Linux Architecture:

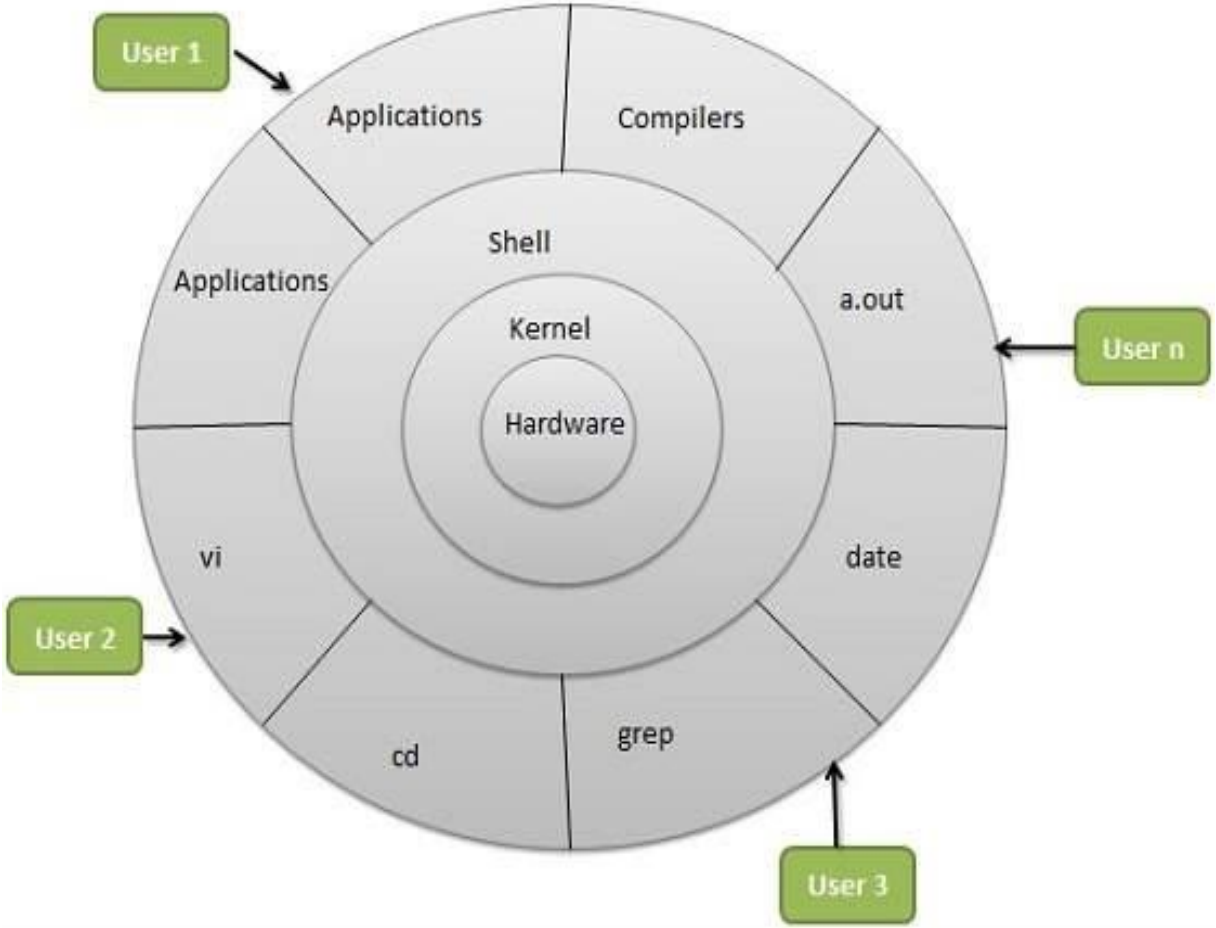
Difference between Windows and Linux

hitbullseye

Table of Contents

Architecture

- The following illustration shows the architecture of a Linux system −



A Linux system's design is made up of the following layers:

- **Hardware layer:** Hardware is made up of all the extra devices, like RAM, HDD, CPU, and so on.
- **The kernel** is the most important part of an operating system. It communicates directly with the hardware and gives low-level services to higher-level parts.
- **Shell** is an interface to the kernel that hides the complexity of how the kernel works from the user. The user gives commands to the shell, which then runs the functions of the kernel.

Utilities: These are programs that help the user do most of what an operating system can do.

The Linux Operating System’s architecture primarily has these components: the Kernel, Hardware layer, System library, Shell, and System utility.

1) The kernel is the most important part of the operating system. It is in charge of all of LINUX's main tasks. This operating system is made up of different modules that talk to the hardware directly. The kernel gives the system the necessary level of separation to hide details about application programs or low-level hardware. Here are the different kinds of Kernels:

Monolithic Kernel Microkernels Exo kernels Hybrid kernels

2). System libraries are special functions that make the operating system work. They don't need the code access rights of kernel modules.

3) System Utility programs can be used to do both simple and complex tasks.

4). The LINUX operating system's hardware layer is made up of devices like RAM, HDD, and CPU.

- The shell is a way for the user to talk to the kernel, and it also lets the kernel do things. It takes instructions from the user and runs the functions of the kernel. There are two kinds of shells: command-line shells and graphical shells. The Shell is a part of many types of operating systems.
- The command-line shells have a command-line interface, while the graphical-line shells have a graphical user interface. Both shells can do things, but the shells with a graphical user interface do things slower than the shells with a command line interface. There are four main types of shells:
- **Shell Korn, The Bourne shell, Shell C, POSIX shell**

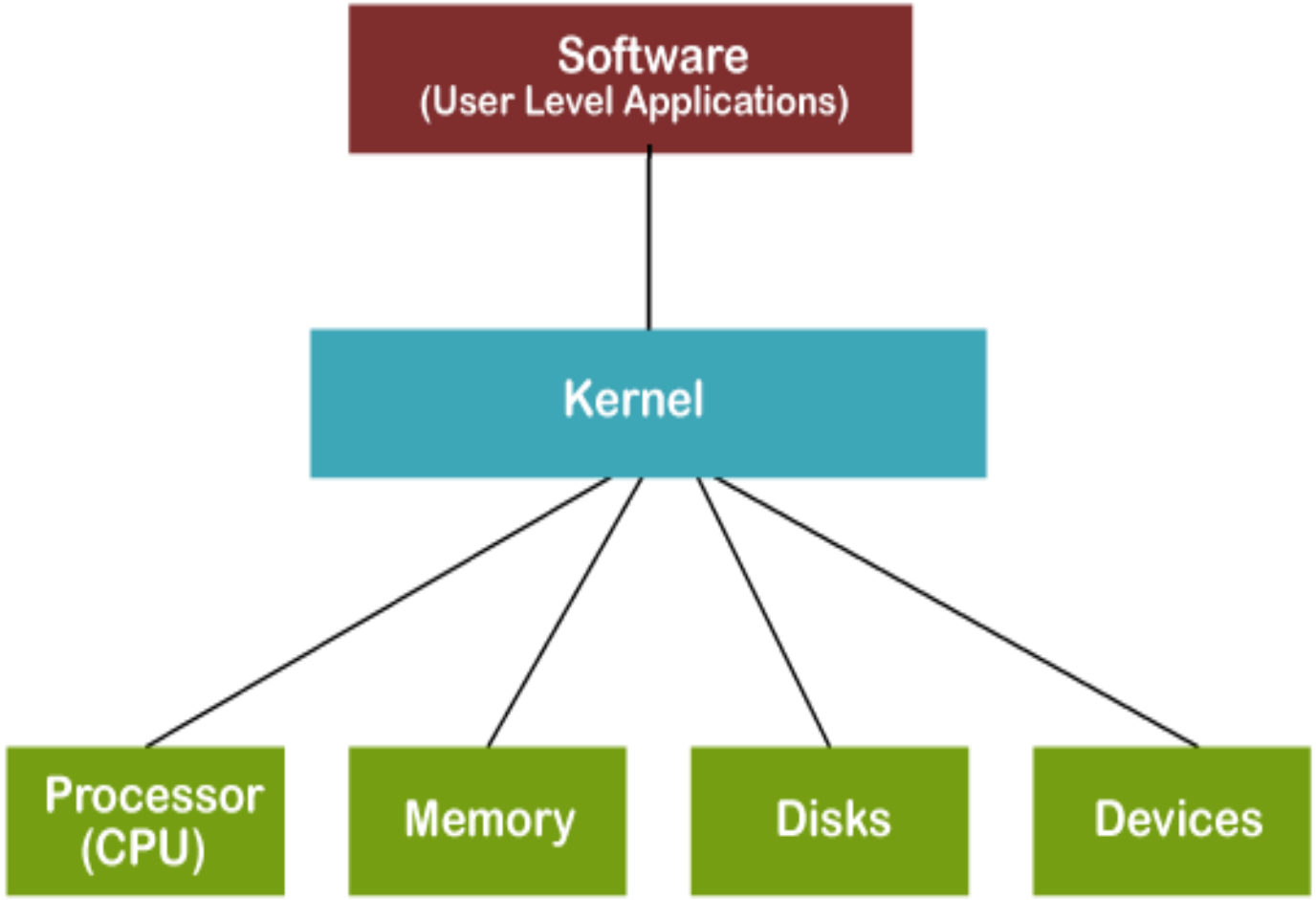
What is Kernel?

- In computer science***, Kernel is a computer program that is a core or heart of an operating system***. Before discussing kernel in detail, let's first understand its basic, i.e., Operating system in a computer.

Operating System:

- An operating system or OS is system software that works as an interface between hardware components and the end-user. It enables other programs to run. Each computer system, whether it is desktop, laptop, tablet, or smartphone, must have an OS to provide basic functionalities for the device. Some widely used operating systems are **Windows, Linux, MacOS, Android, iOS, etc.**

What is Kernel in Operating System?



- As we've already talked about, the Kernel is the most important part of an OS, so it controls everything in the system. The kernel is the part of the computer that is in charge of all hardware and software operations.
- It acts as a link between programs and the data processing that happens at the hardware level. It is the most important part of an OS.
- It is the part of the OS that is always in the computer's memory and lets software and hardware parts talk to each other.

- It is the first program that loads when the computer is turned on (After the bootloader). Once it's loaded, it takes care of all the other start-ups. It also takes care of software requests for memory, peripherals, and I/O. Also, it turns all I/O requests into instructions for the CPU on how to process data. It also takes care of other things, like managing memory, tasks, and disks.

- A kernel is kept and loaded into a protected Kernel space, which is a separate area of memory. It is locked down so that application programs or less important parts of the OS can't get to it.
- Other programs like browsers, word processors, and audio and video players use a separate area of memory called user-space.
- Due to these two separate spaces, user data and kernel data don't interfere with each other and do not cause any instability and slowness.

Functions of a Kernel

- A kernel of an OS is responsible for performing various functions and has control over the system. Some main responsibilities of Kernel are given below:

Device Management

- To perform various actions, processes require access to peripheral devices such as a mouse, keyboard, etc., that are connected to the computer. A kernel is responsible for controlling these devices using device drivers. Here, a device driver is a computer program that helps or enables the OS to communicate with any hardware device.
- A kernel maintains a list of all the available devices, and this list may be already known, configured by the user, or detected by OS at runtime.

Memory Management

- The kernel has full control for accessing the computer's memory. Each process requires some memory to work, and the kernel enables the processes to safely access the memory. To allocate the memory, the first step is known as virtual addressing, which is done by paging or segmentation. Virtual addressing is a process of providing virtual address spaces to the processes. This prevents the application from crashing into each other.

Resource Management

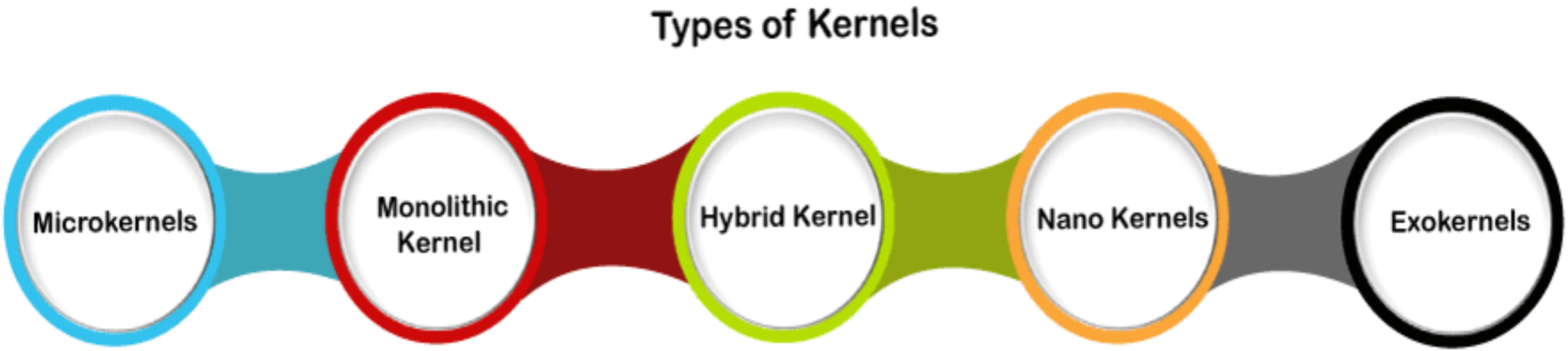
- One of the important functionalities of Kernel is to share the resources between various processes. It must share the resources in a way that each process uniformly accesses the resource.
- The kernel also provides a way for synchronization and inter-process communication (IPC). It is responsible for context switching between processes.

Accessing Computer Resources

- A kernel is responsible for accessing computer resources such as RAM and I/O devices. RAM or Random-Access Memory is used to contain both data and instructions. Each program needs to access the memory to execute and mostly wants more memory than the available. For such a case, Kernel plays its role and decides which memory each process will use and what to do if the required memory is not available. The kernel also allocates the request from applications to use I/O devices such as keyboards, microphones, printers, etc.

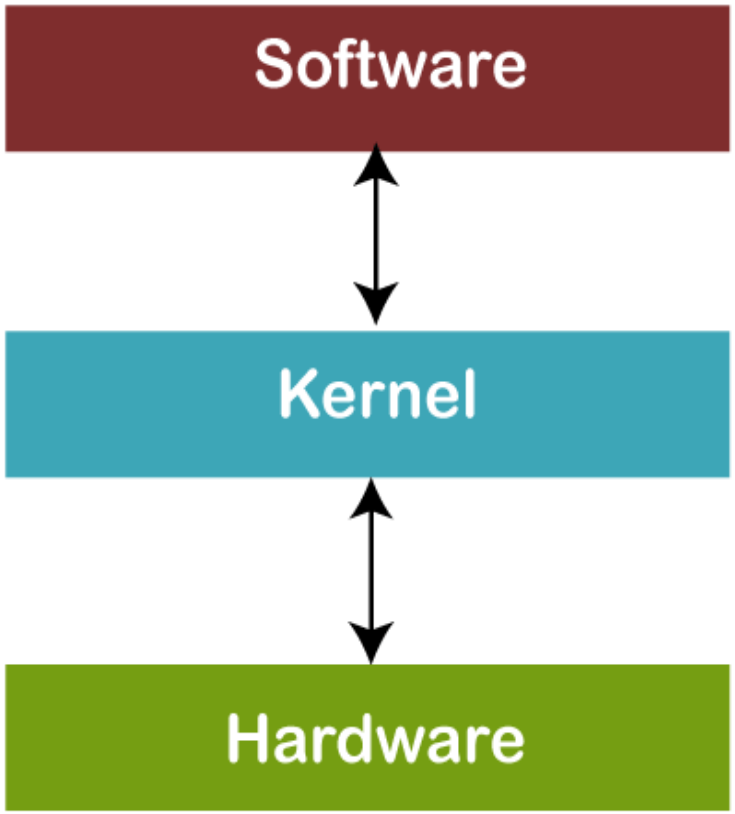
Types of Kernel

There are mainly five types of Kernel, which are given below:



Monolithic Kernels

In a monolithic kernel, the **same memory space is used to implement user services and kernel services.**



- It means, in this type of kernel, there is no different memory used for user services and kernel services. As it uses the same memory space, the size of the kernel increases, increasing the overall size of the OS. The execution of processes is also faster than other kernel types as it does not use separate user and kernel space.

Examples of Monolithic Kernels are **Unix, Linux, Open VMS, XTS-400, etc.**

Advantages:

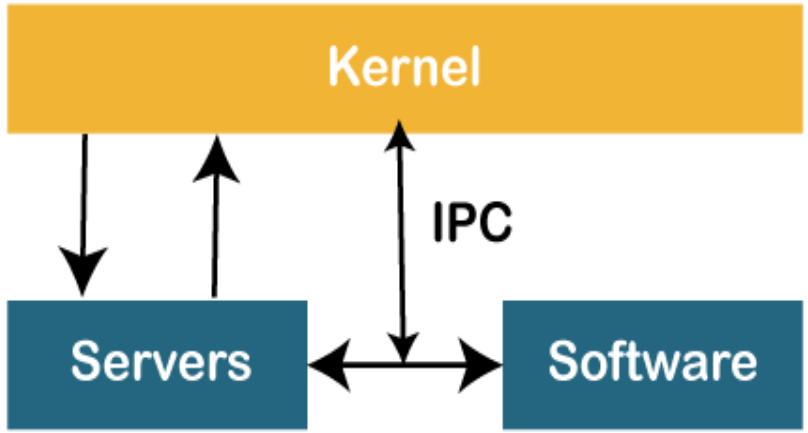
- The execution of processes is also faster as there is no separate user space and kernel space and less software involved.
- As it is a single piece of software hence, it's both sources and compiled forms are smaller.

Disadvantages:

- If any service generates any error, it may crash down the whole system.
- These kernels are not portable, which means for each new architecture, they must be rewritten.
- Large in size and hence become difficult to manage.
- To add a new service, the complete operating system needs to be modified.

Microkernel

- A microkernel is different from a traditional kernel or a Monolithic Kernel. It is sometimes written as K. In this, user services and kernel services are put into action in two different address spaces: user space and kernel space. Since it puts both services in different places, the size of the microkernel is smaller, which also makes the OS smaller.



- Compared to monolithic kernels, microkernels are easier to manage and keep up to date. Still, if there are more system calls and context switches, the system might not work as well because it will be slower. For handling requests from one server to another, these kernels use a message-passing system. Microkernels only do a few important things, like set up memory address spaces, IPC (Interprocess Communication), and manage processes. Kernel doesn't handle other services, like networking, which is done by a user-space program called servers.

Examples of Microkernel are **L4, AmigaOS, Minix, K42**, etc.

Advantages:

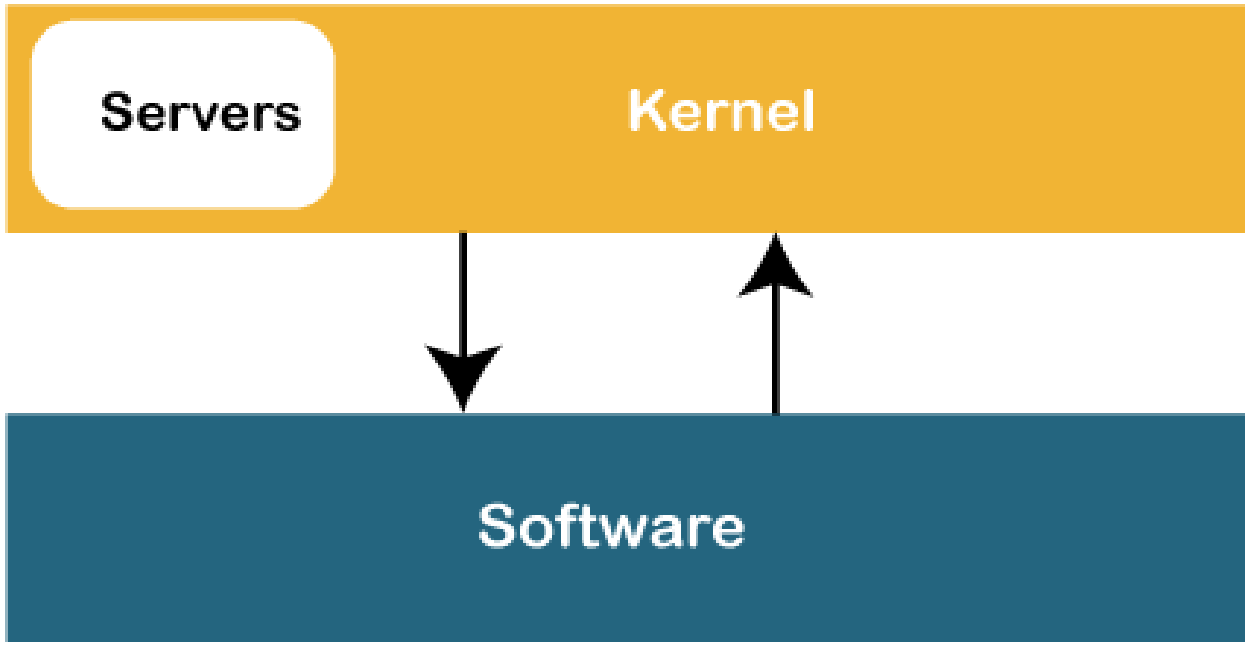
- Microkernels can be managed easily.
- A new service can be easily added without modifying the whole OS.
- In a microkernel, if a kernel process crashes, it is still possible to prevent the whole system from crashing.

Disadvantages:

- There is more requirement of software for interfacing, which reduces the system performance.
- Process management is very complicated.
- The messaging bugs are difficult to fix.

Hybrid Kernel

- Hybrid kernels are also called modular kernels. They are a mix of Monolithic kernels and Microkernels. It uses both the speed of monolithic kernels and the ability of microkernels to be split up into smaller parts.



- A hybrid kernel can be thought of as a version of a microkernel that is bigger and has more features from a monolithic kernel. Many commercial operating systems, like different versions of MS Windows, use these kernels. It is a lot like a microkernel, but it also has some extra code in the kernel space to improve the system's performance. Hybrid kernels let you run some services, like the network stack, in kernel space, which is slower than a traditional microkernel, but you can still run kernel code, like device drivers, as servers in user space.

Examples of Hybrid Kernel are Windows NT, Netware, BeOS, etc.

Advantages:

- There is no requirement for a reboot for testing.
- Third-party technology can be integrated rapidly.

Disadvantages:

- There is a possibility of more bugs with more interfaces to pass through.
- It can be a confusing task to maintain the modules for some administrators, especially when dealing with issues such as symbol differences.

Nanokernel

- As the name suggests, ***in Nanokernel, the complete code of the kernel is very small, which means the code executing in the privileged mode of the hardware is very small***. Here the term nano defines a kernel that supports a nanosecond clock resolution.

Examples of Nanokernel are EROS etc.

Advantages:

- It provides hardware abstractions even with a very small size.

Disadvantages:

- Nanokernel lacks system services.

Exokernel

- Exokernel is still developing and is the experimental approach for designing OS.

- This type of kernel is different from other kernels as in this; resource protection is kept separated from management, which allows us to perform application-specific customization.

Advantages:

- The exokernel-based system can incorporate multiple library operating systems. Each library exports a different API, such as one can be used for high-level UI development, and the other can be used for real-time control.

Disadvantages:

- The design of the exokernel is very complex.

Difference between Linux and Windows Operating Systems

The difference between Linux and Windows OS includes the following.

Linux Operating System	Windows Operating System
Linux is an open-source OS	Windows is not an open-source OS
The file name of Linux is case sensitive	The file name of Windows is case insensitive
It is free	It is commercial
In this OS, a monolithic kernel is used	In this OS, a microkernel is used
Linux is more efficient as compared to windows.	Windows is less efficient

Linux Operating System	Windows Operating System
To separate the directories, a forward slash is used	To separate the directories, the backslash is used
It is more secured	It is not secured as compared to Linux
Linux is extensively used to hack the systems	Windows do not offer much effectiveness in hacking.
Linux uses a hierarchical file system.	Windows uses several data drives namely C: D: E for the purpose of storing files as well as folders.
The considered files in Linux are CD-ROMs, hard drives, & printers	The considered devices in windows are Hard drives, printers, and CD-ROMs.

Linux Operating System	Windows Operating System
The user account types in Linux are 3 types like Regular, Root & Service Account	The user account types in Windows are four types like Administrator, Standard, Child, & Guest
The superuser like Root user of Linux includes all administrative human rights.	The administrator user of Windows includes all administrative human rights of computers.
The naming convention of Linux files is case-sensitive. So, two different files in this OS are sample & SAMPLE.	In Windows OS, you cannot have two files with the similar name within the same folder
For each user, his home directory is created like a home or username.	In windows OS, the default home directory is My Documents

Key Difference between Linux OS and Windows OS

- Linux is faster than Windows, even on older computers.
- In Linux, peripherals are like files, but in Windows, they are called devices.
- In Linux, files with the same name can be in the same folder, while in Windows, two files with the same name can't be in the same folder.
- In Linux, program files and the system are stored in different dictionaries. In Windows OS, program files and the system are usually stored in a C drive.

Key Difference between Linux OS and Windows OS

The key difference between Linux and Windows includes the following.

- Linux is an open-source OS, so the user can easily change the source code as needed. Windows, on the other hand, is a paid OS, so the user doesn't have access to the source code.
- Linux OS is more secure because it finds and fixes bugs. Windows OS, on the other hand, has a large number of users, which makes it a target for hackers.

THANK YOU

hitbullseye