International Journal for Research in Applied Science & Engineering Technology(IJRASET)

Architecture of Windows NT Operating System

Abhishek Mahajan[1], Dhruv Pahuja[2], Akash Verma[3]

Abstract — The architecture of Windows NT, a line of operating systems produced and sold by Microsoft, is a layered design that consists of two main components, user mode and kernel mode. It is a preemptive, reentrant operating system, which has been designed to work with uniprocessor and symmetrical multi-processor (SMP)-based computers. The Microsoft Windows NT operating system was designed and built with fully integrated networking capabilities. These networking capabilities differentiate Windows NT from other operating systems, such as MS-DOS, OS/2, and UNIX, in which network capabilities are installed separately from the core operating system. It was initially hyped as the replacement for all other operating systems for Intel-based PCs, but it was somewhat slow to catch on and was later redirected to the upper end of the market, where it found a niche. It is gradually becoming more popular at the low end as well. This research paper includes the design goals and rationale for the Windows NT operating system.

Index Terms— User mode, kernel mode ,Hardware abstraction layer, Environment subsystems.

I. INTRODUCTION

The Microsoft Windows NT Server Resource Kit for version 4.0 consists of three new volumes and a single compact disc (CD) containing programs for both Windows NT Workstation and Windows NT Server. The architecture of Windows NT, a line of operating systems produced and sold by Microsoft, is a layered design that consists of two main components, user mode and kernel mode. It is a preemptive, reentrant operating system, which has been designed to work with uniprocessor and symmetrical multi-processor (SMP)-based computers.

To process input/output (I/O) requests, they use packet-driven I/O, which utilizes I/O request packets (IRPs) and asynchronous I/O. Starting with Windows 2000, Microsoft began making 64-bit versions of Windows available before this, these operating systems only existed in 32-bit versions.

It was initially hyped as the replacement for all other operating systems for Intel-based PCs, but it was somewhat slow to catch on and was later redirected to the upper end of the market, where it found a niche. It is gradually becoming more popular at the low end as well.

NT is sold in two versions: server and workstation. These two versions are nearly identical and are generated from the same source code. The server version is intended for machines that run as LAN-based file and print servers and has more elaborate management features than the workstation version, which is intended for desktop computing for a single user.

II. THE WHOLE ARCHITECTURE OF WINDOWS NT CAN BE DIVIDED INTO TWO PARTS

A. User mode: User mode is the least privileged mode of Windows NT and it has no direct access to hardware and only restricted access to memory. For example, when programs such as Word and Lotus Notes execute in user mode, they are confined to sandboxes with well-defined restrictions.

The user mode is made up of subsystems which can pass I/O request to the appropriate kernel mode drivers via the I/O manager (which exists in kernel mode). The user mode layer of Windows NT is made up of the Environment subsystems and the Integral subsystem.

The environment subsystems were designed to run applications written for many different types of operating systems. None of the environment subsystems can directly access hardware, and must request access to memory resources through the Virtual Memory Manager that runs in kernel mode.

1. Components of User mode:

- End user Applications
- Environment Subsystems

B. Kernel mode: Windows NT kernel mode has full access to the hardware and system resources of the computer and runs code in a protected memory area. It controls access to scheduling, thread prioritization, memory management and the interaction with hardware. The kernel mode stops user mode services and applications from accessing critical areas of the operating system

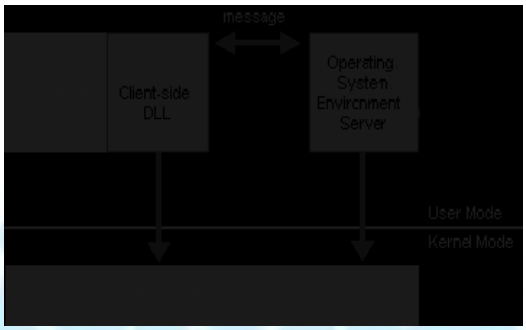
that they should not have access to; user mode processes must ask the kernel mode to perform such operations on their behalf.

2. Components of Kernel mode:

- NT Executive
- NT Kernel (Microkernel)
- Hardware Abstraction Layer (HAL)

III. OPERATING SYSTEM ENVIRONMENTS

NT's operating system environments are implemented as client/server systems. As part of the compile process, applications are bound by a link-time binding to an operating system API that NT's operating system environments export. The link-time binding connects the application to the environment's client-side DLLs, which accomplish the exporting of the API. For example, a Win32 program is a client of the Win32 operating system environment server, so it is linked to Win32's client-side DLLs, including Kernel32.dll, gdi32.dll, and user32.dll. A POSIX program would be linked to the POSIX client-side DLL, psxdll.dll.



Client-side DLLs carry out tasks on behalf of their servers, but they execute as part of a client process. As figure shows, in some cases a client-side DLL can fully implement an API without having to call upon the help of the server; in other cases, the server must help out. The server's aid is usually necessary only when global information related to the environment must be updated. When the client-side DLL requires help from the server, the DLL sends a message known as a local procedure call (LPC) to the server. When the server completes the specified request and returns an answer, the DLL can complete the function and return control to the client. Both the client-side DLL and the server may use NT's native API when necessary. Operating system environment APIs augment the native API with additional functionality or semantics that are specific to themselves.

IV. EXECUTIVE

The NT Executive takes care of the important tasks that are vital to the entire system. This includes services such as object management, virtual memory management, I/O management, and process management. NT's Executive subsystems make up the meatiest layer in kernel mode, and they perform most of the functions traditionally associated with operating systems.The executive is the kernel-mode portion of the Windows NT operating system and, except for a user interface, is a complete operating system unto itself. The executive is never modified or recompiled by the system administrator.

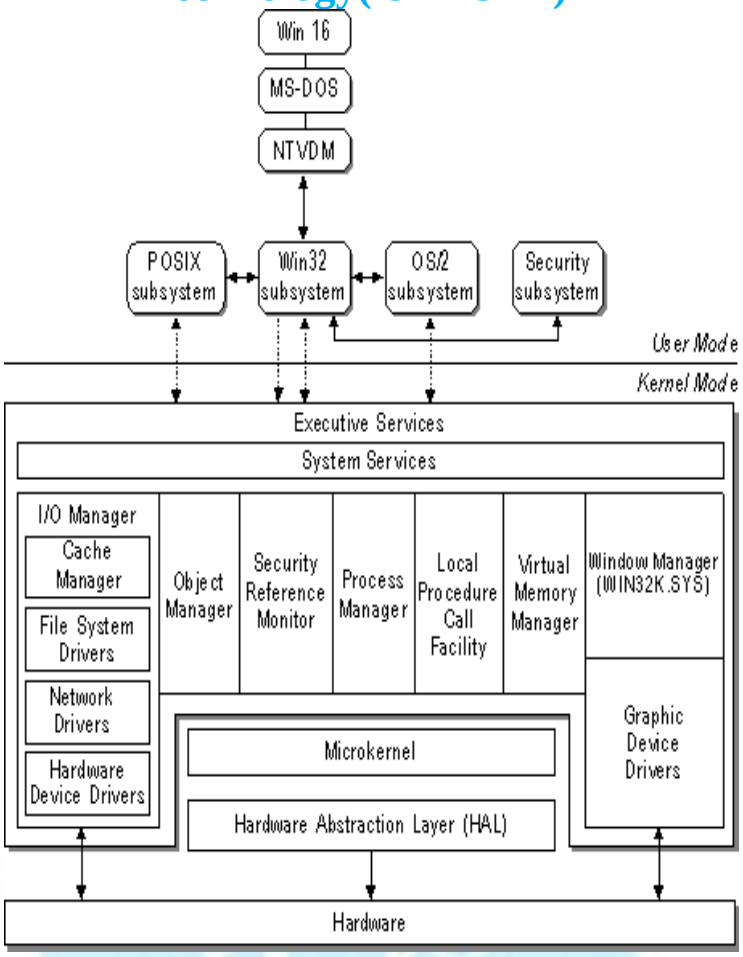


Figure: Windows NT operating system architecture

The executive components are listed below.

I/O Manager
Object Manager
Security Reference Monitor
Process Manager
Local Procedure Call Facility
Virtual Memory Manager
Window Manager
Graphics Device Interface
Graphics Device Drivers

V. KERNEL MODE DRIVERS

Windows NT uses kernel-mode device drivers to enable it to interact with hardware devices. Each of the drivers has well defined system routines and internal routines that it exports to the rest of the operating system. All devices are seen by user mode code as a file object in the I/O manager, though to the I/O manager itself the devices are seen as device objects, which it defines as either file, device or driver objects.

Kernel mode drivers exist in three levels:

- Highest level drivers
- Intermediate level drivers
- Low level drivers

VI. HARDWARE ABSTRACTION LAYER

The Windows NT hardware abstraction layer, or 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 therefore provide a consistent platform on which the kernel is run. The HAL includes hardware-specific code that controls I/O interfaces, interrupt controllers and multiple processors.

However, despite its purpose and designated place within the architecture, the HAL isn't a layer that sits entirely below the kernel, the way the kernel sits below the Executive: all known HAL implementations depend in some measure on the kernel, or even the Executive. In practice, this means that kernel and HAL variants come in matching sets that are specifically engineered to work together.

Its job is to present the rest of the operating system with abstract hardware devices, devoid of the warts and idiosyncracies with which real hardware is so richly endowed. Among the devices modeled are off-chip caches, timers, I/O buses, interrupt controllers, and DMA controllers. By exposing these to the rest of the operating system in idealized form, it becomes easier to port NT to other hardware platforms, since most of the modifications required are concentrated in one place.

VII. CONCLUSION

Windows NT is a symmetric multiprocessing operating system which support multiple operating system environments. It has a Windows graphical user interface and runs Win32, 16-bit Windows, MS-DOS, POSIX, and OS/2 program.

It employs advanced operating system principles such as virtual memory, preemptive multitasking, structured exception handling, and operating system objects. It is secure, powerful, reliable, and flexible.

REFERENCES

[1] Finnel, Lynn (2000). MCSE Exam 70-215, Microsoft Windows 2000 Server. Microsoft Press. ISBN 1-57231-903-8.
[2] Russinovich, Mark (October 1997). "Inside NT's Object Manager". Windows IT Pro.
[3] "Active Directory Data Storage". Microsoft. Retrieved 2005-05-09.
[4] Solomon, David; Russinovich, Mark E. (2000). *Inside Microsoft Windows 2000* (Third Edition ed.). Microsoft Press. ISBN 0-7356-1021-5.
[5] Russinovich, Mark; Solomon, David (2005). Microsoft Windows Internals (4th edition ed.). Microsoft Press. ISBN 0-7356-1917-4.
[6] Schreiber, Sven B. (2001). Undocumented Windows 2000 Secrets. Addison-Wesley Longman. ISBN 978-0201721874.
[7] Siyan, Kanajit S. (2000). Windows 2000 Professional Reference. New Riders. ISBN 0-7357-0952-1.