Operating Systems

Dr. Malek Mouhoub

Mathematics and Computer Science Department

University of Lethbridge

1999

Plan

- Basic Notions
- Process Management
- Storage Management
- Case Studies

Basic Notions

1. Introduction
2. Computer-System Structures
3. Operating-System Structures

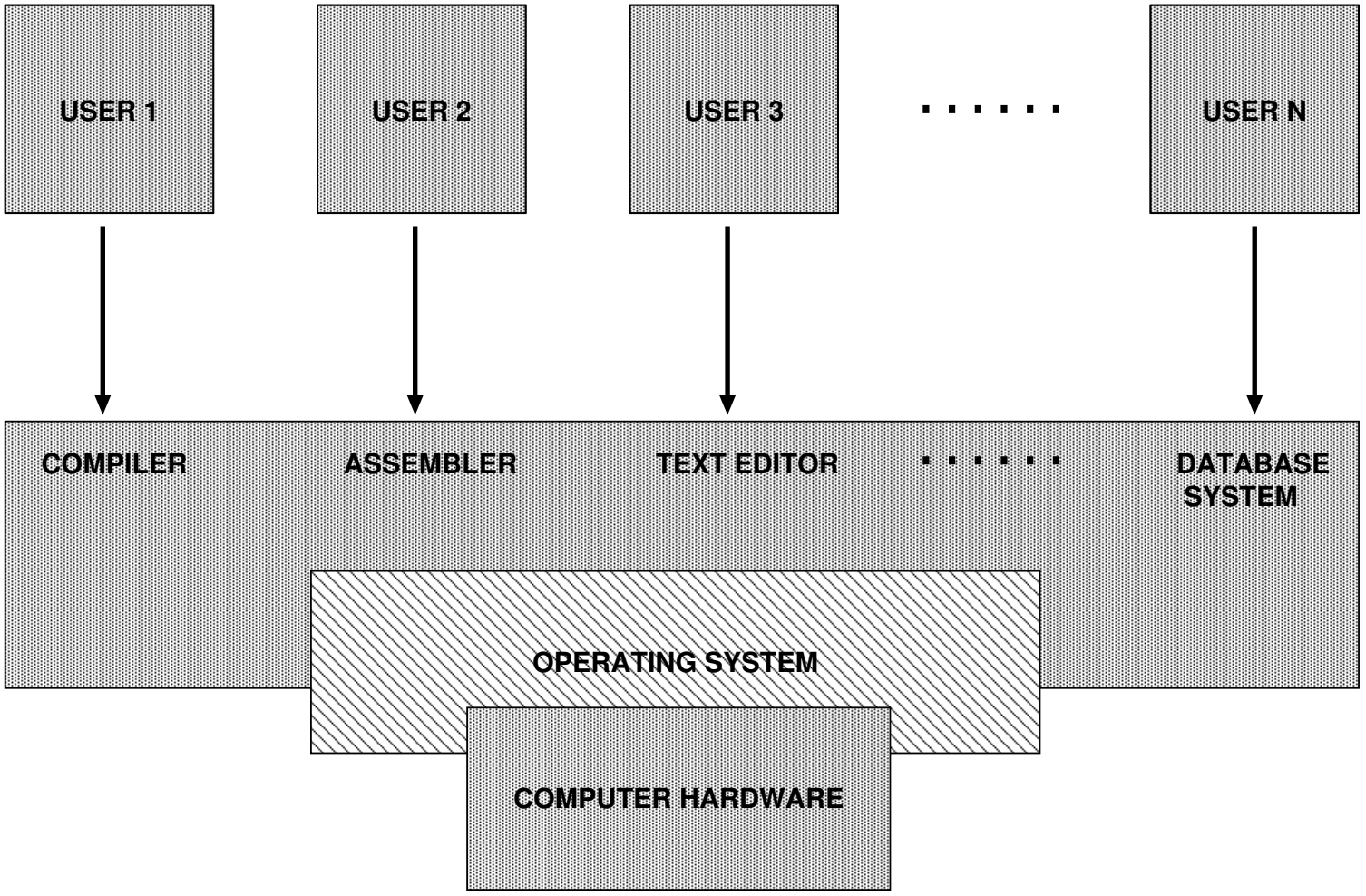
1. Introduction

- What is an operating system ?
- Simple Batch System
- Multiprogramming Batched Systems
- Time-Sharing Systems
- Personal Computer Systems
- Parallel Systems
- Distributed Systems
- Real-Time Systems

What is an Operating System ?

- Interface between the computer and the user.
- Provide the basic environment for working.
- Use the computer **efficiently** and make it **convenient** to use.

Computer System Components



Operating System Definitions

- Resource allocator
  - Conflict when requesting the same resource by many users.
    - Make **efficient** decisions : **optimal** use of resources.
- Controls the various devices (I/O devices) and user programs.
- Set of basic functions (interface) used by the programs or applications.
- The one program running all times : kernel.

Operating System versus Computer Architecture

- Changing the design of the hardware simplify the operating system tasks.
- Improving the operating system for a convenient and efficient use of the hardware.

Simple Batch Systems

- Hire an operator (user $\neq$ operator).
- Add a card reader.
- 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.

Simple Batch Systems (Cont.)

- Problems

  1. How does the monitor know about the nature of the job (e.g., Fortran versus Assembly) or which program to execute ?
  2. How does the monitor distinguish
     (a) job from job ?
     (b) data from program ?

- Solution
  - Introduce control cards.

Simple Batch Systems (Cont.)

Control cards

- Special cards that tell the resident monitor which programs to run : $ JOB, $ FTN, $ RUN, $ DATA, $ END.
- Special characters distinguish control cards from data or programs cards :

  $ in column 1,

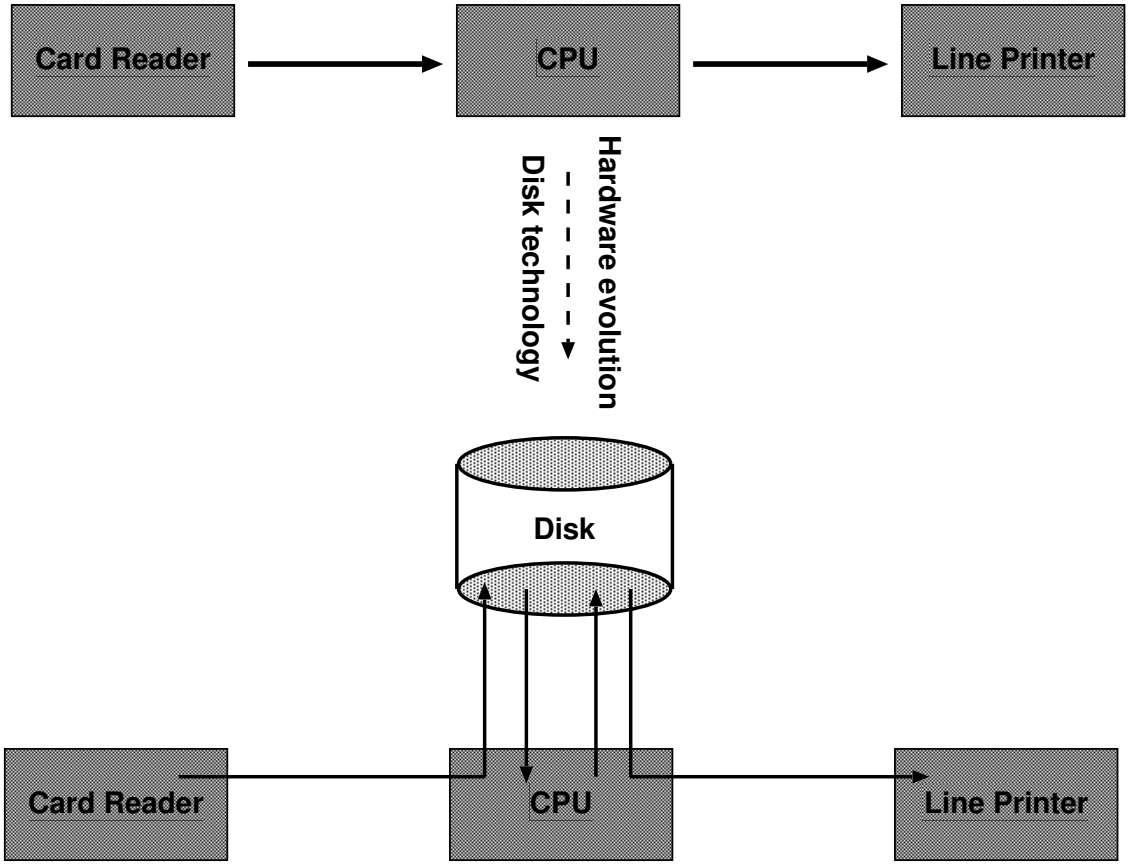
  // in column 1 and 2,

  7-9 in column 1.

Simple Batch Systems (Cont.)

- Parts of resident monitor
  - Control card interpreter : responsable for reading and carrying out instructions on the cards.
  - Loader : loads systems programs and applications programs into memory.
  - Device drivers : know special characteristics and properties for each of the system's I/O devices.
- Problem : Slow Performance - I/O and CPU could not overlap; card reader very slow.

Spooling


Card Reader
CPU
Line Printer
Hardware evolution
Disk technology
Disk
Card Reader
CPU
Line Printer

Spooling

- Overlap I/O of one job with computation of another job. While executing one job, the OS :
  - reads next job from card reader into a storage area on the disk (job queue).
  - outputs printout of previous job from disk to printer.
- Job pool : data structure that allows the OS to select which job to run next in order to increase CPU utilization.

Multiprogrammed Batch Systems

- Increasing CPU utilization by organizing jobs such that the CPU always has one to execute.
- Disk technology $\Rightarrow$ job pool $\Rightarrow$ direct access to the set of jobs to be processed.
- Job scheduling : choose, among the job pool, the set of jobs to be brought into memory.
  - Each job selected is loaded into memory for execution.
  - Memory management : the system must allocate the memory for several jobs.
- CPU scheduling : choose one job among the list of jobs that are ready to run.

Time-Sharing Systems : Interaction Computing

- In multiprogramming there is no **interaction** between the user and and the job when the later is executing.
- In **time-sharing** or **multitasking** multipe jobs are executed by the CPU switching between them so frequently that the users may interact with each program while it is running.
- The CPU is multiplexed among several jobs that are kept in memory and on disk (serving as backing store for main memory : **virtual memory**).
- On-line communication between the user and the system, input is provided by the keyboard and a display screen is used as output.
- Many users can share the computer simultaneously and interactively.
- On-line file system available for users to access data and code.

Personal-Computer Systems

- Personal computers : computer system dedicated to a single user.
- I/O devices : keyboard, mouse, display screen, small printer.
- User convenience and responsiveness.
- Can adopt technology developed for larger operating systems. Often individuals have sole use of computer and do not need advanced CPU utilization or protection features.

Parallel Systems

- Multiprocessor systems with more than one CPU in close communication.
- Tightly coupled system : processors share memory and a clock; communication usually takes place through the shared memory.
- Advantages of parallel systems :
  - Increases throughput
  - Economical
  - Increased reliability
    - graceful degradation
    - fail-soft systems

Parallel Systems (Cont.)

- Symmetric multiprocessing
  - Each processor runs an identical copy of the operating system.
  - Many processors can run at once without performance deterioration.
- Asymmetric multiprocessing
  - Each processor is assigned a specific task; master processor schedules and allocates work to slave processors.
  - More common in extremely large systems.

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 communication lines, such as high-speed buses or telephone lines.
- Advantages of distributed systems :
  - Resource sharing
  - Computation speed up - load sharing
  - Reliability
  - Communication

Real-Time Systems

- Returning results within time constraints.
- Often used as control device in a dedicated application. such as controling scientific experiments, medical imaging systems, industrial control systems, and some display systems.
- Hard real-time system.
  - 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 system :
  - Limited utility in industrial control or robotics.
  - Useful in applications (multimedia, virtual reality) requiring advanced operating-system features.

Process Management

4. Processes
5. CPU Scheduling
6. Process Synchronization
7. Deadlocks

Storage Management

8. Memory Management
9. Virtual Memory
10. File-System Interface
11. File-System Implementation

Case Studies

21. The UNIX System

22. The Linux System

23. Windows NT

24. Historical Perspectives