Outline: Operating Systems

- What is an OS
- OS Functions
  - Multitasking
  - Virtual Memory
  - File Systems
  - Window systems
- PC Operating System Wars: Windows vs. Linux

1

Operating System provides

- a way to boot (start) the computer
- control of the computer hardware: keyboard, display, mouse, printer
- a file system, a way to name and organize files for storage on disk, hence Disk Operating System (DOS)

2

Operating System provides

- a way to load and run user (application) programs
- a way for application programs to use the hardware devices and file system

3

Operating System provides

- a "User Environment"
- a consistent way for application programs to interact with the user
- Much of the UI is provided by Windows, not by individual programs
- Windows have a Graphical User Interface (GUI Interface)

4

Operating System provides

- multiple users - several people may use the computer at one time
- security - to prevent individual programs and users from interfering with each other
- Examples: UNIX Linux Windows NT with MetaFrame

5

Operating System provides

- Ease of use: presents three useful illusions
  - Many separate computers, one for each process
  - Large memory
  - Disks and other secondary storage are organized as collections of files

6

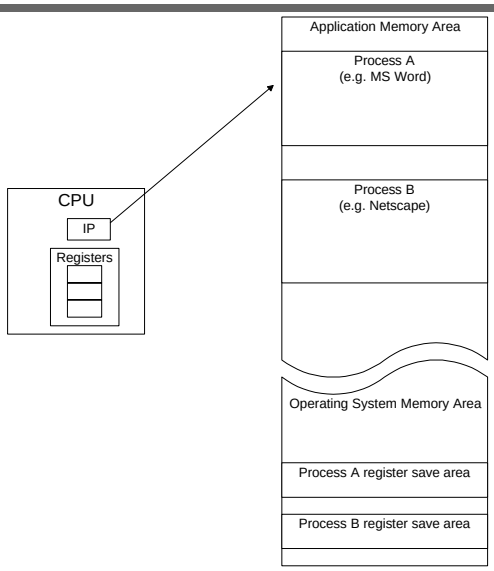
Illusion #1: Multitasking

- **Reality:**
  - One CPU
  - One instruction at a time
- **Illusion:**
  - Several application programs executing concurrently
- **Implementation:**
  - Operating system divides CPU time among application programs (time sharing)
    - each program "thinks" it is the only one running

7

Task Switching Illustrated

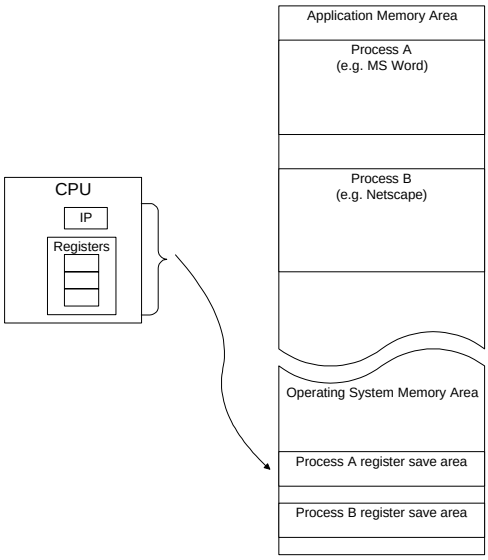
- Initially, the CPU executes MS Word
  - IP points to instruction in MS Word code
  - CPU registers contain data related to MS Word



8

Task Switching Illustrated

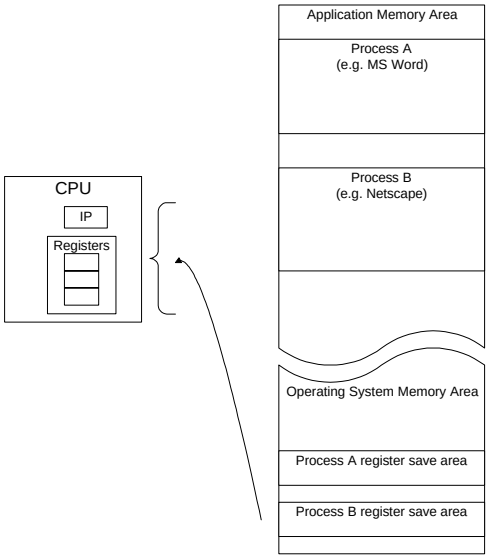
- After a short period of time (a few milliseconds)
  - CPU is interrupted by operating system
  - IP and other CPU registers related to MS Word are saved to a special memory area reserved by the OS



9

Task Switching Illustrated

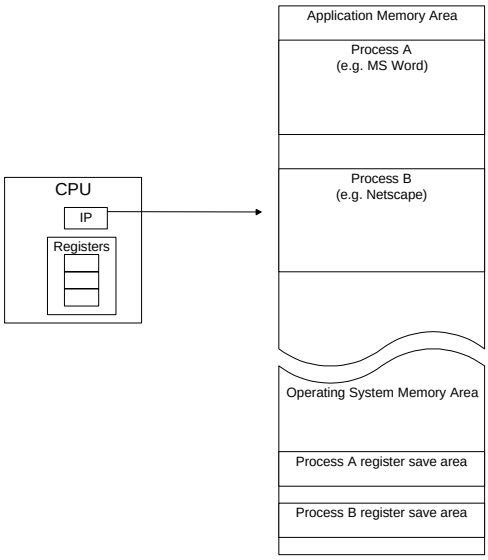
- After a short period of time (a few milliseconds)
  - CPU is interrupted by operating system
  - IP and other CPU registers related to MS Word are saved to a special memory area reserved by the OS
  - IP and other CPU registers related to Netscape are loaded from special memory area



10

Task Switching Illustrated

- This makes the CPU start executing instructions from the Netscape code



11

Cooperative Multitasking

- Used in early versions of Windows
- Application explicitly passes control back to OS
  - A badly behaved application may never pass control back. What happens?

12

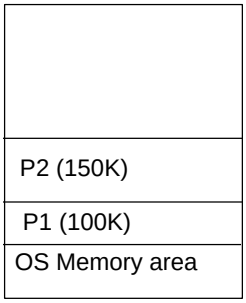
Preemptive Multitasking

- OS/2, UNIX, NT, DOS applications running under Windows
- OS preempts when it wants to take control
  - When application asks for I/O
    - Any I/O request requires a branch to the OS
    - The I/O request won't be handled right away anyway (I/O devices are much slower)
    - So OS initiates I/O request, then saves application state, and gives a time slice to another application
  - Or when application has run longer than a preset limit
    - A special timer interrupt signal causes CPU to branch to an interrupt handler program, which is part of the OS

13

Illusion #2: Memory Management

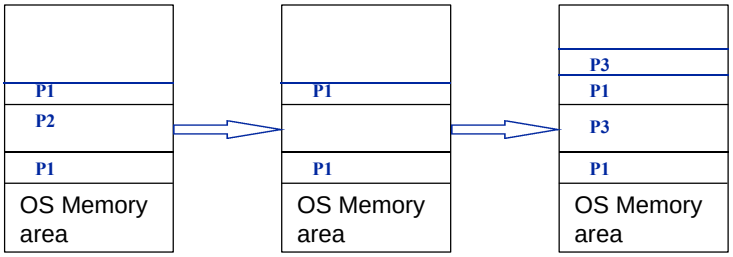
- Assign memory to each process
- Some OS provide memory protection
  - Don't let a process access memory not assigned to it



14

Non-contiguous Memory

- Example: Processes 1, 2, and 3
  - Process 1 starts with 100K of memory
  - Process 2 starts with 150K of memory
  - Process 1 asks for 50K more of memory
  - Process 2 exits
  - Process 3 starts with 200K of memory



15

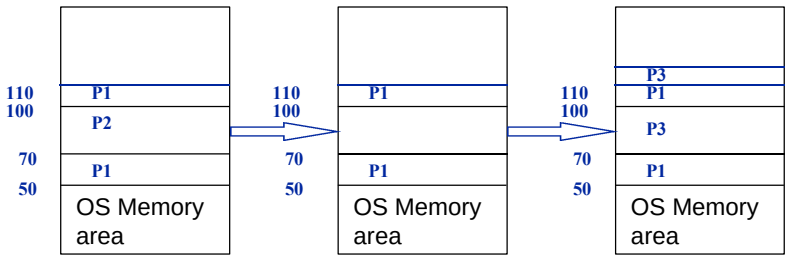
Issues

- OS loads applications in dynamically determined, non-contiguous pieces of memory
- Program code "thinks" it is stored in a fixed, contiguous memory area
  - for example, branch instructions refer to fixed memory addresses
- How can the system reconcile the two views?

16

How to Do it: Paging

- Divide all of main memory into frames
  - Each frame holds 1 page (say 1000 bytes)
- Divide all program code and data into pages
  - page size = frame size
- Place each program page into next available memory frame
  - Process 1 gets frames 51-70
  - Process 2 gets frames 71-100
  - Process 1 gets frames 101-110
  - Process 2 frees 71-100



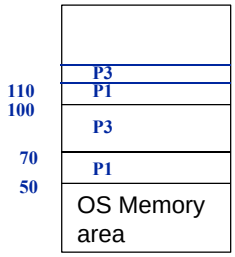
17

Address Translation

- "Virtual Address" consists of page number and location within the page
- Page may be located anywhere in real memory
- Lookup in page table
  - Translate page number to correct frame number
  - Location within the page is unchanged

Page #	1	2			30	31	32
Frame #	71	72			100	111	112

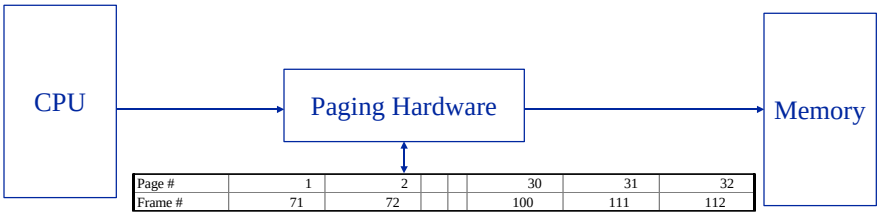
Page table for P3



18

Address Translation Example

- Use decimal numbers for simplicity
- Assume page size = frame size = 1000 bytes
- CPU wants to access memory address 30,127
  - page number = 30, location within page = 127
- Address is intercepted by paging hardware
  - page table indicates that page 30 is stored in frame 100
- Actual address sent to memory hardware is 100,127



19

Virtual Memory

- Some pages are not used for a long time
  - Why keep all of them in memory?
- Copy page to hard disk and use frame for some other page
- Copy pages back from hard disk to main memory frames as they're needed
- Total number of memory pages allocated to processes can exceed total number of memory frames
- Process (and its programmer) not aware that main memory is too small (the big memory illusion)
  - It asks for a main memory location (Page #, offset on page)
  - OS has to get that page into main memory if not already there
  - Called Virtual Memory

20

VM and Caches

- Both techniques exploit a small, fast memory and a big, slow memory
- Caches
  - Ask for address of location in slow memory
  - Check if that location is also in fast memory (cache)
- Virtual Memory
  - Ask for address of location in fast memory
  - Not enough fast memory for all the addresses
    - Some fast memory addresses are actually in slow memory
    - Copy needed page from slow memory to fast memory

21

Illusion #3: File Systems

- Reality:
  - Disks are sets of tracks
  - Tracks are sets of sectors
  - Sectors can store fixed-sized byte blocks
- Illusion:
  - Disks are sets of directories
  - Directories contain other directories or files
  - Files are variable-size byte sequences
  - Directories and files have names

22

The File System

- A DIRECTORY kept by the operating system
  - Keeps track of each file's name
  - Contains information about the file's physical location
  - May keep additional information such as date created, etc.

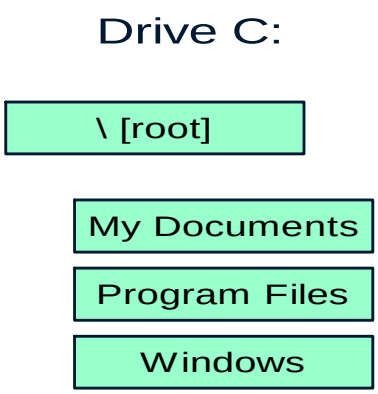
23

The File System

- Special files (subdirectories) are themselves directories
- Directories may contain subdirectories, nesting to multiple levels

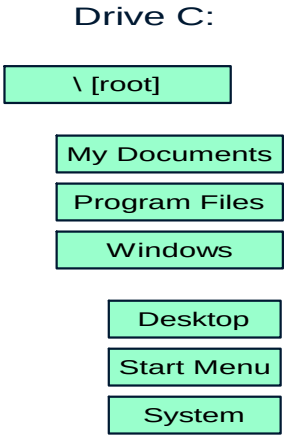
24

Hierarchical directories



25

Hierarchical directories



26

Windows file naming

- Four part names
  - Drive (or device)
  - Path
  - File name
  - File type (extended name)
- C:\Windows\System\WinTrust.hlp

27

C:\Windows\System\WinTrust.hlp

- C: = Drive (hard disk named C:)
- \Windows\System\ = Path
  - begin in "\" (root)
  - go to "Windows" subdirectory
  - go to "System" subdirectory of Windows
- WinTrust = File Name
- hlp = File Type (help file)

28

Windows file types

- Identify the kind of file
- For data files, identify the application that usually opens the file

- EXE - program (executable file)
- COM - command
- DOC - MS Word document
- PPT - PowerPoint presentation
- XLS - Excel spreadsheet
- HLP - Help file

29

File System Abstract Functions

- File operations
  - Set access privileges on file
  - Create or delete a file
  - Open or close
  - Seek to particular position in file
  - Read or write at current position
- Directory operations
  - Create or delete a directory
  - Move a file or directory inside another directory
- General operations
  - Search for a file with given name

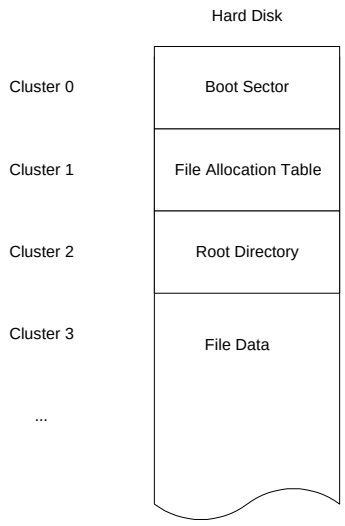
30

File Representation

- Combine disk sectors into clusters
  - 1 cluster ~ 10000 bytes
- Implement file as collection of clusters
  - Clusters need not be contiguous (near each other on the disk)
    - Why is this an advantage?
    - Why is this a disadvantage?
    - How big should the clusters be?
  - Keep a map of the clusters used in each file
  - Which is better: contiguous or fragmented?
- Keep a map of the free clusters

31

DOS Hard Disk Organization



32

Example: DOS Directories and FAT

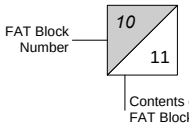
Root Directory

Filename	*Cluster*
file1.txt	2
file2.doc	6

File Allocation Table

0 0	*1* 0	*2* 3	*3* 4	*4* 5
5 12	*6* 7	*7* 8	*8* 9	*9* 10
10 11	*11* 0	*12* 16	*13* 14	*14* 15
15 0	*16* 0	*17* 0	*18* 0	*19* 0

Legend:



Block 10 of the File Allocation Table contains the value 11

33

OS Wars: Windows vs. Linux

- Linux is a type of Unix
- Free operating System
- Original version developed by Linus Torvalds in 1991
- Greatly enhanced and extended by the global "Open Source" software development community

34

Comparison of Windows NT and Linux

- Appearance
- Cost
- Applications
- Scalability
- Software Support
- Technical Support
- Compatibility
- Configurations

35

Appearance

- **NT**
  - **GUI**
- **Linux**
  - **Command Line**

36

Cost

- NT
  - 5 user version $809
  - 10 user version $1129
  - 25 user version $3,999
- Other costs
  - Office Suite, Compiler / Development Tools
  - Netware, Virus Protection, Backup packages, Database operations
- Linux
  - Free

37

Applications

- **Many more programs for NT than for Linux.**
- **Choice of vendors is much larger for NT.**
- **Many of Linux applications are not available in GUI**

38

Scalability

- NT
  - Supports 4GB of RAM
  - NTFS provides 64 bit file system
  - Integrated file cache
- Linux
  - Supports 2 GB of RAM
  - Maximum file size of 2GB

39

Software Support

- NT
  - Wide range of commercial software
  - Single user system
- Linux
  - Windowing protocol

40

Technical Support

- **NT**
  - **Dedicated support network**
  - **Support through partners and OEM's**
  - **350, 000 Microsoft trained professionals**
- **Linux**
  - **Free Online Support**
  - **"Peer to Peer" support**

41

Compatibility

- NT
  - Difficult to access UNIX file systems
  - No UNIX programs run under NT
  - NT Backups only run on NT
- Linux
  - Can access DOS/Windows/NT file systems
  - Compatible with any version of UNIX
  - Many window programs run under Linux
  - Backups are compatible between distributions of Linux, and UNIX but not with NT

42

Configurations

- NT
  - Admin user is not authorized to make all changes
  - Configuration
- Linux
  - Easily changed to meet needs
  - Interfaces
  - Wide range of free tools to configure system

43

Security

- NT
  - Access Permissions
  - Bound to machine and Domain
  - Have to check by hand for any file changes
- Linux
  - Security is tied to the file
  - Can easily tell if a file has been changed
  - Firewall functionality is built in to the server

44