Information Communication Technologies

Lecture 5. Operating system software

Kassymova Aizhan Bakhytzhanovna
PhD, Associate professor
a.kassymova@satbayev.university

Agenda

1 Layers of Software

2 The BIOS: Life at the Bottom

3 Process Control

4 Device Management and Configuration

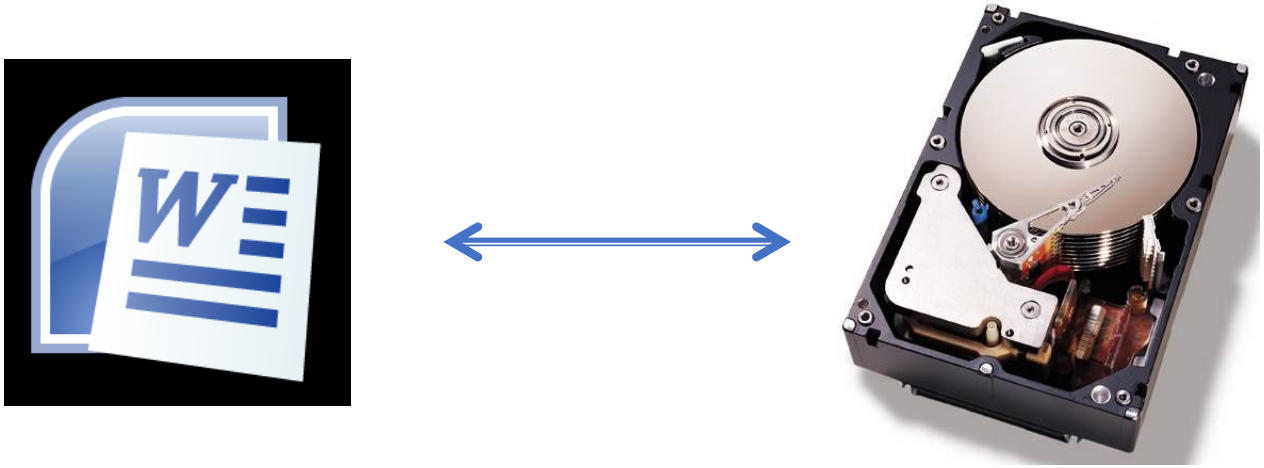
5 Interrupt Handling

Layers and Process Management

- Farmer does not know what the wheat will be used for.
- The bakery does not know how the wheat was harvested
- You do not even need to know that croutons contain wheat!

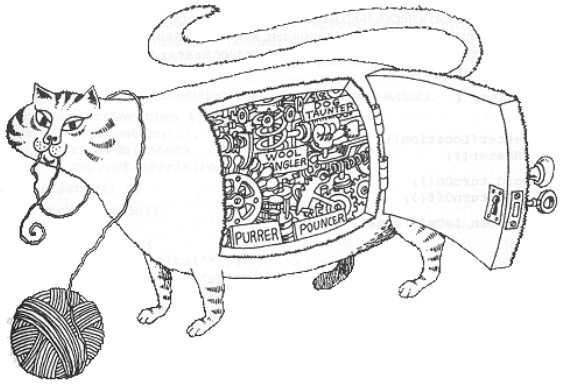
Layers and Process Management

- Your word processing program does not need to know how to control disk drives in order to be able to open and save files; there are layers of software below it that handle those details.

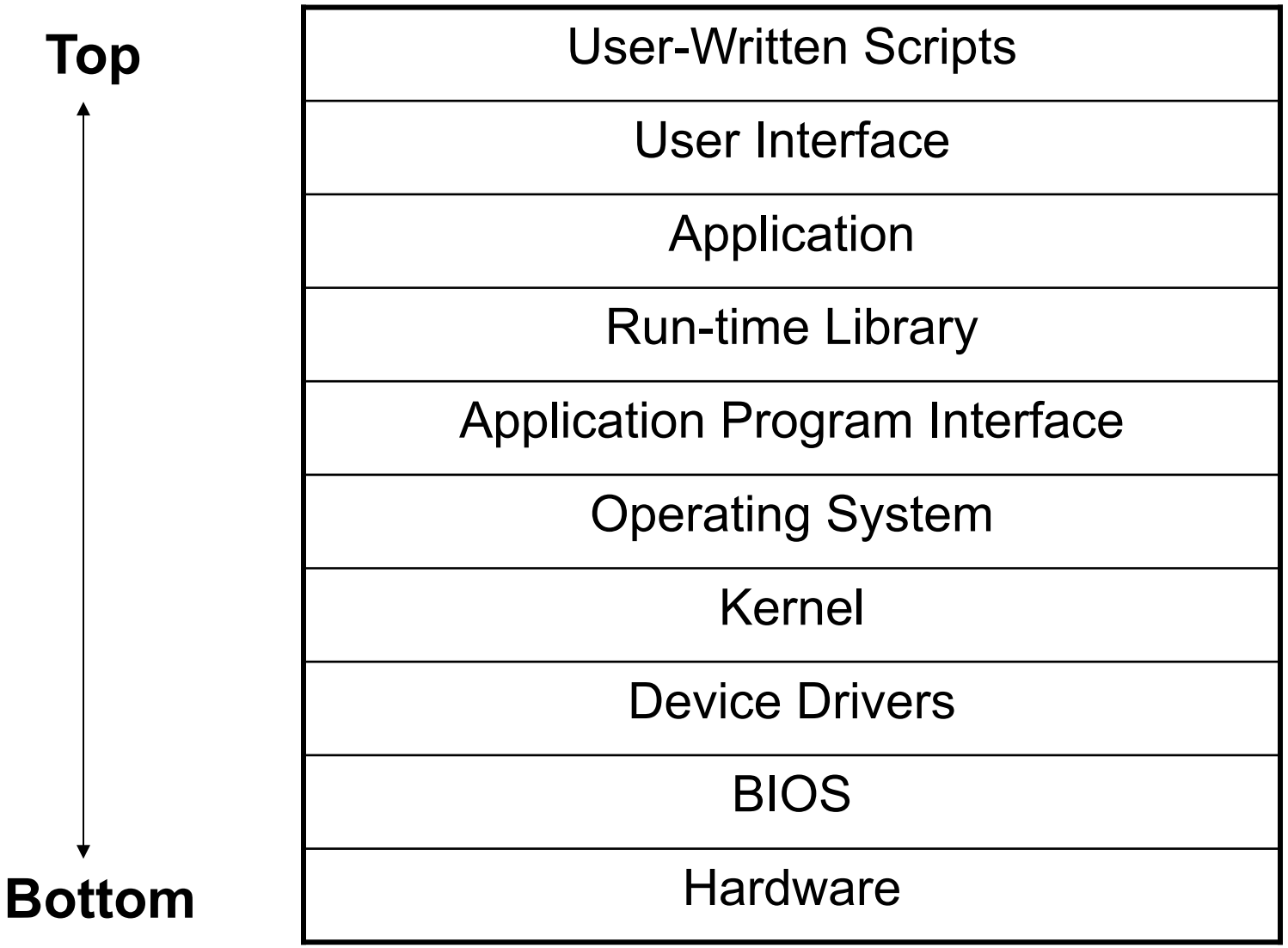


Encapsulation and Abstraction

- ***Encapsulation*** means that each layer needs only a limited amount of knowledge to do its job
  - Organizing software into layers that are relatively independent keep the entire system manageable, and afford greater efficiencies
- ***Abstraction*** means how the software layers communicate, beyond the view of the user



Layers of Software



Layers of Software (continued)

- ***Hardware***:
  - Lowest level of the computer
  - The physical components from which the computer system is constructed
- ***BIOS*** or ***Basic Input/Output System***:
  - Most primitive level of software
  - Deals directly with the signals that control each hardware component
  - Most of its work is done when the system first **boot**s up

Bootstrap program

1. Power up. When you turn on the power switch, the power light is illuminated, and power is distributed to the computer circuitry.

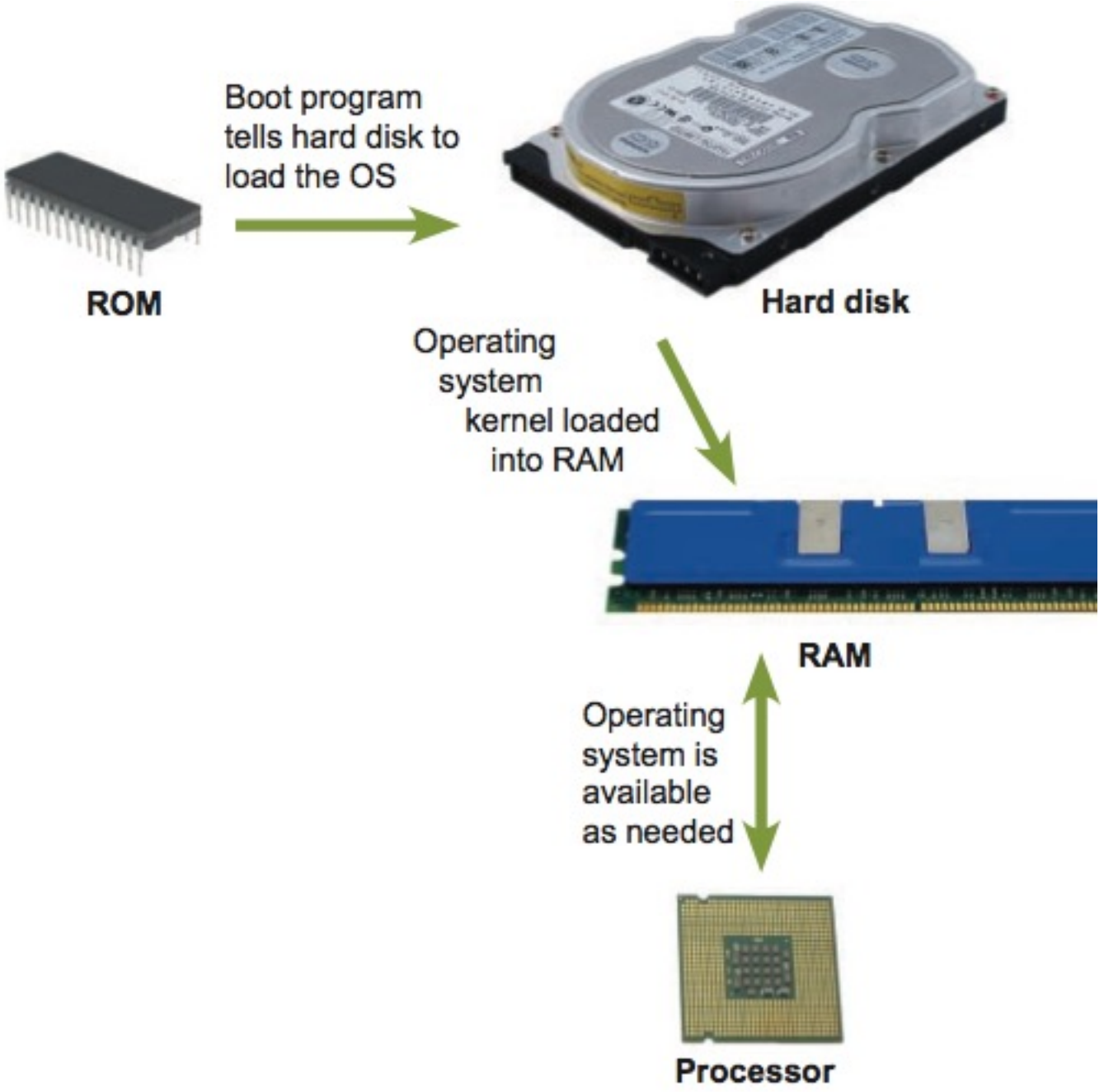
2. **Start boot program.** The microprocessor begins to execute the bootstrap program that is stored in ROM.

3. **Power-on self-test.** The computer performs diagnostic tests of several crucial system components.

4. **Identify peripheral devices.** The computer identifies the peripheral devices that are connected and checks their settings.

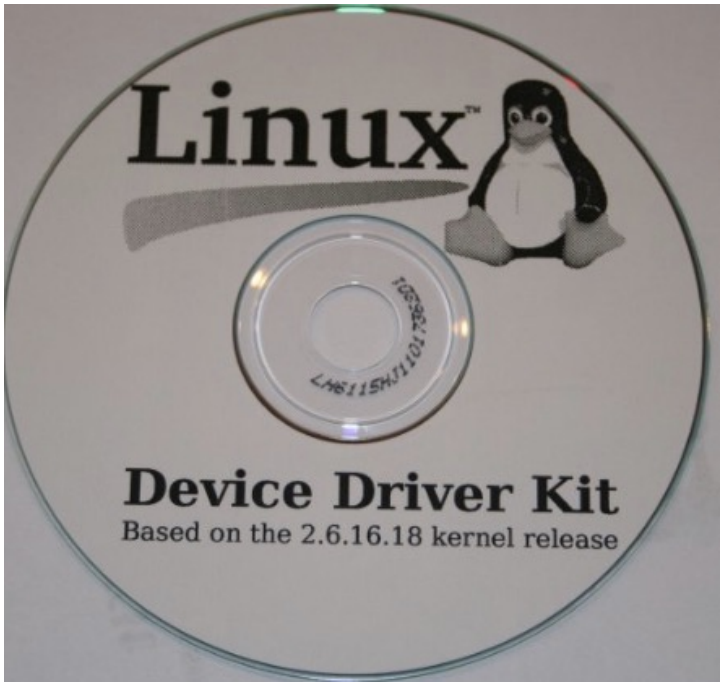
5. **Load operating system.** The operating system is copied from the hard disk to RAM.

6. Check configuration and customization. The microprocessor reads configuration data and executes any customized startup routines specified by the user.


Boot program tells hard disk to load the OS
ROM
Hard disk
Operating system kernel loaded into RAM
RAM
Operating system is available as needed
Processor

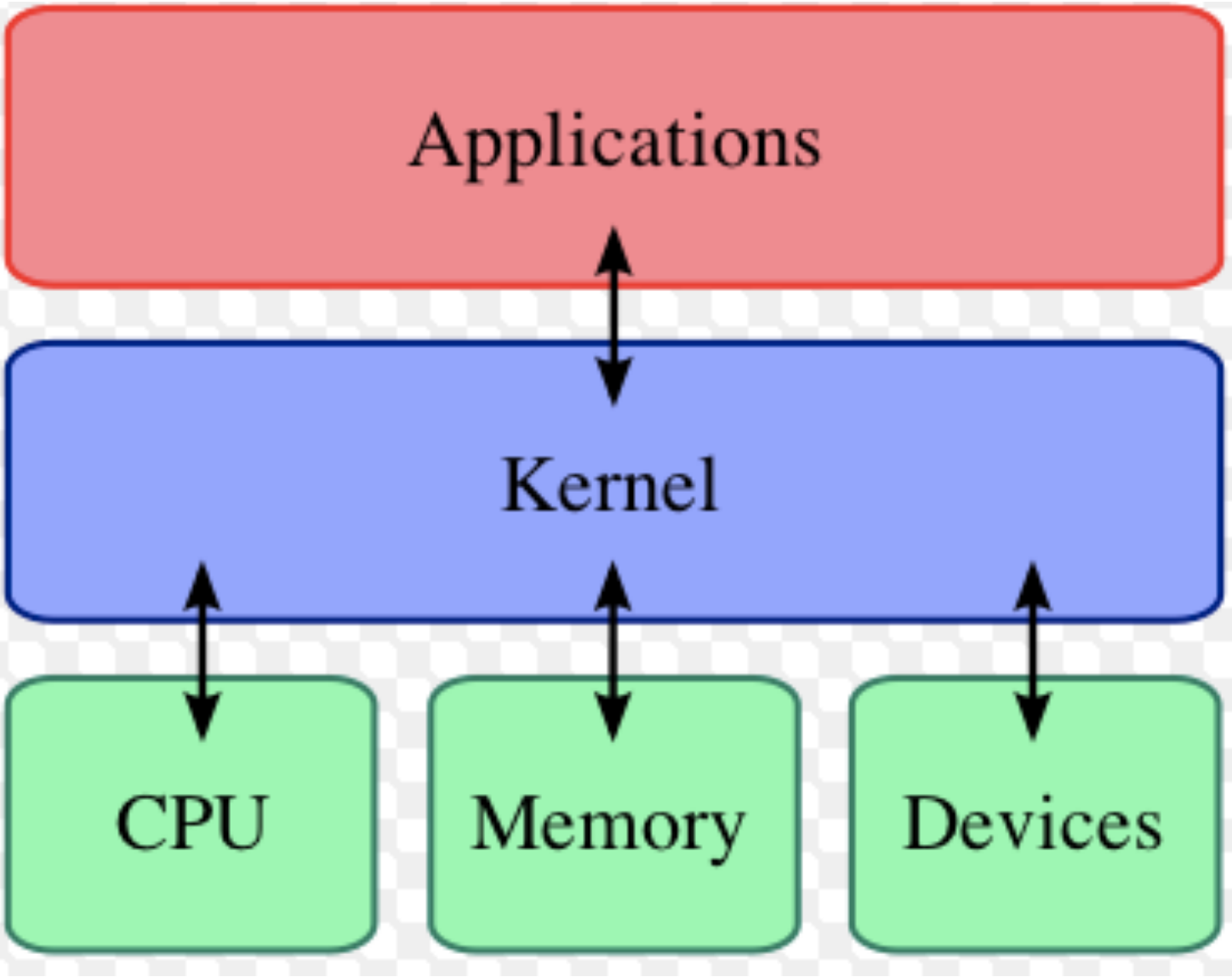
Layers of Software (continued)

- ***Device drivers***:
  - **Helper programs used by Operating systems to control a device**
  - In order to add new hardware, the appropriate device driver must be present.
  - Windows OS may sometimes have such drivers stored.



Layers of Software (continued)

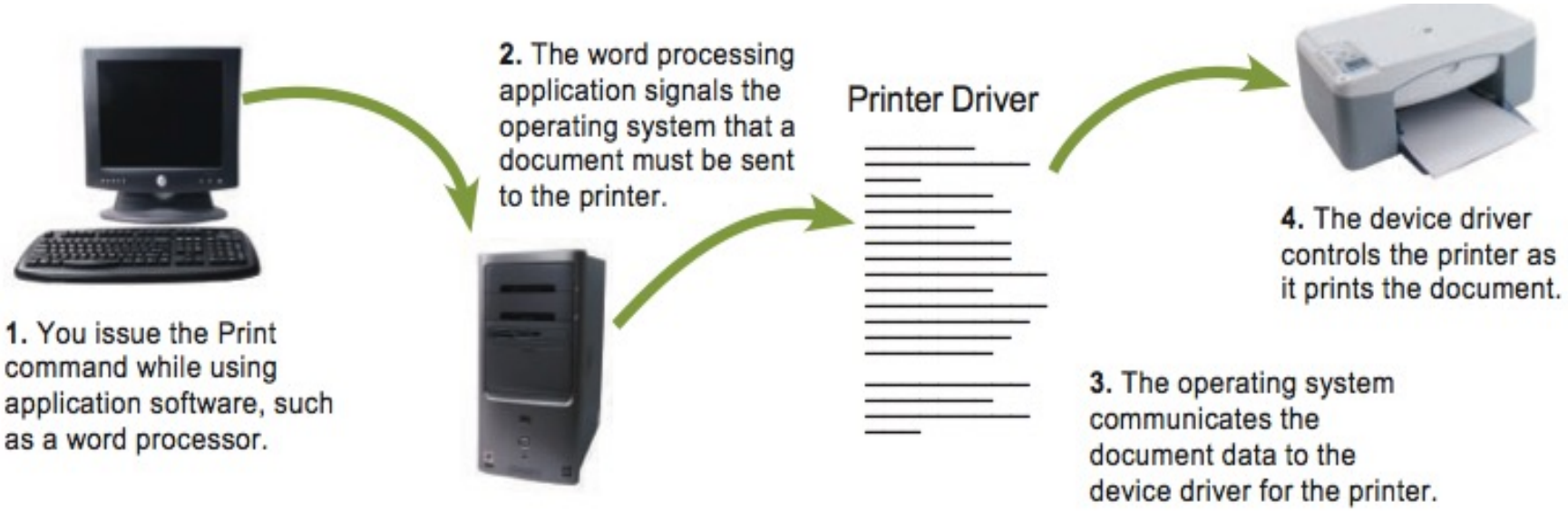
- ***Kernel***:
  - Performs functions that are critical to maintaining the operating system
  - The kernel is responsible for
    - **Memory management**
    - **Process and task management**
    - **Disk management (File System)**
    - **Interrupt Handling**
  - Resident in RAM at all times
  - Because space is a factor, kernel is kept small


Applications
Kernel
CPU
Memory
Devices

Layers of Software: Operating System

- ***Operating System (OS)***: is a type of system software that acts as the master controller for all activities that take place within a computer system.

A command to print a document is relayed through various levels of software, including the OS, until it reaches the printer.



Operating System Tasks

Manage processor resources

Manage memory

Keep track of storage resources

Ensure that input and output proceed in an orderly manner

Establish basic elements of the user interface

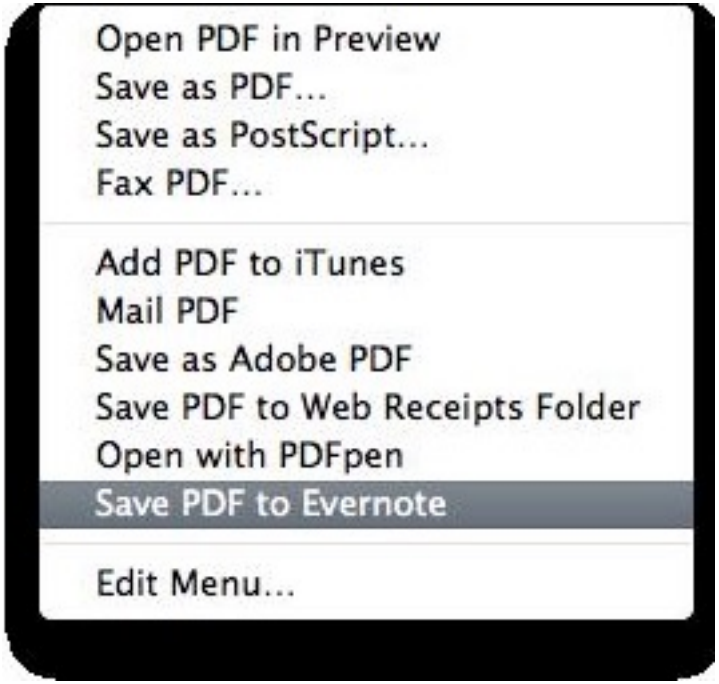
Layers of Software: Operating System (continued)

Different operating systems have different user interfaces and run compatible applications

- Examples: Microsoft Windows, UNIX, LINUX, Mac OS

Layers of Software: Operating System (continued)

- External Services of OS:
  - Help users start programs
  - Manage stored data
  - Maintain security of the system
  - Provides ways to select an application program, find, rename and delete documents and other data stored on disk



Layers of Software: Operating System (continued)

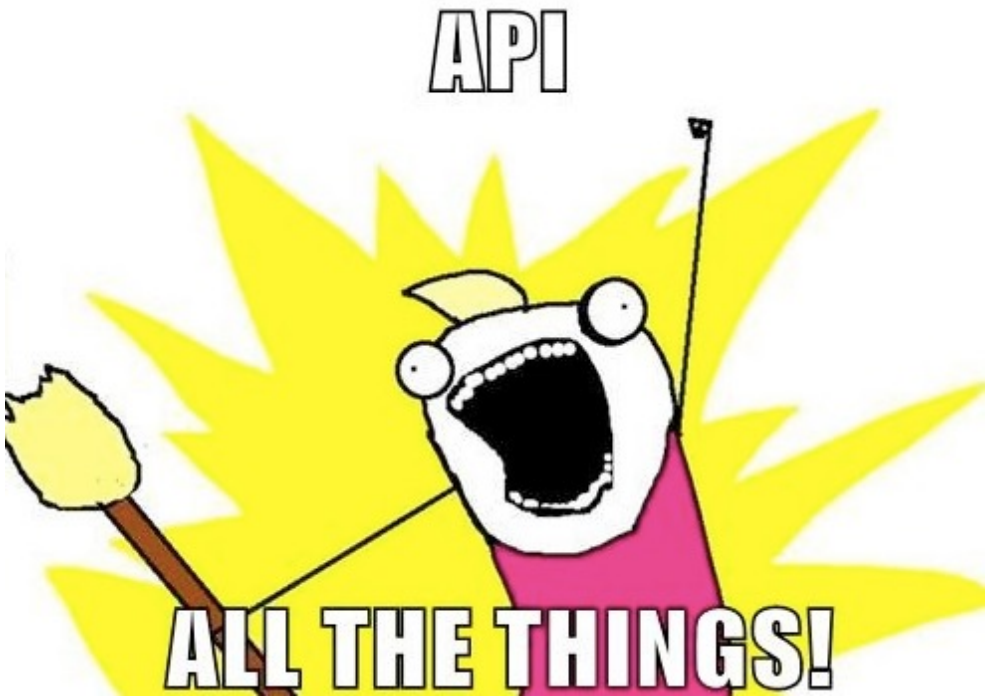
- Internal services of OS:
  - controls input and output
  - Allocates system resources (e.g. memory, disk drive capacity, processor time)
  - Manages storage for programs and data
  - Detects equipment failures

Layers of Software: Operating System (continued)

- Operating systems can be classified into two categories.
- A ***server operating system*** is designed for computers that provide centralized storage facilities and communications capabilities for networks and Web sites (Mac OS X Server, Windows Server 2008 R2, Linux)
- A ***desktop operating system*** is designed for a single-user microcomputer.
- **MacOS, Microsoft** – for personal computers
- **Linux , UNIX, Solaris** – for high-end workstations and servers

Layers of Software (continued)

- ***Application Program Interface (API)***:
  - A set of routines, protocols, and tools for building software applications
  - Application communicates with OS through the **API calls**.

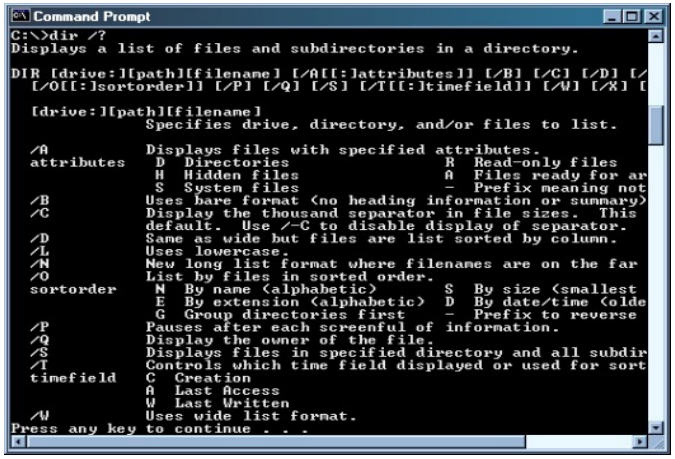


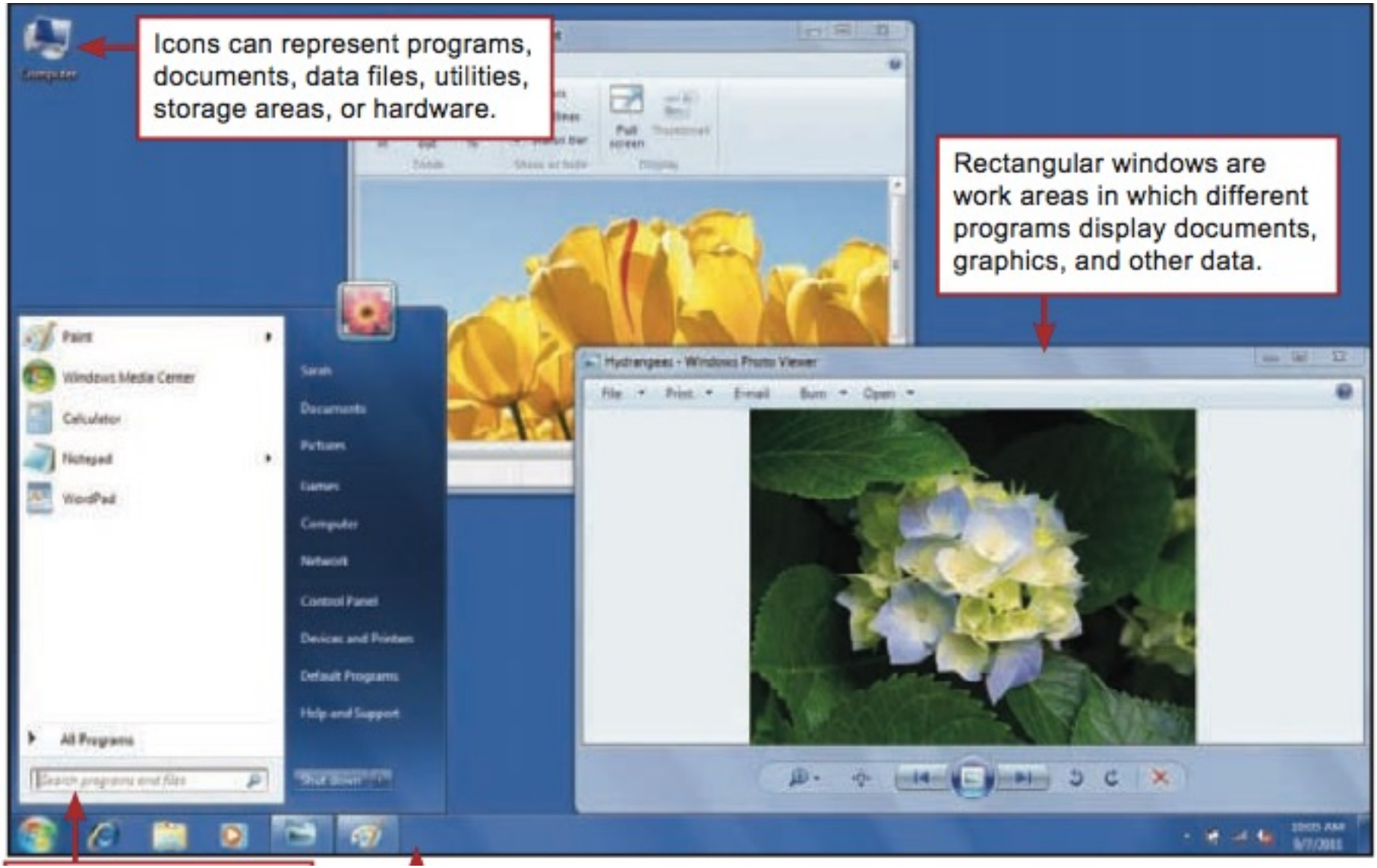
Layers of Software (continued)

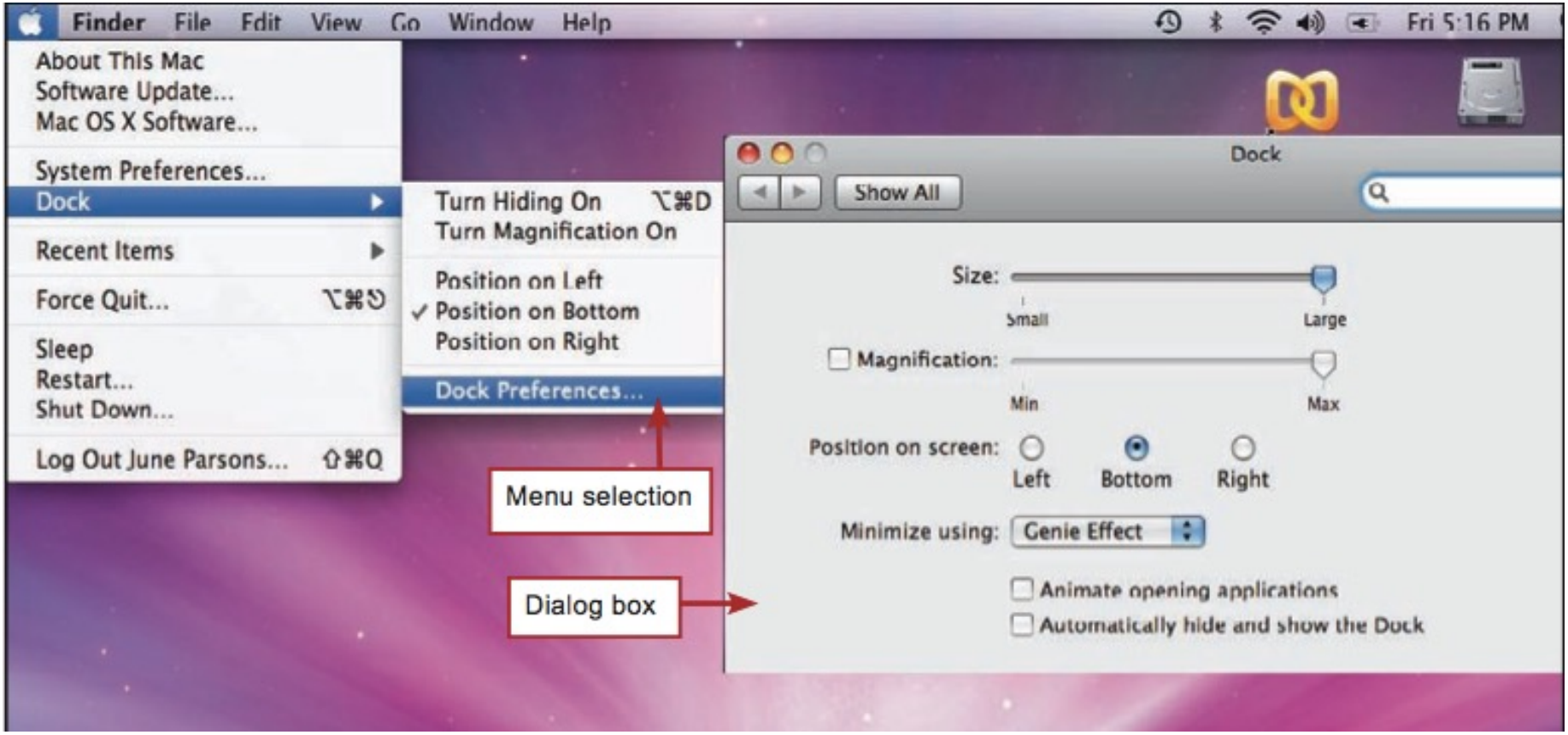
- ***Run time Libraries***:
  - A library of routines that are bound to the program during execution
  - Collection of Software routines which application programs rely on
  - Functions that make appropriate API calls needed to enable the OS to perform accordingly
- ***Application:***
  - Layer where the routines perform tasks the application is designed to do.

Layers of Software (continued)

- ***User interface***:
  - Responsible for the communication between the application and the user
  - Typically it is a ***GUI***, composed of buttons and pull-down menus
  - The GUI passes the information on to the application




Icons can represent programs, documents, data files, utilities, storage areas, or hardware.
Rectangular windows are work areas in which different programs display documents, graphics, and other data.
Paint
Windows Media Center
Calculator
Notepad
WordPad
All Programs
Documents
Pictures
Games
Computer
Network
Control Panel
Devices and Printers
Default Programs
Help and Support
Hydrangeas - Windows Photo Viewer
File
Print
E-mail
Burn
Open


Finder File Edit View Go Window Help
Fri 5:16 PM
About This Mac
Software Update...
Mac OS X Software...
System Preferences...
Dock
Recent Items
Force Quit...
Sleep
Restart...
Shut Down...
Log Out June Parsons...
Turn Hiding On
Turn Magnification On
Position on Left
Position on Bottom
Position on Right
Dock Preferences...
Menu selection
Dialog box
Dock
Show All
Size:
Small
Large
Magnification:
Min
Max
Position on screen:
Left
Bottom
Right
Minimize using:
Genie Effect
Animate opening applications
Automatically hide and show the Dock

Layers of Software (continued)

- ***Scripts*** or ***macros***:
  - Routines that many applications use to allow users to create documents **using the application's built-in commands**.
  - **Allow users to automate sequences of actions they perform frequently**
  - Can perform any function that does not require additional user input

Layers of Software (continued)

- The computer industry relies on specialists in each layer of software.
- Some people make their living writing BIOS software. Others concentrate on improving GUI technology.
- **The greatest number of programmers are found at the application level.**

Windows OS

- **Windows 95/98/NT/ME/2000/XP/Vista/7**
  - Widely used in PCs
  - Supports a vast array of applications and peripheral devices
  - Provides icons, buttons, menus and various other graphical objects that can be manipulated by a mouse
  - Provides a command-line interface
  - Supports ***multitasking*** (running more than one program at a time)

Mac OS X

- Introduced By APPLE computers (1984)
- Contains **graphical user interface** featuring menus and icons
- OS interface through keyboard or mouse
- **Software applications that are compatible with Mac OS are called Mac software.**

Mac OS (continued)

- **Fewer software are compatible with Mac OS** than Windows OS.
- Mac OS has good graphical application software support.
- Some hardware and software add-ons enable Windows software to run on Mac OS.

Mac OS (continued)

- In 1984, Apple Computer introduced the revolutionary Lisa computer.
- The Macintosh computer, with its graphical user interface, ***Mac OS***, was a major factor contributing to its success.

The Macintosh Computer 128K

Mac OS (continued)

- MacBook notebooks (first level)
- MacBook Pro notebooks (high level)
- Ultraportable MacBook Air
- Desktop computers iMac
- Workstation MacPro

DOS

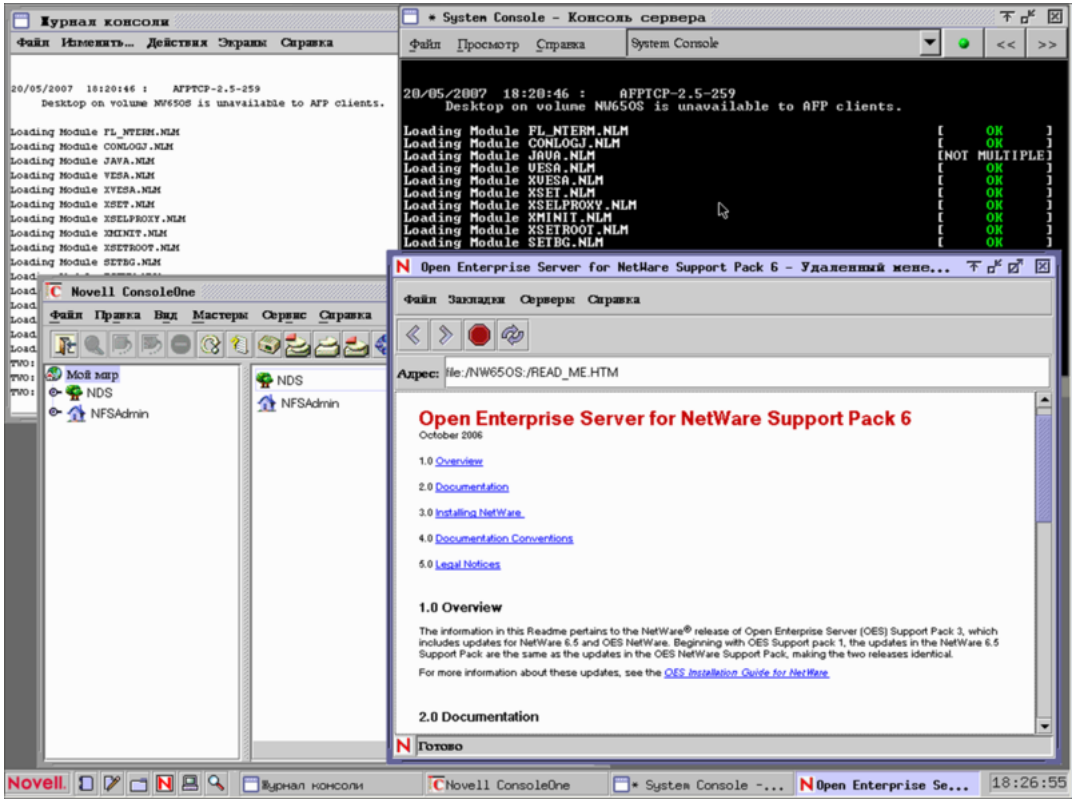
- ***DOS*** stands for **d**isk **o**perating **s**ystem.
  - Developed by Microsoft and introduced in IBM-PC.
  - **Uses command-line interface**. Has been incorporated into Windows OS.

- Operates behind the scenes so Windows users do have to memorize and type complex commands

Windows Server, NetWare, UNIX and LINUX

- Computer providing **centralized storage** and **communications services** requires Server OS
- Differs in managing large flow of data on large networks and web sites
- **Novell NetWare**: developed for micro computer network
  - Used to access documents and data files in a centralized storage

Novell NetWare OS



Windows Server, NetWare, UNIX and LINUX (continued)

- **UNIX and LINUX**:
  - Developed for mini and micro computer networks and web servers of all sizes.
  - Variation of UNIX (1971) is LINUX, which is has a stable and secure OS.
  - UNIX and LINUX are suitable for servers and high-end workstations.
  - A GUI environment

Utilities

- **A program that performs a very specific task, usually related to managing system resources.**
- Operating systems contain a number of utilities for managing disk drives, printers, and other devices.

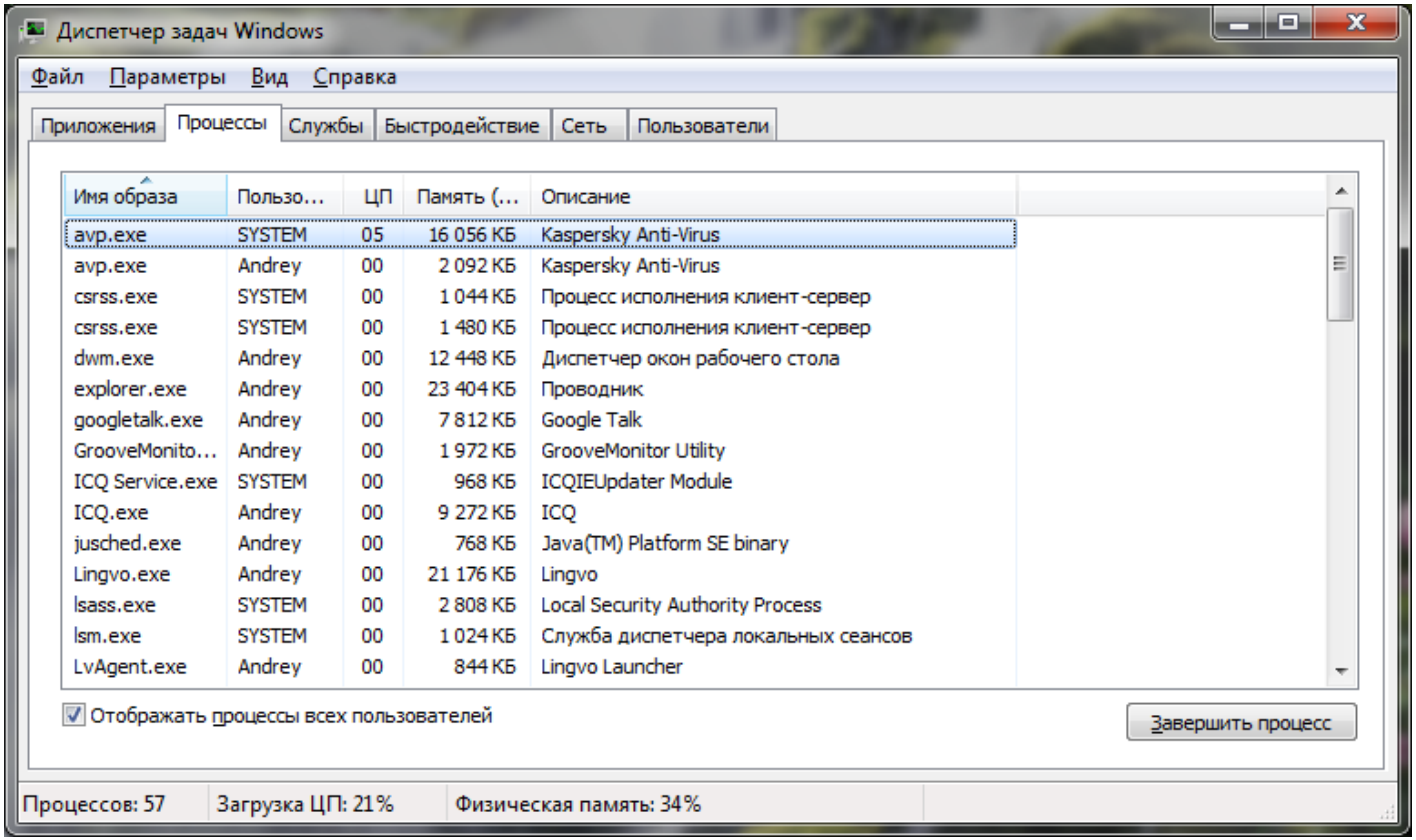
Utilities (continued)

- **Norton Utilities**:
  - Retrieves data from damaged disk, encrypt it and helps troubleshoot the problem
- **OS provides disk formatting utility to format disks.**
- Formatting disk:
  - Creating electronic storage shelves for data (address table)
- Norton Disk Doctor, Registry Cleaner, Disk Cleaner, Disk Defragmenter, Restore Center, System Optimizer, Process Viewer и Performance Test.

Process Control

- Keeps track of all the processes running
- **Process is an instance of a running program**
- Process can be in three states:
  - Running
  - Runnable
  - Blocked
- Kernel maintains a **queue** of processes (**run queue**)

Process Control



Interrupts

- **A signal informing a program that an event has occurred and needs attention or change in execution**.
- Interrupt signals can come from a variety of sources.
  - Hardware interrupts
  - Software interrupts
- To avoid losing of data, interrupts are handled in less than a thousandth of a second.
- PCs support 256 types of software interrupts and 15 hardware interrupts.

Interrupts (continued)

- **Processor receives an IRQ(Interrupt ReQuest)**
- When the processor receives a interrupt, it
  - Stops executing the current application
  - Saves the address of the last instruction executed
  - Jumps to a fixed memory location (e.g. address of keyboard interrupt handler)
  - Starts executing the instructions it finds there in the new memory location
  - Processor's registers contain data the user program was manipulating at the time.

Interrupt Priority and Nested Interrupts

- Processor assigns priorities to different types of interrupts.
  - **Low speed devices have low priority.**
  - **High Speed devices have high priority.**
- Interrupts cannot nest infinitely.
- Interrupt handler can only be interrupted by a higher priority interrupt.

Trap

- Event similar to an interrupt
- **Triggered by the execution of processor instructions**
- Processor traps the errors similar to an interrupt but without time pressure.

- Example: a division-by-zero operation

- **Trap handler** responds to an error either by printing a message, terminate the user program, or continuing with the program.

Fault

- Occurs when:
  - **The hardware is asked to perform a task that is not possible for the hardware devices**
    - For example, non-existent memory location
  - Memory correction circuitry detects an uncorrectable error
  - Attempting to divide by zero
  - Program contains an illegal machine instruction

End of Lecture 5