Operating Systems

Martin Hynes

Room 200

Martin.hynes@gmit.ie

Ext 2215

Course Assessment Breakdown

- 60% - End of Semester Final Examination
- 20% - In Class Assessment
- 20% - Project Based Assignment

Lecture & Laboratory Times

- Lectures
  - Wednesday 11am – 12noon Room 995
  - Thursday 5pm – 6pm Room 995
- Laboratories Sessions
  - Group B – Monday 1pm – 2pm CR1 & Tuesday 11am – 1pm CR1
  - Group A – Tuesday 3pm – 4pm CR7 & Wednesday 4pm – 6pm CR2
  - Group C – Tuesday 4pm – 6pm CR5 & Thursday 1pm– 2pm CR2

Course contents (Subject to Change)

- Process Management
- Thread programming
- Process Communication & Synchronization
- Memory Management
- I/O & File Management
- Virtualization

Course goals

- Understand how an operating system works and its architecture.
- Learn how OS concepts are implemented in a real operating system.
- Introduction to Thread Programming.
- Understand how Operating System virtualization works.

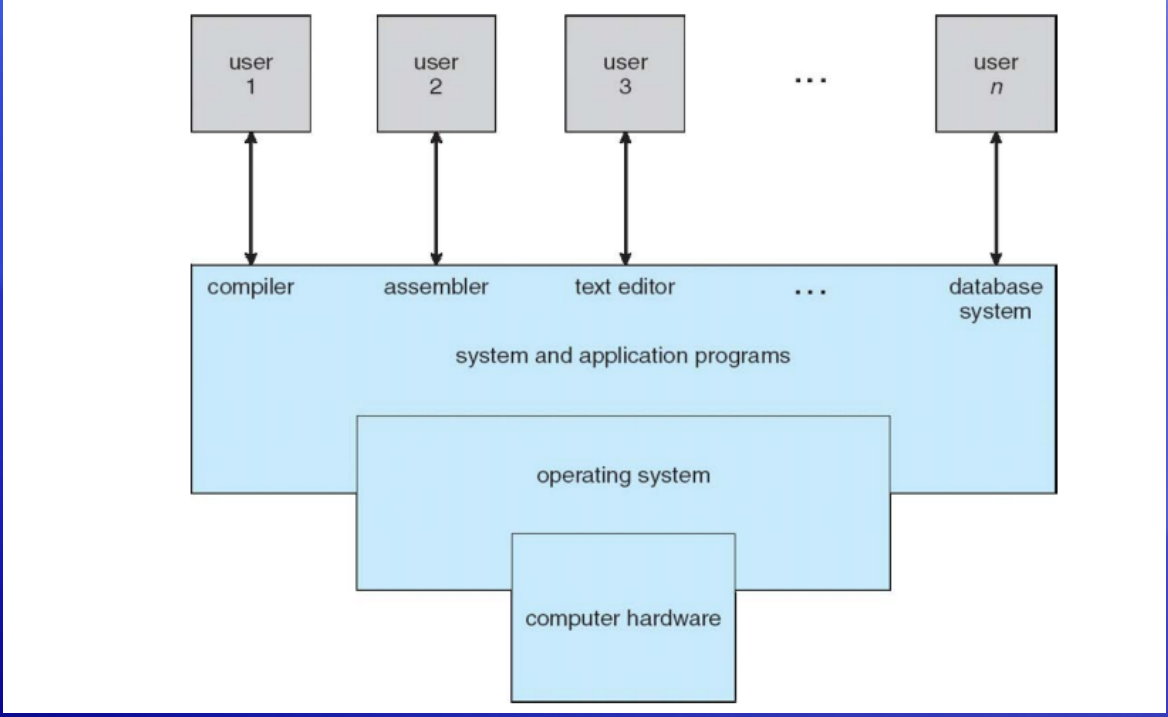
Text Books

- A. McIver-McHoes & I Flynn, Understanding Operating Systems.
- A. Silberschatz, P Galvin & Greg Gagne, Applied Operating Systems.
- G. Nutt, Operating Systems A Modern Perspective.
- Deitel, Deitel & Choffnes, Operating Systems

What is an Operating System

- A program that acts as a bridge between a user of a computer and the computer hardware.
- Operating Systems Goals
  - Execute user programs and make solving user problems easier.
  - Make the computer system convenient to use.
  - Use the computer hardware as efficiently as possible.

What is an Operating System



Example Operating System Functions

Action	Operating System Functions
Login	Process Creating, Authentication, Resource Allocation
Creating a file	Application Launching, File Management
Running a program (e.g. Microsoft Word)	Scheduling, resource (e.g. Memory, disk) allocation

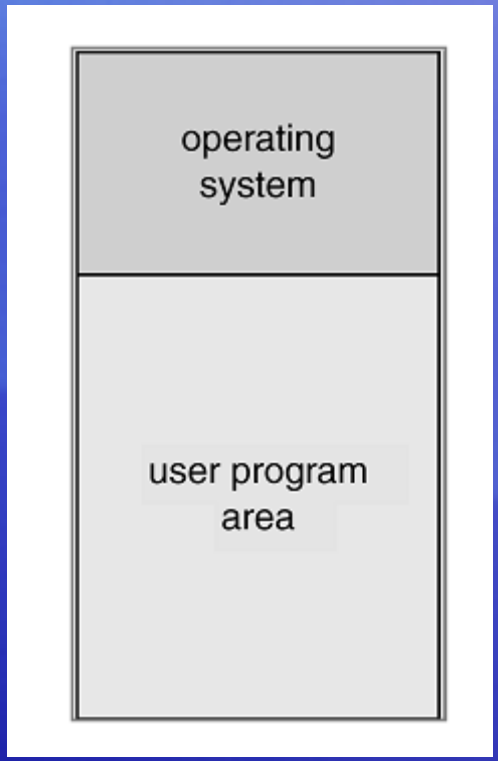
Operating System Definitions

- Resource allocator – manages and allocates resources
- Control program – controls the execution of user programs and operations of I/O devices
- Kernel – the one program running at all times (all else being application programs)

Mainframe Systems

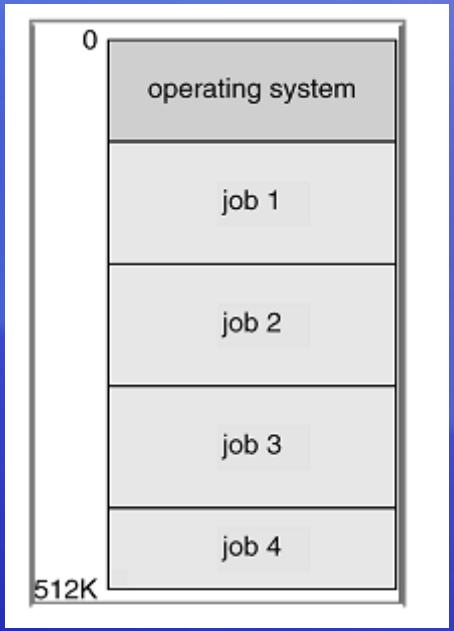
- Reduce setup time by batching similar jobs
- Automatic job sequencing – automatically transfers control from one job to another. First rudimentary operating system
- Resident monitor
  - initial control in monitor
  - control transfers to job
  - when job completes control transfers back to monitor

Memory Layout for a Simple Batch System

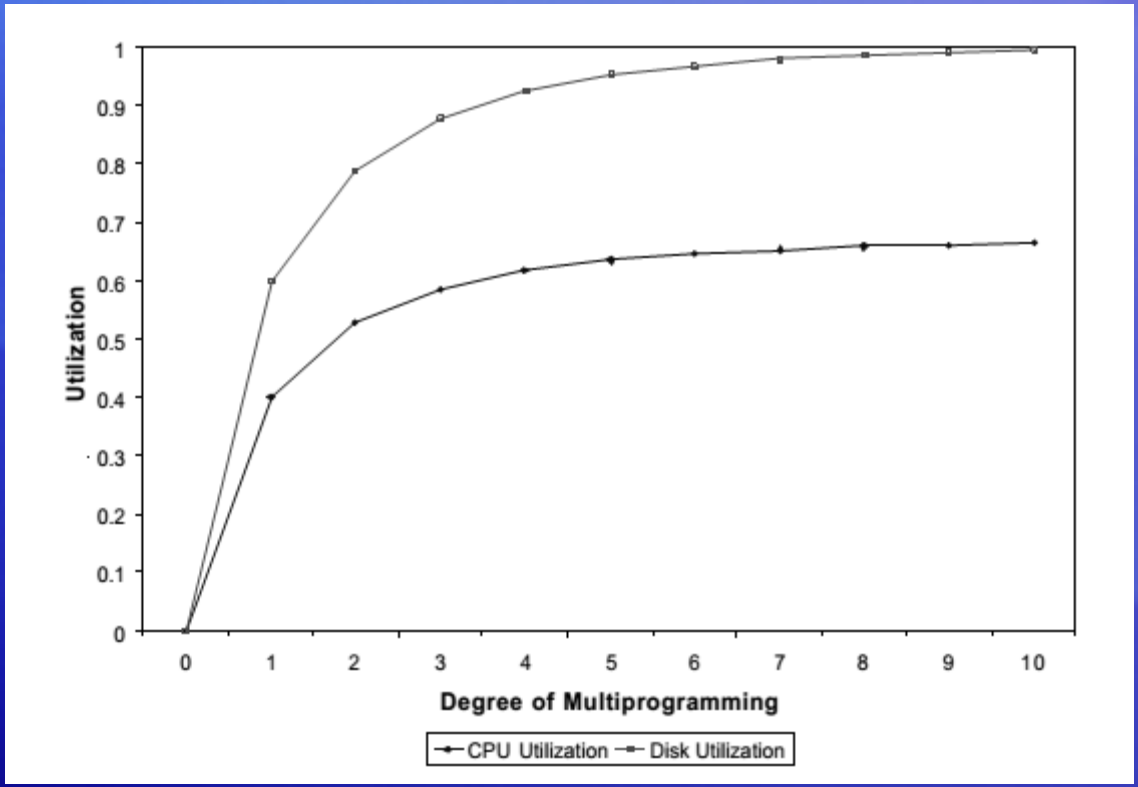


Multiprogrammed Batch System

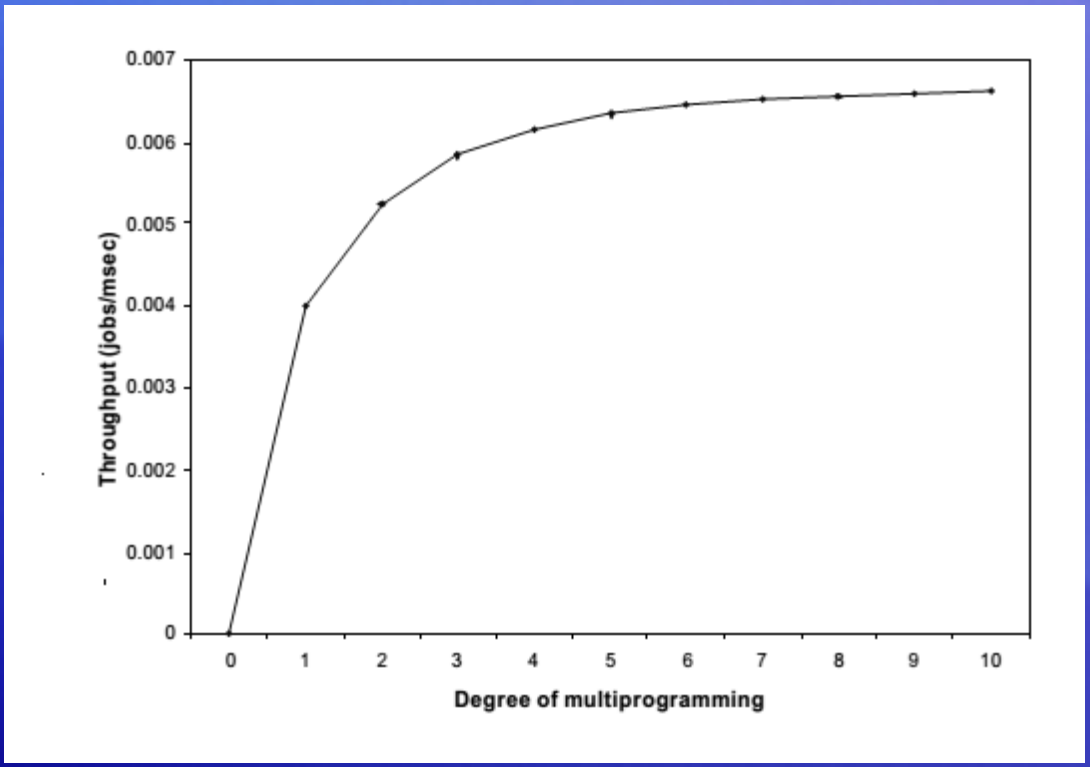
- Several jobs are kept in main memory at the same time, and the CPU is multiplexed between them.



Utilization in Multiprogamming Systems



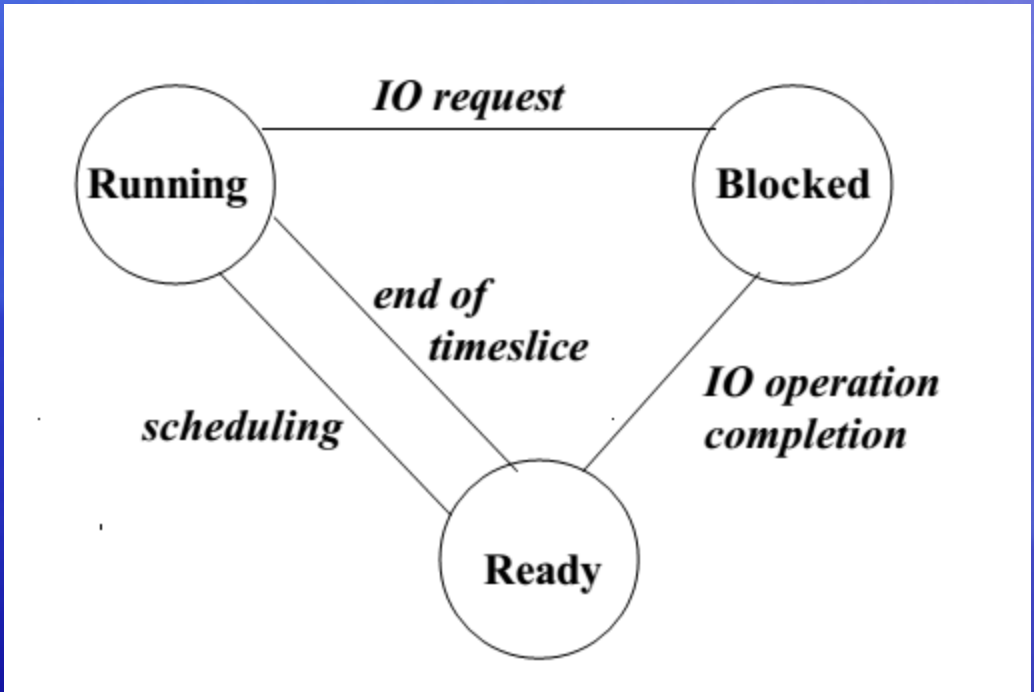
Throughput in Multiprogamming Systems



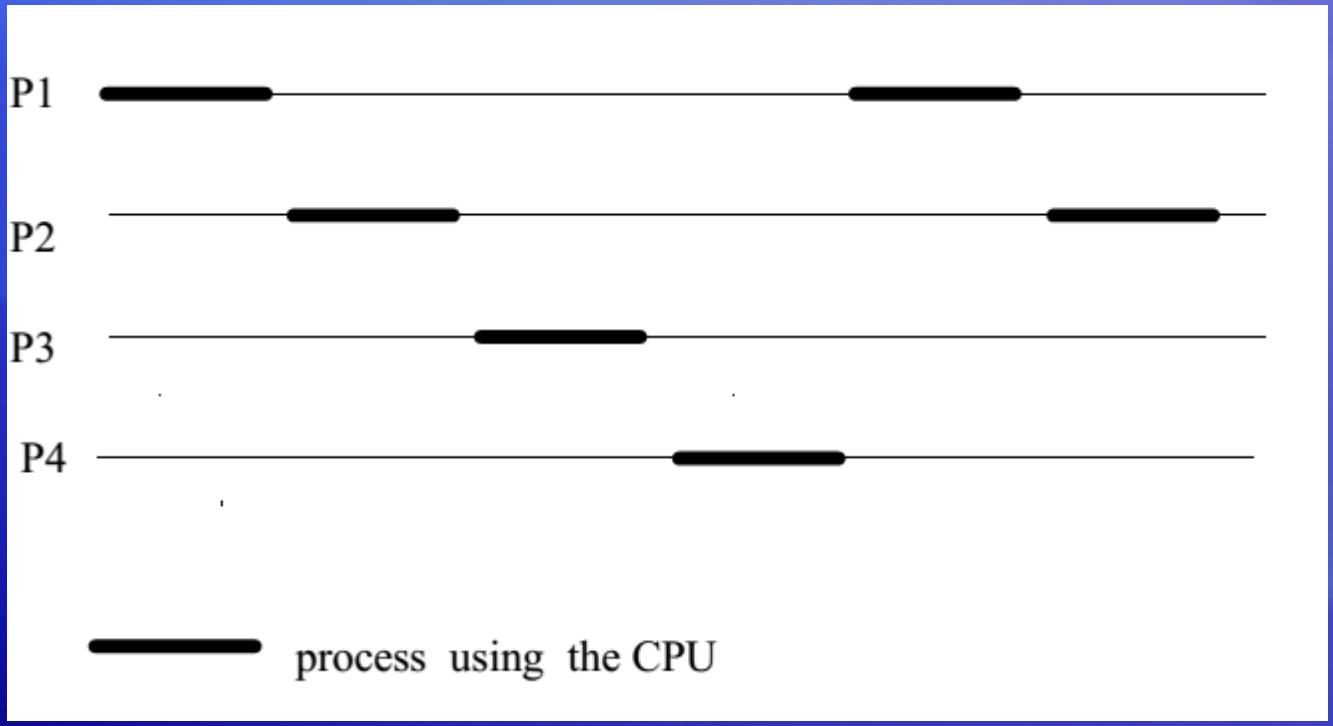
OS Features Needed for Multiprogramming

- I/O routine supplied by the system
- Memory Management – the system must allocated the memory to several jobs
- CPU scheduling – the system must choose among the several jobs to run.
- Allocation of devices.

Process States in a Multiprogrammed OS



Multiprogramming



Time Sharing Systems – Interactive Computing

- The CPU is multiplexed among several jobs that are kept in memory and on disk (the CPU is allocated to a job only if the job is in memory)
- The switching between jobs occurs so frequently that the users can interact with each program while it is running.
  - As switching occurs rapidly it gives the user the impression that the computer system is dedicated to that user.
- An interactive system provides direct communication between the user and the system.
- The user gives instruction to the operating system / program using the keyboard/mouse and waits for immediate results
  - Therefore the response time needs to be short.

Multiprogramming

- The OS gives each process a certain timeslice to run.
- Control is passed to another process if:
  - Running process ends before the timeslice expires
    - Running process leaves the system
  - Running process needs an I/O operation
    - Running process joins the I/O device queue
  - Time Slice Expires
    - Running process goes back to the CPU queue

Desktop Systems

- Personal computers – computer system dedicated to a single user
- I/O devices – keyboards, mice, display screens, small printers
- User convenience and responsiveness
- Can adopt technology developed for larger operating system
- Often individuals have sole use of computer and do not need advanced CPU utilization of protection features
- May run several different types of operating systems (Windows, MacOS, UNIX, Linux)

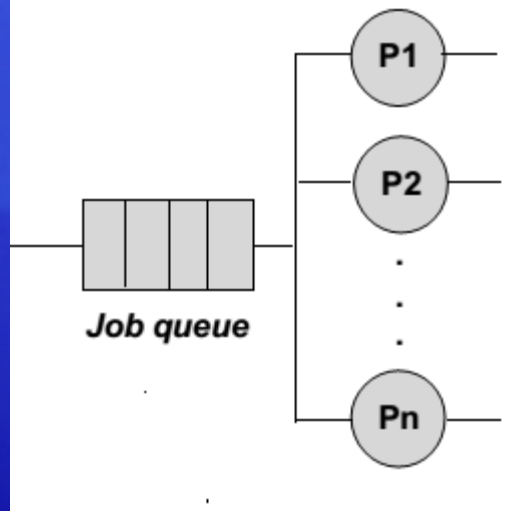
Parallel Systems

- Systems with more than one CPU in close communication
- Also known as multiprocessor systems
- Tightly coupled system – processors share memory and a clock; communication usually takes place through the shared memory

Advantages of parallel system:

- Increased throughput
- Economical
- Increased reliability (in some cases)
  - graceful degradation
  - fail- soft systems

Parallel Systems



Prob. System is Up = Prob. at least one processor is up
= 1 – Prob (all processors are down)
= 1 - p^n

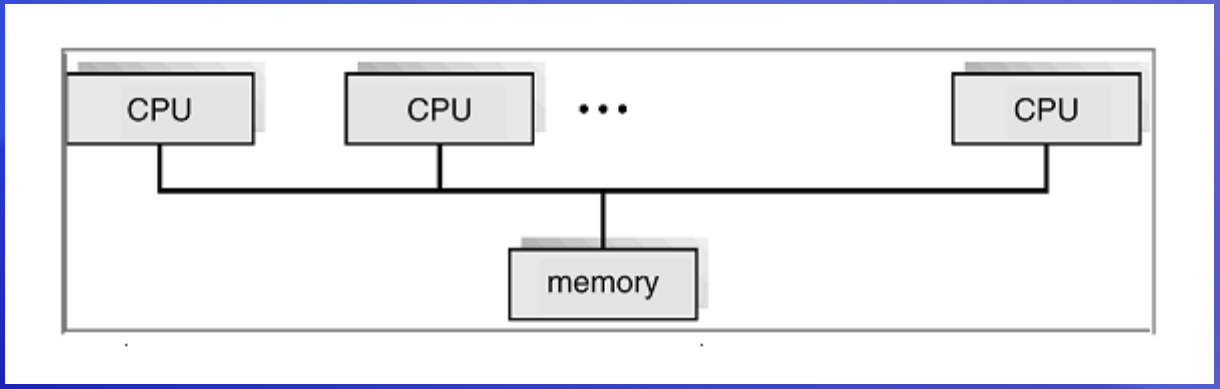
n	*1- p^n*
1	0.200
2	0.360
3	0.488
4	0.590
5	0.672
6	0.738
7	0.790
8	0.832
9	0.866
10	0.893

Parallel Systems

- Asymmetric multiprocessing
  - Each processor is assigned a specific task; master processor schedules and allocated work to slave processors
  - More common in extremely large systems

- Symmetric multiprocessing (SMP)
  - Each processor runs and identical copy of the operating system
  - Many processes can run at once without performance deterioration
  - Most modern operating systems support SMP

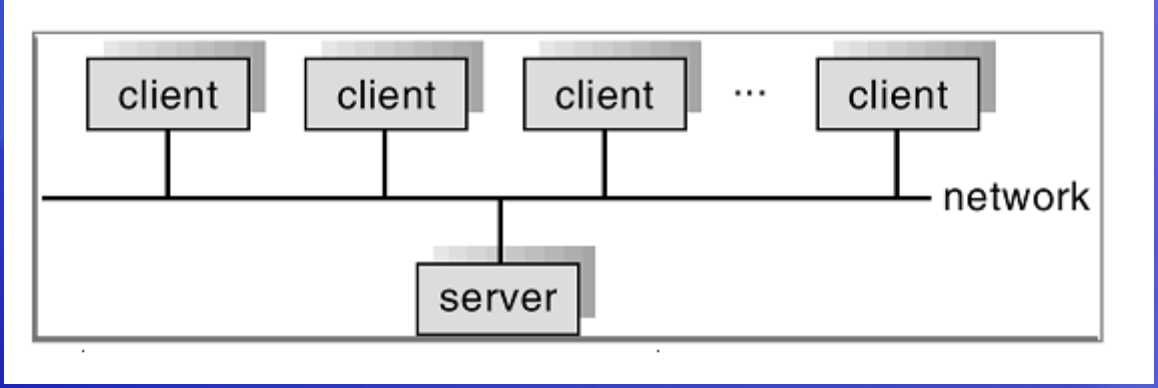
Symmetric Multiprogramming Architecture



Distributed Systems

- Distribute the computation among several physical processors
  - Loosely coupled system – each processor has its own local memory; processors communicate with one another through various communications lines, such as high speed buses or telephone lines
- Advantages of distributed systems
  - Resources Sharing
  - Computation speed up – load sharing
  - Reliability
  - Communications
- Requires networking infrastructure
  - Local area networks (LAN) or Wide area networks (WAN)
  - May be either client -server or peer-to -peer systems

General Structure of Client Server



Clustered Systems

- Clustering allows two or more systems to share storage
- Provides high reliability
- Asymmetric clustering: one server runs the application or applications while other servers standby
- Symmetric clustering: all N hosts are running the application or applications

Real Time Systems

- Often used as a control device in a dedicated application such as controlling scientific experiments, medical imaging systems, industrial control systems, and some display systems
- Well-defined fixed-time constraints
- Real- Time systems may be either hard or soft real –time

Real Time Systems

- Hard real-time:
  - Secondary storage limited or absent, data stored in short term memory, or read-only memory (ROM)
  - Conflicts with time- sharing systems, not supported by general-purpose operating systems
- Soft real-time
  - Limited utility in industrial control of robotics
  - Integrate -able with time- share systems
  - Useful in applications (multimedia, virtual reality) requiring tight response times

OPERATING SYSTEM STRUCTURES

PROCESS MANAGEMENT

A **process** is a **program** in execution: (A program is passive, a process active.)

A process has resources (CPU time, files) and attributes that must be managed.

Management of processes includes:

- Process Scheduling (priority, time management, . . .)
- Creation/termination
- Block/Unblock (suspension/resumption)
- Synchronization
- Communication
- Deadlock handling
- Debugging

OPERATING SYSTEM STRUCTURES

MAIN MEMORY MANAGEMENT

- Allocation/de-allocation for processes, files, I/O.
- Maintenance of several processes at a time
- Keep track of who's using what memory
- Movement of process memory to/from secondary storage.

FILE MANAGEMENT

A file is a collection of related information defined by its creator. Commonly, files represent programs (both source and object forms) and data.

The operating system is responsible for the following activities in connections with file management:

- File creation and deletion.
- Directory creation and deletion.
- Support of primitives for manipulating files and directories.
- Mapping files onto secondary storage.
- File backup on stable (nonvolatile) storage media.

OPERATING SYSTEM STRUCTURES

I/O MANAGEMENT

The I/O subsystem consist of:

- A memory management component that includes buffering, caching and spooling
- Generic device driver code
- Drivers for each device - translate read/write requests into disk position commands.

SECONDARY STORAGE MANAGEMENT

- Disks, tapes, optical, ...
- Free space management (paging/swapping)
- Storage allocation (what data goes where on disk)
- Disk scheduling

OPERATING SYSTEM STRUCTURES

NETWORKING

- Communication system between distributed processors.
- Getting information about files/processes/etc. on a remote machine.
- Can use either a message passing or a shared memory model.

PROTECTION

- Of files, memory, CPU, etc.
- Means controlling of access
- Depends on the attributes of the file and user

SYSTEM PROGRAMS

- Command Interpreters -- Program that accepts control statements (shell, GUI interface, etc.)
- Compilers/linkers
- Communications (ftp, telnet, etc.)