# OPERATING SYSTEMS

**Course Teacher**

S. M. Sabit Bananee

Lecturer

Department of Computer Science & Engineering (CSE)

Islamic University of Technology (IUT).

# Types of Operating Systems

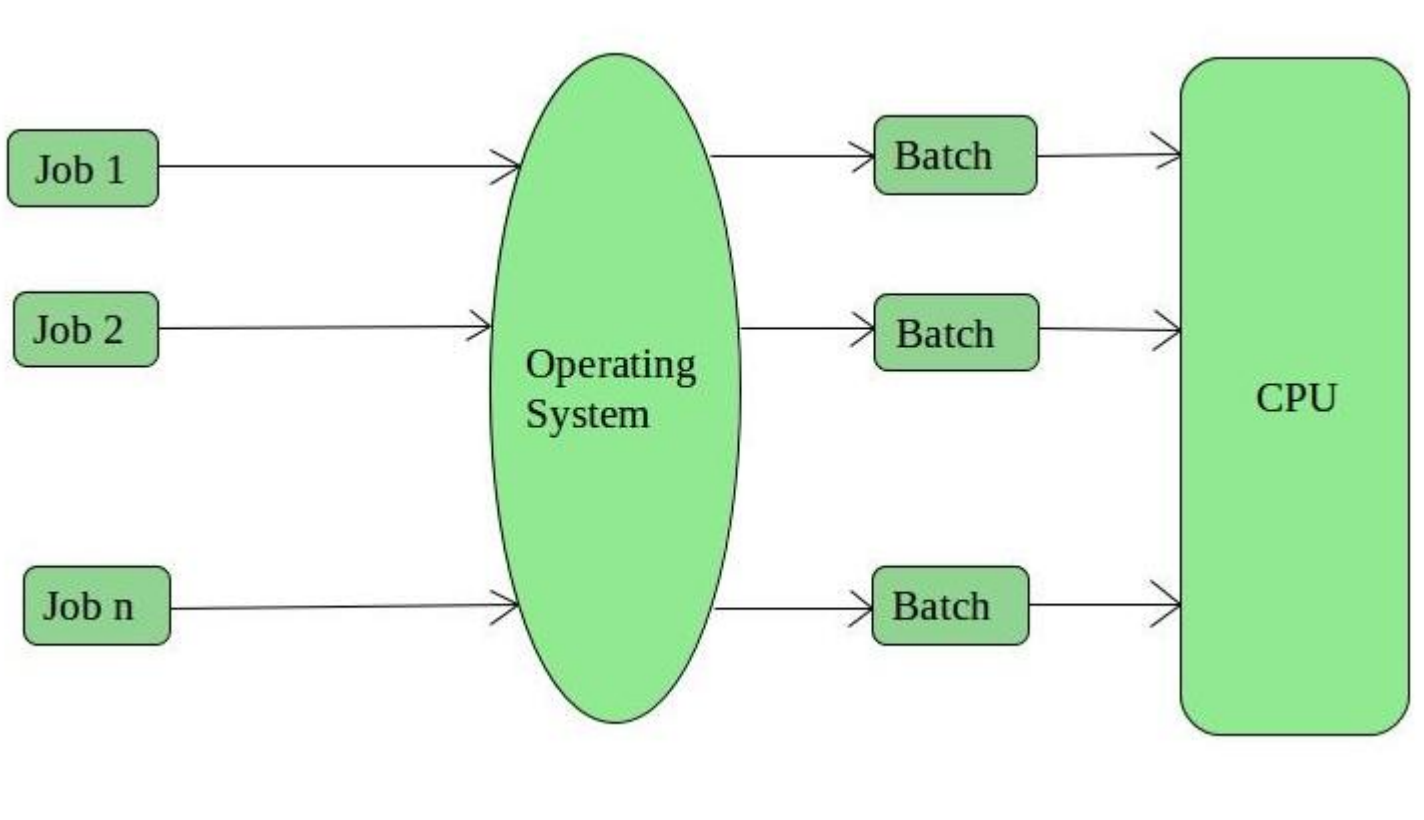
- Batch Operating System
- Time-sharing Operating System
- Distributed Operating System
- Network Operating System
- Real-time Operating System

# Batch Operating System

- This type of operating system does not interact with the computer directly. There is an operator which takes similar jobs having same requirement and group them into batches. It is the responsibility of operator to sort the jobs with similar needs.

- **Examples of Batch based Operating System:** Payroll System, Bank Statements etc.

# Batch Operating System



# Advantages of Batch Operating System

- It is very difficult to guess or know the time required by any job to complete. Processors of the batch systems know how long the job would be when it is in queue
- Multiple users can share the batch systems
- The idle time for batch system is very less
- It is easy to manage large work repeatedly in batch systems

# Disadvantages of Batch Operating System

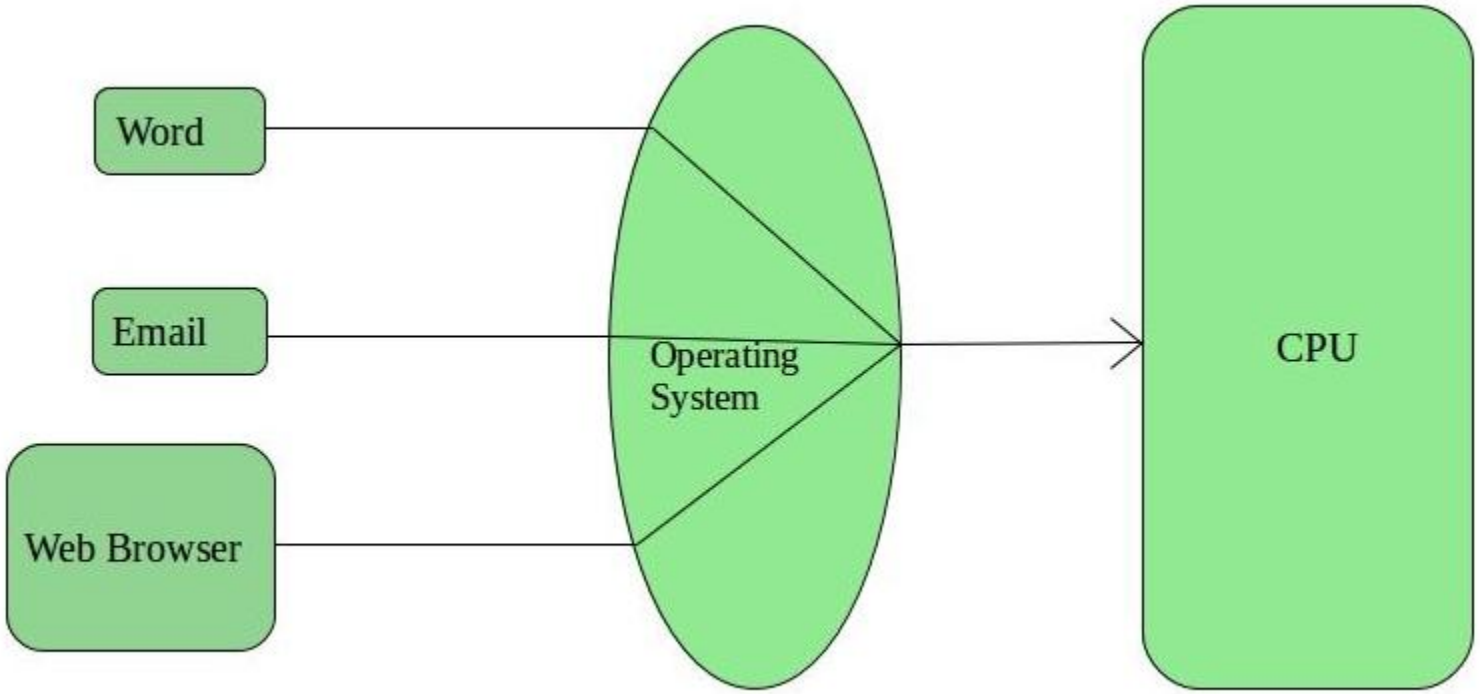
- Batch systems are hard to debug
- It is sometime costly
- The other jobs will have to wait for an unknown time if any job fails

# Time-sharing Operating System

- Each task is given some time to execute, so that all the tasks work smoothly. Each user gets time of CPU as they use single system. These systems are also known as Multitasking Systems

- **Examples of Time-Sharing OSs are:** Multics, Unix etc.

# Time-sharing Operating System



# Advantages of Time-sharing Operating System

- Each task gets an equal opportunity
- Less chances of duplication of software
- CPU idle time can be reduced

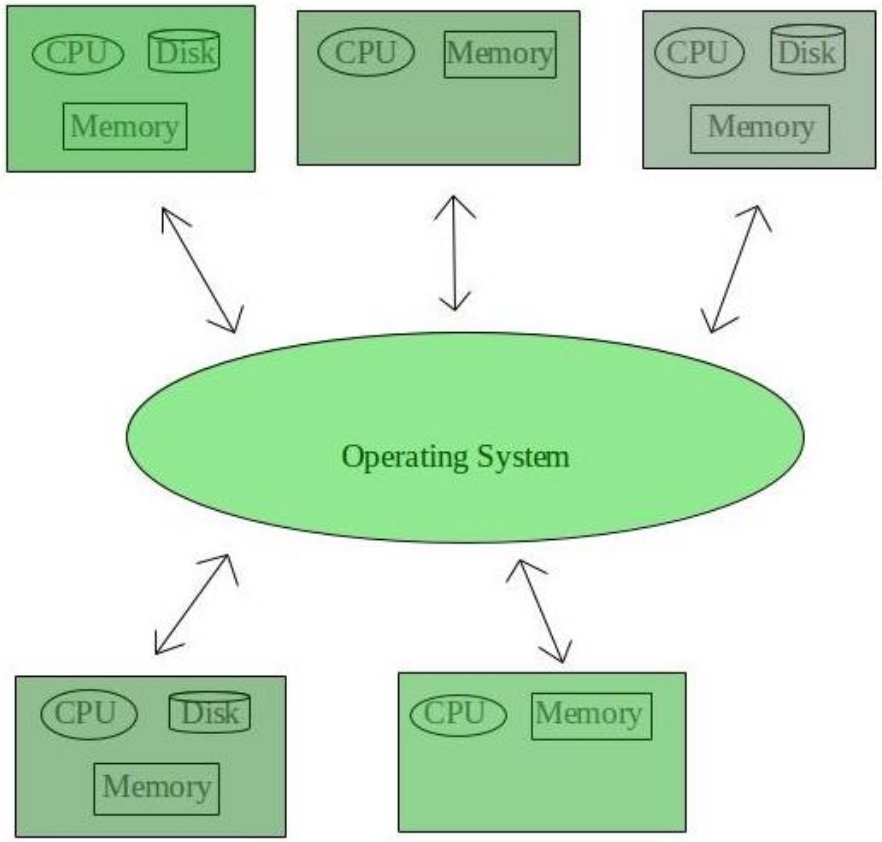
# Disadvantages of Time-sharing Operating System

- Reliability problem
- One must have to take care of security and integrity of user programs and data
- Data communication problem

# Distributed Operating System

- Various autonomous interconnected computers communicate each other using a shared communication network. Independent systems possess their own memory unit and CPU. These are referred as **loosely coupled systems** or distributed systems.
- The major benefit of working with these types of operating system is that it is always possible that one user can access the files or software which are not actually present on his system but on some other system connected within this network.
- **Examples of Distributed Operating System are-** LOCUS etc.

# Distributed Operating System



# Advantages of Distributed Operating System

- Failure of one will not affect the other network communication, as all systems are independent from each other
- Electronic mail increases the data exchange speed
- Since resources are being shared, computation is highly fast and durable
- Load on host computer reduces
- These systems are easily scalable as many systems can be easily added to the network
- Delay in data processing reduces

# Disadvantages of Distributed Operating System

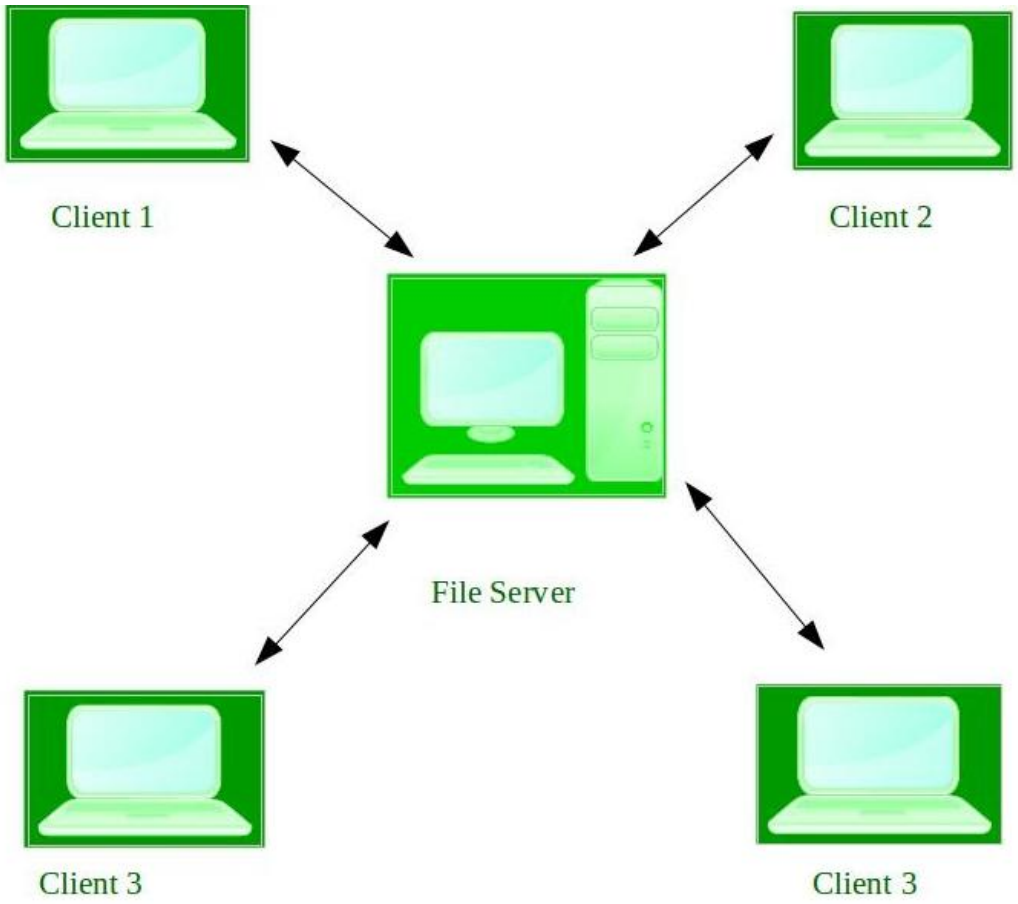
- Failure of the main network will stop the entire communication
- To establish distributed systems the language which are used are not well defined yet
- These types of systems are not readily available as they are very expensive. Not only that the underlying software is highly complex and not understood well yet

# Network Operating System

- These systems run on a server and provide the capability to manage data, users, groups, security, applications, and other networking functions. These type of operating systems allow shared access of files, printers, security, applications, and other networking functions over a small private network.

- **Examples of Network Operating System are:** Microsoft Windows Server 2003, Microsoft Windows Server 2008, UNIX, Linux, Mac OS X, Novell NetWare, and BSD etc.

# Network Operating System



# Advantages of Network Operating System

- Highly stable centralized servers
- Security concerns are handled through servers
- New technologies and hardware up-gradation are easily integrated to the system
- Server access are possible remotely from different locations and types of systems

# Disadvantages of Network Operating System

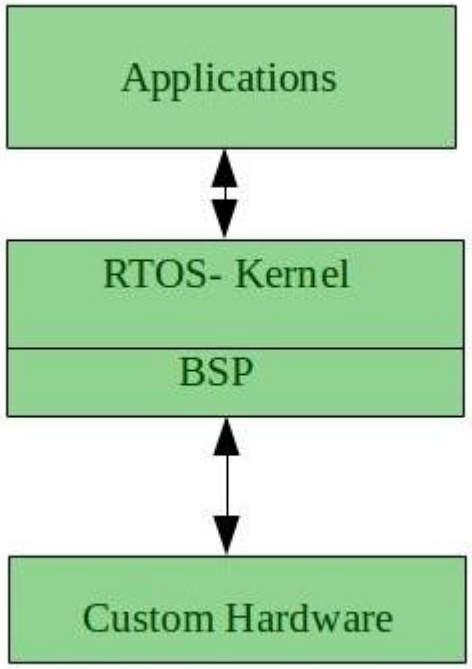
- Servers are costly
- User has to depend on central location for most operations
- Maintenance and updates are required regularly

# Real-time Operating System (RTOS)

- These types of OSs serves the real-time systems. The time interval required to process and respond to inputs is very small. This time interval is called **response time**.

- **Examples of Real-Time Operating Systems are:** Scientific experiments, medical imaging systems, industrial control systems, weapon systems, robots, air traffic control systems, etc.

# Real-time Operating System (RTOS)



# Advantages of RTOS

- **Maximum Consumption:** Maximum utilization of devices and system,thus more output from all the resources
- **Task Shifting:** Time assigned for shifting tasks in these systems are very less. For example in older systems it takes about 10 micro seconds in shifting one task to another and in latest systems it takes 3 micro seconds.
- **Focus on Application:** Focus on running applications and less importance to applications which are in queue.
- **Real time operating system in embedded system:** Since size of programs are small, RTOS can also be used in embedded systems like in transport and others.
- **Error Free:** These types of systems are error free.
- **Memory Allocation:** Memory allocation is best managed in these type of systems.

# Disadvantages of RTOS

- **Limited Tasks:** Very few tasks run at the same time and their concentration is very less on few applications to avoid errors.
- **Use heavy system resources:** Sometimes the system resources are not so good and they are expensive as well.
- **Complex Algorithms:** The algorithms are very complex and difficult for the designer to write on.
- **Device driver and interrupt signals:** It needs specific device drivers and interrupt signals to response earliest to interrupts.
- **Thread Priority:** It is not good to set thread priority as these systems are very less prone to switching tasks

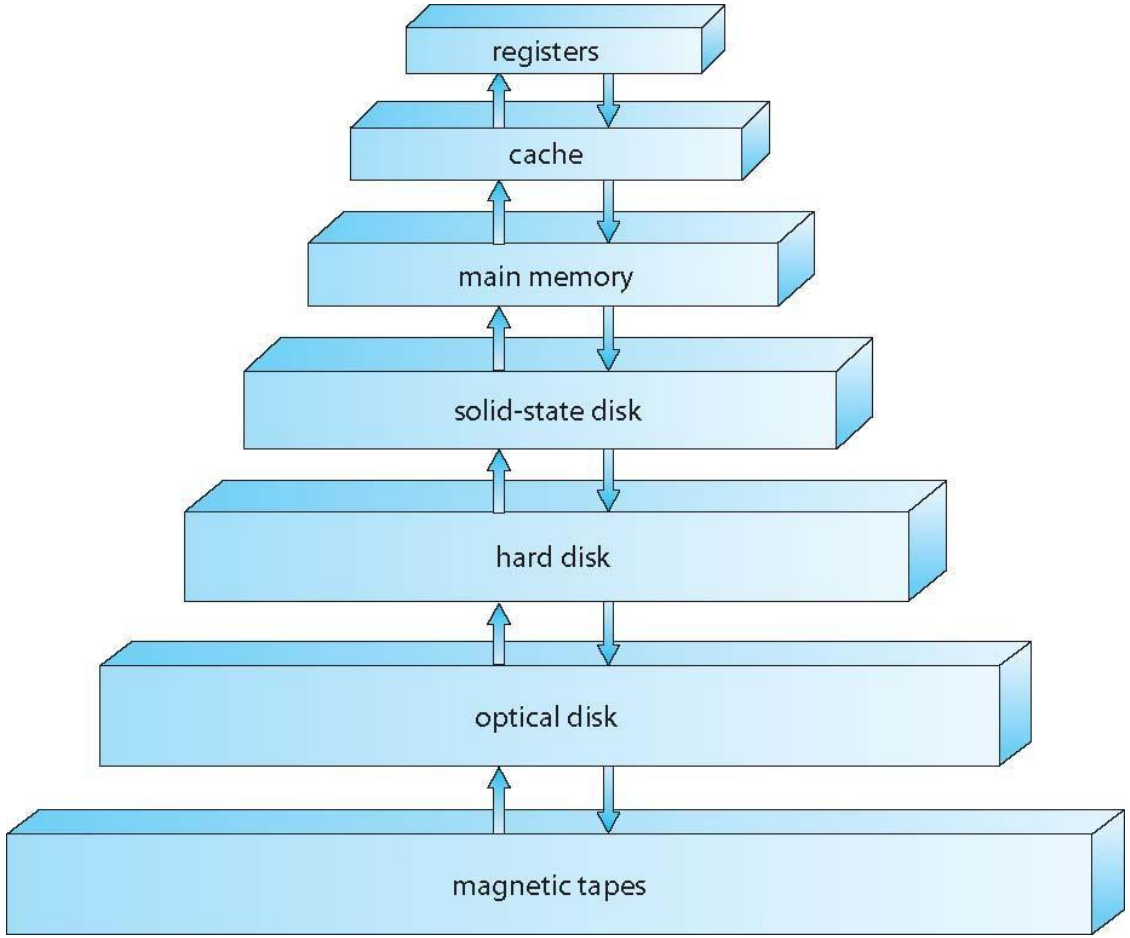
# Storage Structure

- 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

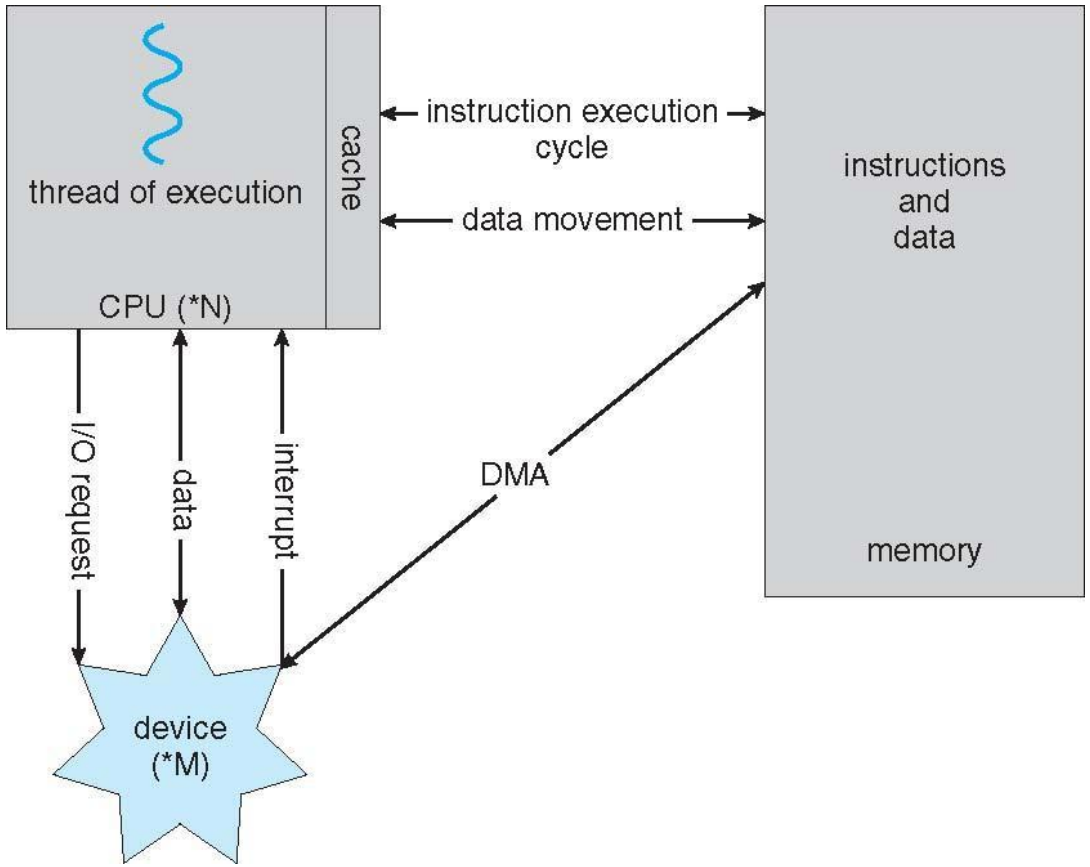
# Storage-Device Hierarchy



# Direct Memory Access Structure

- Used for high-speed I/O devices able to transmit information at close to memory speeds
- Device controller transfers blocks of data from buffer storage directly to main memory without CPU intervention
- Only one interrupt is generated per block, rather than the one interrupt per byte

# How a Modern Computer Works

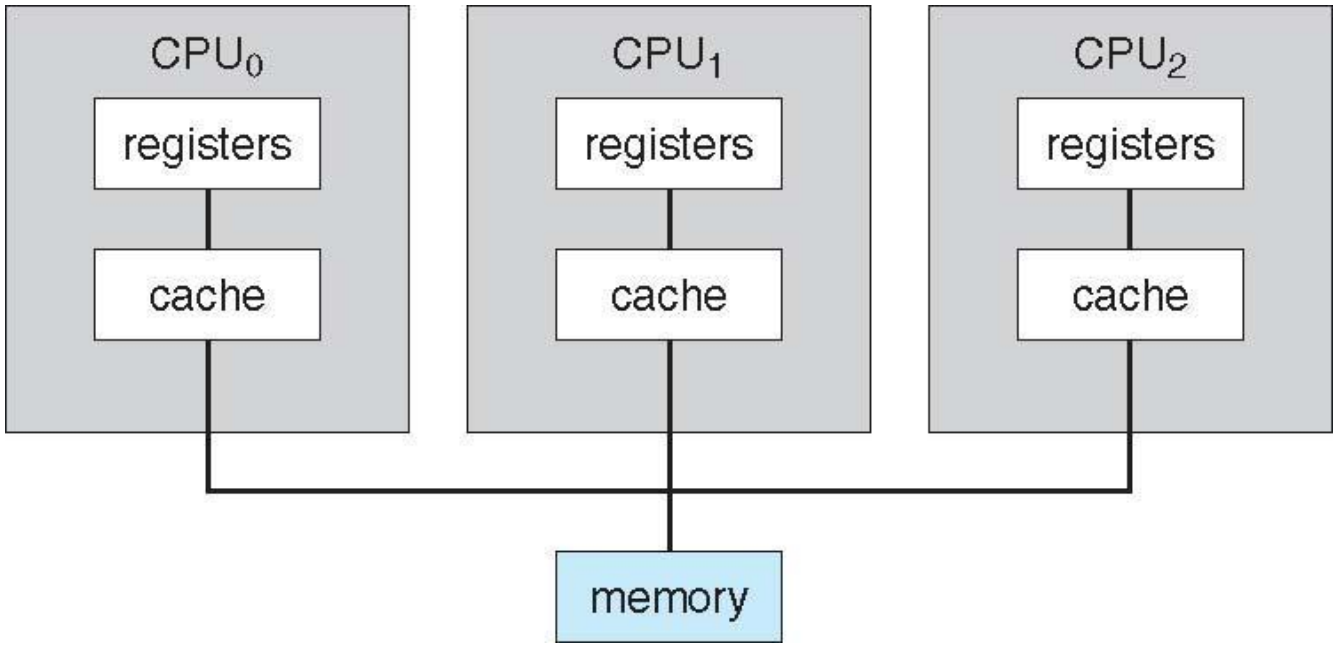



*A von Neumann architecture*

# Computer-System Architecture

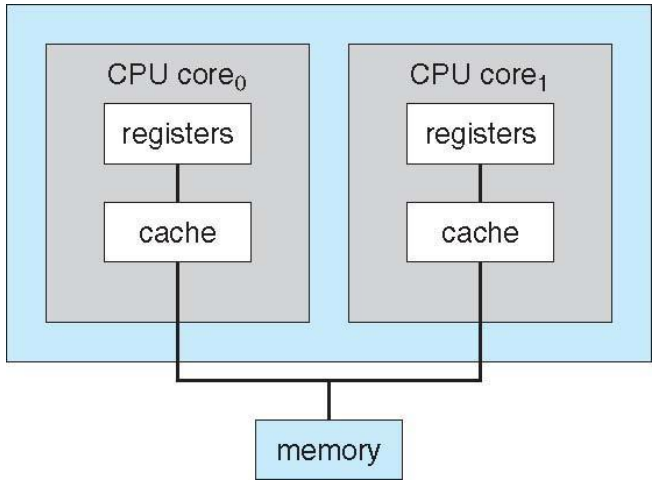
- Most systems use a single general-purpose processor
  - Most systems have special-purpose processors as well
- **Multiprocessors** systems growing in use and importance
  - Also known as **parallel systems**, **tightly-coupled systems**
  - Advantages include:
    1. **Increased throughput**
    2. **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

# Symmetric Multiprocessing Architecture

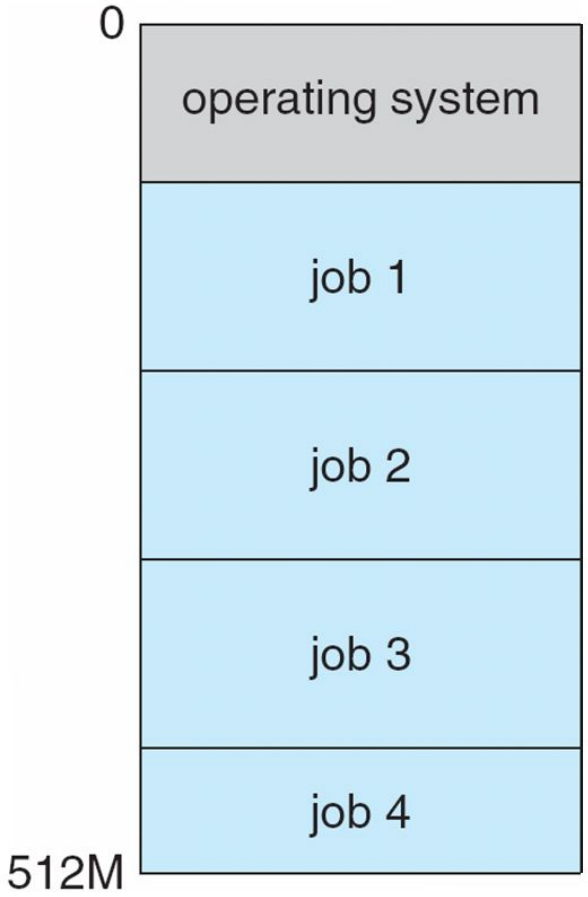


# A Dual-Core Design

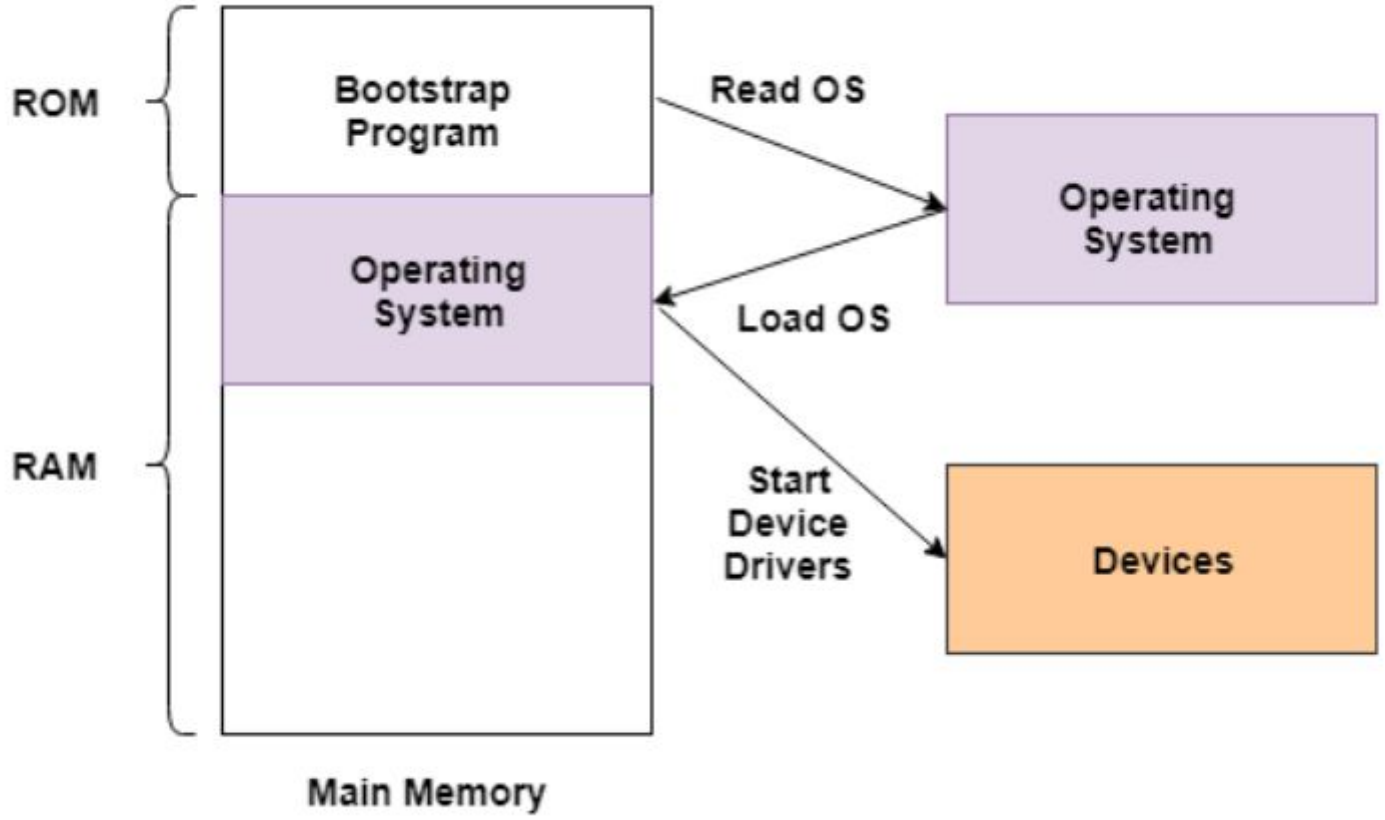
- Multi-chip and **multicore**
- Systems containing all  chips
  - Chassis containing multiple separate systems



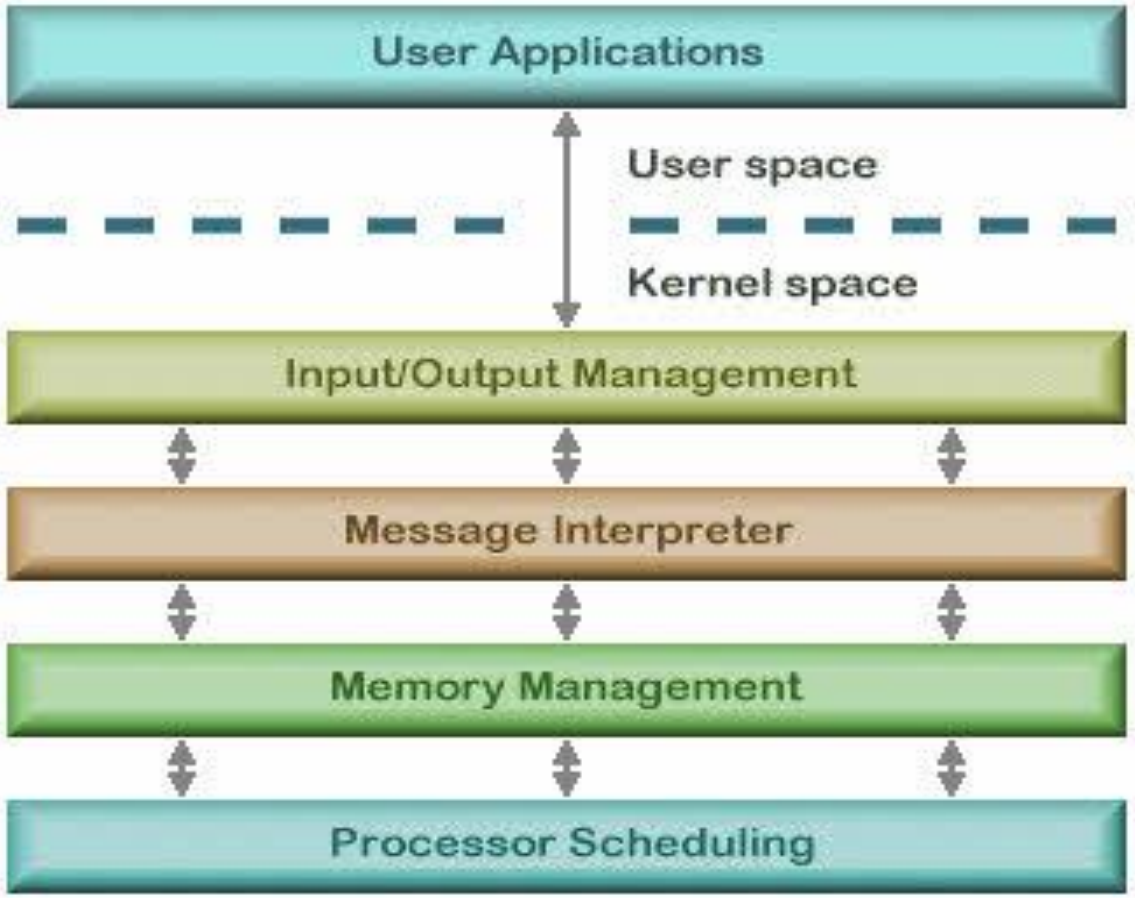
# Memory Layout for Multiprogrammed System



# Bootstrap Program



# OS Architecture



# Thank you