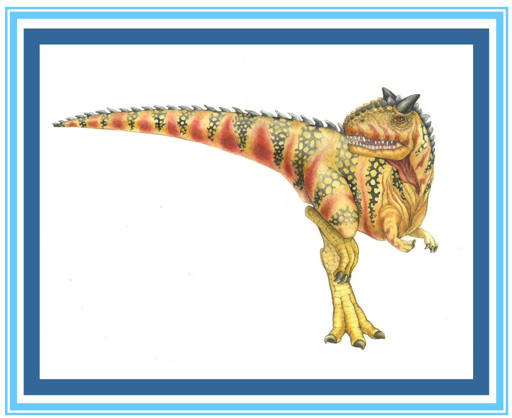
Chapter 1: Introduction

Lecture #02



Recap

- What is an Operating System
- What Operating Systems Do
- Computer-System Organization
- Computer-System Architecture
- Operating-System Structure

Objectives

- Storage Structure
- Operating System Operations
- Operating System architecture

Storage Definitions and Notation Review

The basic unit of computer storage is the **bit**. A bit can contain one of two values, 0 and 1. All other storage in a computer is based on collections of bits. Given enough bits, it is amazing how many things a computer can represent: numbers, letters, images, movies, sounds, documents, and programs, to name a few. A **byte** is 8 bits, and on most computers it is the smallest convenient chunk of storage. For example, most computers don't have an instruction to move a bit but do have one to move a byte. A less common term is **word**, which is a given computer architecture's native unit of data. A word is made up of one or more bytes. For example, a computer that has 64-bit registers and 64-bit memory addressing typically has 64-bit (8-byte) words. A computer executes many operations in its native word size rather than a byte at a time.

Computer storage, along with most computer throughput, is generally measured and manipulated in bytes and collections of bytes.
A **kilobyte**, or **KB**, is 1,024 bytes
a **megabyte**, or **MB**, is $1{,}024^2$ bytes
a **gigabyte**, or **GB**, is $1{,}024^3$ bytes
a **terabyte**, or **TB**, is $1{,}024^4$ bytes
a **petabyte**, or **PB**, is $1{,}024^5$ bytes

Computer manufacturers often round off these numbers and say that a megabyte is 1 million bytes and a gigabyte is 1 billion bytes. Networking measurements are an exception to this general rule; they are given in bits (because networks move data a bit at a time).

Storage Structure

- General purpose computers run their programs from random access memory called main memory.
- Interaction with main memory is achieved through a series of load and store instructions to specific memory addresses.
  - Load instruction moves a word from main memory to an internal register of the CPU.
  - Store instruction moves content of a register to main memory.
  - The CPU automatically loads instructions from main memory for execution.
- Instruction execution cycle works as:
  - Fetch instruction from memory and stores instruction in the instruction register.
  - Decodes instruction and may cause operands to be fetched from memory and store in the internal register.
  - After instruction on operands executed, result is stored back in memory.

Storage Structure

- The memory unit only sees a stream of memory addresses; it does not know how they are generated.
- We are only interested in sequence of memory addresses generated by the running program.
- Ideally we want all the programs and data to reside in the main memory permanently, but it is not possible for the following two reasons:
  - Main memory is too small to store all the programs and data permanently.
  - Main memory is volatile storage device that loses its contents when power is turned off.

Storage Structure

- For this reason most computer systems provide secondary storage as an extension of main memory.
  - The main requirement of secondary storage is that it can hold large quantity of data.
  - Most of the secondary storage devices are magnetic disks which provide storage for both programs and data.
  - There are other types of storage devices as well for which the speed, cost, size and volatility differ.

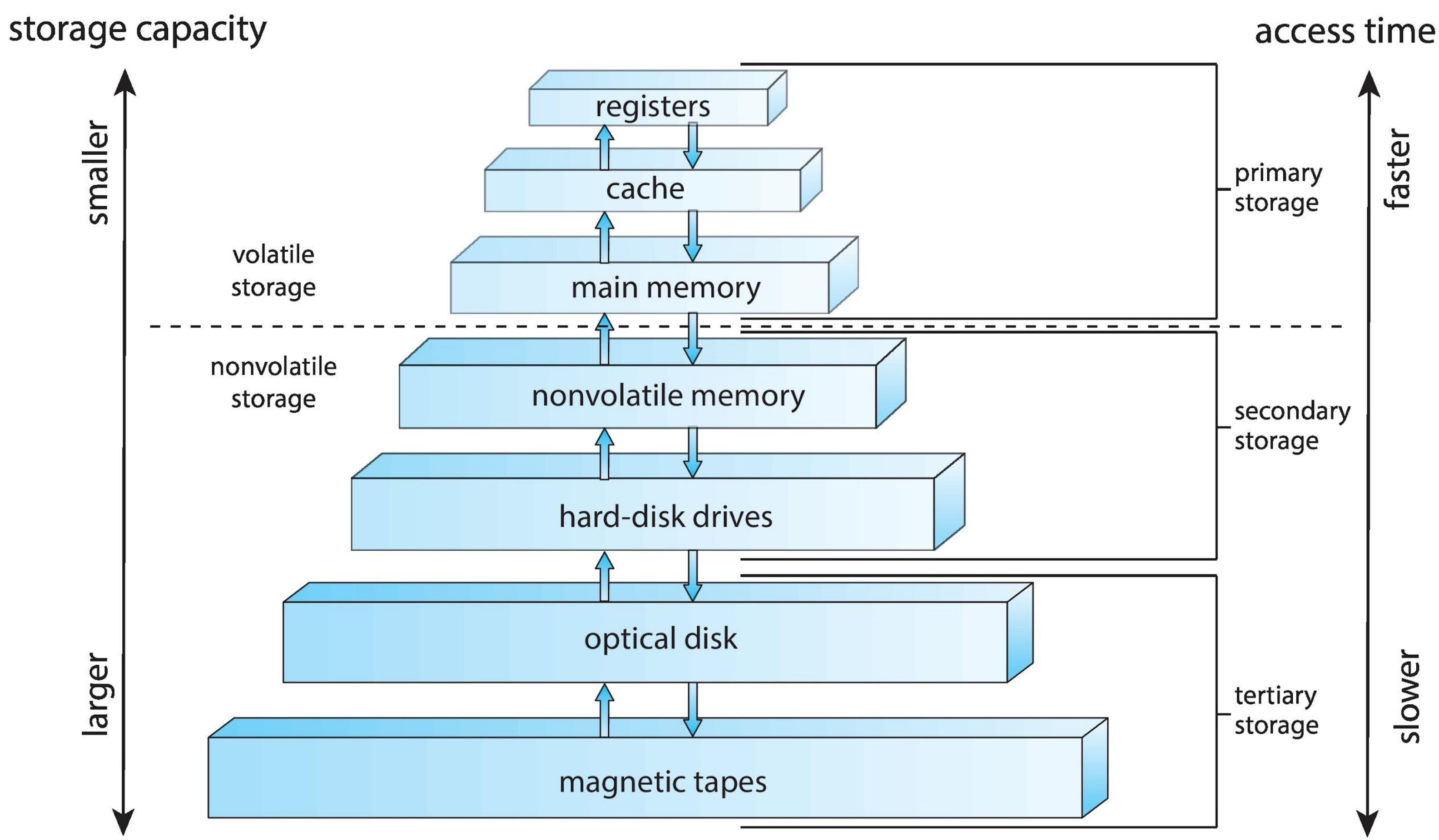
Storage Structure

- Main memory – only large storage media that the CPU can access directly
  - **Random access**
  - Typically **volatile**
- Secondary storage – extension of main memory that provides large **nonvolatile** storage capacity
- Hard 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
- **Solid-state disks** – faster than hard disks, nonvolatile
  - Various technologies
  - Becoming more popular

Storage Hierarchy

- Storage systems organized in hierarchy
  - Speed
  - Cost
  - Volatility
- **Caching** – copying information into faster storage system; main memory can be viewed as a cache for secondary storage
- **Device Driver** for each device controller to manage I/O
  - Provides uniform interface between controller and kernel

Storage-Device Hierarchy



Caching

- Important principle, performed at many levels in a computer (in hardware, operating system, software)
- Information in use copied from slower to faster storage temporarily
- Faster storage (cache) checked first to determine if information is there
  - If it is, information used directly from the cache (fast)
  - If not, data copied to cache and used there
- Cache smaller than storage being cached
  - Cache management important design problem
  - Cache size and replacement policy

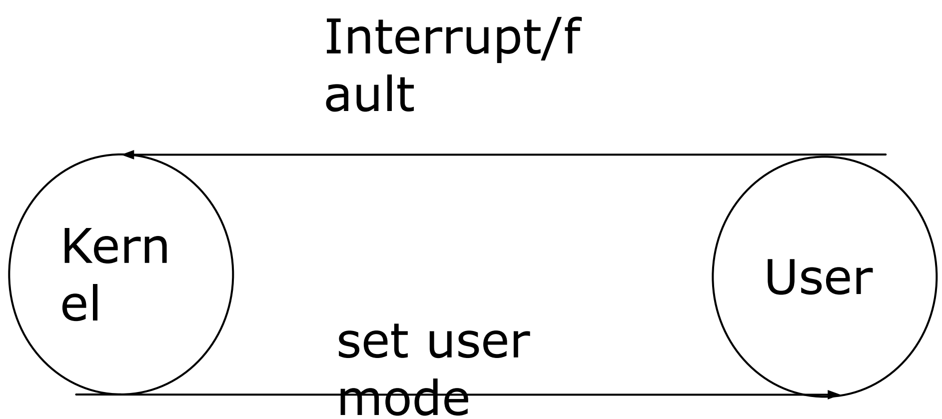
Operating-System Operations

Operating-System Operations

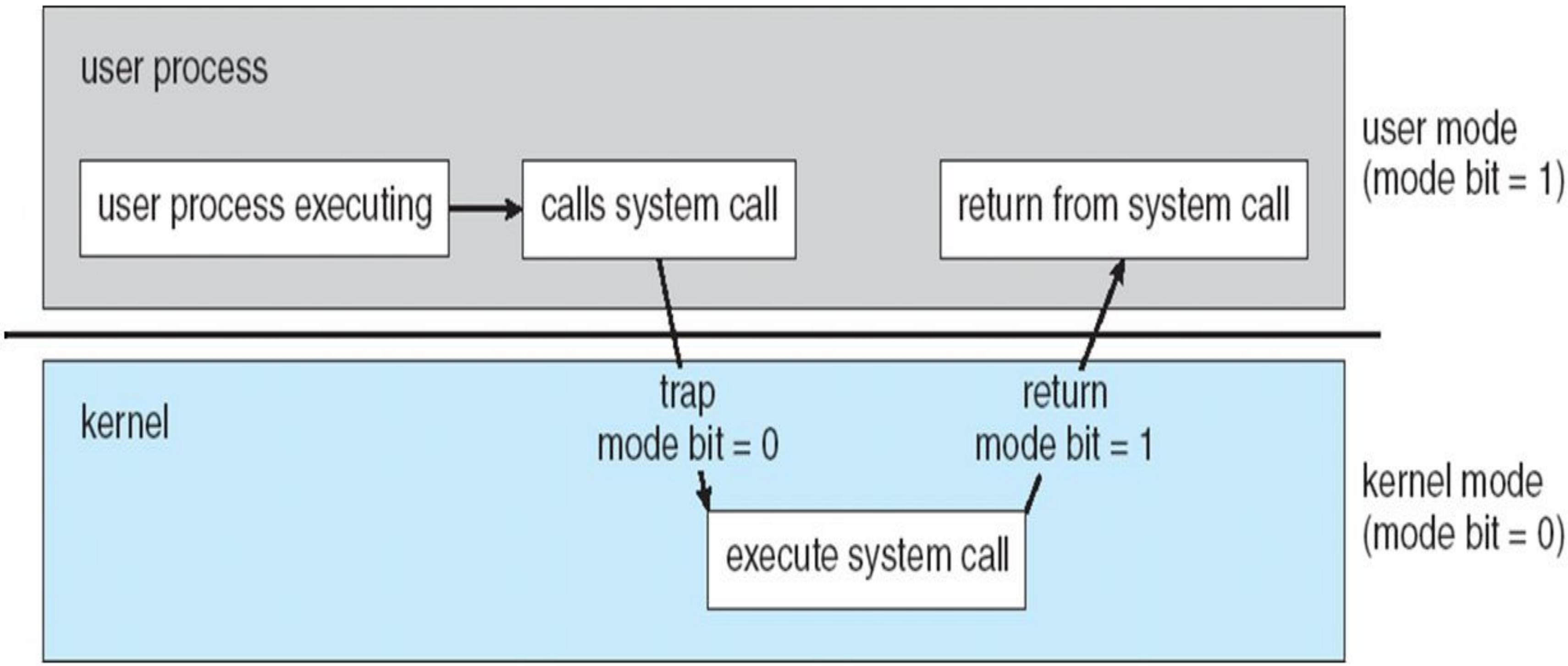
- **Dual-mode** operation allows OS to protect itself and other system components
  - **User mode** and **kernel mode**
- **User Mode**
  - The system is in user mode when the operating system is running a user application such as handling a text editor. The transition from user mode to kernel mode occurs when the application requests the help of operating system or an interrupt or a system call occurs.
  - The mode bit is set to 1 in the user mode. It is changed from 1 to 0 when switching from user mode to kernel mode.
- **Kernel Mode**
  - The system starts in kernel mode when it boots and after the operating system is loaded, it executes applications in user mode. There are some privileged instructions that can only be executed in kernel mode.
  - These are interrupt instructions, input output management etc. If the privileged instructions are executed in user mode, it is illegal and a trap is generated.
  - The mode bit is set to 0 in the kernel mode. It is changed from 0 to 1 when switching from kernel mode to user mode.

Operating-System Operations (cont.)

- **Mode bit** provided by hardware to indicate the current mode: kernel (0) or user (1).
  When an interrupt or fault occurs hardware switches to monitor/kernel mode
  - ✔ Provides ability to distinguish when system is running user code or kernel code
  - ✔ Some instructions designated as **privileged**, only executable in kernel mode
  - ✔ System call changes mode to kernel, return from call resets it to user



Transition from User to Kernel Mode

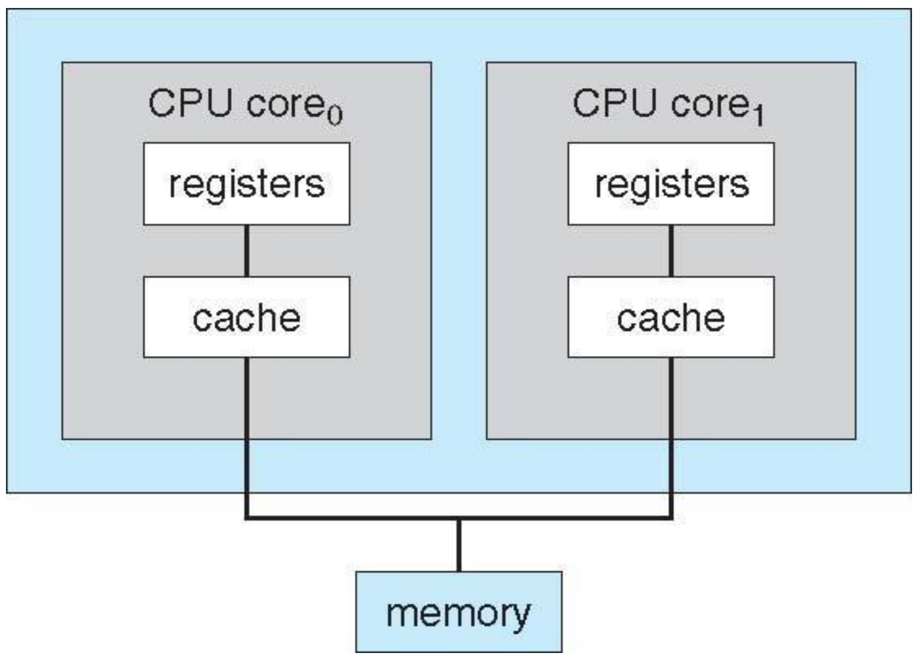


Computer-System Architecture

- Most systems use a single general-purpose processor
  - Most systems have special-purpose processors as well
- **Multiprocessors** systems growing in use and importance
  - Also known as **parallel systems**, **tightly-coupled systems**
  - Advantages include:
    1. **Increased throughput**
    2. **Economy of scale**
    3. **Increased reliability** – graceful degradation or fault tolerance
  - Two types:
    1. **Asymmetric Multiprocessing** – each processor is assigned a specie task.
    2. **Symmetric Multiprocessing** – each processor performs all tasks

A Dual-Core Design

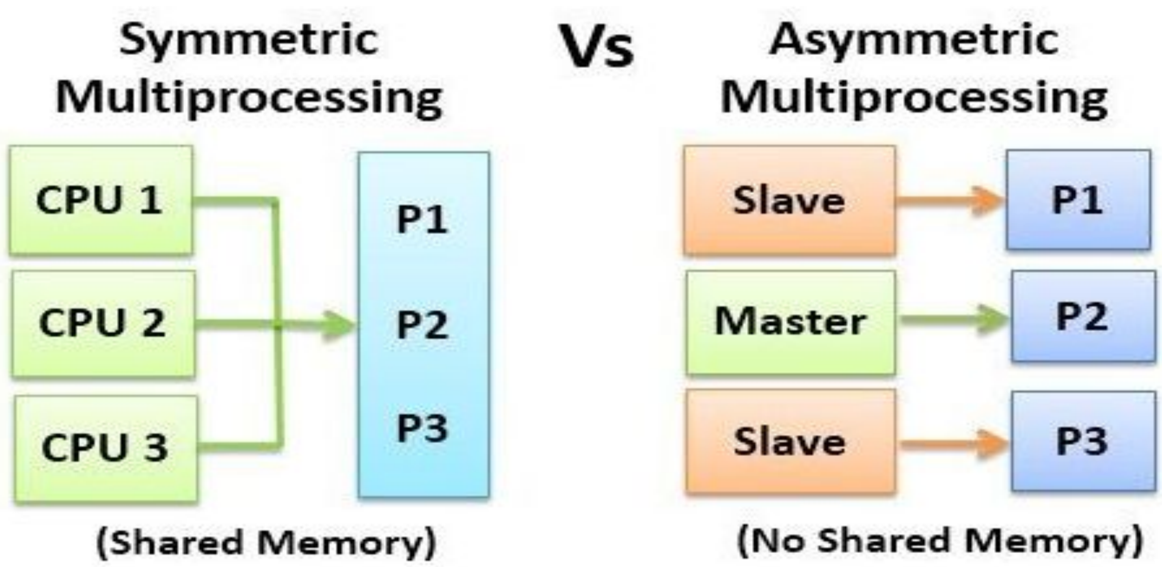
- Multi-chip and **multicore**
- Systems containing all chips
  - Chassis containing multiple separate systems



Types of Multiprocessors

Two types:

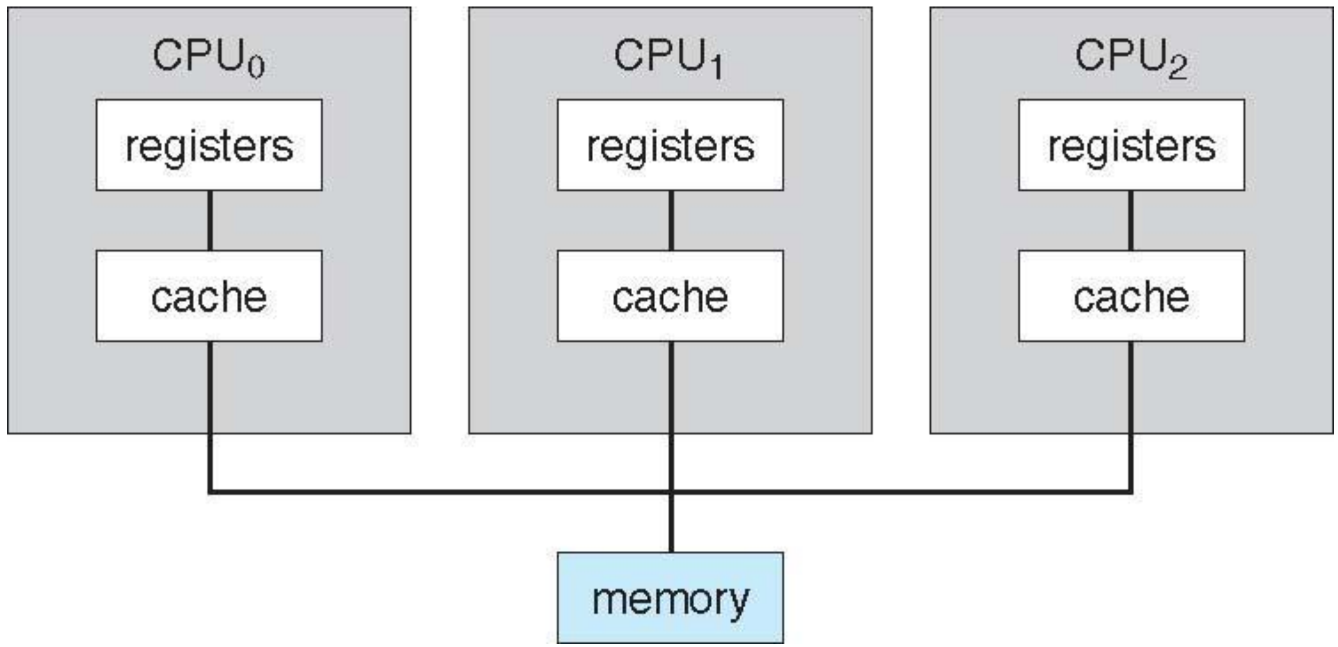
1. **Asymmetric Multiprocessing** – each processor is assigned a specie task.
2. **Symmetric Multiprocessing** – each processor performs all tasks



Types of Multiprocessors

BASIS FOR COMPARISON	SYMMETRIC MULTIPROCESSING	ASYMMETRIC MULTIPROCESSING
Basic	Each processor run the tasks in Operating System.	Only Master processor run the tasks of Operating System.
Process	Processor takes processes from a common ready queue, or there may be a private ready queue for each processor.	Master processor assign processes to the slave processors, or they have some predefined processes.
Architecture	All processor in Symmetric Multiprocessing has the same architecture.	All processor in Asymmetric Multiprocessing may have same or different architecture.
Communication	All processors communicate with another processor by a shared memory.	Processors need not communicate as they are controlled by the master processor.
Failure	If a processor fails, the computing capacity of the system reduces.	If a master processor fails, a slave is turned to the master processor to continue the execution. If a slave processor fails, its task is switched to other processors.

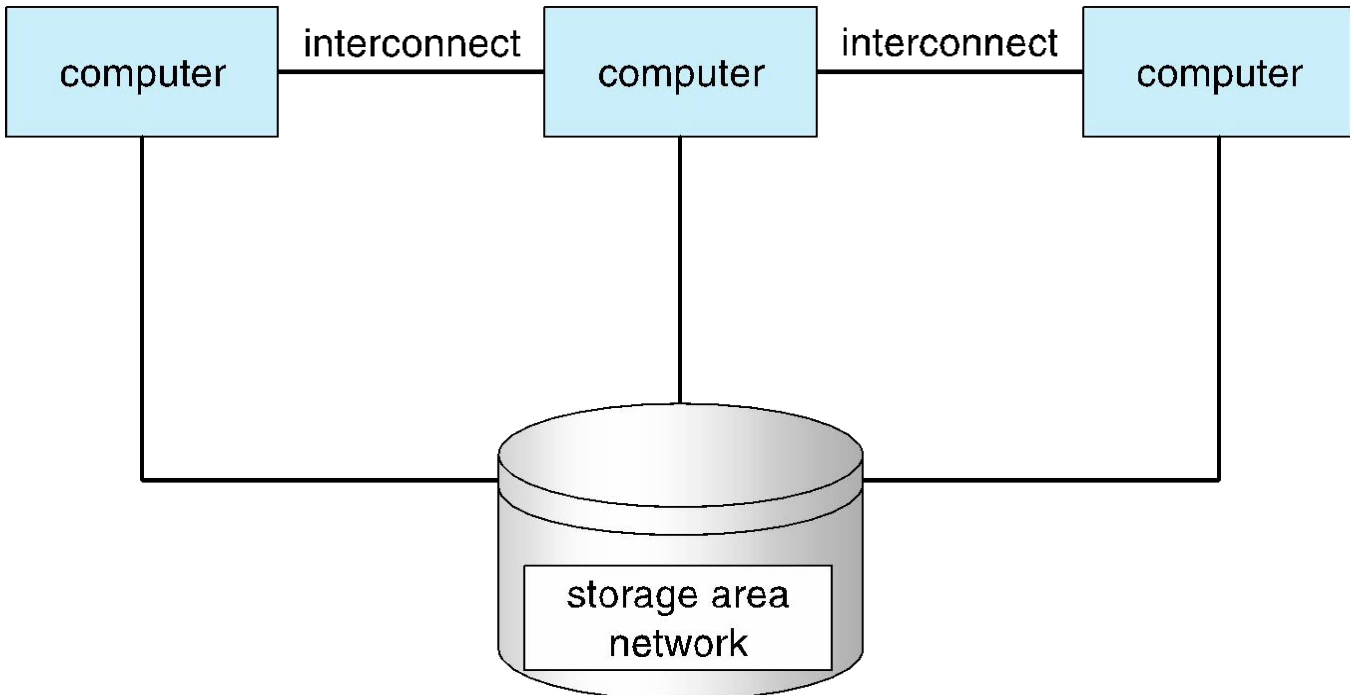
Symmetric Multiprocessing Architecture



Clustered Systems

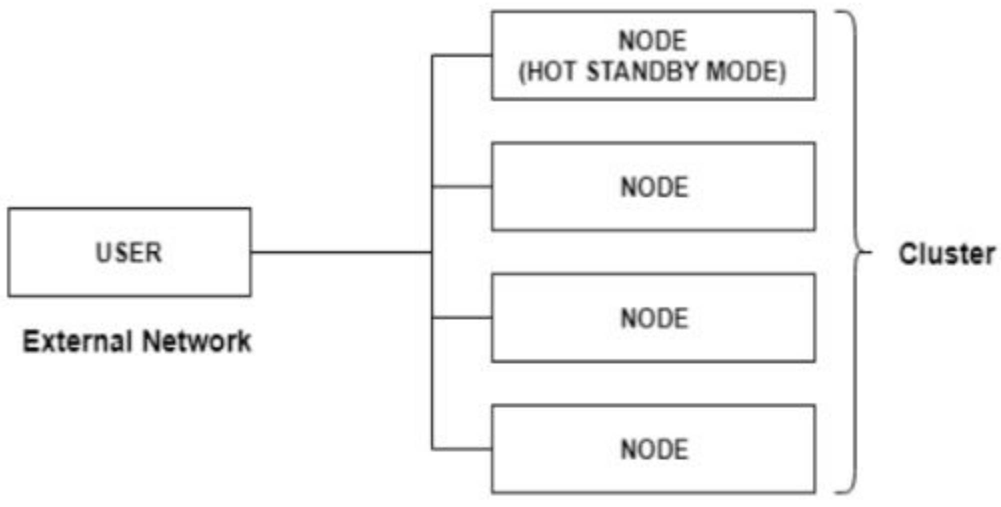
- Like multiprocessor systems, but multiple systems working together
  - Usually sharing storage via a **storage-area network (SAN)**
  - Some clusters are for **high-performance computing (HPC)**
- Applications must be written to use **parallelization**
- Clustered systems are similar to parallel systems as they both have multiple CPUs.
- However a major difference is that clustered systems are created by two or more individual computer systems merged together. Basically, they have independent computer systems with a common storage and the systems work together.
- Provides a **high-availability** service which survives failures
  - **Asymmetric clustering** has one machine in hot-standby mode
  - **Symmetric clustering** has multiple nodes running applications, monitoring each other

Clustered Systems



Asymmetric Clustering

- In this system, one of the nodes in the clustered system is in hot standby mode and all the others run the required applications.
- The hot standby mode is a failsafe in which a hot standby node is part of the system . The hot standby node continuously monitors the server and if it fails, the hot standby node takes its place.



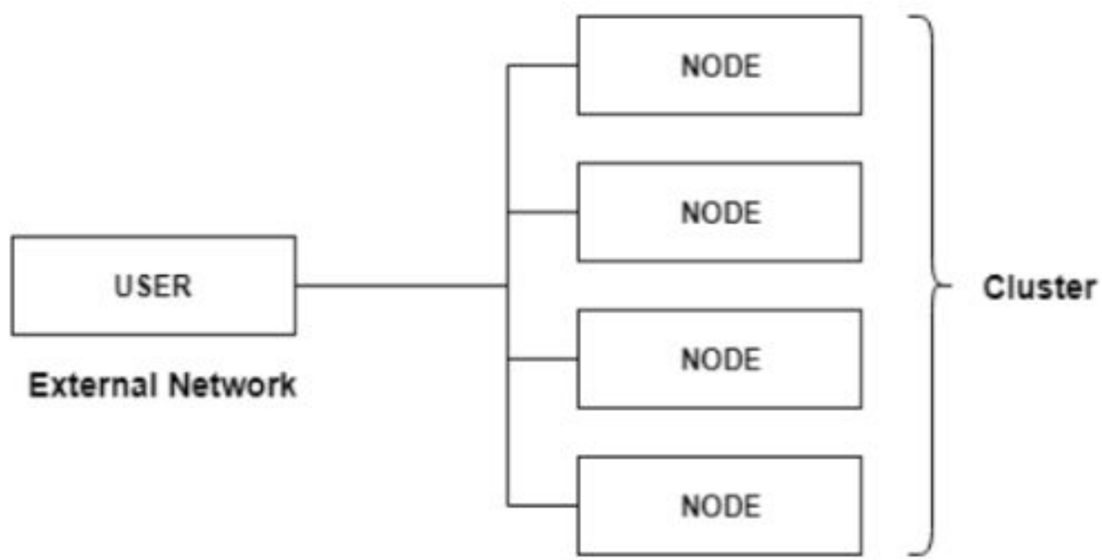
Asymmetric Clustering System

Asymmetric Clustering

- There is a master node in asymmetric clustering that directs all the slaves nodes to perform the tasks required. The requests are delegated by the master node.
- A distributed cache is used in asymmetric clustering to improve the performance of the system.
- Resources such as memory, peripheral devices etc. are divided between the nodes of the asymmetric clustering system at boot time.

Symmetric Clustering

In symmetric clustering system two or more nodes all run applications as well as monitor each other. This is more efficient than asymmetric system as it uses all the hardware and doesn't keep a node merely as a hot standby



Symmetric Clustering System

End of Lecture #02