Operating System (Recap)

A program that acts as an intermediary between a user of a computer and the computer hardware

The goals of an operating system are to:

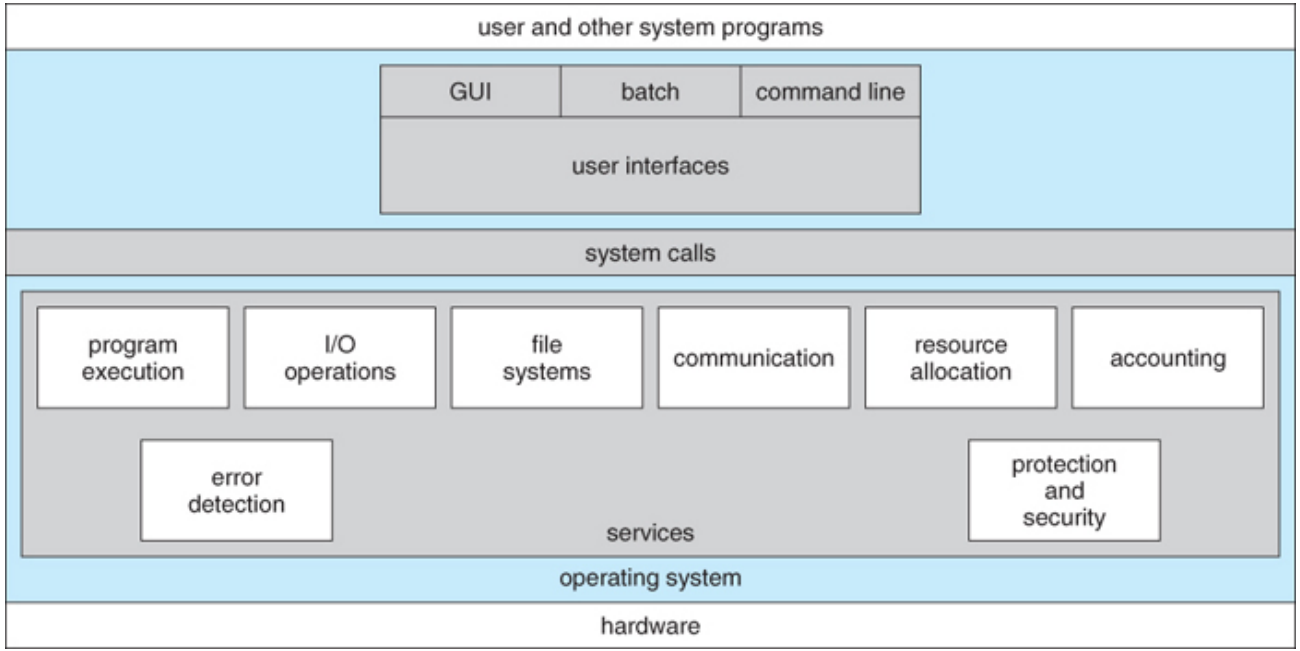
- Execute and control user programs
- Make computer easy to use
- Manage hardware (and other) resources

Operating System (Recap)

OSes provide environments in which programs run, and services for the users of the system

User Interfaces
Program Execution
I/O Operations
File-System Manipulation
Communications
Error Detection

Operating System Services (Recap)



View of operating system services (Silberschatz, Gagne, Galvin)

Operating System Services

User Interfaces - Means by which users can issue commands to the system. Depending on the system these may be a command-line interface (e.g. sh, csh, ksh, tcsh, etc.), a GUI interface (e.g. Windows, X-Windows, KDE, Gnome, etc.), or batch command system

Program Execution - The OS must be able to load a program into RAM, run the program, and terminate the program, either normally or abnormally.

I/O Operations - The OS is responsible for transferring data to and from I/O devices, including keyboards, terminals, printers, and storage devices

(Silberschatz, Gagne, Galvin)

Operating System Services

File-System Manipulation - In addition to raw data storage, the OS is also responsible for maintaining directory and subdirectory structures, mapping file names to specific blocks of data storage, and providing tools for navigating and utilizing the file system.

Communications - Inter-process communications, IPC, either between processes running on the same processor, or between processes running on separate processors or separate machines.

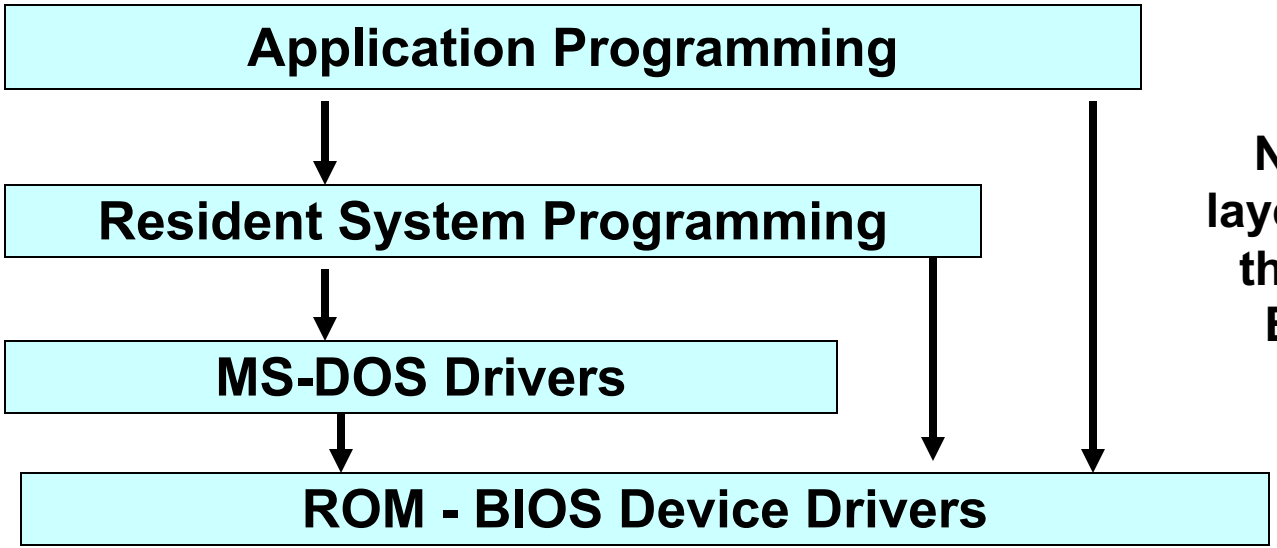
Error Detection - Both hardware and software errors must be detected and handled appropriately, with a minimum of harmful repercussions. Debugging and diagnostic tools.

(Silberschatz, Gagne, Galvin)

OS Structures (Recap)

A SIMPLE STRUCTURE:

Example of MS-DOS



Note how all layers can touch the hardware. Bad News!!

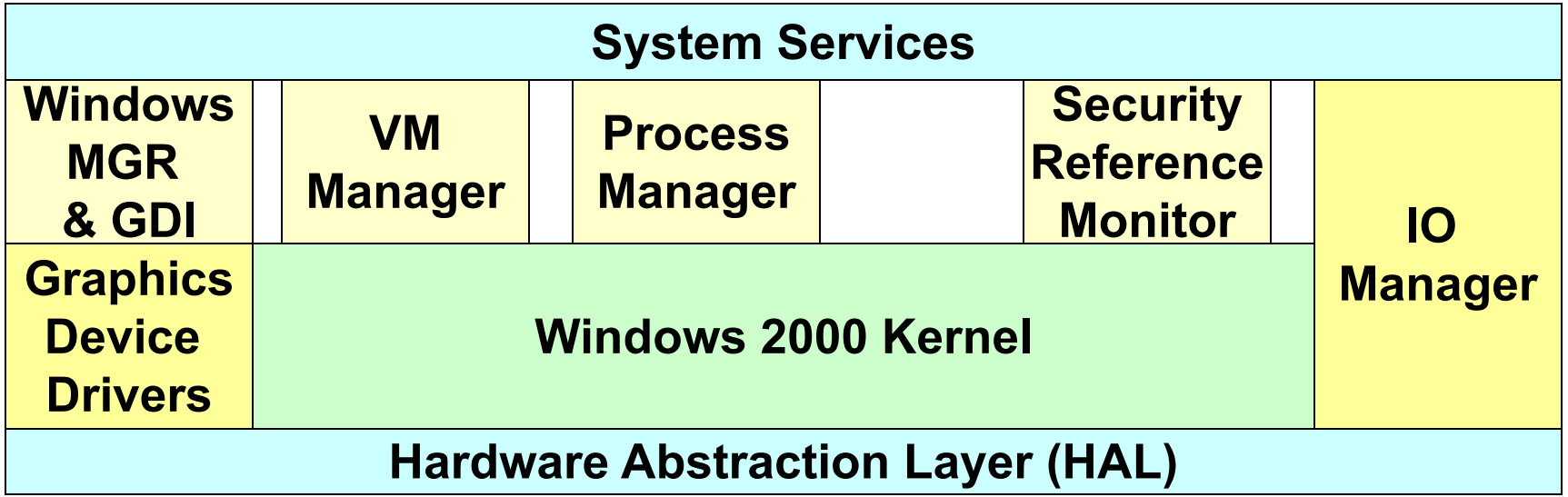
Source: WPI CS 502 Operating Systems, & Silberschatz, Gagne, Galvin

OS Structures (Recap)

How An Operating System Is Put Together

A LAYERED STRUCTURE:

Example of Windows 2000.

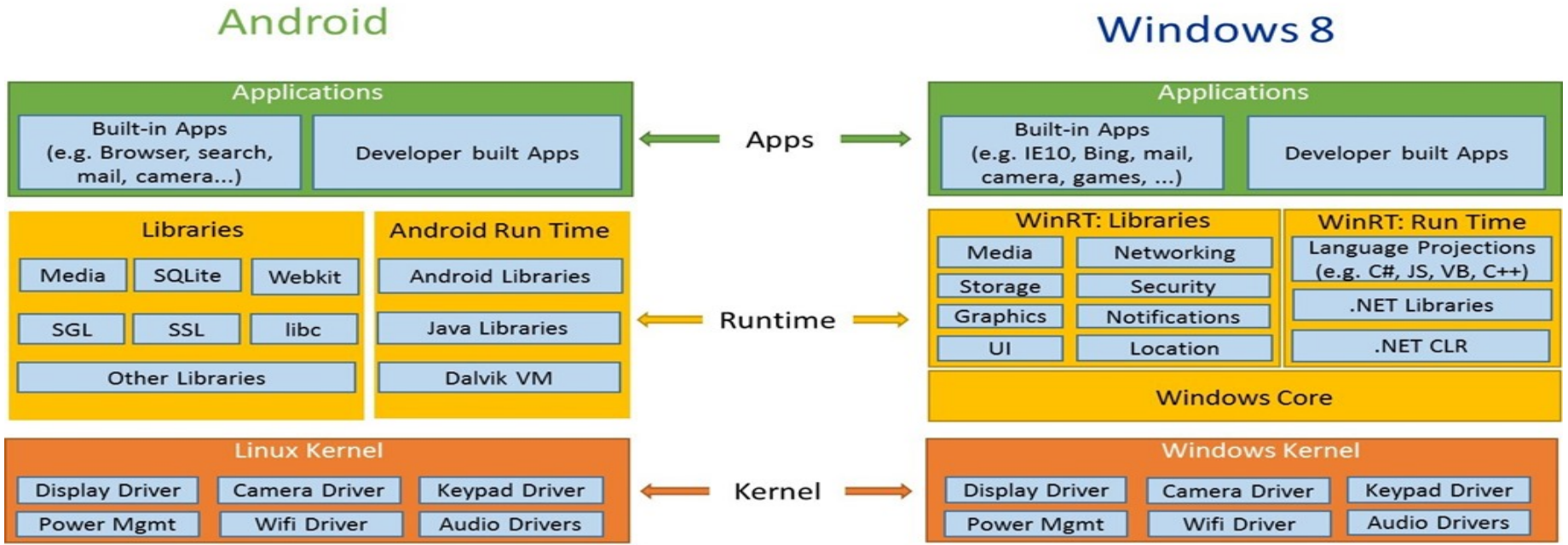


Source: WPI CS 502 Operating Systems, & Silberschatz, Gagne, Galvin

OS Structures (Recap)

How An Operating System Is Put Together

A LAYERED STRUCTURE:



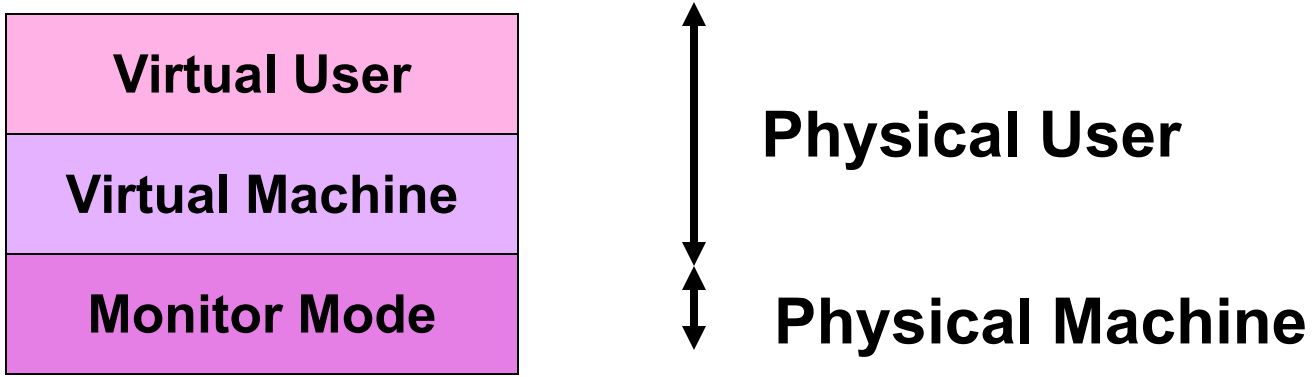
Source: WPI CS 502 Operating Systems, & Silberschatz, Gagne, Galvin

OS Structures (Recap)

Virtual Machine

In a Virtual Machine - each process "seems" to execute on its own processor with its own memory, devices, etc.

- The resources of the physical machine are shared. Virtual devices are sliced out of the physical ones. Virtual disks are subsets of physical ones.
- Useful for running different OS simultaneously on the same machine.



Source: WPI CS 502 Operating Systems, & Silberschatz, Gagne, Galvin

Virtual Machine (Recap)

Provides an interface identical to the underlying bare hardware

Virtual machine implementation intercepts operations and interprets them

Several virtual machines may share the resources of a physical computer:

CPU scheduling: create illusion that users have their own processor

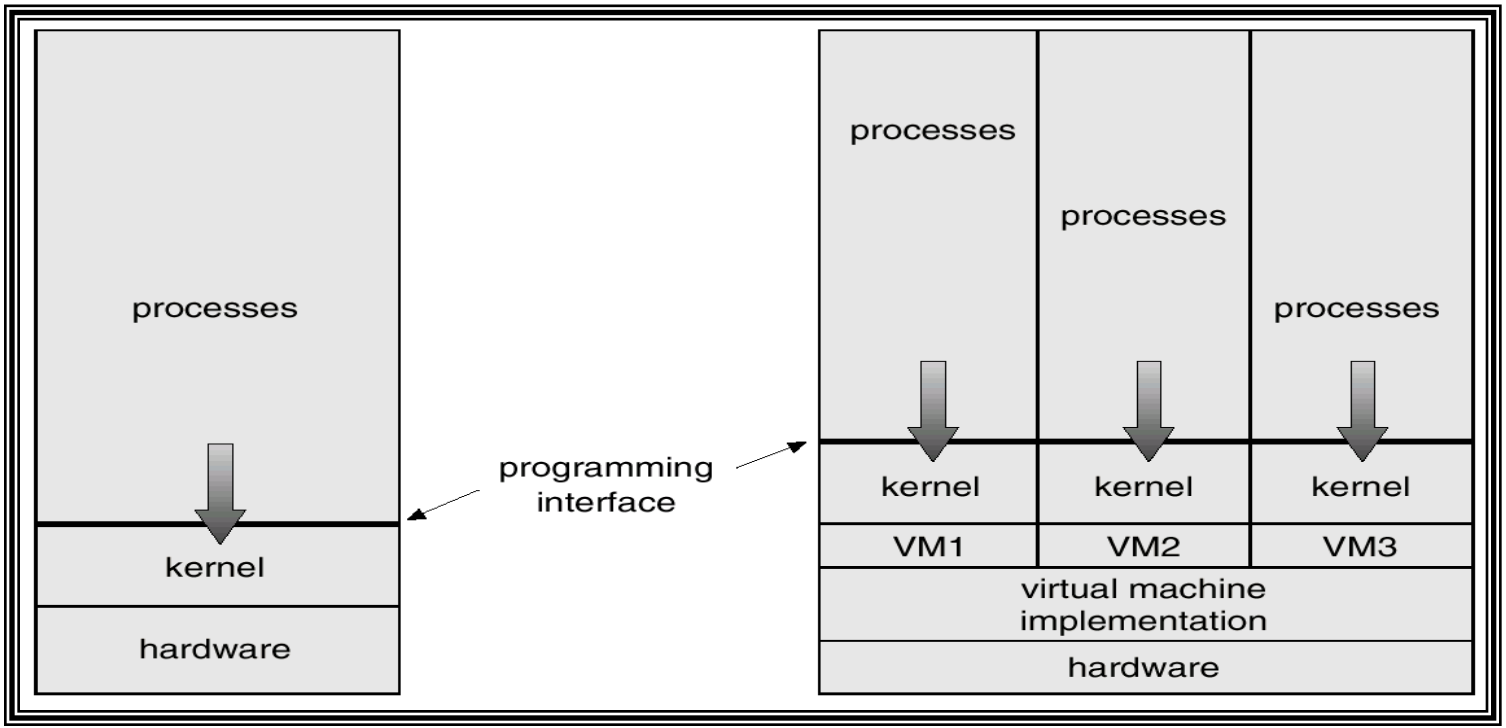
Virtual disks with virtual file systems on physical disk / file system

A normal user time-sharing terminal serves as the virtual machine operator’s console

Possible to run multiple and different OS’s on the same physical computer

OS Structures (Recap)

Virtual Machine

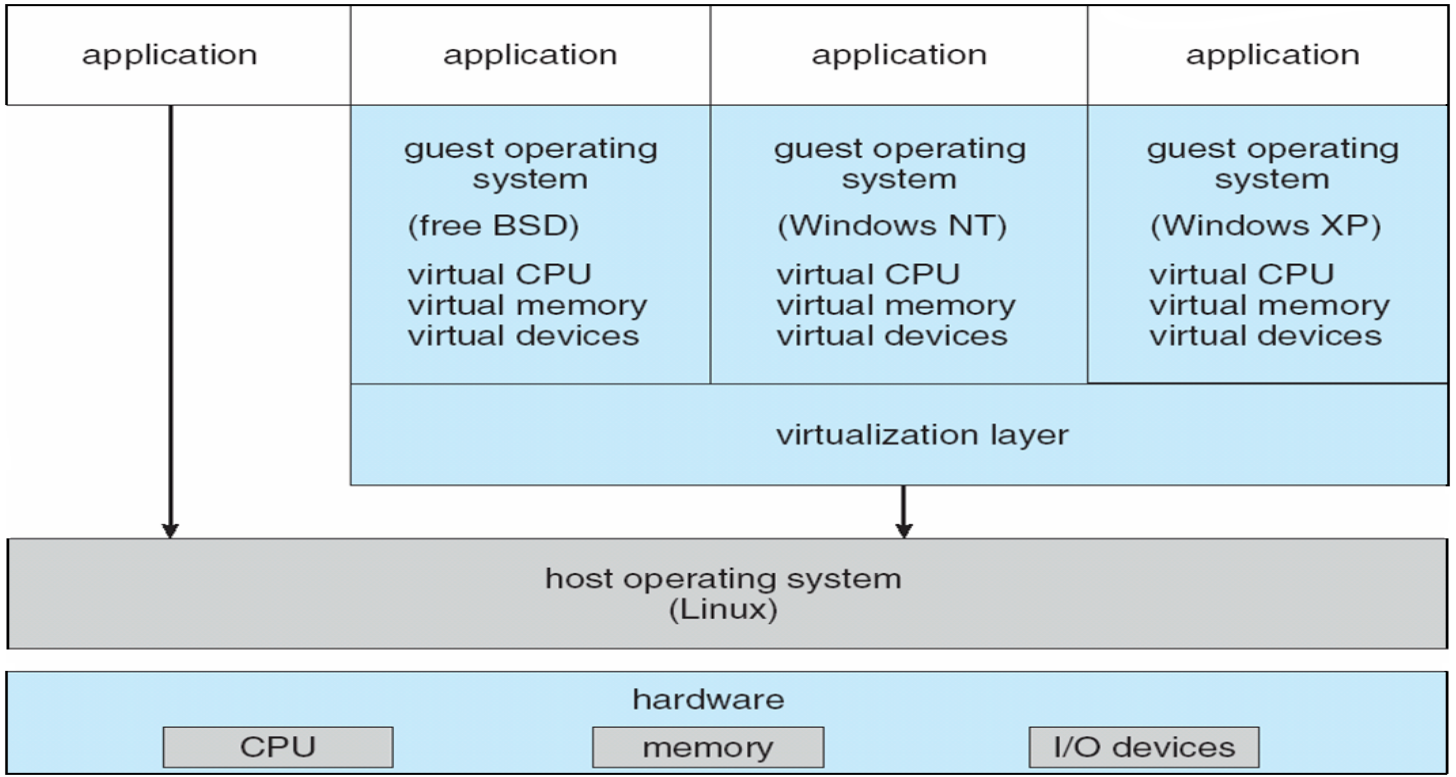


(Silberschatz, Gagne, Galvin)

OS Structures (Recap)

Virtual Machine

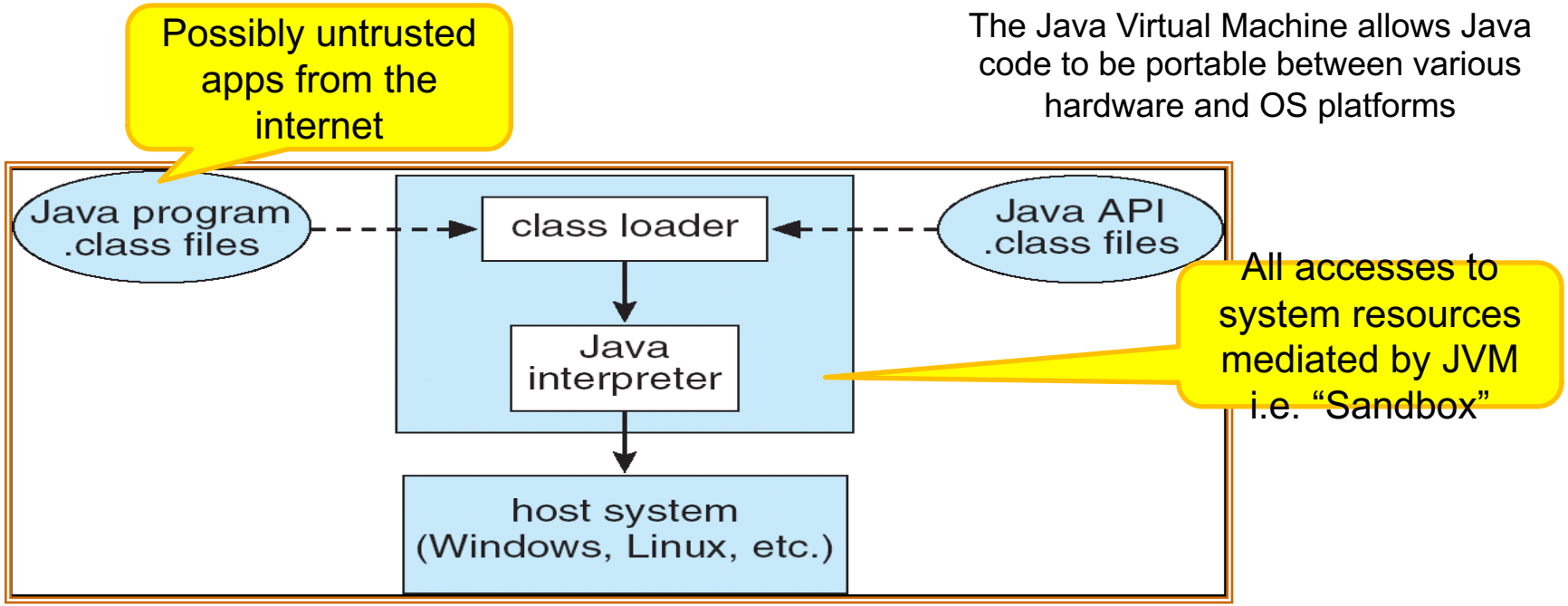
VMware Example



(Silberschatz, Gagne, Galvin)

OS Structures (Recap)

Example of Java Virtual Machine



(Silberschatz, Gagne, Galvin)

Java Virtual Machine (Recap)

Programming environment virtualization

Write once, run anywhere!

Each Java object compiled into architecture-neutral bytecode

JVM class loader loads the bytecode and executes it

Includes garbage collection to reclaim memory no longer in use

Made faster by just-in-time (JIT) compiler that turns bytecode into native code and caches them