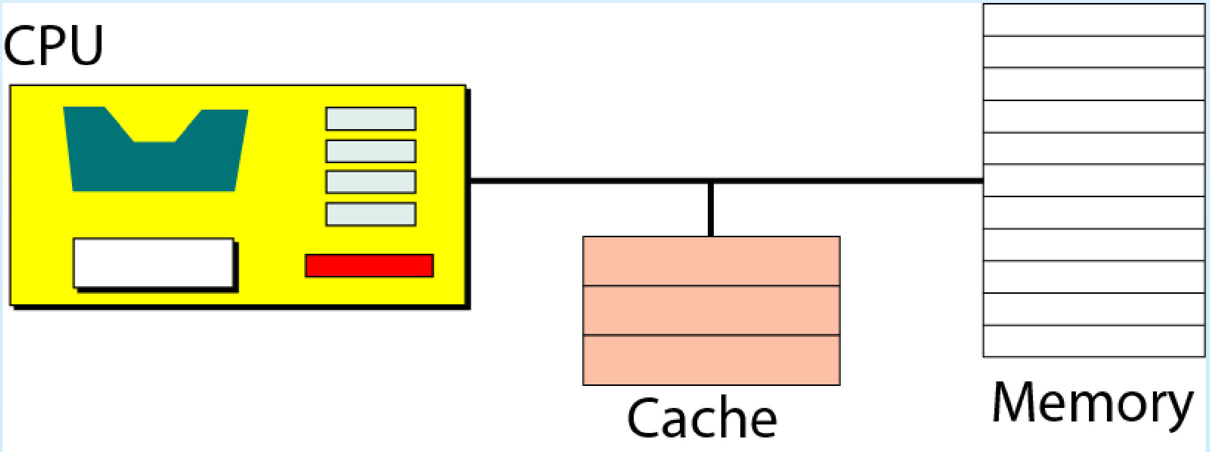




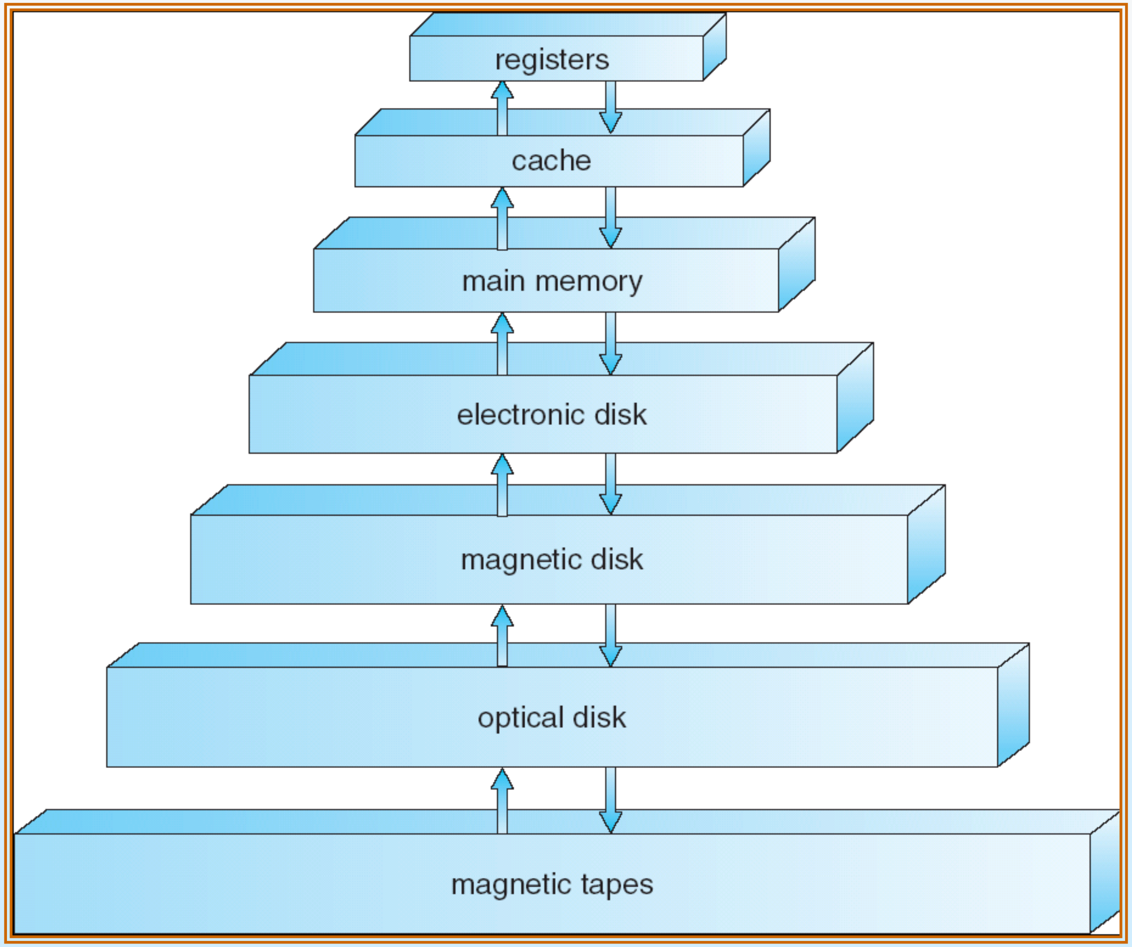


Storage Hierarchy

- Storage systems organized in **hierarchy**.
  - Speed
  - Cost
  - Volatility
  - Size
- *Caching* – Information in use copied from slower to faster storage temporarily

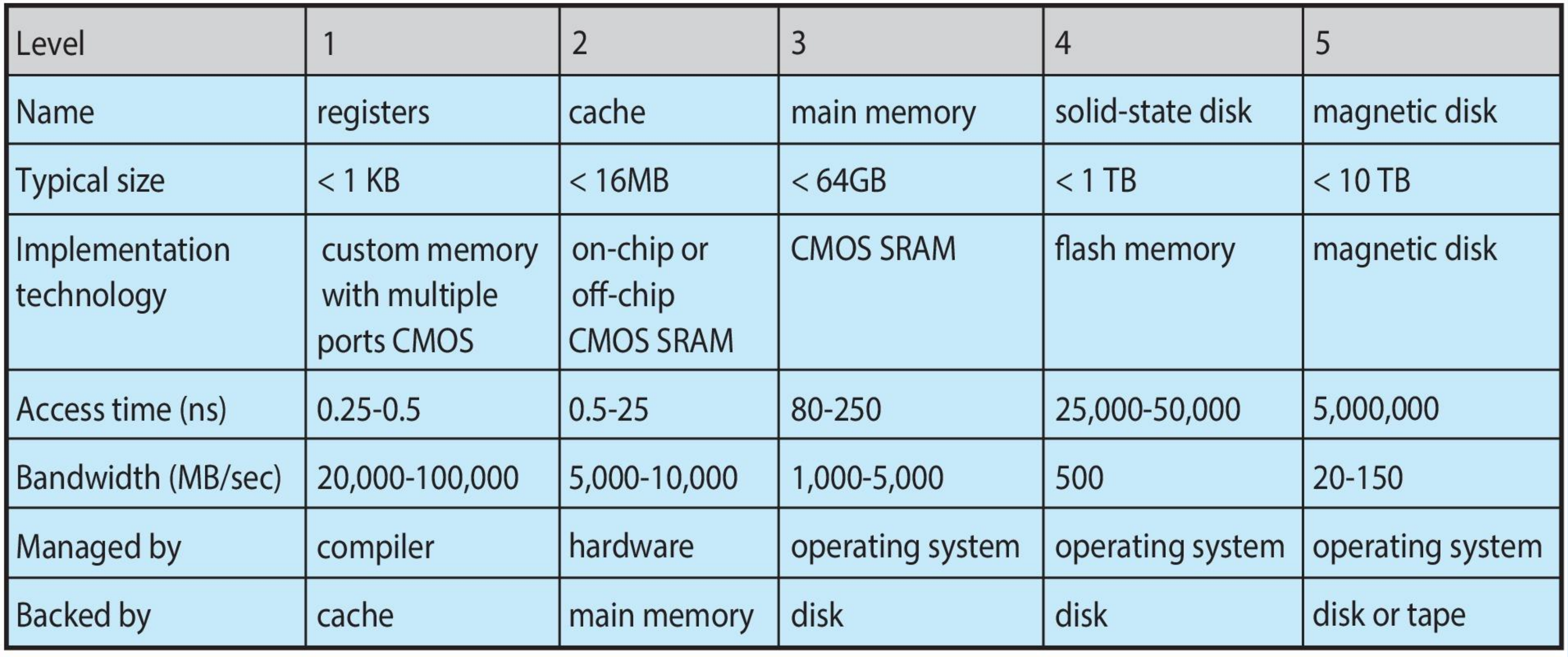


Storage-Device Hierarchy



Performance of Various Levels of Storage

- Movement between levels of storage hierarchy can be explicit or implicit



Level	1	2	3	4	5
Name	registers	cache	main memory	solid-state disk	magnetic disk
Typical size	< 1 KB	< 16MB	< 64GB	< 1 TB	< 10 TB
Implementation technology	custom memory with multiple ports CMOS	on-chip or off-chip CMOS SRAM	CMOS SRAM	flash memory	magnetic disk
Access time (ns)	0.25-0.5	0.5-25	80-250	25,000-50,000	5,000,000
Bandwidth (MB/sec)	20,000-100,000	5,000-10,000	1,000-5,000	500	20-150
Managed by	compiler	hardware	operating system	operating system	operating system
Backed by	cache	main memory	disk	disk	disk or tape

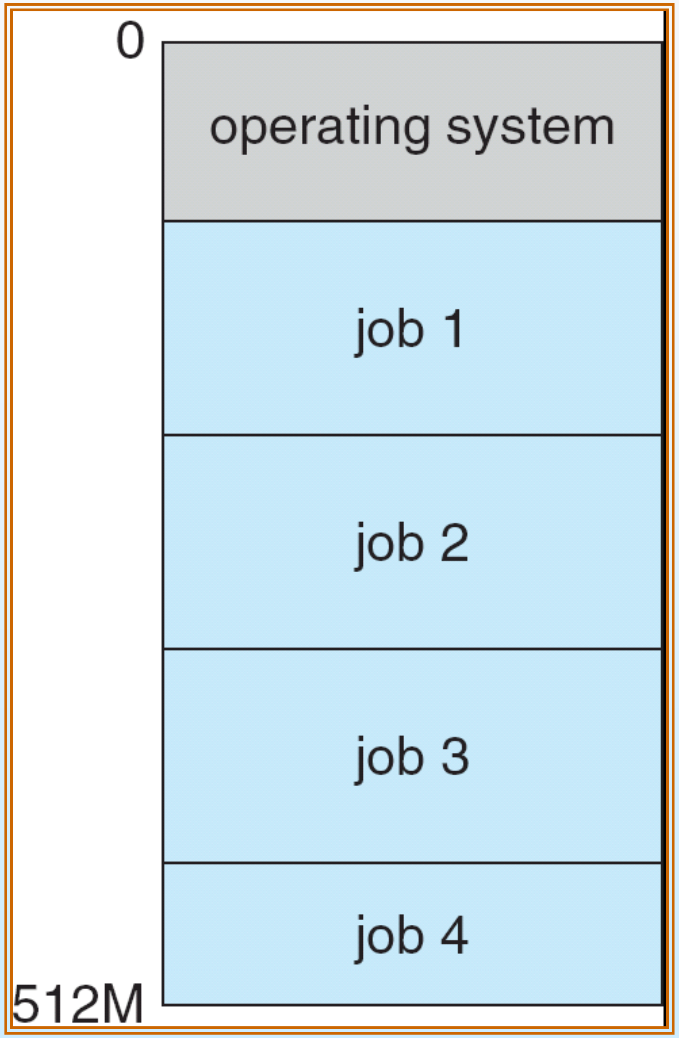
Operating System Structure

- **Multiprogramming** needed for efficiency
  - Single user cannot keep CPU and I/O devices busy at all times
  - Multiprogramming organizes jobs (code and data) so CPU always has one to execute
  - A subset of total jobs in system is kept in memory
  - One job selected and run via **job scheduling**
  - When it has to wait (for I/O for example), OS switches to another job

Operating System Structure (cont)

- **Timesharing (multitasking)** is logical extension in which CPU switches jobs so frequently that users can interact with each job while it is running, creating **interactive** computing
  - **Response time** should be < 1 second
  - Each user has at least one program executing in memory ⇨**process**
  - If several jobs ready to run at the same time ⇨ **CPU scheduling**
  - If processes don’t fit in memory, **swapping** moves them in and out to run
  - **Virtual memory** allows execution of processes not completely in memory

Memory Layout for Multiprogrammed System

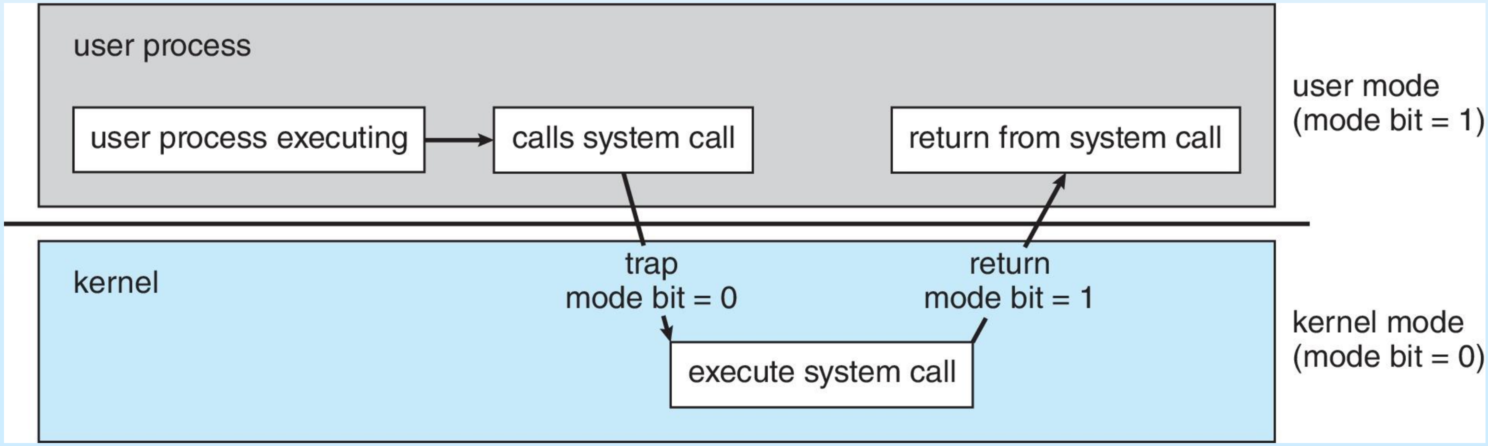


Hardware Protection

- With sharing many processes could be adversely affected by a **bug** in one program. So, a properly designed OS must ensure that an **incorrect program** can not run and also can not cause other programs to execute incorrectly.
- Many errors detected by hardware can be handled by OS.
- Hardware generates **interrupt**.
- Software error handled by **exception** or **trap.**

Hardware Protection

- **Dual-mode** operation allows OS to protect itself and other system components
  - **User mode (1)** and **Monitor/System mode (0)**
  - **Mode bit** provided by hardware
    - Provides ability to distinguish when system is running user code or system code
    - Some instructions designated as **privileged**, only executable in system mode



Hardware Protection

- At system boot time, the hardware starts in **kernel mode.**
- The operating system is then loaded and starts user applications in **user mode**.
- Whenever a trap or interrupt occurs, the hardware switches from **user mode to kernel mode** (that is, changes the state of the mode bit to 0).
- Thus, whenever the operating system gains control of the computer, it is in **kernel mode**.
- The system always **switches to user mode** (by setting the mode bit to 1) before passing control to a user program.

Hardware Protection

- I/O Protection
  - To prevent a user from performing illegal I/O, we define all I/O instructions to be privileged instruction.

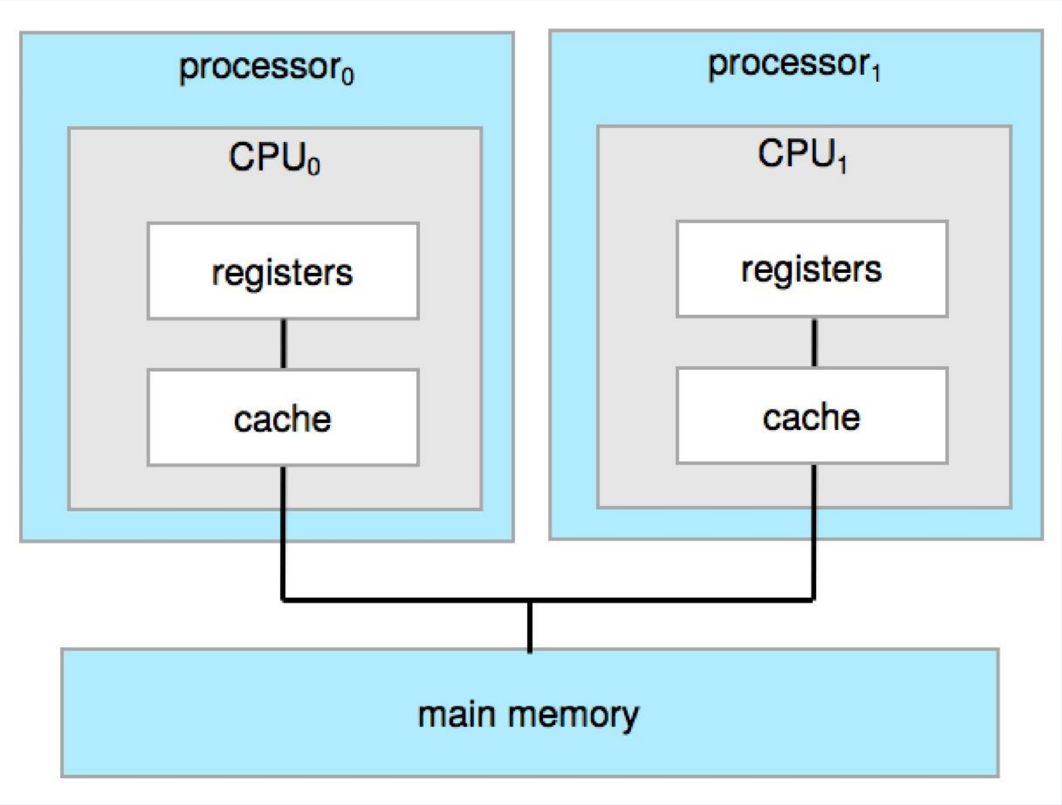
Hardware Protection

- Memory Protection
  - A user program might overwrite instructions in the interrupt service routine. So, we must protect the **interrupt vector** from modification by a user program.

Hardware Protection

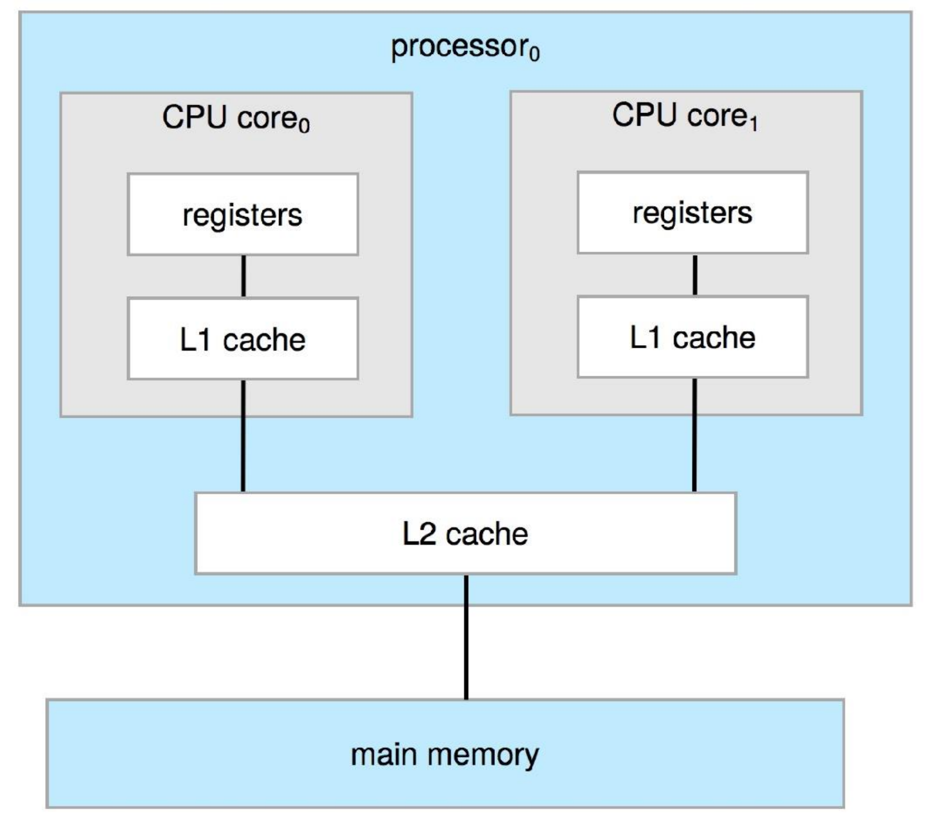
- CPU Protection
  - We must prevent a user program getting stuck in an **infinite loop** and never returning the control to the OS. To accomplish this goal, we can use a timer.

Symmetric Multiprocessing Architecture



A Dual-Core Design

- Multi-chip and **multicore**
- Systems containing all chips
  - Chassis containing multiple separate systems



Chapter 2

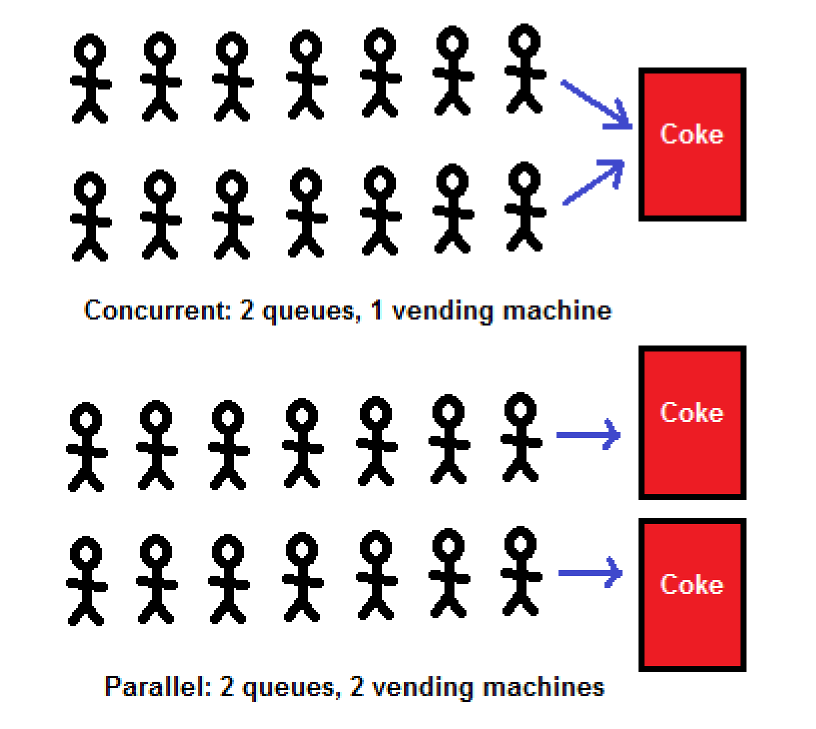
OS STRUCTURES

Process Management

- **A process is a program in execution**. It is a unit of work within the system. Program is a *passive entity*, process is an *active entity*.
- Process needs resources to accomplish its task
  - CPU, memory, I/O, files
  - Initialization data
- Process termination requires reclaim of any reusable resources
- Processes maybe single-threaded or multi-threaded
  - Process executes instructions sequentially, one at a time, until completion
- Typically a system has many processes running concurrently on one or more CPUs
  - Concurrency by multiplexing the CPUs among the processes / threads

Concurrency

www.questpond.com
CONCURRENCY
S. KSHIRSAGAR



Concurrent: 2 queues, 1 vending machine

Parallel: 2 queues, 2 vending machines

Process Management Activities

The operating system is responsible for the following activities in connection with process management:

□ Creating and deleting both user and system processes

□ Suspending and resuming processes

□ Providing mechanisms for process synchronization

□ Providing mechanisms for process communication

□ Providing mechanisms for deadlock handling

Memory Management

- All **data** in memory before and after processing
- All **instructions** in memory in order to execute
- Memory management determines what is in memory when
  - Optimizing CPU utilization.
- **Memory management activities**
  - Keeping track of which parts of memory are currently being used and by whom
  - Deciding which processes (or parts thereof) and data to move into and out of memory
  - Allocating and de-allocating memory space as needed

Storage Management

- OS provides **uniform, logical view** of information storage
  - Abstracts physical properties to logical storage unit - **file**
  - Each medium is controlled by device (i.e., disk drive, tape drive)
    - Varying properties include access speed, capacity, data-transfer rate, access method (sequential or random)
- **File-System management**
  - Files usually organized into directories
  - Access control on most systems to determine who can access what
  - **OS activities** include
    - Creating and deleting files and directories
    - Primitives to manipulate files and directories
    - Mapping files onto secondary storage
    - Backup files onto stable (non-volatile) storage media

Mass-Storage Management

- Because main memory is too small **to accommodate all data and programs, and its data is lost when power is lost**, the computer system must provide secondary storage to back up main memory.
- Proper management of disk storage is of central importance to a computer system.
- **OS activities**
  - Mounting and Unmounting
  - Free-space management
  - Storage allocation
  - Disk scheduling
  - Partitioning
  - Protection

Protection and Security

- **Protection** – refers to a mechanism for controlling the access of programs, processes or user's to the resources defined by a computer systems.
- **Security** – defense of the system against internal and external attacks
    - Huge range, including denial-of-service, worms, viruses, identity theft, theft of service
- Systems generally first distinguish among users, to determine who can do what
    - User identities (**user IDs**, security IDs) include name and associated number, one per user
    - User ID then associated with all files, processes of that user to determine access control

The picture can't be displayed.

End of Lecture 1