31268_WEB SYSTEMS– LECTURE 1

Operating Systems – Part 1

What is an operating system?

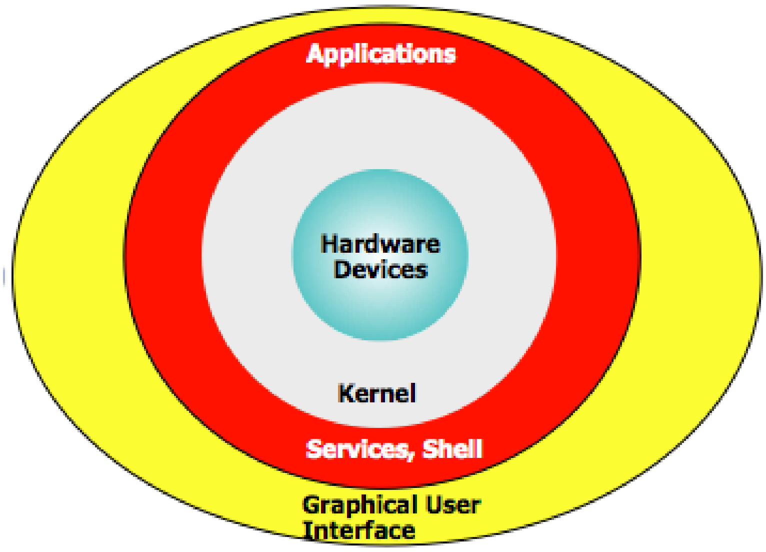
- A piece of software that sits between all programs and the computer's hardware
- Manages computer
- Runs programs
- Interface between user and hardware
- Provides services to programs and users
- Protects users and programs from each other

Common Operating Systems

- Large systems: Mainframes pioneered operating systems since the 1960s; IBM still run mainframes; Handles 1000s of users (IBM mainframes also run Linux); Supercomputers now tend to run Linux
- Minicomputers: OpenVMS; Unix and Unix-like operating systems (such as Linux, BSD, Solaris, Mac OSX)
- Personal computers: Linux, Microsoft Windows, Mac OSX (which is acutally Unix!)
- Embedded systems: Military, telecommunications (suh as, QNZ, Windows CE, VxWorks)

Operating System Structure: Unix "onion" model"

- Hardware
  - CPU – central processing unit
  - Memory
  - Input/Output – Mouse, keyboard, display, printer
  - Storage – Flash, hard drive DVD
- Kernel
  - Controls the hardware directly – device drivers, firmware
  - Provides resources and services to applications – e.g. CPU, memory, storage, video, mouse, memory
  - Manages access to privileged resources
- Applications
  - Programs to do "something" for the user



- Services
  - Service are programs that run "behind the scenes"
  - Usually provides system support – e.g. security, networking
- Shell
  - Also known as Command Line Interface, Terminal, etc.
  - A program that makes a set of commands available to the user
- GUI
  - A user-friendly interface on top of the operating system

Where does the OS fit in?

- An OS sits between the users and their programs on one side, and the computer and hardware on the other
- User → User program → Operating systems → Hardware devices

Operating Systems – User Interfaces

Psychology for User Interfaces

- Cognitive scientists analyse how people think
- Designing a user interface is hard work, and rarely done right the first time

Computer Interfaces

- Command Line Interface
    - Interact through the keyboard and a monitor which only prints text
    - E.g. sh 1969, CPM, cmd.exe

Strengths	Weaknesses
- Greater flexibility	- Hard to learn → cryptic commands and parameters
- Find turning → parameters	- Multiple options → more than 1 way to do things
- Essential for system administration	- Output often cryptic or non-existent
- Faster, less overhead	- Inconsistent commands → diff. versions
- Runs on simple hardware	- No graphics
- Can run remotely	
- Robust – difficult to crash	

- Graphical User Interface
    - Interact via windows, icons, menu, pointer device
    - E.g. Apple – Mac OSX, Microsoft Windows, Unix – Gnome, KDE

Strengths	Weaknesses
- Little experience required	- Can't do everything
- Good for graphics → artwork, desktop publishing	- Can crash the system
- User friendly	- User is unsure of what the OS is really doing
- Hides complexity from users	- Slows computer down
	- Needs better hardware
	- Hides complexity from users

-

Operating Systems – UNIX

Where is UNIX used?

- Used on most of the computers running the Internet
  - Web servers, domain name servers, email servers, web hosting
  - Unix isn't popular for ordinary users, however MAC OSX is passed on Unix
- Based from 2 original versions
  - System V – the original version from AT&T
  - BSD – From the uni of California

Popular current versions of Unix

Unix Verion	Description
Solaris	Sun's version of Unix (based on system V)
Linux	Free verion on Unix (based on system V and BSD)
Mac OS/X	Runs on the latest Macs (based on BSD
AIX	Version of Unix from IBM (based on system V)

Unix Irregularities

- Ad hoc Development
  - Quite a lot of Unix, especially the various scriptoing languages and the individual commands grew up in an ad-hoc and unregulated, haphazard fashion
  - While this resulted in a much more powerful and versatile operating system, it also results in being rather confusing at the user level

Why Unix has survived

- No one owns these ideas
  - Unix is a set of ideas, none of which are secret
  - Any person or group is free to implement these ideas.
- Unix is based on simple concepts: Files, processes, permissions and users
  - Even hardware devices e.g. dev/mouse are represented as files
  - This has simplified the conceptual picture of Unix (if not the internal code)
  - It has also allowed Unix to incorporate new ideas and technologies quite easily
- Unix is portable
  - Unix is written in the programming language C → Any computer with a C compiler can usually compile the source code
  - The technology of computer hardware has evolved enormously since 1970, but it is still comceptually the same
- Some Unix varieties are free
- Unix is efficient, stable and relatively secure
  - Fast and stable (system crashes are rare)
  - Designed for security for multi-user systems – files have owners, security permissions are tight → fewer viruses
- Unix as a set of tools approach
  - The Unix CLI has some very powerful features. Specifically simple commands, pipes and I/O redirection
  - You can create vary powerful ad hoc tools → by passing the output of one command to another command

Operating Systems – File System

File systems and file manipulation

- A file systemis a part of the OS that manages data storage and access
- There are many components of a file system: Disk physical structure, disk logical structure, file allocationg methods, file management

Disk Physical Structure:

- A physical disk is organised into:
  - Tracks: Concentric rings on the platter
  - Heads: Reads data from a platter
  - Cylinders: Collection of all tracks on platters (which are horizontally in the same position)
  - Sectors: Part of a track for data

Disk Structure

- A disk is a stack of magnetic platters
  - This stack is divided into cylinders
  - Each cylinder contains circular tracks (which are in turn, divided into sectors)
- Read/write operations are provided by the disk
  - These move concurrently along the fixed disk arm
- The disk itself rotates with constant angular velocity to provide access to every sector

Disk Formatting

- Formatting is the operation which creates the physical disk structure
- Formatting is organising and marking the surface of a disk into tracks, sectors, and cylinders
- It is also sometimes incorrectly a term used to signify the action of writing a filesystem to a disk

Disk Logical structures

- Partitions: Disks can be subdivided into partitions – each is an independent storing device
- Blocks: The OS views all the disk space as an array of fixed size logical blocks (a logical block is the smallest unit of data to transfer

File allocation methods

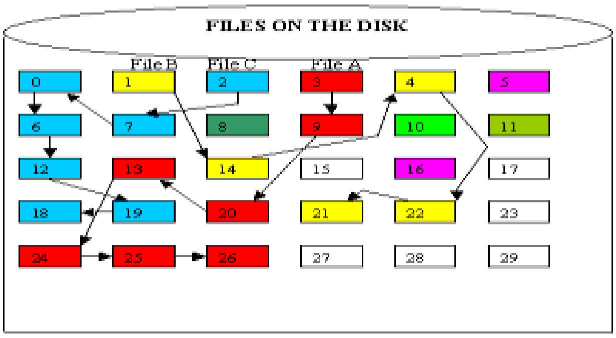
- Block: Space is allocated to a file as one or more blocks
- Directory: A table of information that the OS uses to locate blocks associated with files on a disk
- There are three types of file allocation: Contiguous, chained/linked, indexed

Contiguous allocation

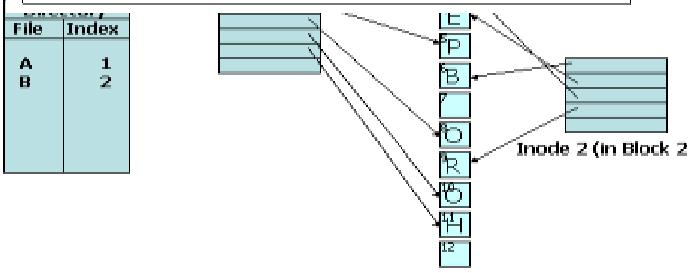
- A single contiguous set of blocks is allocated to a file at the time of file creation
- Supports random access: You know exactly where every block is after the starting block
- Fragmentation of unused space (external fragmentation) will occur, needs compaction
- Often used in magnetic tapes rather than disks

Chained allocation

- File is written as a collection of non-contiguous blocks
- File is implemented as a linked list of blocks
- Each block contains a (pointer to) the address of the next block
  - Last block contains invalid (negative) number (acts as an end-of-file marker)
- Directory entry contains the head (starting) block number and length of the file
- Chained is good for sequential access, bad for random access



Indexed allocation



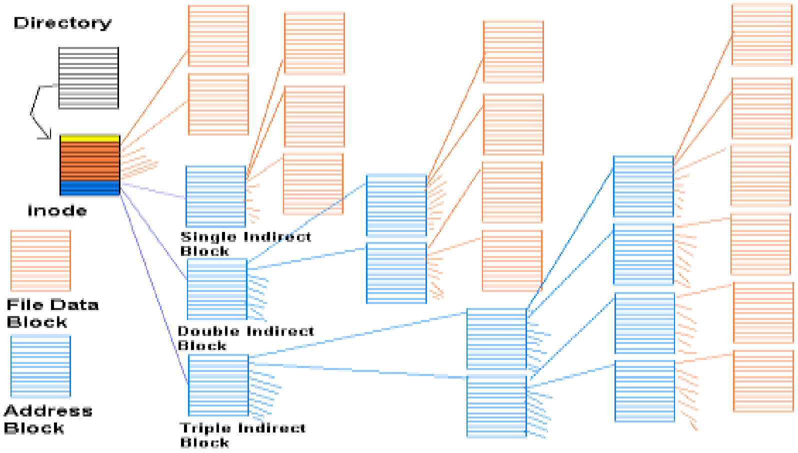
- File is stored as a data structure, called an inNode
- Each file/directory is referenced by an iNode
- Very efficient use of space and fast to read blocks
- iNodes and directories in Unix

iNodes and Directories in Unix

- The iNode system is used in Unix, where all iNodes are numbered
- In addition to Inodes, there are specialized blocks on a file known as directories. Directories contain the names of files and the iNode number for the file
- Notice that the iNode structure is a tree
- An iNode contains all the relevant data about a file. It also contains the locations (addreses) of the first 10 or so blocks of data.

What does an iNode store?

- Length of the file in bytes
- Device ID (which device contains the file)
- User ID (the owner of the file)
- Group ID (the group the file belongs to)
- File mode (file permissions)
- Timestamps (ctime, mtime and atime)
- Reference count (how many hard links point to it)
- Pointers to the disk blocks that store the files contents



File management: Files and directories

- A file manager performs functions related to storage and file management
  - Displays a list of files on a disk
  - Displays the amount of used or free space on a disk
  - Organising, copying, renaming, deleting, moving, sorting files
  - Creating/deleting directories/folders