Operating Systems

1. Introduction to Operating Systems

Gaurav Somani
gaurav@lnmiit.ac.in

Structure

- OS Concepts

  Process, File, I/O and Memory Management

- OS Case Studies

  Windows, UNIX, UNIX derivatives and some handheld and Embedded Operating Systems.

- Practical Implementation

  Shell, Processes, Threads, Virtual Memory, File Systems and other experiments (Linux Platform)

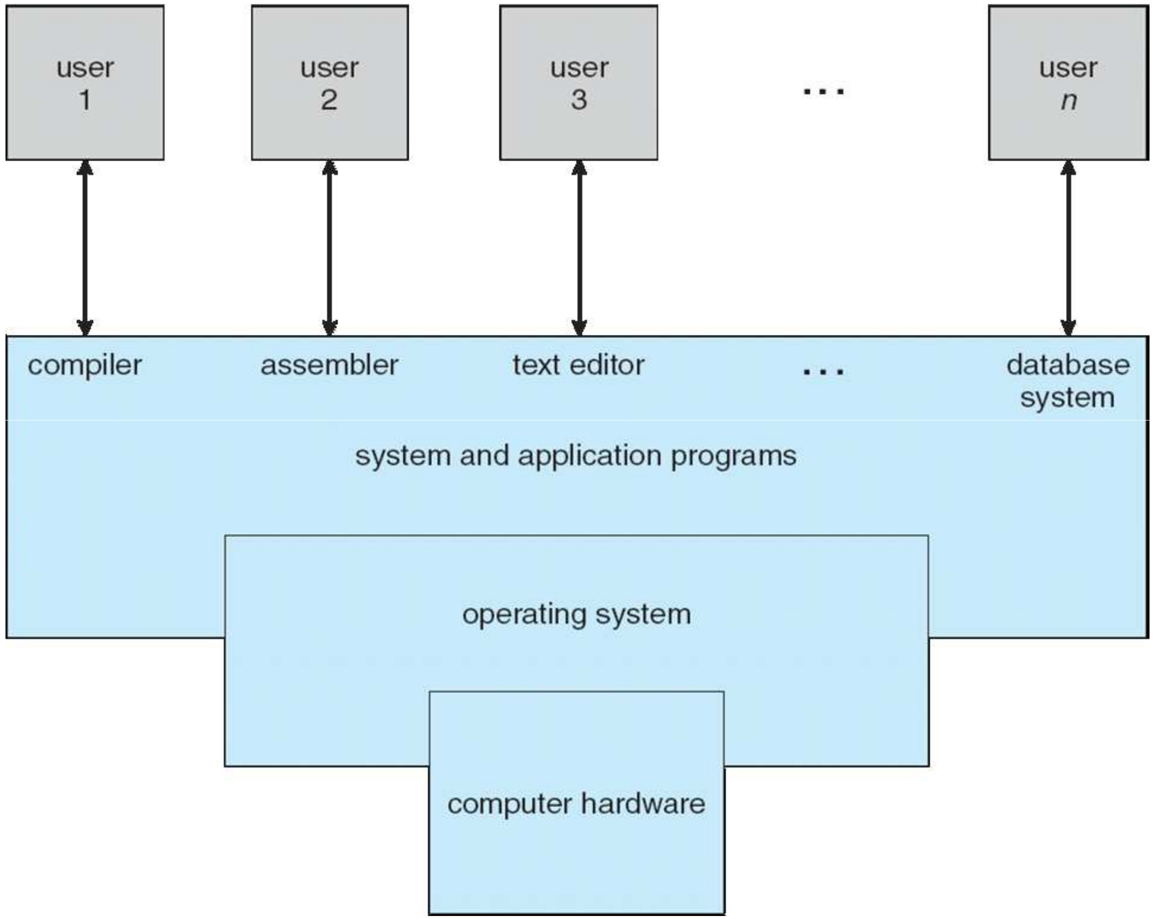
Outline

- What Operating Systems Do
- Computer-System Organization
- Computer-System Architecture
- Operating-System Structure
- Operating-System Operations
- Process Management
- Memory Management
- Storage Management
- Protection and Security
- Distributed Systems
- Special-Purpose Systems
- Computing Environments
- Open-Source Operating Systems

Computer System Structure

- Computer system can be divided into four components
  - Hardware - provides basic computing resources
    - CPU, memory and I/O devices
  - Operating system
    - Controls and coordinates use of hardware among various applications and users
  - Application/System 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 and video games
  - Users
    - People, machines and other computers

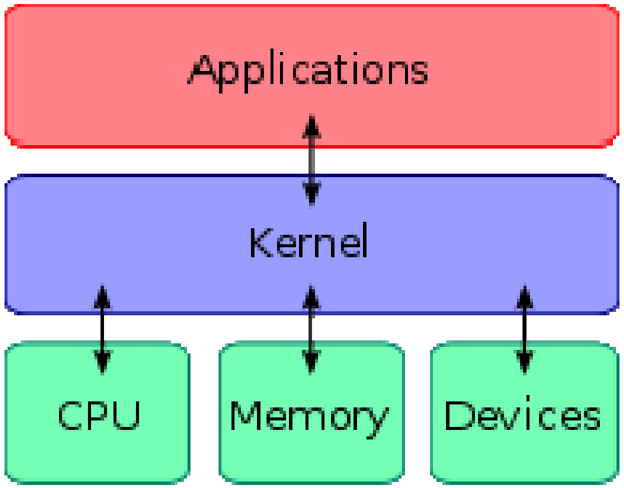
Computer System Structure



Definition of an OS

- No single Definition
- An Interface between computer hardware and user
- It is an extended machine
  - Hides the messy details which must be performed.
  - Presents user with a *virtual machine*, easier to use.
- It is a resource manager
  - Each program gets time with the resource.
  - Each program gets space on the resource.

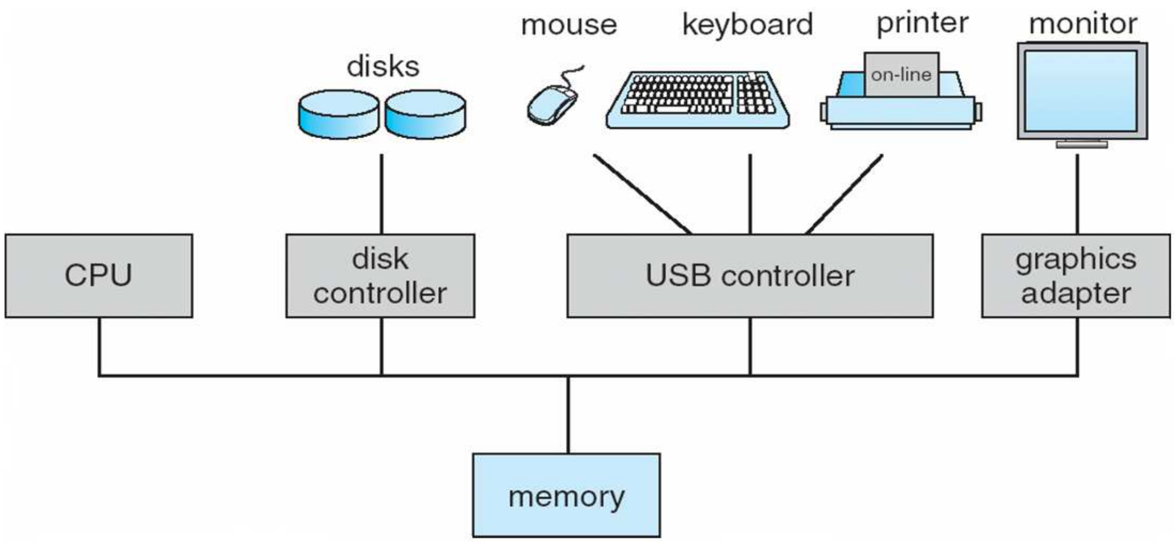
Kernel



“Kernel is traditionally used to denote the part of the operating system that is mandatory and common to all other software.” - Jochen Liedtke (Developer of L4 Microkernel)

Computer System Organization

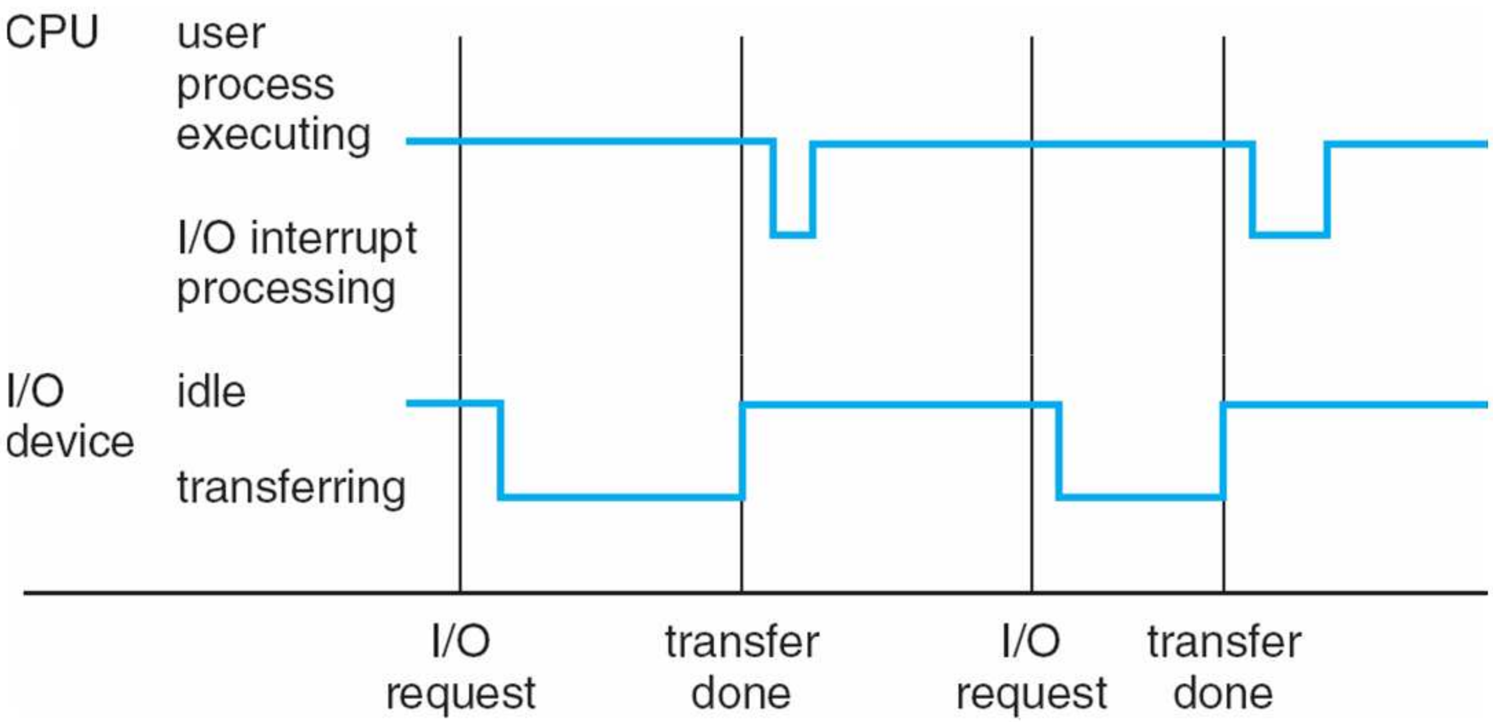
- Computer-system operation
  - One or more CPUs, device controllers connect through common bus providing access to shared memory
  - Concurrent execution of CPUs and devices competing for memory cycles



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
- I/O is from the device to local buffer of controller
- Device controller informs CPU that it has finished its operation by causing an *interrupt*

Interrupt Timeline



Common Functions of Interrupts

- Interrupt transfers control to the interrupt service routine generally, through the **interrupt vector**, which contains the addresses of all the service routines
- Interrupt architecture must save the address of the interrupted instruction
- Incoming interrupts are *disabled* while another interrupt is being processed to prevent a *lost interrupt*
- A *trap* is a software-generated interrupt caused either by an error or a user request
- An operating system is **interrupt driven**

Interrupt Handling

- The operating system preserves the state of the CPU by storing registers and the program counter
- Determines which type of interrupt has occurred:
  - **polling**
  - **vectored** interrupt system
- Separate segments of code determine what action should be taken for each type of interrupt

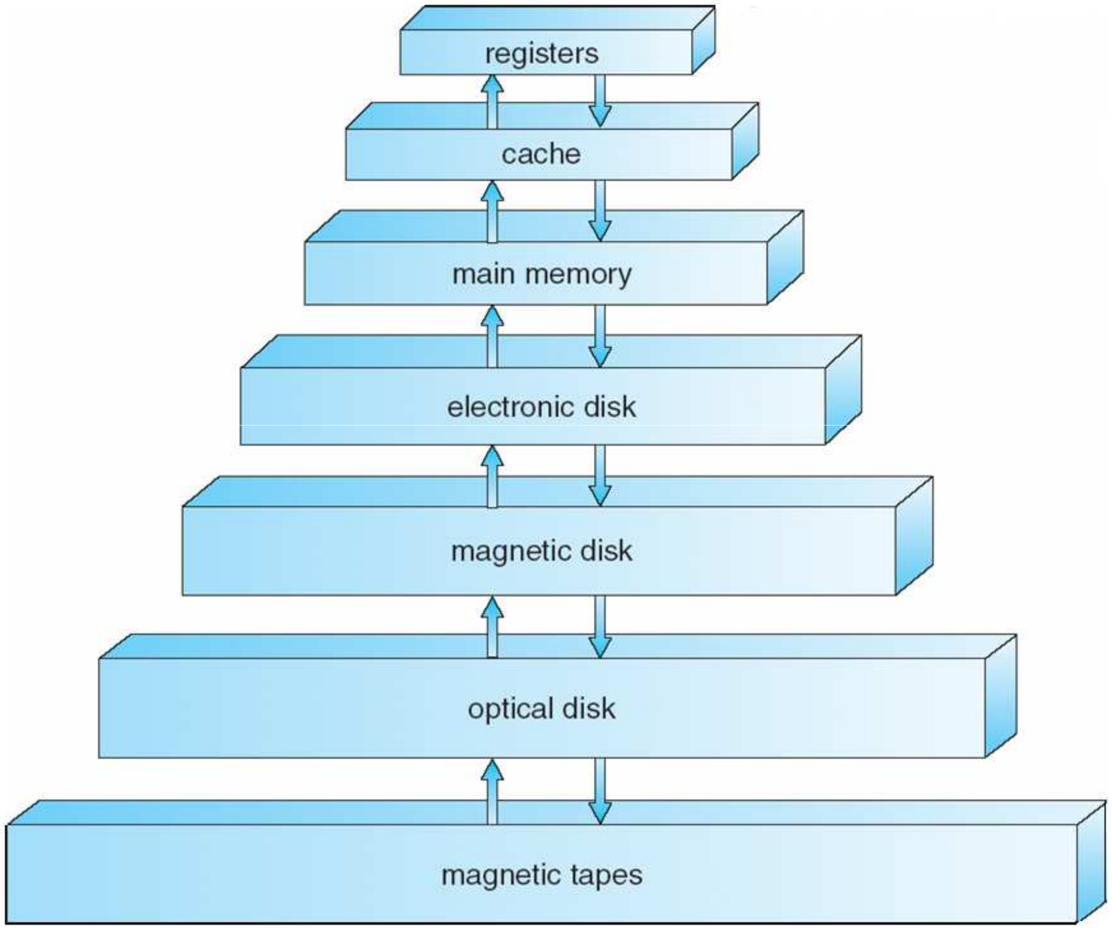
Storage Structure

- 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

Storage Hierarchy

- Storage systems organized in hierarchy
  - Speed
  - Cost
  - Volatility
- **Caching** – copying information into faster storage system; main memory can be viewed as a last *cache* for secondary storage

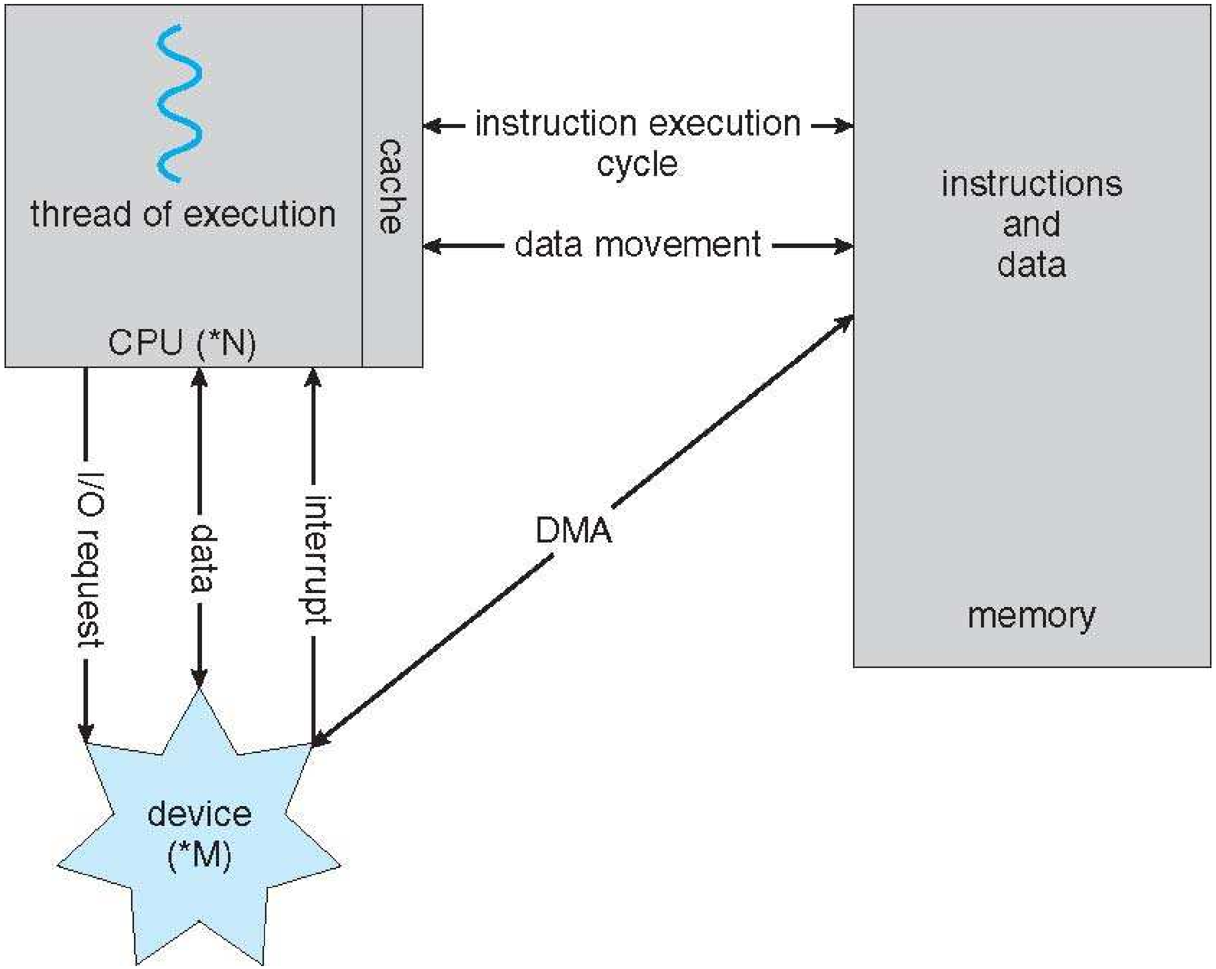
Storage-Device Hierarchy



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

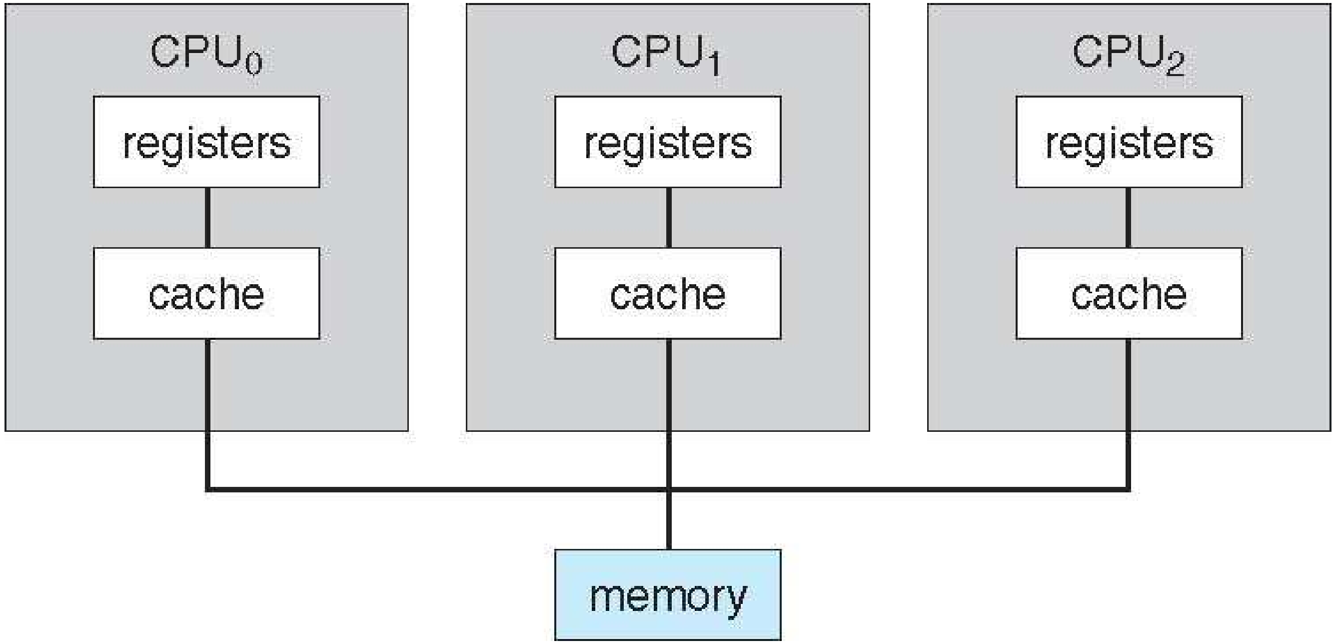
How does a Modern Computer Work?



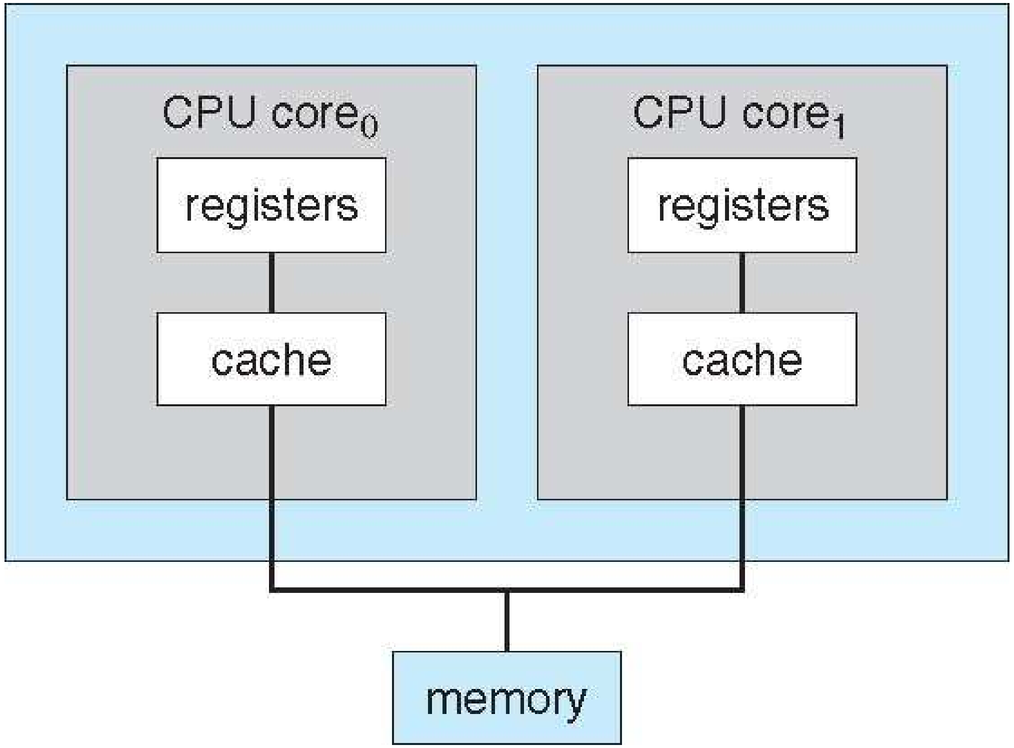
Computer-System Architecture

- Most systems use a single general-purpose processor (PDAs through mainframes)
  - 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
    2. Symmetric Multiprocessing

Symmetric Multiprocessing Architecture



A Dual-Core Design



Clustered Systems

- Like multiprocessor systems, but multiple systems working together
  - Usually sharing storage via a storage-area network (SAN)
  - 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
  - Some clusters are for high-performance computing (HPC)
    - Applications must be written to use parallelization

Modern Operating Systems

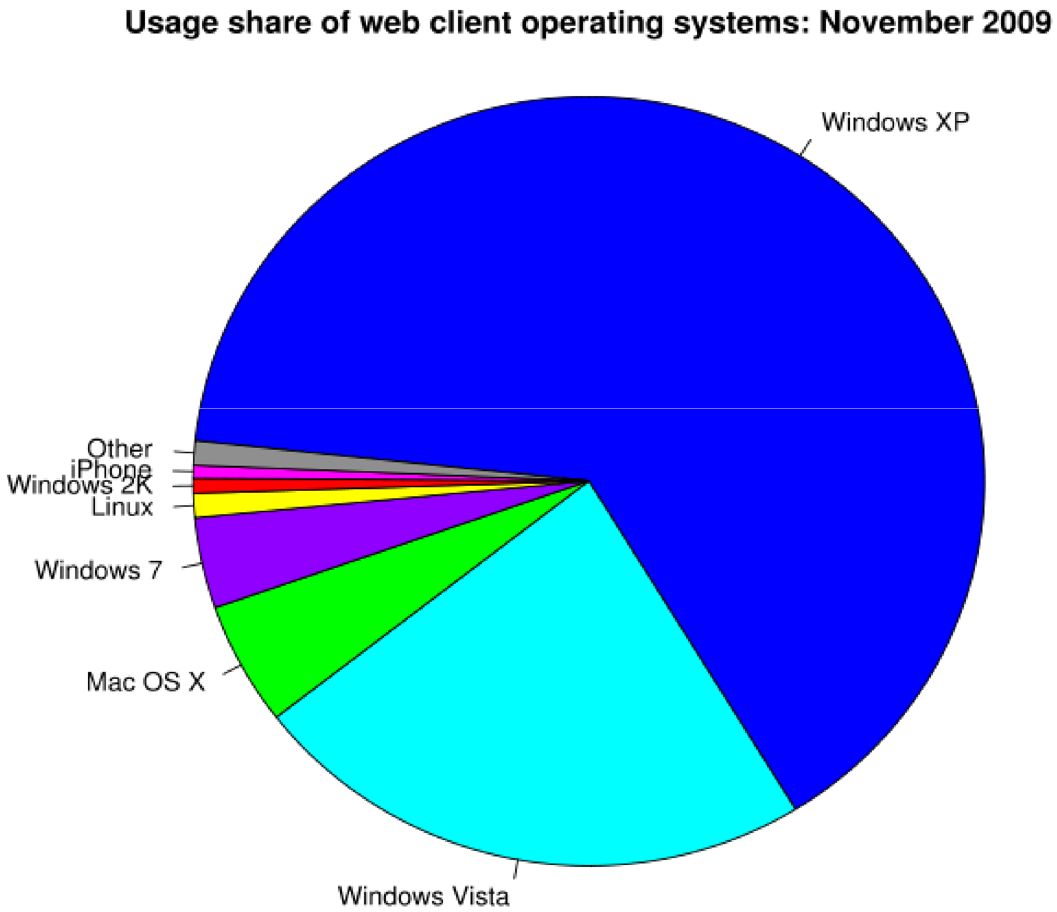
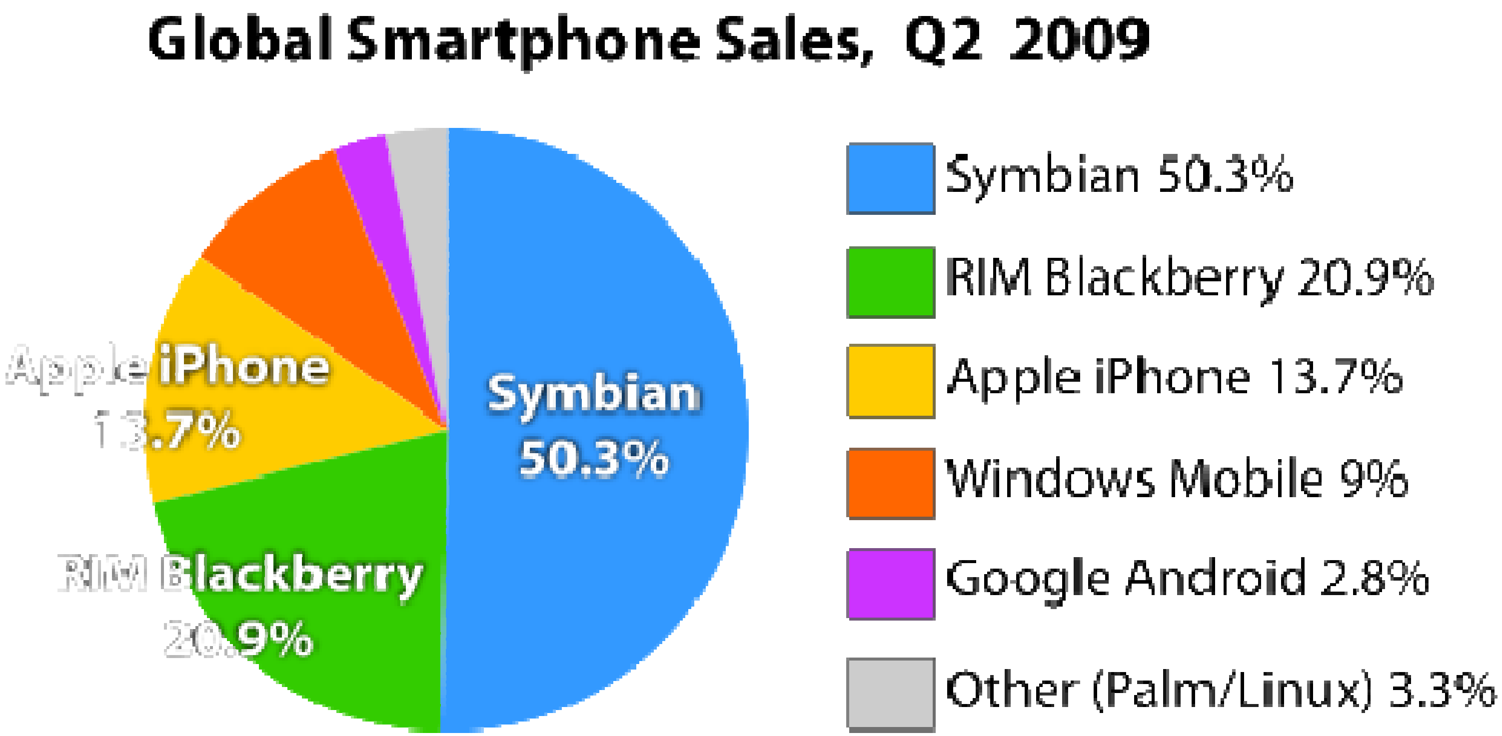


Image Source - Canalys: iPhone outsold all Windows Mobile phones in Q2 2009

Modern Operating Systems



[Image Source - Canalys: iPhone outsold all Windows Mobile phones in Q2 2009]()

Supercomputers

OS	Count	Share	Rmax (GF)	Rpeak (GF)	Processor
Linux	426	85.20%	4897046	7956758	970790
Windows	6	1.20%	47495	86797	12112
Unix	30	6.00%	408378	519178	73532
BSD	2	0.40%	44783	50176	5696
Mixed	34	6.80%	1540037	1900361	580693
MacOS	2	0.40%	28430	44816	5272
Totals	500	100%	6966169	10558086	1648095

http://www.top500.org/stats/list/30/osfam Nov 2007

Rmax – The highest score measured using the LINPACK benchmark suite. This is the number which is used to rank the computers. Measured in Giga-Flops.
Rpeak – This is the theoretical peak performance of the system. Measured in Gflops.

Modern Operating Systems

See some web operating systems

1. Glide OS – x
2. EyeOS - eyeos.org
3. Ghost Operating System

http://g.ho.st/vc.html?ghostId=guest

4. gOS -http://www.thinkgos.com/gos/download.html

Thanks