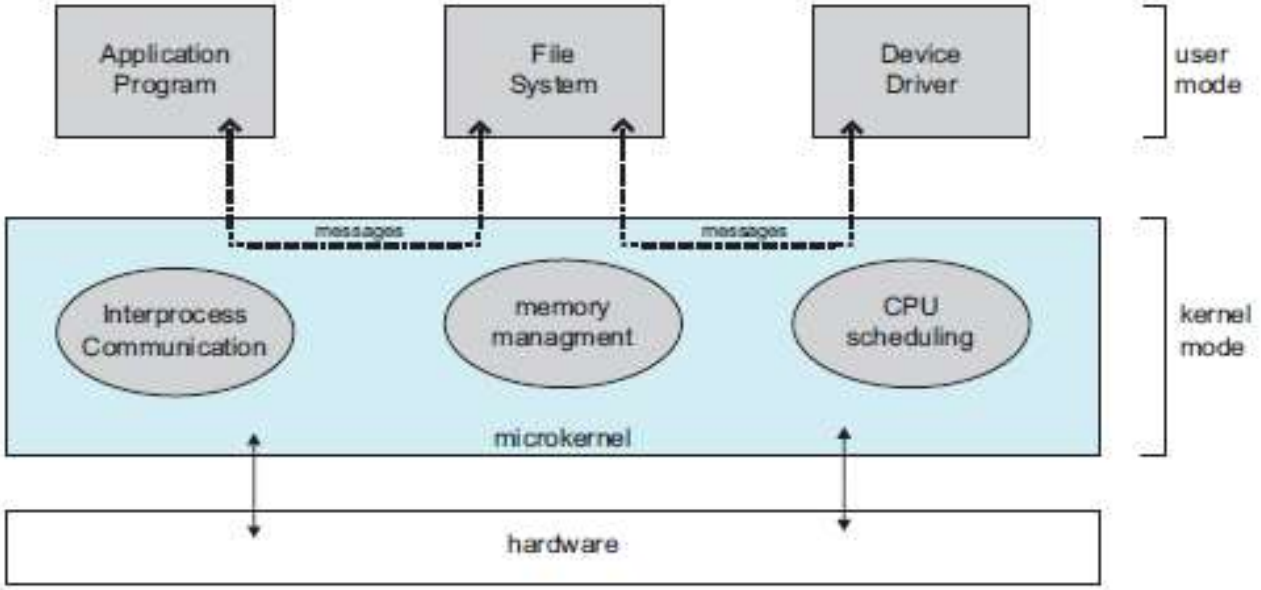
Operating Systems

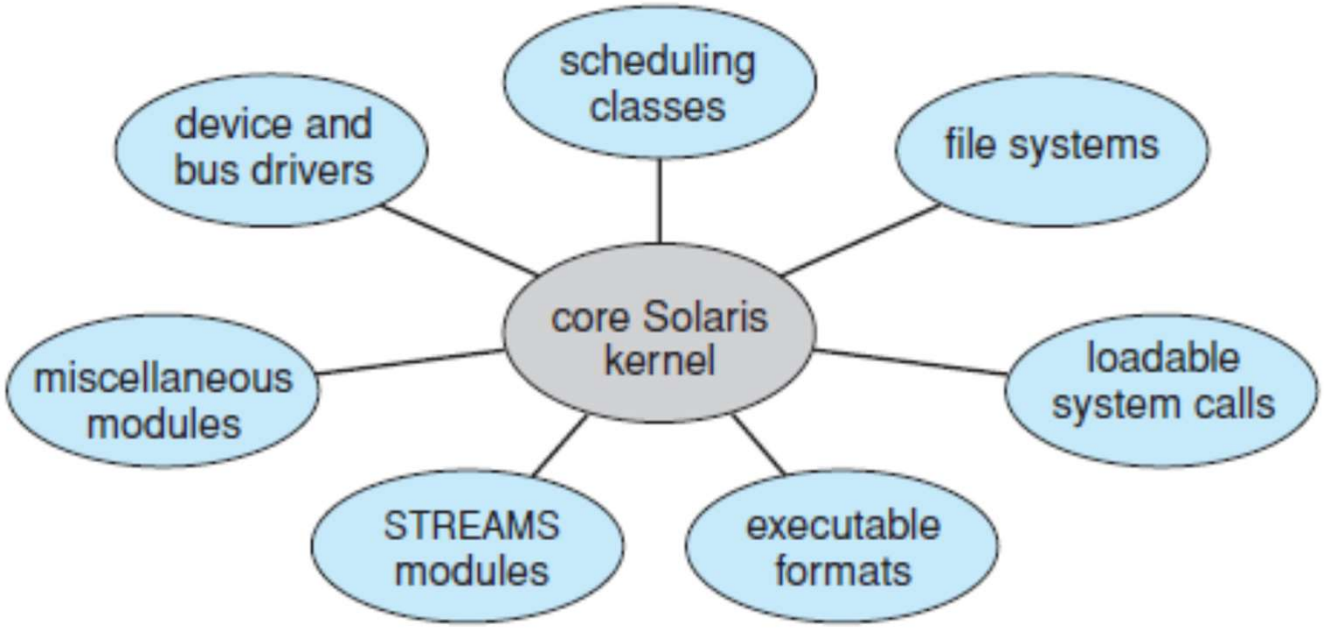
Instructor: Dr. Shachi Sharma

Introduction -2

Layered Architecture (Mach OS)

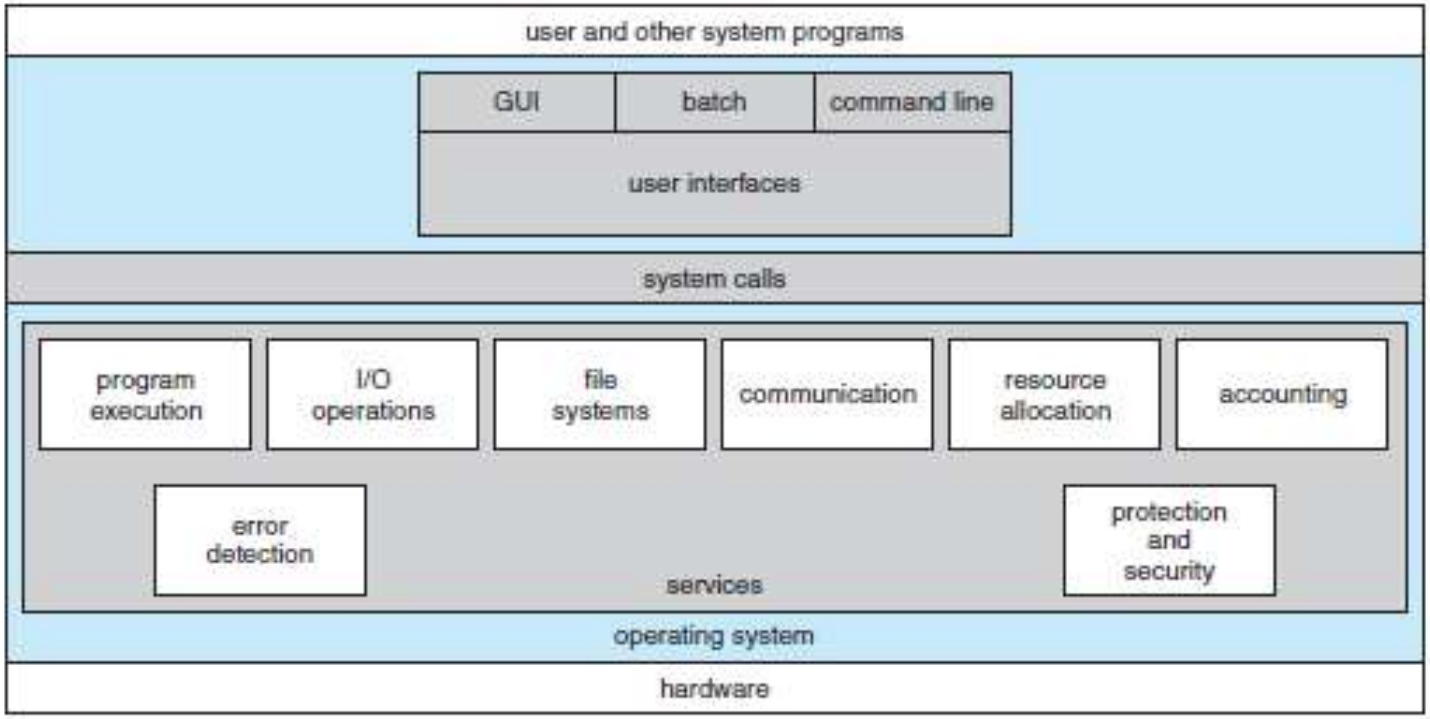


Module Based Architecture



Windows, Unix, Mac OS, Linux, Solaris follow module based architecture

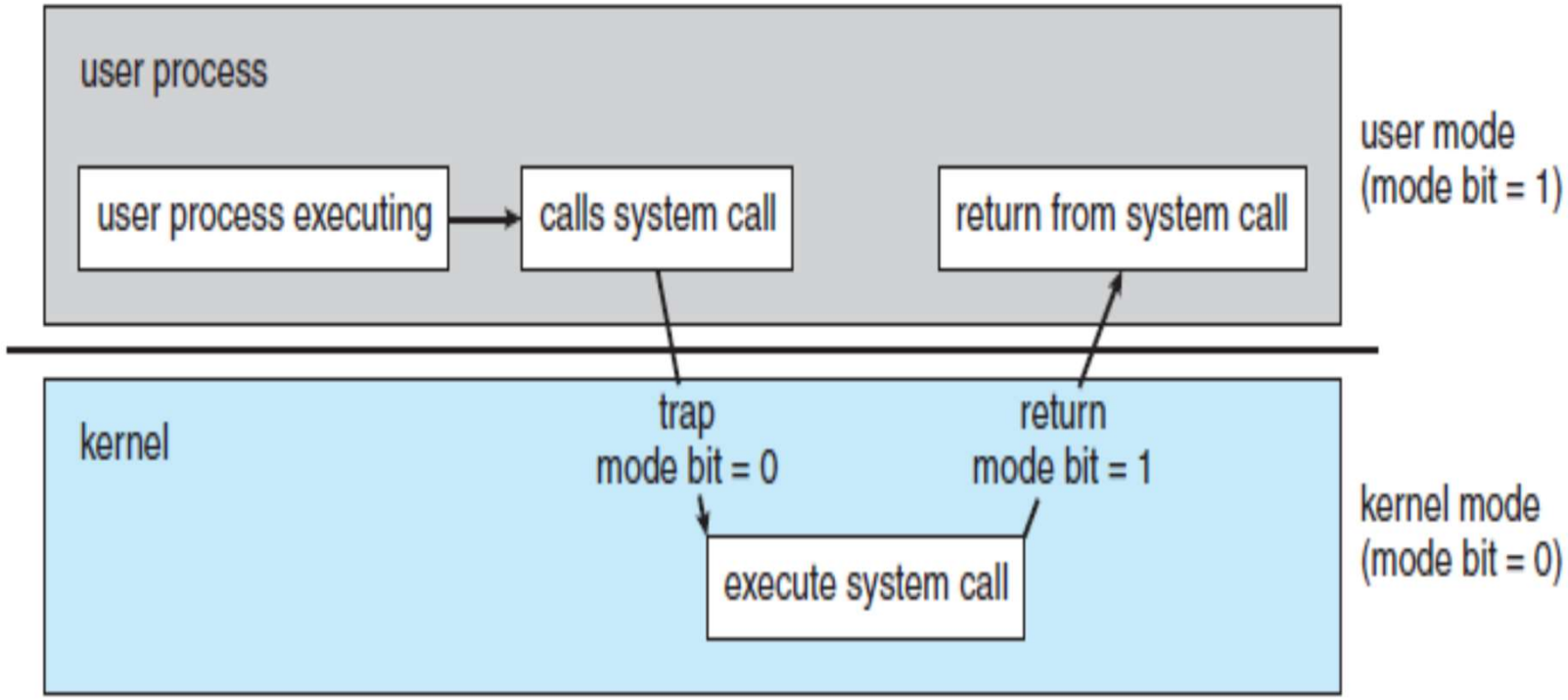
Operating System Services



System Calls

- User requests a service from OS using system call
- Mostly C and C++ programs
- Types of System Calls
  - Process control
  - File manipulation
  - Device manipulation
  - Information maintenance
  - Communications
  - Protection

Interrupt Handling Modes



Interrupts and Interrupt Handling (Heart of the OS)

- Interrupts:
  - All devices send a signal to the CPU whenever they require to ``communicate'' with the CPU.
  - CPU is switched to Interrupt Handler program (part of the OS/supplied externally).
  - The Interrupt Handler (IH)``saves'' various information with respect to the running program and it is jumped to a fixed location in memory – the IH routine.
  - The IH routine – binaries (programs) that involve instructions and commands to communicate with the device.
  - The IH routine may either handle the device interrupt and jump back to executing the saved task
  - In case of task scheduler timer clock interrupt the CPU scheduler handles the tasks/process and handles it.
  - Software induced interrupt is called trap or exception
    - Invalid memory address, division by zero

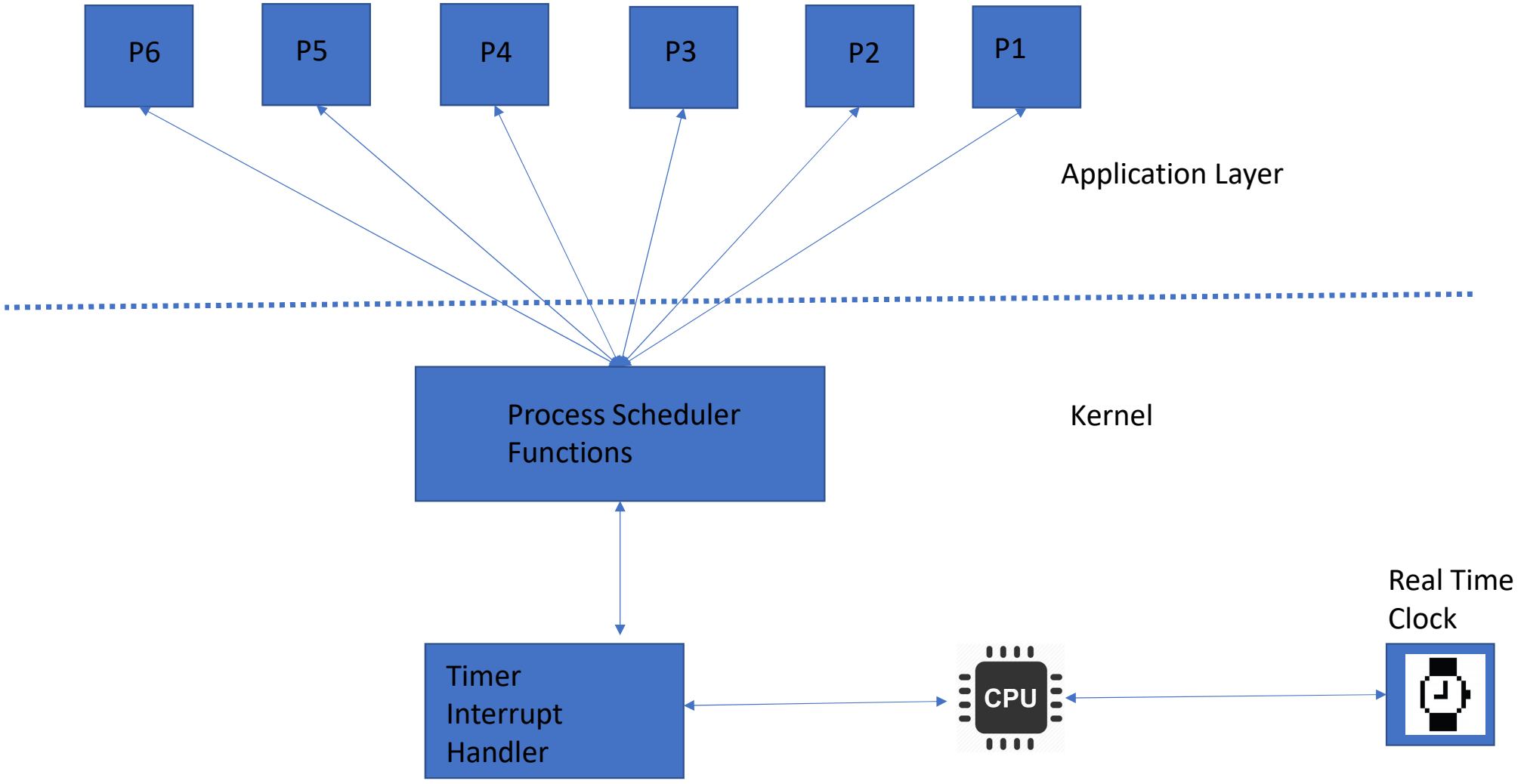
Processes/Tasks/Threads

- Running program binary
- The OS loads up the binary
  - Runtime linker supplies the necessary address bindings – the address of memory where local variables are stored – MMU Used.
  - Saves ``state'' of the running program in Process Control Block (PCB) for each process.
  - Loads the ``state'' of the new program and starts executing it.
- Every so often a timer interrupt ``interrupts'' the CPU to switch the process/task.
- Timer interrupt handler → invokes → process scheduler.

Multi-Processing / Multi-Programming

- Multiple processes running in tandem giving the illusion that multiple programs are running.
- Process executions synchronized with the help of the process/task scheduler.
  - Scheduler is invoked periodically by the timer interrupt handler which saves the state of a task and loads up another.
  - Gives illusion of multi-processing.
- Task state –
  - Every running program has a task / process state which includes:
    - Process ID (PID)
    - CPU registers
    - Memory addresses for code, stack, data, heap and BSS
    - I/O operations in progress.
    - Interrupts that are unhandled.
    - Previous process running and next process that needs to be run (in the process queue).

Processes


P6
P5
P4
P3
P2
P1
Application Layer
Process Scheduler Functions
Kernel
Timer Interrupt Handler
CPU
Real Time Clock

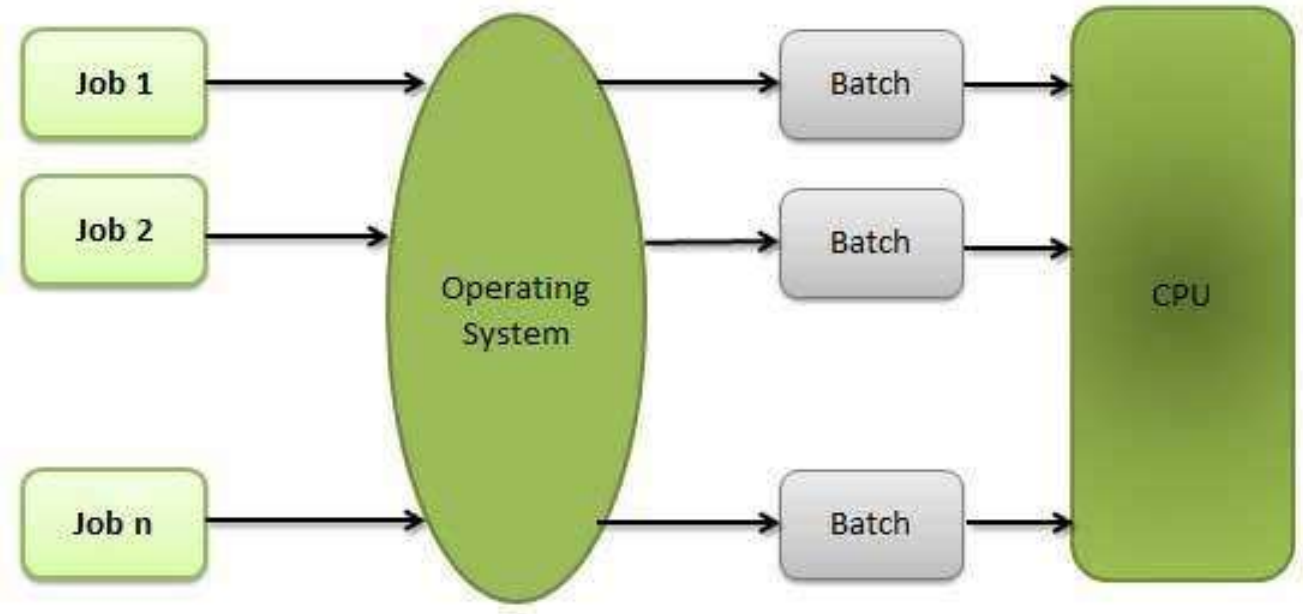
Multi-Programming

- **Advantages**
  - High and efficient CPU utilization.
  - User feels that many programs are allotted CPU almost simultaneously.
- **Disadvantages**
  - CPU scheduling is required.
  - To accommodate many jobs in memory, memory management is required.

Batch Processing

- Operating System collects the programs and data together in a batch before processing starts. An operating system does the following activities related to batch processing −
  - The OS defines a job which has predefined sequence of commands, programs and data as a single unit.
  - The OS keeps a number a jobs in memory and executes them without any manual information.
  - Jobs are processed in the order of submission, i.e., first come first served fashion.
  - When a job completes its execution, its memory is released and the output for the job gets copied into an output spool for later printing or processing.

Batch Processing



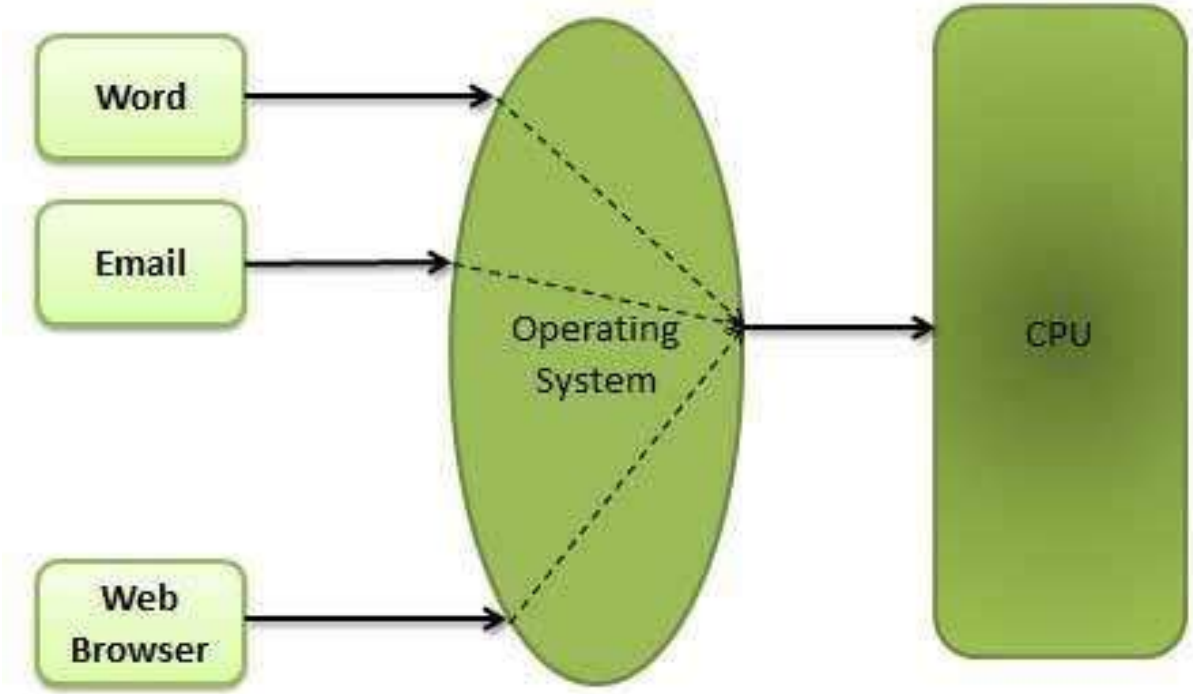
Batch Processing

- **Advantages**
  - Batch processing takes much of the work of the operator to the computer.
  - Increased performance as a new job get started as soon as the previous job is finished, without any manual intervention.
- **Disadvantages**
  - Difficult to debug program.
  - A job could enter an infinite loop.
  - Due to lack of protection scheme, one batch job can affect pending jobs.

Multi-Tasking/Time-Sharing

- Multitasking is when multiple jobs are executed by the CPU simultaneously by switching between them. Switches occur so frequently that the users may interact with each program while it is running. An OS does the following activities related to multitasking −
  - The user gives instructions to the operating system or to a program directly, and receives an immediate response.
  - The OS handles multitasking in the way that it can handle multiple operations/executes multiple programs at a time.
  - Multitasking Operating Systems are also known as Time-sharing systems.
  - These Operating Systems were developed to provide interactive use of a computer system at a reasonable cost.
  - A time-shared operating system uses the concept of CPU scheduling and multiprogramming to provide each user with a small portion of a time-shared CPU.
  - Each user has at least one separate program in memory.

Multi-Tasking/Time-Sharing



Next Lecture

- Process and Process Management