**Storage Device Hierarchy**

Size vs Speed & Cost  
Higher levels are faster but have smaller size ie cache, register

**Cache**

There is cache on the CPU, Hard Disc, ect not only on memory

Caching, speed up a slower storage medium. Reduce bottlenecks.  
-transfer data form a slower to faster storage medium temporarily.

Cache Line: Entry in cache memory

**DMA**

Only for IO devices that can transfer close to memory speed.

Old: Disc -> Buffer -> CPU - > Memory  
  
DMA allows buffer to go directly to memory.

Therefore less CPU interrupts.

**Computer-System Architecture**

Most systems use a single general purpose processor

Multiple processors/Cores allows parallel work to be done.

Advantage:  
-Increase throughput  
-Economy of scale  
-Increased Reliability

**2 Types of Multiprocessing**

-Asymmetric: Allocate processor for specific task  
-Symmetric: Each processor performs all tasks  
  
  
**Multi-Core**:Multiple CPUs, in one CPU. Each core is a “cpu”

**NBNB  
-**Socket: One chip – ie the one CPU on motherboard.  
-Cores: Physical Processors  
-Logical Processors (hardware thread): Allows each core to work “simultaneously” on multiple threads (Multithreading by Intel). This is makes it so you don’t have to wipe the registers when getting a new instruction, can switch between 2 registers instead.  
Instruction pointer/program counter: register that holds the address of the next instruction that must be executed  
L1 cache is smaller and closer to core than L2

**Operating System Structures**

2 types:

Multiprogramming (batch system – **non pre-emptive design**): Best for efficiency. Loads many jobs in memory

Timesharing (multitasking): Shares time cycles between many jobs.

**Operating system operations**

**Interupt Driven**: Can be created by hardware and software  
-Clock in computer can interrupt  
  
Software interrupts are normally for errors.

Microsoft Made to run in ring levels of protection. **(N)**-ring 0, closest to core with most privileges.

Level 0: Kernal mode (bottom Dark blue in CMD)  
Level 3: User Mode (top Light blue in CMD)

This separation helps with making sure developers code does not interfere with the OS.

Virtual machine manager (VMM), creates a level -1, allows multiple operating systems to be run simultaneously.

**Transition from User to Kernal Mode**

System Call (Terminology **NBNB):** An instruction that goes from user mode to kernal mode, and the result is returned to user mode.

Timer exists, to stop a single program from hogging resources. After timer counted down to 0, interrupt occurs, goes to next process.

**Process Management**

**Process:** A Running program  
-A program is only a **process**, when it is running.

Single threaded processes have one program counter, while mutti threaded have one per thread.

**Process Management Activities**

Suspending: temporarily stop running

Synchronization: When one process finished something, its processed data is passed to another.

Communication:

Deadlock: Process grabs holds of a resource, but does not release it due to waiting for something/another process data.

**Mass-Storage Management**

Long term storage

**Cache Coherency**: All values of ‘A’ on hard disk to cache, have same value… a change on hard disk, is reflected in cache

**Protection and Security**

**Protection: A** process is protected from other processes

**Security**: Defence of system from internal and external attacks

**Computing Environments – Distributed**

Allows a task to be distributed between multiple systems.  
-Allows separate systems to work together.  
ie LAN,  
eg For sharing of files.

Client-Server  
Peer-to-Peer  
Virtualization: Needed for cloud computation  
-OS natively compiled for CPU, run guest OS  
-VMM manages this layer.

Emulation: Emulating a device/CPU through software

**Computing Environments – Cloud Computing**

Public cloud: available to anyone  
Private cloud: Only available to company  
Hybrid: mix

Software as a Service (Saas): one/more applications available via the internet.  
Platform as a Service (Paas): Software stack ready for application use via the internet. Eg Database – all software installed already  
Infrastructure as a Service (IaaS): Servers or storage available over internet. Eg Storage for backups – install from OS level

**Computing Environments – Real-Time Embedded Systems**

The OS must have the ability to **prioritise certain** processes, and give all resources to that process.