Operating Systems Fundamentals

Classifying Operating Systems

Introduction

- Many different types of Operating Systems exist

- Different types have different focuses
  - This impacts the amount of attention given to the basic OS components already detailed

- Operating systems can be classified in a number of ways

Operating System Types

- By the **number of users that can access a machine at once**
  - Single-user; Multi-user
- By the **operational configuration**
  - Stand-alone (e.g. your personal PC/laptop);
  - Networked (e.g. All the PCs in the college);
  - Embedded (e.g. The microwave or … your phone!)
- By the **resource sharing policies**
  - Pre-emptive; Co-operative
    - This is further developed when looking at processes and memory management
- By the **types of hardware systems the OS will be run on**
  - E.g. number of processors

Single-User Operating System

- Designed to run with only one user on the machine at a time
  - DOS, Windows 95/98

- They will have some common features
  - Allowing software to run
  - Providing a disk file system
  - Handling hardware devices
  - Providing a user interface

Any operating system must do the above.

The following systems have extra features as well.

Multi-User Operating Systems

- Designed to allow multiple users perform operations on the same physical computer at the same time
  - Each user should be unaware of the other users presence
    - Security and Protection. Is this required for a single user system?
- Builds on the functionality of a single-user operating system
  - At a basic level, it will provide the single-user functionality to multiple users at the same time

Multi-User Operating Systems, continued

- Additional operating system functionality is required to support multiple users correctly
  - User access restrictions (e.g. Usr1 cant see Usr2's files/dirs)
  - Process isolation (e.g. Usr1 processes run in separate memory area than Usr2)

- Most modern operating systems are multi-user operating systems
  - Unix/Linux
  - Windows NT/2000/XP/Vista/7/8/10
  - Mac OS (which is actually based on ….. BSD Unix)

- Should allow controlled access to resources.

Operational Configuration Modes

- Operating Systems can also be classified on the configuration that they are used in:

1. Standalone

  - The computer can function and operate correctly with just the Operating System installed

Operational Configuration Modes, cont'd

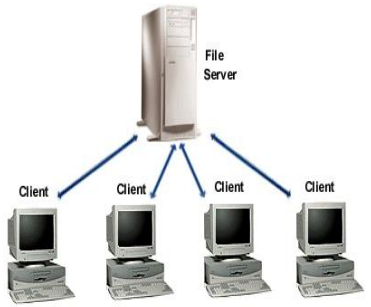
2. Networked

- External / Networked systems are required for correct operation
- External file storage and drives used to store information

- Designed to access and use network resources with ease
  - Unix, Windows NT-based OS's

- Extra features include:
  - Ability to access files, printers and other resources across the Network
  - Network logons supported, allowing a user to access the computer from anywhere

- Common features in modern Operating Systems



Operational Configuration Modes, cont'd

3. Embedded

- Approximately 1% of processors are in PCs
  - Embedded Devices constitute the bulk of the processors in use
    - Mobile Phones, Cameras, Cars, Washing machines, etc
  - Embedded OSs run on constrained devices
    - Note phones are no longer constrained much – but they used to be!
    - Think back to your Nokia candy bar phone
  - Traditionally only required to perform a few operations
    - Phones have changed that but a huge proportion of embedded OS's still only have to do simple tasks (e.g. microwaves/washing machines/building heating&lighting controllers)

- RT OS's (used on subset of embedded devices) tend to use less memory, provide good response to Real Time signals, and don't req disks, keyboards and screens

Real Time OS designed to meet time constraints against functionality (e.g. airbags/ABS)

Resource Sharing Policy

- Not all Operating Systems share resources fairly
- OS functionality concerned with managing access from user applications to the underlying hardware
- How an OS performs this management can be used to help classify the OS

Resource Sharing Policy, cont'd

1. Pre-emptive OS (think… jump the Q in Disney)

- The OS can
  - stop any running process,
  - save its ***state*** and
  - then run other applications as required

- Each running process has sole use of available computer resources during its time window

- The process is ***pre-empted*** either by its time window expiring or an external event occurring that needs to be processed
  - Network Communications
  - Interrupt

Resource Sharing Policy, cont'd

2. Co-operative OS

- Each running process cedes the available resources to each other
- Requires a high level of control between applications to ensure correct OS operations
- Badly written applications could lead to the system 'hanging'

Types of Computer Systems – number of processors

- Based on the number of processors that the OS will have to support

  - **Single Processor OS**
    - One main CPU/core
    - Older PCs and simple embedded systems fall into this category

  - **Multiprocessor OS**
    - >=2 processors/cores in one system
    - Modern PC's, Solaris, Windows XP\Vista\7\8\8.1\10, MAC OS X, Linux
    - All modern smartphones

  - **Cluster OS**
    - >=2 individual systems coupled together
    - Individual systems connected by a LAN
    - Exchange data with each other and typically have a single storage solution shared by the systems

Summary

- Different means to classify Operating Systems have been presented

- These include
  - The number of users of the computer
  - The environment it operates in
  - The manner in which resources are shared and managed
  - The types of hardware systems the OS will be run on

Question

- I run a process (that uses all the CPU resources i.e. 100%) on our Lab PC's
- When I run 2 of these, I expect both processes to get 50% (i.e. if I run a tool that tells me what cpu % each process is using, I should see they both get 50%)

- But….
  - When I run the tool
  - Both process say they have 100% CPU
    - WHY???