



Dinosaurs? & Dinosaur Wars

CSCI 6730 / 4730 Operating Systems



Operating Systems Overview



Maria Hybinette, UGA

Frederick P. Brooks' Mythical Man-Month (1975). Description of the software crises - likens large scale programming to a tarpit

No scene from prehistory is quite so vivid as that of the mortal struggles of great beasts in the tar pits. In the mind's eye one sees dinosaurs, mammoths, and saber toothed tigers struggling against the grip of the tar. The fiercer the struggle, the more entangling the tar, and no beast is so strong or so skillful but that he ultimately sinks.

Large-scale programming has over the past decade been such a tar pit, and many great and powerful beasts have thrashed violently in it. Most have emerged with running systems - few have met goals, schedules, and budgets.



Maria Hybinette, UGA

2

Evolution of our Textbook



Maria Hybinette, UGA

3

Outline & Questions

- What is an Operating Systems (OS)?
- What does an OS do?
- What is an OS and what is it not?
- How do I run an OS?
- How does an Operating System run?
- What is the basic structure?
- Computer System Component Architecture

Maria Hybinette, UGA

© 1984, Reed of Lubbock, Tex. Inc., Los Angeles County Natural History Museum.

4

Questions?

- What are the major operating system components?
- What are basic computer system organizations?
- How do you communicate with the operating systems?
- What services are (need to be) provided?

Maria Hybinette, UGA

5

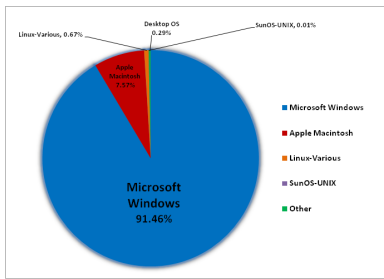
Poll?

- What desktop/laptop OS do you have?
- Which desktop/laptop OSs are you familiar with?
- What do you think the market share (%) is (portion of different OSs)?

Maria Hybinette, UGA

6

Popularity: The Desktop OS Market Share



Desktop OS market share as of January, 2008

http://en.wikipedia.org/wiki/Image:Os_as_of_2008.png

Maria Hyömette, UGA

7

What is an Operating System?

- A hardware manager
- A program that acts as an **intermediary** between a user of a computer and the computer hardware.
 - » Execute user programs and make solving user problems easier.
- Operating system goals:
 - » Make the computer system **convenient** to use.
 - » Use the computer hardware in an **efficient** manner.
 - » Combination of the above.
 - » Handhelds (convenience), Mainframes/Servers (efficiency)

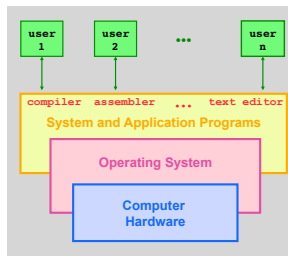
Maria Hyömette, UGA

8

Where is the OS? Computer System Layers

Computer system can be divided roughly in four components:

- Hardware
- Operating system
- Application programs
- Users



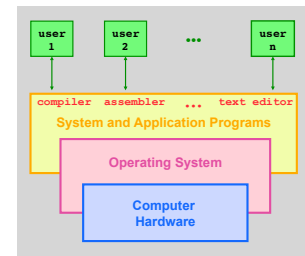
Maria Hyömette, UGA

9

Computer System Layers

Computer system can be divided roughly in four components:

- Hardware:
 - » provides basic computer resources:
 - CPU, Memory, I/O Devices



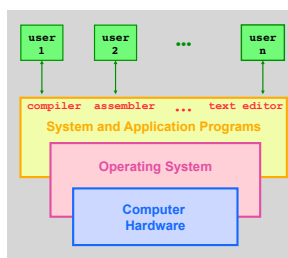
Maria Hyömette, UGA

10

Computer System Layers

Computer system can be divided roughly in four components:

- Hardware
- Operating system:
 - » Controls and coordinates use of hardware among various applications and users.



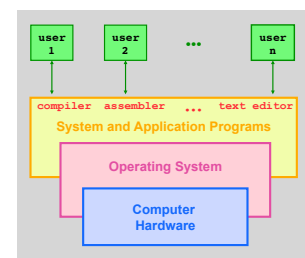
Maria Hyömette, UGA

11

Computer System Layers

Computer system can be divided roughly in four components:

- Hardware
- Operating system
- Application programs
 - » define the ways in which the system resources are used to solve the computing problems of the users
 - Word processors, compilers, web browsers, database systems, video games



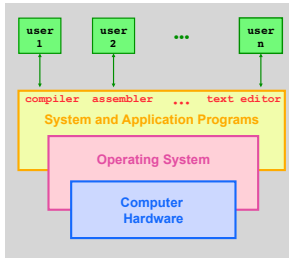
Maria Hyömette, UGA

12

Computer System Layers

Computer system can be divided roughly in four components:

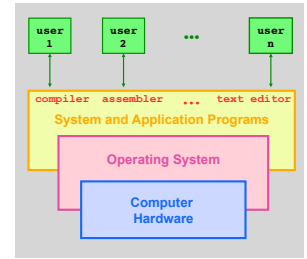
- Hardware
- Operating system
- Application programs
- Users
 - » People, machines, other computers



Computer System Layers

Computer system can be divided roughly in four components:

- Hardware
- Operating system
- Application programs
- Users



What Does Operating Systems Do?

- A Space/Time Controller
- It *allocates resources*
 - » Manages all resources
 - » Decides between conflicting requests for efficient and fair resource use
- It *controls execution*
 - » Controls execution of programs to prevent errors and improper use of the computer

What "Makes up" the Operating System?

- *No universally accepted definition!*
- *"Everything a vendor ships when you order an operating system" used to be a good approximation... But varies wildly*
- Operating System is the **"Kernel"**
 - » the one program that runs at all times on the computer
 - » everything else is either a
 - system program (ships with the operating system) or an
 - application program

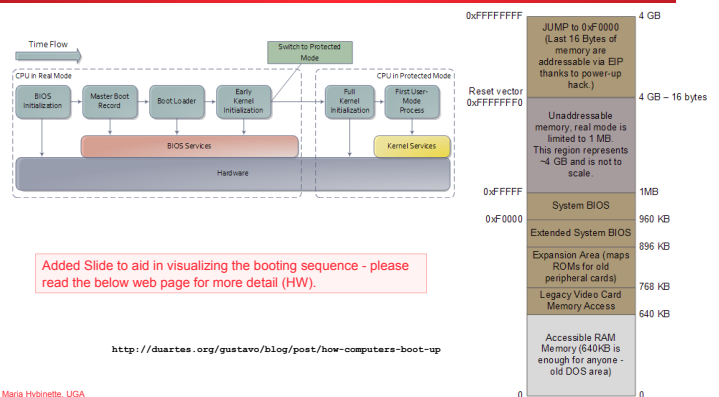


The Computer Startup Process

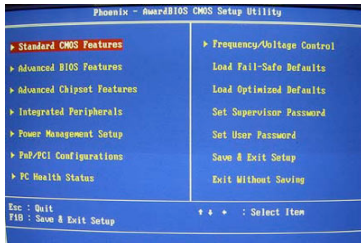
- A *bootstrap program* (initial program) is loaded at power-up or reboot
 - » Stored in firmware in ROM/ EEROM
 - » Stored on a chip on the motherboard ('parentboard')
- Initializes all aspects of system
- Loads operating system kernel (e.g., from disk) and starts execution
- Pentium initial boot strap program is called the system Basic Input Output System or BIOS.



Visual of the Time-Line of the Booting Sequence



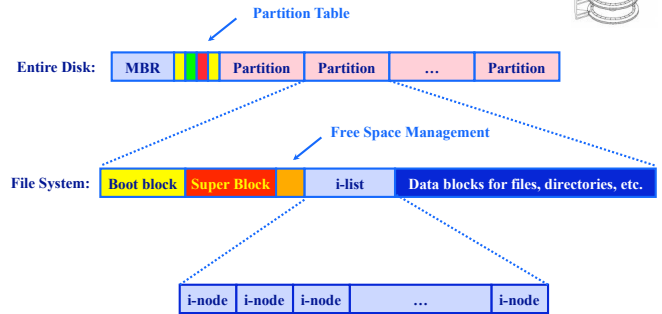
CMOS/BIOS Configuration Utility



Booting from hard disk



Disk is divided into 1+ partitions: one file system per partition



Entire Disk & Booting Computer

- **Master Boot Record (sector 0) - Pentium**
 - » used to boot computer
- **Partition Table**
 - » starting and ending address of each partition
- **“A program (e.g., the system Basic Input Output System or BIOS for Pentiums)” reads in and executes the MBR**
 - » searches for first active partition (noted in the partition table)
 - » reads in its first block (the boot block) and executes it.

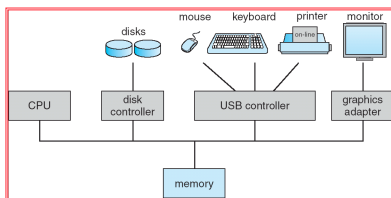


Partition Layout

- **Boot block:**
 - » contains a hardware specific program that is called automatically to load “UNIX” at system startup time
- **Super block:**
 - » file system type, #blocks in file system
- **Free space management (two lists):**
 - » a chain of free data block numbers
 - » a chain of free i-node numbers
- **i-list/i-node table:**
 - » administrative information about a file (meta-data: name, type, location, size, protection bits, ...) structured into an array: inode table or simply the i-list
 - » An i-node number:
 - uniquely identifies a file in a file system
 - is an index to the i-node table



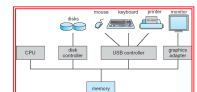
Computer System Organization



- **One or more CPUs, device controllers connect through common bus providing access to shared memory**
- **Concurrent execution of CPUs and devices competing for memory cycles**

Computer System Operations

- **CPU:** the processor that perform the actual computation
- **I/O controllers:**
 - » take commands in registers, generate flags and interrupts
 - » each device controller
 - is in charge of a particular device type
 - has a local buffer for I/O
 - Examples: audio – output device, mouse – input, disk – I/O.
- CPU moves data from/to main memory to/from local buffers.
- I/O is from the device to local buffer of controller.
- Device controller informs CPU that it has finished its operation by causing an interrupt.



Communication: Interrupts



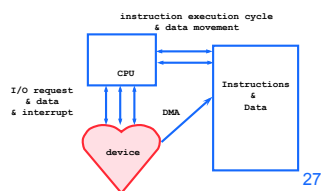
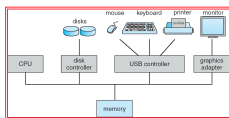
- How do devices communicate to the OS? For example telling the OS to (when) check for user input?

An interrupt is a signal to the processor to temporarily suspend execution because some system event needs handling (alert!).

- Occurrence of an event is **signaled** by interrupts either by software or hardware
 - » A **trap** is a software-generated interrupt caused either by an **error** or a **user request**.
- Modern operating systems are **interrupt driven**.
- OS Stops what it is doing, preserve the current state and then handles the interrupt (=overhead) .

Communication: Direct Memory Access Structure

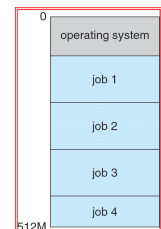
- Used for high-speed I/O devices able to transmit information at close to memory speeds
- Device **controller transfers blocks** of data from buffer storage directly to main memory **without CPU intervention**.
- Only one interrupt is generated per block, rather than the one interrupt per byte.



Multi-programming

Multiprogramming 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



memory layout

User View: Timesharing (Multitasking)

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

Definition of Terms

- **Uniprogramming**
 - » one process at the time.
- **Multiprogramming**
 - » multiple processes (with separate address spaces) concurrently on a machine (more on this later)
- **Multiprocessing**
 - » running programs on a machine with multiple processors.
- **Multithreading**
 - » multiple threads per address space (later).
- **Multitasking**
 - » a single user can run multiple processes.

Storage Structure

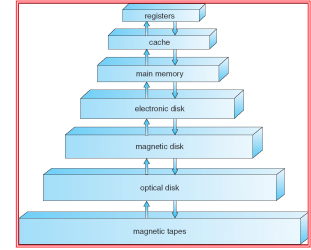
- **Main memory** – only large storage media that the CPU can access directly.
- **Secondary storage** – extension of main memory that provides large nonvolatile storage capacity.
- **Magnetic disks** – rigid metal or glass platters covered with magnetic recording material
 - » Disk surface is logically divided into *tracks*, which are subdivided into *sectors*.
 - » The *disk controller* determines the logical interaction between the device and the computer.

Maria Hyömette, UGA

31

Storage Hierarchy

- Storage systems organized in hierarchy.
 - » Speed
 - » Cost
 - » Volatility
- **Caching** – copying information into faster storage system; main memory can be viewed as a last cache for secondary



Maria Hyömette, UGA

32

Performance of Various Levels of Storage

Level	1	2	3	4
Name	registers	cache	main memory	disk storage
Typical size	< 1 KB	> 16 MB	> 16 GB	> 100 GB
Implementation technology	custom memory with multiple ports, CMOS	on-chip or off-chip CMOS SRAM	CMOS DRAM	magnetic disk
Access time (ns)	0.25 – 0.5	0.5 – 25	80 – 250	5,000,000
Bandwidth (MB/sec)	20,000 – 100,000	5000 – 10,000	1000 – 5000	20 – 150
Managed by	compiler	hardware	operating system	operating system
Backed by	cache	main memory	disk	CD or tape

- Putting this in perspective: Mac Book Pro 2007
 - L2 Cache – 6MB
 - Memory – 4GB (667 MHz)
 - Bus speed 800 MHz, CPU 2.6 Ghz
 - Disk – 233 GB (1.5 Gigabit)

Maria Hyömette, UGA

33

Caching

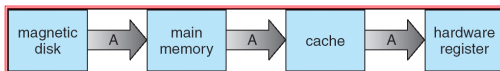
- Use of high-speed memory to hold recently-accessed data.
- Performed at many levels in a computer (in hardware, operating system, software)
- Information in use copied from slower to faster storage temporarily
- **Faster storage (cache)** checked first to determine if information is there
 - » If it is, information used directly from the cache (fast)
 - » If not, data copied to cache and used there
- **Cache smaller than storage being cached**
 - » Cache management important design problem
 - » Cache size and replacement policy

Maria Hyömette, UGA

34

Migration of Integer 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

Maria Hyömette, UGA

35

An Operating System's Core Tasks

- Process Managements
- Memory Managements
- File Managements
- I/O System Managements
- Protection System

Maria Hyömette, UGA

36

Process Management

- A process is a program in execution (an active entity, i.e. a running program)
 - » Basic unit of work on a computer
 - » Examples: compilation process, word processing process
 - » A process needs certain resources:
 - e.g. CPU time, memory, files, I/O devices to accomplish its task
- Each user can run many processes at once (e.g. using &)
 - » One process:
 - cat file1 file2 &
 - » Two processes:
 - ls | wc -l
- A time sharing system (such as UNIX) run several processes by multiplexing between them

Maria Hyönnelie, UGA

37

Process Management Activities

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

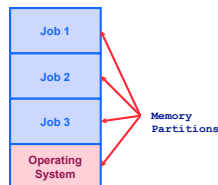
- Creating and deleting both user and system processes
- Suspending and resuming processes
- Providing mechanisms for process synchronization
- Providing mechanisms for process communication
- Providing mechanisms for deadlock handling

Maria Hyönnelie, UGA

38

Memory Management

- Programs become **processes** when they are loaded into memory and start executing.
 - » All data in memory before and after processing
 - » All instructions in memory in order to execute
- Memory management determines what is in memory 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

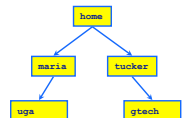


Maria Hyönnelie, UGA

39

File Management

- OS provides uniform, logical view of information storage
 - » Abstracts physical properties to logical storage unit
 - A File : OS maps logical files to physical devices
 - » 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)
- 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 dirs
 - Mapping files onto secondary storage
 - Backup files onto stable (non-volatile) storage media



Maria Hyönnelie, UGA

Mass-Storage Management

- Main memory is volatile and limited in size
 - » Use disks to store 'overflow' and data that needs to be persistent.
- Disks are *slower* than main memory and processors
 - » Entire speed of computer operation hinges on disk subsystem and its algorithms
- OS mass storage management activities:
 - » Free-space management
 - » Storage allocation
 - » Disk scheduling
- Some storage need not be fast
 - » Tertiary storage includes optical storage, magnetic tape
 - » Still must be managed
 - » Varies between WORM (write-once, read-many-times) and RW (read-write)

Maria Hyönnelie, UGA

41

I/O Subsystem Management

- 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

Maria Hyönnelie, UGA

42

Protection and Security

- **Protection** – any mechanism for controlling access of processes or users to resources defined by the OS
- **Security** – defense of the system against internal and external attacks
 - » Huge range, including denial-of-service, worms, viruses, identity theft, theft of service
- Systems generally first distinguish among users, to determine who can do what
 - » User identities (user IDs, security IDs) include name and associated number, one per user
 - » User ID then associated with all files, processes of that user to determine access control
 - » Group identifier (group ID) allows set of users to be defined and controls managed, then also associated with each process, file
 - » Privilege escalation allows user to change to effective ID with more rights

Maria Hyömette, UGA

43

Computing Environments

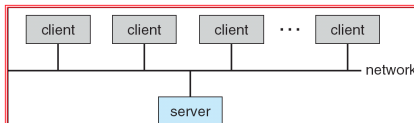
- **Traditional computer**
 - » Blurring over time
 - » Office environment
 - PCs connected to a network, terminals attached to mainframe or minicomputers providing batch and timesharing
 - Now portals allowing networked and remote systems access to same resources
 - » Home networks
 - Used to be single system, then modems
 - Now firewalled, networked

Maria Hyömette, UGA

44

Computing Environments (Cont.)

- **Client-Server Computing**
 - » Dumb terminals supplanted by smart PCs
 - » Many systems now servers, responding to requests generated by clients
 - Compute-server provides an interface to client to request services (i.e. database)
 - File-server provides interface for clients to store and retrieve files



Maria Hyömette, UGA

45

Peer-to-Peer Computing

- **Another model of distributed system**
- **P2P does not distinguish clients and servers**
 - » Instead all nodes are considered peers
 - » May each act as client, server or both
- **Node must join P2P network**
 - » Registers its service with central lookup service on network, or
 - » Broadcast request for service and respond to requests for service via discovery protocol
- **Examples include Napster and Gnutella**

Maria Hyömette, UGA

46

Web-Based Computing

- Web has become ubiquitous
- PCs most prevalent devices
- More devices becoming networked to allow web access
- New category of devices to manage web traffic among similar servers: **load balancers**
- Use of operating systems like Windows 95, client-side, have evolved into Linux and Windows XP, which can be clients and servers

Maria Hyömette, UGA

47

Summary

- **An Operating System (from here on OS) is a software (a program) that performs two functions:**
 - » it extends the “use” of the computer hardware and
 - » it manage the computer system resources



Maria Hyömette, UGA

48