Operating Systems Concepts

Introduction

INFO 2603
Platform Technologies

Week 2: 10-Sept-2018

Operating System (OS)

An Operating System is the layer of software that manages a computer's resources for its users and their applications.

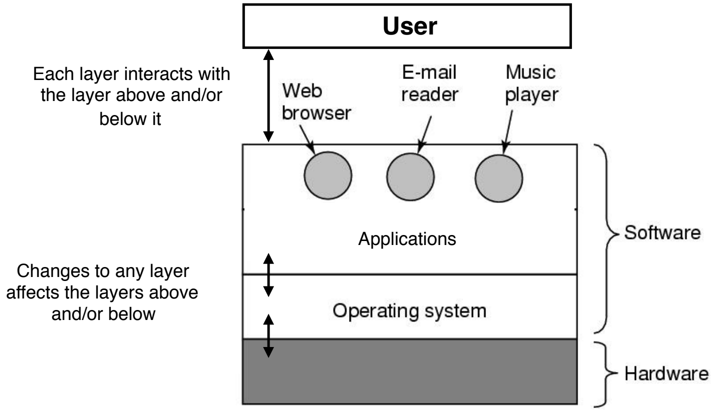
It abstracts the use of these resources and provides a standardised interface to simplify development and use of applications.

Operating System - Managing Resources

An OS manages system resources and typically deals with:

- CPU scheduling
- Process management
- Memory management
- Input/Output device management
- Storage device management (hard disks, CD/DVD drives, etc)
- File System Management

Operating System - Layers



Each layer interacts with the layer above and/or below it

Changes to any layer affects the layers above and/or below

Fig. 1 Operating System Relative to Hardware and Software Layers

Tanenbaum, Modern Operating Systems 3 e, (c) 2008 Prentice-Hall, Inc. All rights reserved.

Operating System Modes

- A typical operating system has two modes of operation.
  - User Mode
  - Kernel Mode

User Mode

- The User Mode is concerned with the actual interface between the user and the system.

- It controls things like running applications and accessing files.

Kernel Mode

- The Kernel Mode is concerned with everything running in the background.

- It controls things like:

  - accessing system resources

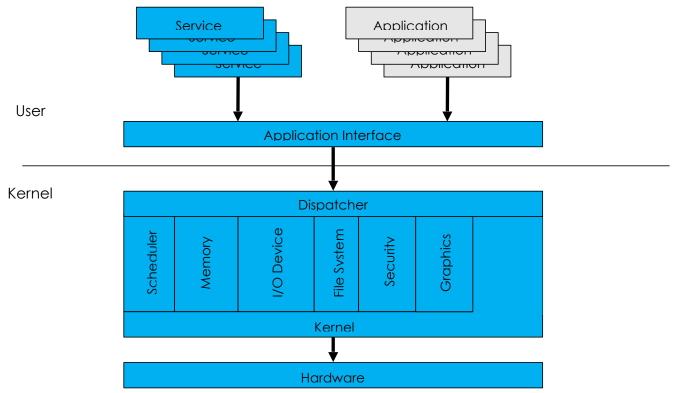
  - controlling hardware functions

  - processing program instructions.

- The Kernel forms the core of the operating system.

  - acts like a supervisor for everything that is happening in the computer.

Typical OS Architecture



Service

Application

User

Application Interface

Kernel

Dispatcher

Scheduler

Memory

I/O Device

File System

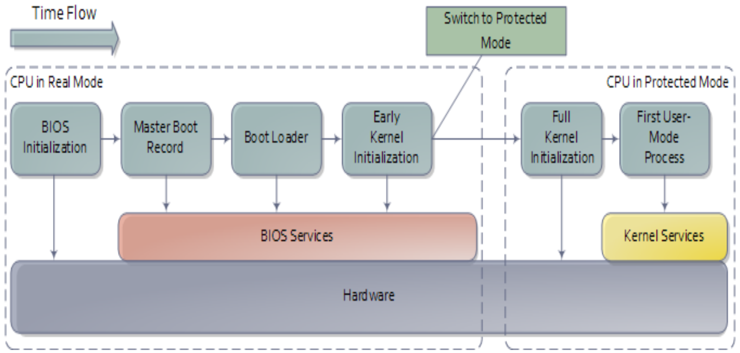
Security

Graphics

Kernel

Hardware

Starting an Operating System

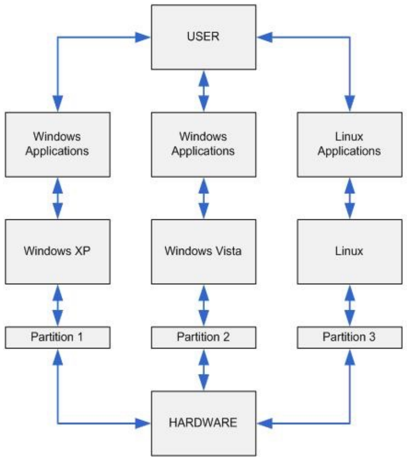


Types of Operating Systems

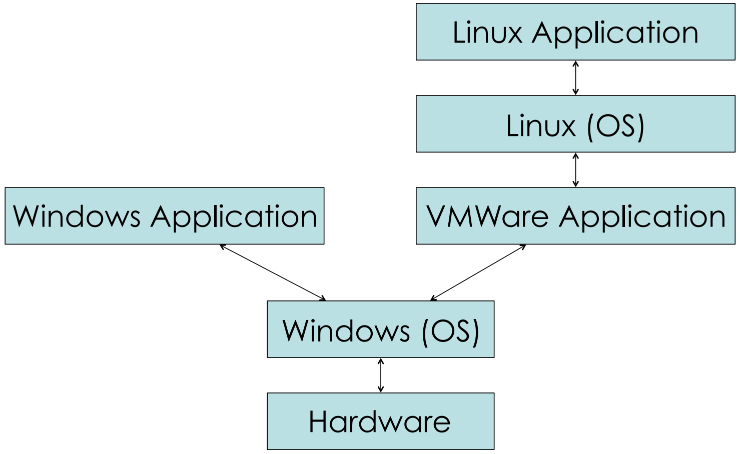
- Mainframe operating systems
- Server operating systems
- Multiprocessor operating systems
- Personal computer operating systems
- Handheld operating systems
- Embedded operating systems
- Sensor node operating systems
- Real-time operating systems
- Smart card operating systems

Operating System Examples

Running Multiple Operating Systems



Virtual Machine



Linux Application

Linux (OS)

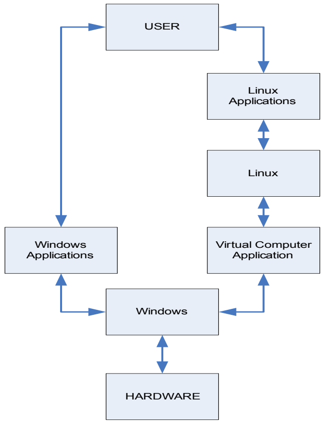
Windows Application

VMWare Application

Windows (OS)

Hardware

Running a Virtual Machine



USER

Linux Applications

Linux

Windows Applications

Virtual Computer Application

Windows

HARDWARE

Figure 2.3
Layers with a virtual operating system

User Interaction with the OS

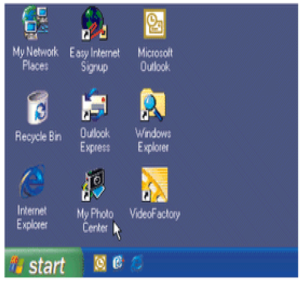
An operating system operates the functions of a computer. It also provides a way for users to interface with, or access, a computer's applications, resources and hardware.

There are two main types of user interfaces for an operating system:

1. Graphical user interface (GUI)

2. Command line interface

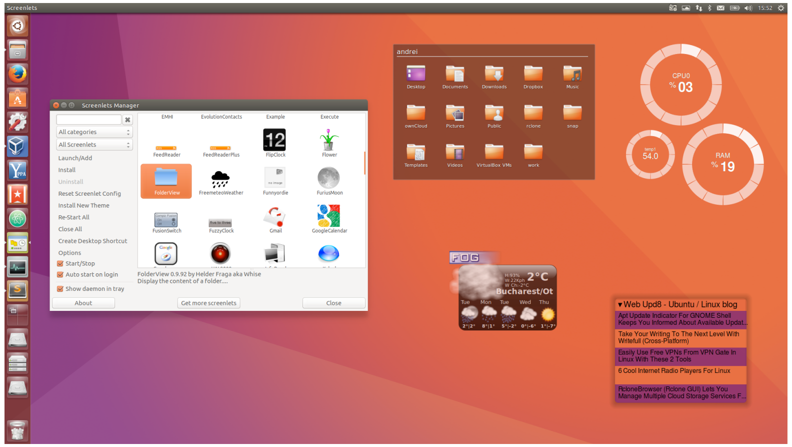
Graphical User Interface



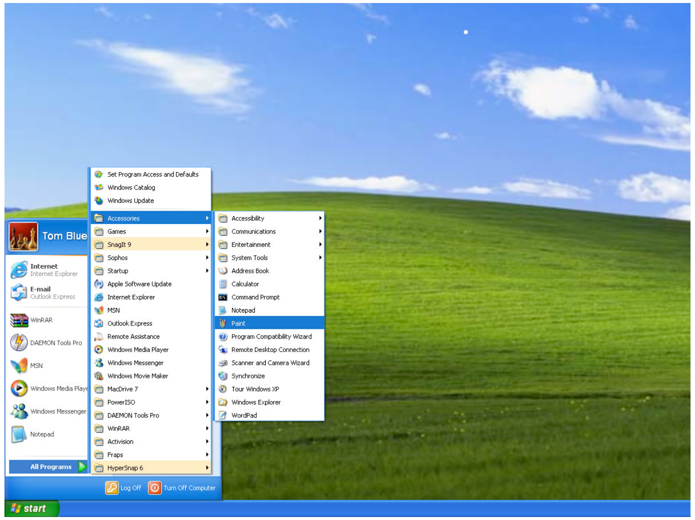
A graphical user interface features menus and icons that you can manipulate with the click of a mouse.

A GUI uses graphics (or pictures) and menus to help the user access resources and issue commands.

Windows XP, Linux and Mac OS X are examples of GUI operating systems.



Ubuntu Desktop



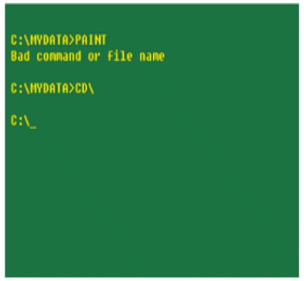
Windows Desktop

Command Line Interface

A command line interface uses typed commands to issue instructions to the computer.

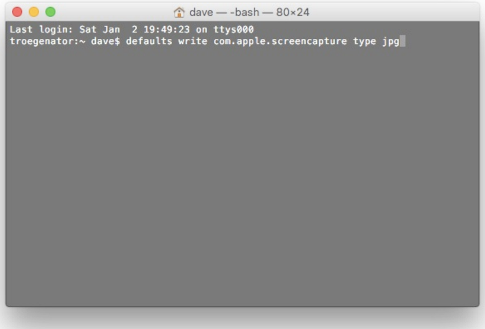
It can be more difficult to use because the user must type the precise commands and locations of files.

DOS (Disk Operating System) and UNIX are examples of command line interface operating systems.



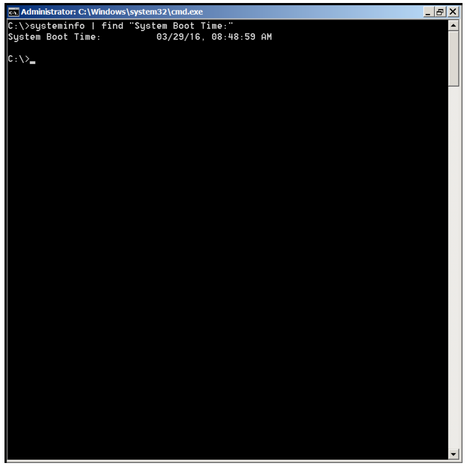
```
C:\MYDATA>PAINT
Bad command or file name

C:\MYDATA>CD\

C:\_
```

A command-line interface requires you to memorize and type commands.



```
Last login: Sat Jan  2 19:49:23 on ttys000
troegenator:~ dave$ defaults write com.apple.screencapture type jpg
```

MacOs Bash Shell



Administrator: C:\Windows\system32\cmd.exe

```
C:\>systeminfo | find "System Boot Time:"
System Boot Time:          03/29/16, 08:48:59 AM

C:\>_
```

Window Command Prompt

Command Line Interpreter

Applications are accessed at the User Mode level. This means that they do not have the authority to directly access system resources that are controlled at the Kernel Mode level.

When a user types a command (in a command line interface) or performs a task within an application (using a GUI), *processes* are initiated.

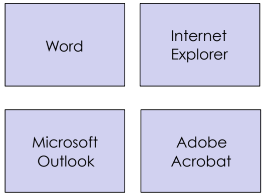
Since those processes usually require access to system resources, the command line interpreter converts them into system actions (called *system calls*). Most interpreters execute applications to perform the system calls.

Managing System Resources

- An operating system needs to manage a wide range of system resources:
  - CPU scheduling and process management
  - Memory (RAM)
  - Access to peripheral devices
  - File system management.

CPU Scheduling

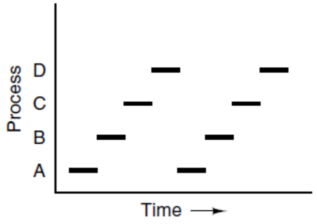
- Decide when applications should run; does the order matter?
- Are the applications active or sleeping?
- Multiple applications at the same time



Applications

Process

A process is an abstraction of a running program.
The CPU executes only one process at a time.



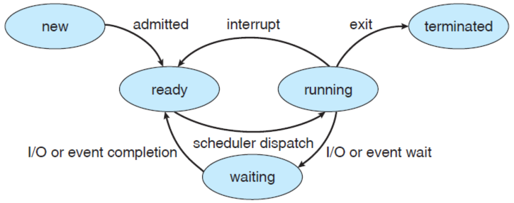
Process States

Fives states a process may be in:

1. **Running** (actually using the CPU at that instant).
2. **Ready** (runnable; temporarily stopped to let another process run).
3. **Blocked** (unable to run until some external event happens).
4. **New** (Process is being created).
5. **Terminated** (Process has finished execution)

Process States

Diagram of process state



The Process Table

- Keeping track of all of the processes is done with the Process Table.
- The Process Table lists:
  - Processes that are currently being run
  - Processes that are waiting to be executed
  - Processes that have been temporarily suspended
  - It also keeps track of the current status, or state, of each process.

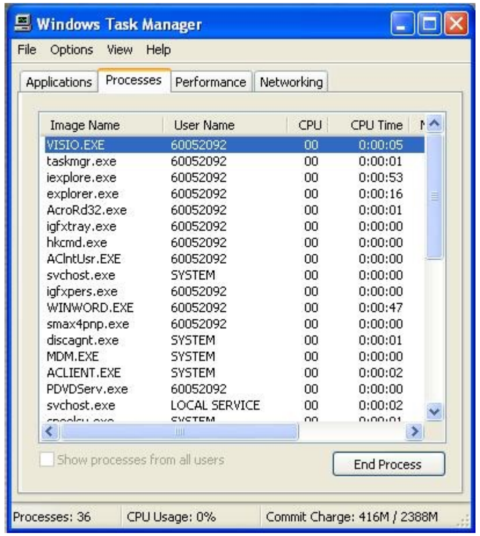


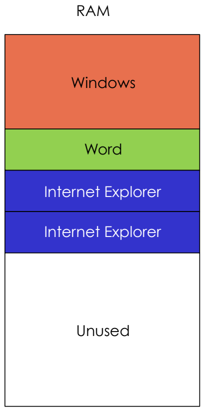
Image Name	User Name	CPU	CPU Time
VISIO.EXE	60052092	00	0:00:05
taskmgr.exe	60052092	00	0:00:01
iexplore.exe	60052092	00	0:00:53
explorer.exe	60052092	00	0:00:16
AcroRd32.exe	60052092	00	0:00:01
igfxtray.exe	60052092	00	0:00:00
hkcmd.exe	60052092	00	0:00:00
AClntUsr.EXE	60052092	00	0:00:00
svchost.exe	SYSTEM	00	0:00:00
igfxpers.exe	60052092	00	0:00:00
WINWORD.EXE	60052092	00	0:00:47
smax4pnp.exe	60052092	00	0:00:00
discagnt.exe	SYSTEM	00	0:00:01
MDM.EXE	SYSTEM	00	0:00:00
ACLIENT.EXE	SYSTEM	00	0:00:02
PDVDServ.exe	60052092	00	0:00:00
svchost.exe	LOCAL SERVICE	00	0:00:02
spoolsv.exe	SYSTEM	00	0:00:01

□ Show processes from all users

Processes: 36 CPU Usage: 0% Commit Charge: 416M / 2388M

Memory Management

- Give memory to each application as needed.
- Protect applications from each other
- Protect operating system from ‘bad’ applications


RAM

Windows

Word

Internet Explorer

Internet Explorer

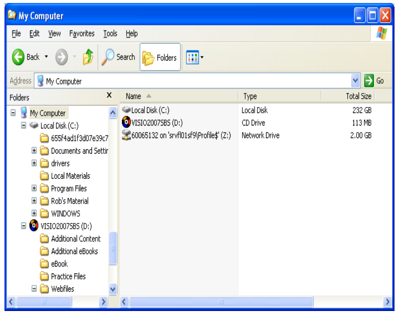
Unused

Peripherals

- Devices are very difficult to program
- OS Provides Drivers
- Applications speak to drivers
- Drivers communicate to device

File System

- Manage Information (files and folders)
- Provide common view of storage devices



My Computer
Local Disk (C:)
655f4ad1f3d07e39c7
Documents and Settir
drivers
Local Materials
Program Files
Rob's Material
WINDOWS
VISIO2007SBS (D:)
Additional Content
Additional eBooks
eBook
Practice Files
Webfiles

Name	Type	Total Size
Local Disk (C:)	Local Disk	232 GB
VISIO2007SBS (D:)	CD Drive	113 MB
60065132 on 'srvfl01sf9\Profile$' (Z:)	Network Drive	2.00 GB

References

- https://spectrum.ieee.org/semiconductors/design/bespoke-processors-a-new-path-to-cheap-chips
-