Introduction

Computer systems' crucial operating systems (OS) serve as a bridge between hardware and software programmes. They control system resources like memory, file systems, and processes to make ensuring a computer system runs smoothly and effectively. The three essential operating system components that have the biggest direct effects on a computer system's performance and functioning are process control, file management, and memory management. Effective process control guarantees that tasks are completed on time and effectively without expending excessive resources. File organisation and management provide accurate and safe data storage and retrieval. Memory allocation and management guarantee that there is adequate memory accessible for running programmes. They also release memory when it is no longer required. Two of the most popular operating systems, Windows and macOS, provide a variety of management functions aimed at maximizing these three elements. Operating systems make sure that computers run smoothly and effectively by maximizing process control, file management, and memory management. This enables users to do their activities quickly and conveniently.

Process Management

Microsoft created Windows, an operating system that manages processes using preemptive multitasking. The scheduler in Windows allocates CPU time to processes based on their assigned priority levels, ensuring that higher-priority processes receive more CPU time than lower-priority ones. Furthermore, Windows employs threads as the fundamental unit of operation, allowing multiple threads to operate simultaneously within a single process. This means that developers can create, manage, and synchronize processes and threads with ease, thanks to the numerous APIs provided by the OS. These APIs provide granular control over the behavior of processes and threads, allowing for efficient use of system resources.

In addition to these process management capabilities, Windows includes a range of built-in security tools to help protect the system from unauthorized access and modification. One of the most notable tools is User Account Control (UAC), which prompts the user for permission before allowing certain actions to be taken. This prevents malware from making unwanted changes to the system, as the user must approve any changes before they can be made. By implementing such security measures, Windows provides a secure and stable platform for users and developers alike.

macOS, Apple's operating system, is based on the Unix operating system and employs a similar preemptive multitasking strategy to Windows. The Mach kernel, which underlies macOS, provides sophisticated process management tools such as lightweight threads and inter-process communication. With these tools, developers can create, control, and synchronize processes and threads with ease, thanks to the many APIs provided by the OS. This allows developers to write efficient and scalable applications that take full advantage of the system's capabilities.

In addition to its robust process management capabilities, macOS includes several built-in security tools to protect against malware and other threats. Gatekeeper, for example, verifies the legitimacy of apps before installation, preventing malicious software from infiltrating the system. Similarly, macOS comes with FileVault, which encrypts the entire hard drive, providing an additional layer of security. By incorporating such security measures, macOS provides a safe and secure environment for users and developers alike.

Overall, both Windows and macOS offer robust process management capabilities, including preemptive multitasking and support for threads and APIs. Additionally, both operating systems include built-in security tools to help protect against malware and unauthorized modifications to the system. These features make Windows and macOS reliable, secure, and efficient operating systems for a wide range of users and developers.

File Management

Windows: Windows' default file system is the New Technology File System (NTFS). Advanced features including file compression, encryption, and access control are supported by NTFS. Additionally, it offers journaling, which safeguards the file system's integrity by documenting modifications before they are committed. Windows offers a hierarchical folder structure and classifies file types based on file extensions. Developers may create, read, write, and remove files and directories using the OS's different file management APIs.

Along with these capabilities, Windows also has Storage Spaces, which lets users build customizable storage pools out of several drives, and File History, which lets users restore past versions of data. The OS supports the Server Message Block (SMB) protocol for file and folder sharing over a network, enabling users to access shared resources on other machines. The ability to sync files between devices and access them from any location is provided by Windows' built-in support for cloud storage services like OneDrive.

Apple File System (APFS) is the default file system for macOS. The APFS file system includes functions including encryption, snapshots, and space sharing and is designed for solid-state discs (SSDs). The file system's dependability is further ensured by built-in support for data integrity and crash prevention. A hierarchical folder system is also used by macOS, and file extensions are used to indicate the nature of files. Similar to Windows, macOS has file management APIs that let developers carry out a variety of file operations.

Time Machine, which enables users to backup and restore their whole system, and Finder, which offers sophisticated file search and organisation capabilities, are two tools that come with macOS. Users can access shared resources on other machines thanks to the operating system's support for file sharing over a network utilizing SMB and the Apple Filing Protocol (AFP). iCloud Drive, a cloud storage platform that enables users to sync files across devices and view them from any location, is also integrated with macOS.

Memory Management

Windows: The virtual memory management system used by Windows divides physical memory into pages and assigns them to programmes as needed. The operating system uses a demand-paging technique to only load pages into memory when they are needed. A page replacement mechanism is also used by Windows when memory has to be freed up. Developers can allocate, deallocate, and manage memory using the OS's memory management APIs.

Windows has functions like ReadyBoost that let users use external storage devices as extra memory caches to increase system speed. The OS also supports SuperFetch, a feature that speeds up startup time by preloading commonly used software into memory. The Windows Display Driver Model (WDDM), which controls memory allocation for graphics processing in Windows, makes sure that graphics-intensive programmes have access to the resources they require for optimum performance. Non-Uniform Memory Access (NUMA) architectures, which can enhance performance on systems with many CPUs, are also supported by Windows.

macOS: The Mach kernel manages memory deallocation and allocation in the virtual memory management system used by macOS. Similar to Windows, macOS manages memory effectively by using demand-paging and a page replacement mechanism. For developers to work with memory allocation and deallocation, the OS provides memory management APIs.

Memory compression, which decreases the amount of physical memory used by compressing idle data, and App Nap, which saves energy by reducing the speed of idle programmes' background operations, are features included in macOS. Additionally, the OS offers Automatic Reference Counting (ARC), an approach to memory management that makes it easier for developers to manage memory by automatically retiring objects when they are no longer required. The Metal framework, which offers low-level access to the GPU for better speed, is used by macOS to handle memory allocation for graphics processing. The NUMA architectures are also supported by macOS, which improves performance on computers with many processors.

Comparison Table

Feature	**Windows**	**macOS**

Process Control	Preemptive multitasking, priority-based scheduling, thread-based execution, UAC	Preemptive multitasking, Mach kernel, thread-based execution, Gatekeeper
File Management	NTFS, hierarchical folder structure, file extensions, file management APIs, File History, Storage Spaces	APFS, hierarchical folder structure, file extensions, file management APIs, Time Machine, Finder
Memory Management	Virtual memory, demand-paging, page replacement algorithm, memory management APIs, ReadyBoost	Virtual memory, Mach kernel, demand-paging, page replacement algorithm, memory management APIs, memory compression, App Nap

Conclusion

In conclusion, operating systems like Windows and macOS are essential to the smooth operation of computer systems. Process control, file management, and memory management are the three key operating system elements that are directly related to how well a computer system performs and operates. Strong process management features are offered by both Windows and macOS, including preemptive multitasking, thread and API support, and integrated security measures. These security solutions shield the system from viruses and unauthorised changes. With a hierarchical folder structure, file extensions for categorization, and APIs for file operations, Windows manages files using the New Technology File System (NTFS). For file management, macOS employs the Apple File System (APFS), which has capabilities that are comparable to those of NTFS. File sharing via a network and cloud storage services are supported by both Windows and macOS. Both Windows and macOS use their respective memory management APIs to allocate, deallocate, and manage memory. For a variety of consumers and developers, Windows and macOS provide operating systems that are dependable, safe, and effective.