Windows Architecture

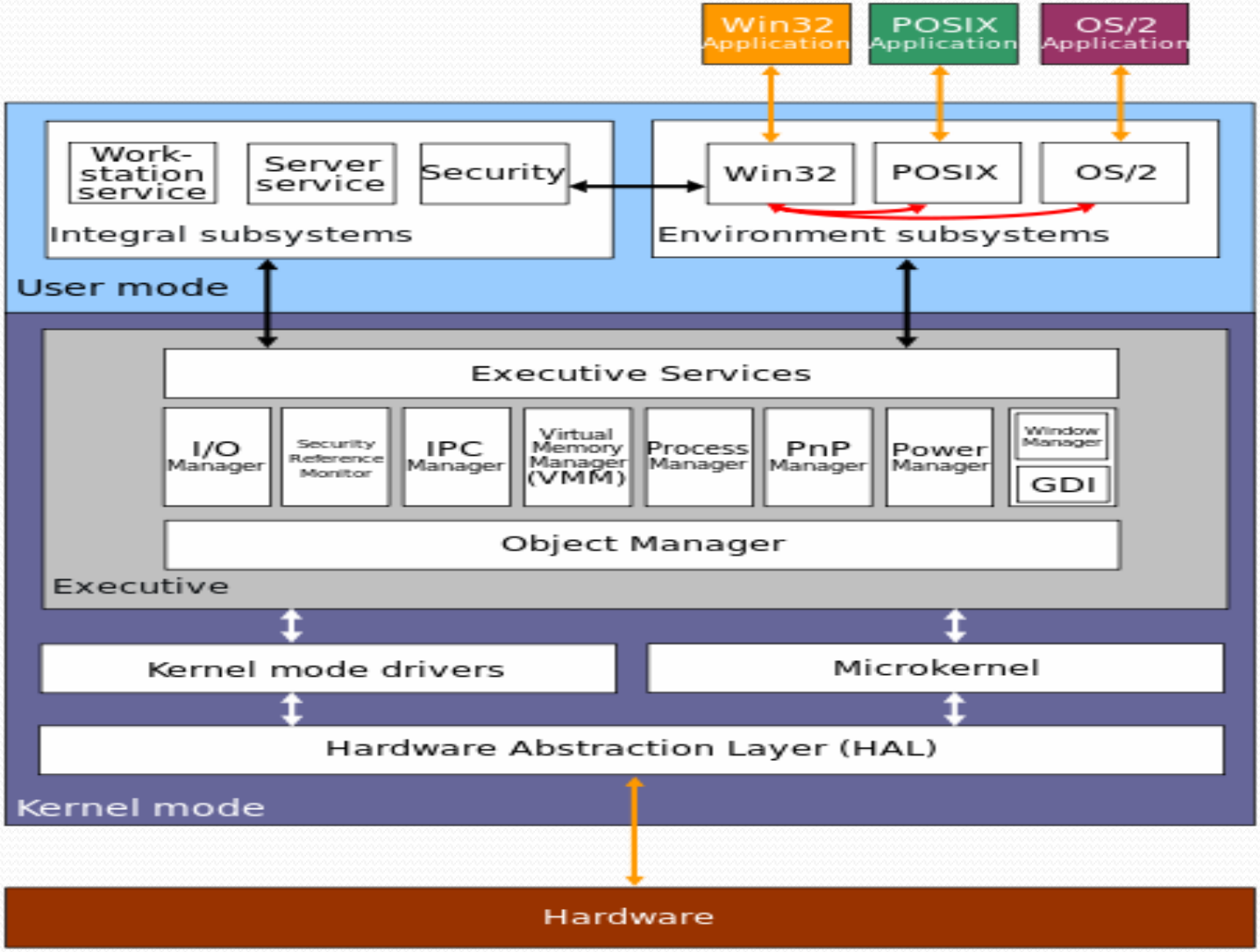
Compiled By: Editorial Staff

For more visit: www.stacksol.com

Windows Architecture

- The **architecture of Windows** , a line of produced and sold by Microsoft, is a layered design.
- Layered design consist of two main components user mode and kernel mode.
- Starting with Windows 2000, Microsoft began making 64-bit versions of Windows available; before this, these operating systems only existed in 32-bit versions.
- This structure is a **modular** structure, composed of several simple modules. These modules are:

- Hardware Abstraction layer
- Kernel/microkernel
- Executive Services
- Environment Subsystem
- Integral subsystem


Win32 Application
POSIX Application
OS/2 Application
Work-station service
Server service
Security
Integral subsystems
Win32
POSIX
OS/2
Environment subsystems
User mode
Executive Services
I/O Manager
Security Reference Monitor
IPC Manager
Virtual Memory Manager (VMM)
Process Manager
PnP Manager
Power Manager
Window Manager
GDI
Object Manager
Executive
Kernel mode drivers
Microkernel
Hardware Abstraction Layer (HAL)
Kernel mode
Hardware

Modes of Windows operating system

Program and application run in OS in two modes

- Protected mode/Kernel mode

➢ Kernel is known as a hybrid kernel. The architecture comprises HAL, driver, microkernel, executive Services.

➢ In Kernel mode, the executing code has complete and unrestricted access to the underlying hardware. It can execute any CPU instruction and reference any memory address. Kernel mode is generally most trusted functions of the operating system. Crashes in kernel mode are terrible, they will halt the entire PC.

- User mode

Programs and subsystems in user mode are limited in terms of to what system resources they have access.(can not directly access hardware).

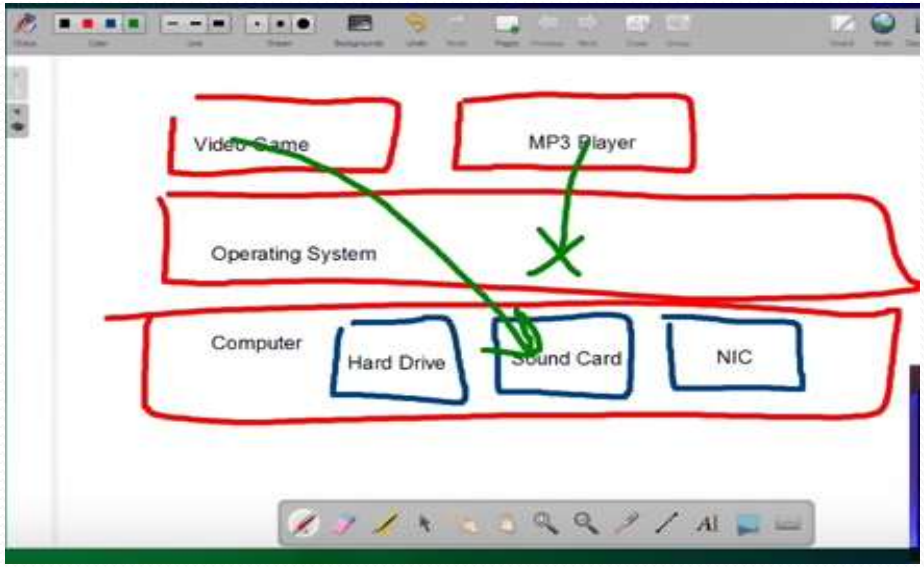
In User mode, the executing code has no ability to *directly* access hardware or reference memory. Code running in user mode must delegate to system APIs to access hardware or memory. Due to the protection afforded by this sort of isolation, crashes in user mode are always recoverable.

Hardware Abstraction Layer

HAL, is a layer between the physical hardware of the computer and the rest of the operating system. It was designed to hide differences in hardware and provide a consistent platform on which the kernel is run. The HAL includes hardware-specific code that controls I/O interfaces, Interrupt controller and multiple processors.

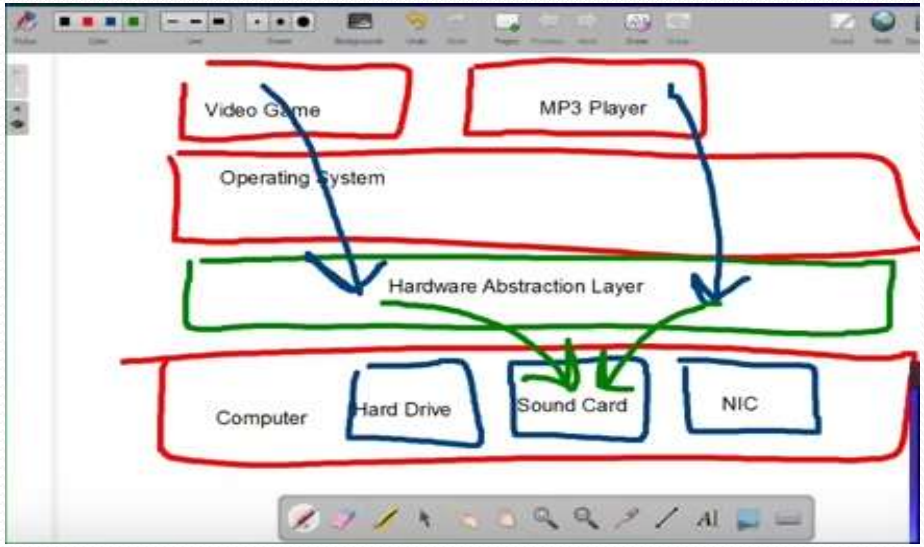
- With out HAL

If a Program access Hardwere then second one is unable to access.



- With HAL

Program access HAL and here Is driver (teach HAL) that is used to tell HAL which hardware access. Therefore we can run multiple program.



Kernel

- Windows Kernel is known as hybrid kernel.
- The **Kernel** works very closely with the **HAL**
- It schedules the activities to be performed by the CPU.
- kernel synchronizes activity among processors to optimize performance.
- kernel mode stops user mode services and applications from accessing critical areas of the operating system that should not have access to.
- Kernel mode drivers exist in three levels.

1) Highest level drivers
2) Intermediate level drivers
3) Low level drivers

Microkernel:

- The Microkernel is a collection of programs that can provide tasks such as **address space management**, **thread management** and **inter-process communication** (**IPC**).
- The Microkernel along with the Windows kernel to make the operating system work efficiently.

Four main responsibilities of kernel

- thread scheduling
- interrupt handling
- low-level processor synchronization
- recovery after a power failure

Executive Services

- The **Executive Services**, which includes the kernel and the HAL, provides a set of common services that the user can use. This section interacts with **Input/output devices**, **object management**, **process management** and the **system security**. Each group of services is managed by one of the components of the executive services, which are as follows:

- Object Manger
- Power Manger
- Process Manager
- I/O Manager
- Virtual memory management
- Local Procedure Call Facility
- Cache Manager
- Security Reference Monitor
- Plug and Play Monitor
- Device Drive Manager

Object Manager

- The **Object Manager** provides rules for **retention**, **naming** and **security** of objects. Objects can be, for example, files and folders saved in the file system. It also removes the duplicate object resources.

  Creation and **insertion** of objects can be done in this section.

  Objects are manipulated by a standard set of methods, namely create, open, close, delete, query name and security

Power Manger

- The **Power Manger** deals with power events like **power-off**, **stand-by**, and **hibernate**. Windows 2000 supports all of the latest standards in Power Management including the **Advanced Power Management** (**AMP**) and **Advanced Configuration and Power Interface** (**ACPI**).
- Consequently, network devices can be powered off when not in use and dynamically reactivated when network access is required.
- Check network security.

Process Manager

- The **Process Manager** manages the creation and deletion of processes. It provides a standard set of services for creating and using processes
- Process is started via the `Create Process` routine which loads any dynamic link libraries that are used by the process, and creates a *primary thread*

- Every dynamic link library or executable file that is loaded into the address space of a process is identified by an *instance handle*

I/O Manager

- The **I/O Manager** manages all the **input** and **output** for the operating system. It supports all file system drivers, hardware device drivers and network drivers, . The I/O Manager provides a common interface that all drivers, such as **FAT** file system driver. This allows the I/O Manager to communicate with all drivers in the same way, without any knowledge of how the devices they control actually work.

Local Procedure Call Facility

- The executive system implements a message passing facility called a **Local Procedure Call** (**LPC**). Applications communicate with the environment subsystems by passing messages via the LPC facility.

Cache Manager

The **Cache Manager** is a part of the **I/O** architecture. It handles caching for the entire I/O system. Caching is used to improve the performance of the I/O systems. Instead of reading and writing directly to disk, frequently used files are temporarily stored in a cache in memory, and read and write operations are performed to these files in the memory.

Security Reference Monitor

- The **Security Reference Monitor** (**SRM**) is responsible for enforcing the **access validation** and **audit-generation policy** defined by the local security subsystem.

Plug and Play Manager

- **Plug and Play**, which made its first appearance with Microsoft **Windows 95**, is now a feature of **Windows 2000**. Changes have been made within the system architecture of Windows 2000 to accommodate this facility.

Device Manager

- **Device Manager** allows you to check the status of your **hardware devices** and to update **device drivers** for the hardware installed on your computer.

Environment subsystem

- Environment subsystem allow Window2000 to run application written for different operating system. The environment subsystem accept the API call made by the application, convert the API call into a format that is understood by Window 2000, and then pass the converted API to executive components running in Kernel mode.
- Main three environment subsystem
  - The Win32 subsystem
  - An OS/2 subsystem
  - POSIX subsystem
-

- **The Win32**

  The main subsystem in windows is Win32. It controls 32 based application and provides an environment for Win16 and Microsoft MS-DOS based application.

- **The OS/2**

  During the early development of window, the window environment become the default. Consequently window provided only limited facilities in the os/2 environment subsystem. OS/2 1.x character based application run on only on Window on Intel86 computer. Real code OS/2 can run on all platform using MS-DOS environment.

- **The POSIX**

The POSIX environment subsystem supports applications that are strictly written to either the POSIX.1 standard that is IEEE standard or the related ISO/IEC standards. This subsystem has been replaced by Interix, which is a part of Windows Services for UNIX. This was in turn replaced by the Windows Subsystem for Linux.

Integral subsystem

- Integral subsystem perform essential operating system function.
- **Security**

❑Creates security token and rights. Permission to user account.

❑Accept user login request and initiates authentication.

- **Workstation services**

❑The workstation services allow a Windows 2000 computer to access the network.

❑Provide an API to access the network redirection