SNS COLLEGE OF TECHNOLOGY

Coimbatore-35.
An Autonomous Institution

Accredited by NBA – AICTE and Accredited by NAAC – UGC with 'A+' Grade
Approved by AICTE, New Delhi & Affiliated to Anna University, Chennai

COURSE NAME : 19CST201 – OPERATING SYSTEMS

II YEAR/ IV SEMESTER

UNIT – I OVERVIEW AND PROCESS MANAGEMENT

Topic: Operating System Structure

Dr.A.Sumithra

Associate Professor

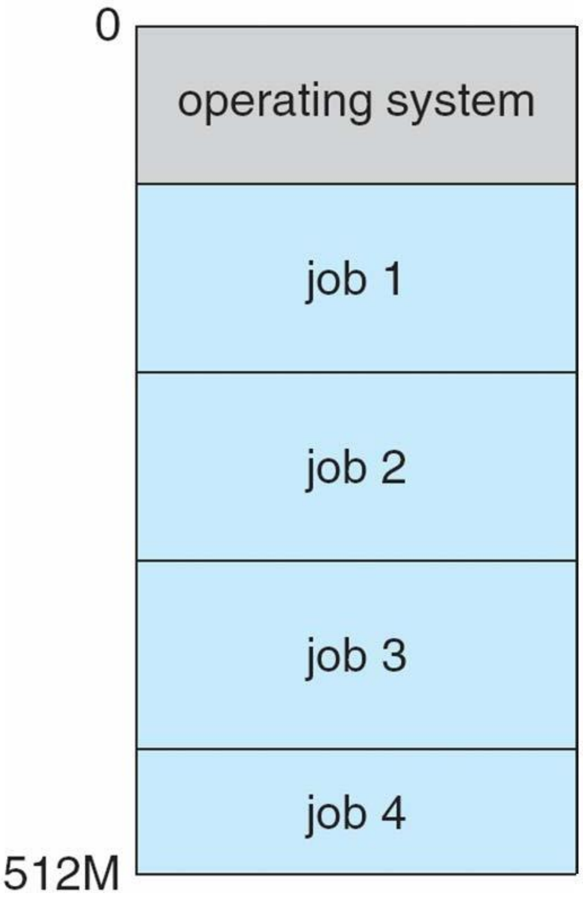
Department of Computer Science and Engineering

Operating System Structure

- □ **Multiprogramming** (**Batch system**) needed for efficiency
  - □ Single user cannot keep CPU and I/O devices busy at all times
  - □ Multiprogramming organizes jobs (code and data) so CPU always has one to execute
  - □ A subset of total jobs in system is kept in memory
  - □ One job selected and run via **job scheduling**
  - □ When it has to wait (for I/O for example), OS switches to another job

- □ **Timesharing** (**multitasking**) is logical extension in which CPU switches jobs so frequently that users can interact with each job while it is running, creating **interactive** computing
  - □ **Response time** should be < 1 second
  - □ Each user has at least one program executing in memory **process**
  - □ If several jobs ready to run at the same time **CPU scheduling**
  - □ If processes don’t fit in memory, **swapping** moves them in and out to run
  - □ **Virtual memory** allows execution of processes not completely in memory

Memory Layout for Multiprogrammed System



Operating-System Operations

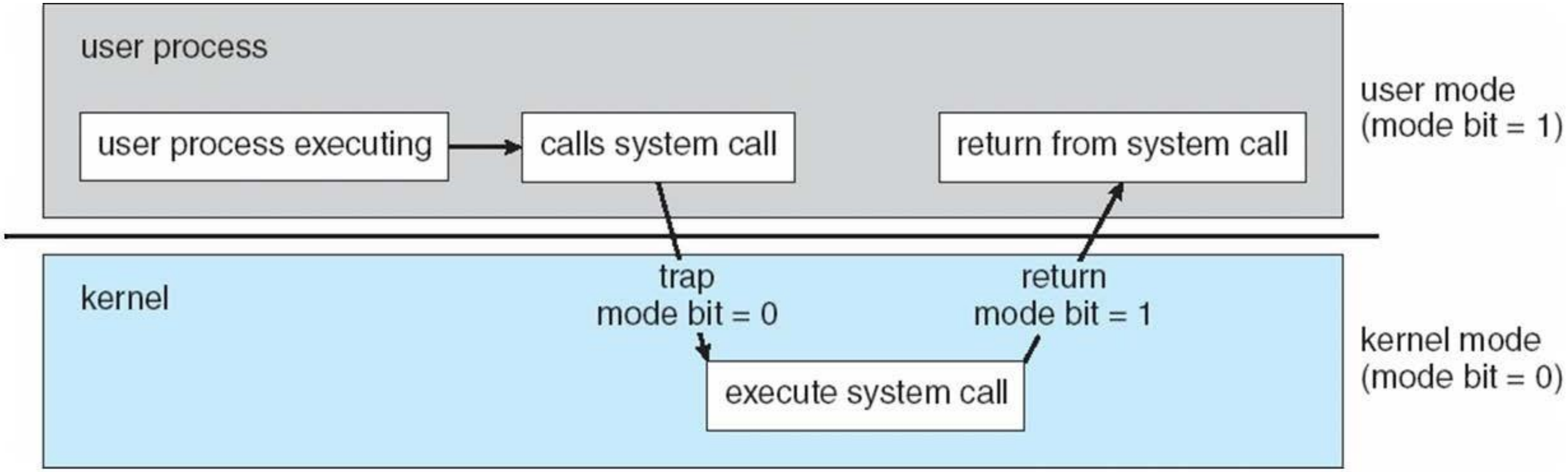
- **Interrupt driven** (hardware and software)
  - Hardware interrupt by one of the devices
  - Software interrupt (**exception** or **trap):**
    - Software error (e.g., division by zero)
    - Request for operating system service
    - Other process problems include infinite loop, processes modifying each other or the operating system

Operating-System Operations (cont.)

- **Dual-mode** operation allows OS to protect itself and other system components
  - **User mode** and **kernel mode**
  - **Mode bit** provided by hardware
    - Provides ability to distinguish when system is running user code or kernel code
    - Some instructions designated as **privileged**, only executable in kernel mode
    - System call changes mode to kernel, return from call resets it to user
- Increasingly CPUs support multi-mode operations
  - i.e. **virtual machine manager** (**VMM**) mode for guest **VMs**

Transition from User to Kernel Mode

- □ Timer to prevent infinite loop / process hogging resources
  - □ Timer is set to interrupt the computer after some time period
  - □ Keep a counter that is decremented by the physical clock.
  - □ Operating system set the counter (privileged instruction)
  - □ When counter zero generate an interrupt
  - □ Set up before scheduling process to regain control or terminate program that exceeds allotted time



Process Management

- □ A process is a program in execution. It is a unit of work within the system. Program is a ***passive entity***, process is an ***active entity***.
- □ Process needs resources to accomplish its task
  - □ CPU, memory, I/O, files
  - □ Initialization data
- □ Process termination requires reclaim of any reusable resources
- □ Single-threaded process has one **program counter** specifying location of next instruction to execute
  - □ Process executes instructions sequentially, one at a time, until completion
- □ Multi-threaded process has one program counter per thread
- □ Typically system has many processes, some user, some operating system running concurrently on one or more CPUs
  - □ Concurrency by multiplexing the CPUs among the processes / threads

Process Management Activities

The operating system is responsible for the following activities in connection with process management:

- □ Creating and deleting both user and system processes
- □ Suspending and resuming processes
- □ Providing mechanisms for process synchronization
- □ Providing mechanisms for process communication
- □ Providing mechanisms for deadlock handling

Memory Management

- To execute a program all (or part) of the instructions must be in memory
- All (or part) of the data that is needed by the program must be in memory.
- Memory management determines what is in memory and when
  - Optimizing CPU utilization and computer response to users
- Memory management activities
  - Keeping track of which parts of memory are currently being used and by whom
  - Deciding which processes (or parts thereof) and data to move into and out of memory
  - Allocating and deallocating memory space as needed

Storage Management

- OS provides uniform, logical view of information storage
  - Abstracts physical properties to logical storage unit - **file**
  - Each medium is controlled by device (i.e., disk drive, tape drive)
    - Varying properties include access speed, capacity, data-transfer rate, access method (sequential or random)

1. File-System management

- Files usually organized into directories
- Access control on most systems to determine who can access what
- OS activities include
  - Creating and deleting files and directories
  - Primitives to manipulate files and directories
  - Mapping files onto secondary storage
  - Backup files onto stable (non-volatile) storage media

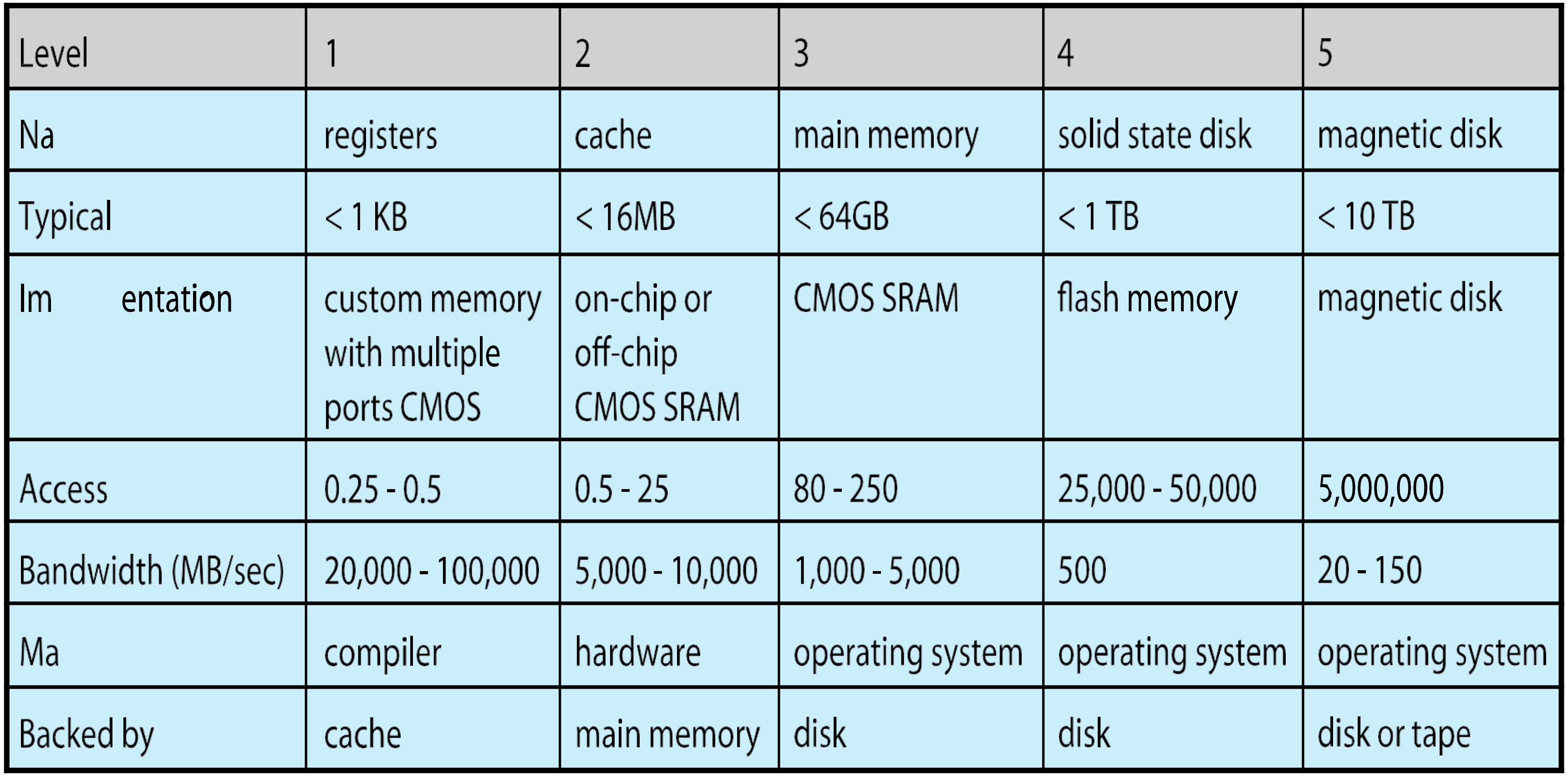
Storage Management

2. Mass-Storage Management

- □ Usually disks used to store data that does not fit in main memory or data that must be kept for a “long” period of time
- □ Proper management is of central importance
- □ Entire speed of computer operation hinges on disk subsystem and its algorithms
- □ OS activities
  - □ Free-space management
  - □ Storage allocation
  - □ Disk scheduling

Storage Management

3. Caching → Performance of Various Levels of Storage

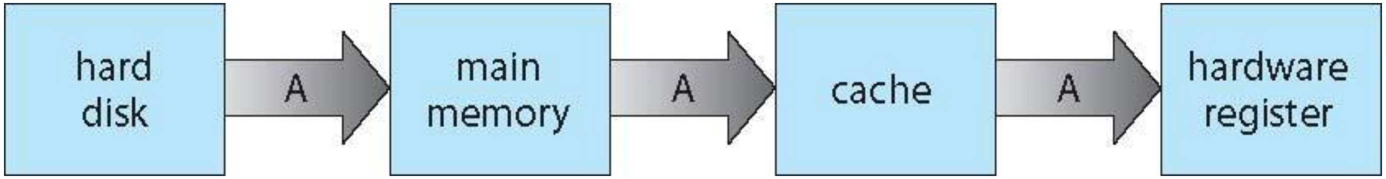


Level	1	2	3	4	5
Na	registers	cache	main memory	solid state disk	magnetic disk
Typical	< 1 KB	< 16MB	< 64GB	< 1 TB	< 10 TB
Im entation	custom memory with multiple ports CMOS	on-chip or off-chip CMOS SRAM	CMOS SRAM	flash memory	magnetic disk
Access	0.25 - 0.5	0.5 - 25	80 - 250	25,000 - 50,000	5,000,000
Bandwidth (MB/sec)	20,000 - 100,000	5,000 - 10,000	1,000 - 5,000	500	20 - 150
Ma	compiler	hardware	operating system	operating system	operating system
Backed by	cache	main memory	disk	disk	disk or tape

Movement between levels of storage hierarchy can be explicit or implicit

Migration of data “A” from Disk to Register

- □ Multitasking environments must be careful to use most recent value, no matter where it is stored in the storage hierarchy



- □ Multiprocessor environment must provide **cache coherency** in hardware such that all CPUs have the most recent value in their cache
- □ Distributed environment situation even more complex
  - □ Several copies of a datum can exist
  - □ Various solutions covered in Chapter 17

Storage Management

4. I/O Subsystem

- □ One purpose of OS is to hide peculiarities of hardware devices from the user
- □ I/O subsystem responsible for
  - □ Memory management of I/O including buffering (storing data temporarily while it is being transferred), caching (storing parts of data in faster storage for performance), spooling (the overlapping of output of one job with input of other jobs)
  - □ General device-driver interface
  - □ Drivers for specific hardware devices

REFERENCES

TEXT BOOKS:

T1 Silberschatz, Galvin, and Gagne, “Operating System Concepts”, Ninth Edition, Wiley India Pvt Ltd, 2009.)

T2. Andrew S. Tanenbaum, “Modern Operating Systems”, Fourth Edition, Pearson Education, 2010

REFERENCES:

R1 Gary Nutt, “Operating Systems”, Third Edition, Pearson Education, 2004.

R2 Harvey M. Deitel, “Operating Systems”, Third Edition, Pearson Education, 2004.

R3 Abraham Silberschatz, Peter Baer Galvin and Greg Gagne, “Operating System Concepts”, 9th Edition, John Wiley and Sons Inc., 2012.

R4. William Stallings, “Operating Systems – Internals and Design Principles”, 7th Edition, Prentice Hall, 2011