

A person in a suit

Description automatically generated

ICL-1302 Computer Systems

TOBY BEEVERS

*Cambria ID: 21075206 Bangor ID: 500704883*

APPLIED DATA SCIENCE DEGREE APPRENTICESHIP BSC

Contents

[Introduction 2](#_Toc157856716)

[The Kernal 2](#_Toc157856717)

[The Operating System 2](#_Toc157856718)

[Input/Output (I/O) 2](#_Toc157856719)

[Memory 2](#_Toc157856720)

[Storage 2](#_Toc157856721)

[Processor 3](#_Toc157856722)

[Applications, Operating Systems & Device Drivers 3](#_Toc157856723)

[Conclusion 3](#_Toc157856724)

[Appendix 3](#_Toc157856725)

[Operating Systems 3](#_Toc157856726)

[Device Drivers 3](#_Toc157856727)

[Applications 4](#_Toc157856728)

[References 4](#_Toc157856729)

# Introduction

In Assignment Two we looked at various fundamental hardware components and how they differ across mobile and desktop technologies, in this assignment we will take a deeper look at each of the hardware components and how they interact with the Operating System (OS).

Computer systems use the OS as a bridge between the user and the computer hardware, at the centre of these operations is the Kernal which is an important part that communication between the OS and hardware components.

# The Kernal

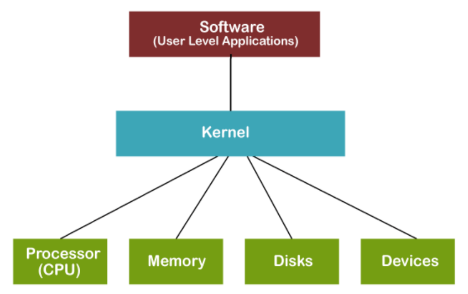


Figure 1: A basic diagram showing the Kernal layer and how it sits between the User Layer and the Hardware Layer.

*“In computer science, Kernel is a computer program that is a core or heart of an operating system.”* (Javatpoint, 2024)

The Kernal can be considered a translation layer that fundamentally manages the Read/Write of the I/O as well as being the building block of the OS to manage the CPU, GPU, Memory and other key components.

In his YouTube channel *SavvyNik* explains the *Kernal* and how it forms a building block in Linux, as well as explaining how people often confuse the terms Linux and Linux Distribution in the same context. *SavvyNik* considers the *Kernal* as part of Linux which is use along with other programs to create Linux Distributions (SavvyNik, 2022)*.*

(Hudek, et al., 2024)

# The Operating System

The OS is a piece of software that loads when the computer boots, the purpose of an OS is to control the hardware components. The OS uses various Manager like;

* The **device manager** which allocates resources to external hardware devices and allows them to be used by applications.
* The **memory manage**r which controls the allocation of memory between applications.
* The **file manage**r which controls the opening, reading, and writing of files in storage.
* And the **process manager** controls the allocation of central processing unit (CPU) cycles to multiple running applications.

### Input/Output (I/O)

Input/Output devices primarily use the Device Manager to facilitates communication between peripherals and the CPU, managing input from devices such as keyboards, cameras, and output to devices like monitors, speakers.

### Memory

Memory Manager is used to allocate and deallocate memory space, managing virtual memory, and ensuring efficient use of RAM.

### Storage

File manager organises and manages data on storage devices, handling file access, and ensuring data integrity.

### Processor

Processing primarily uses the Task Manager to allocate CPU resources to processes, managing task priority, and ensuring fair use of the CPU.

# Applications, Operating Systems & Device Drivers

In his 2022 blog, Jack Denver makes a clear differentiation between Driver Software and Application Software stating.

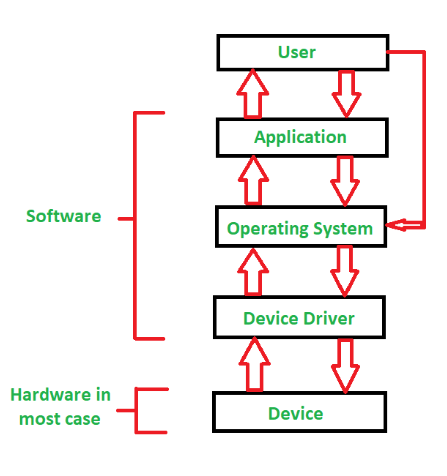


Figure 2: Illustration of the interaction between the user, OS, Device driver, and the devices.

“*Application software is a type of software that helps users solve a particular problem or perform a specific task. Driver software is a type of software that accompanies a hardware component and helps it communicate with the rest of the computer system*” (Denver, 2022)*.*

Making this clear differentiation helps to define how a user logged into a PC is using Applications which form part of the Operating System, working with the Kernal to translate via the Driver Software to the hardware components.

# Conclusion

In conclusion, this assignment provides a comprehensive exploration of the intricate relationship between computer hardware components and the operating system. The Kernel, often likened to a translation layer, serves as the link for communication between the OS and critical hardware components. The OS itself functions as a vital piece of software, seamlessly loading during the computer's boot process to exert control over hardware components.

Managers within the OS, such as the device manager, memory manager, file manager, and process manager, work cohesively to allocate resources, control memory, manage files, and oversee CPU cycles, respectively.

To distinguish further, the conclusion draws from Jack Denver's insights, highlighting the clear differentiation between application software and driver software. Applications, integral to the OS, are user-oriented and task-specific, while driver software accompanies hardware components, aiding communication within the computer system. This demarcation reinforces the understanding of how users interact with applications, operating systems, and, crucially, driver software to facilitate communication with hardware components.

# Appendix

### Operating Systems

Software that manages hardware resources and provides essential services to other software applications. This serves as an intermediary between the user and hardware, ensuring the smooth functioning of the computer. E*xample, Windows, Linux, macOS.*

### Device Drivers

Software that facilitates communication between the operating system and specific hardware devices. Drivers enable the OS to recognize, control, and interact with hardware components. *Example, NVIDIA graphics drivers, Realtek audio drivers.*

### Applications

Applications are user-facing programs designed for specific tasks, such as word processors, web browsers, or games. They interact directly with users to provide functionality and achieve specific goals. *Examples, Microsoft Word, Google Chrome, Photoshop.*

# References

Denver, J., 2022. *Difference Between Driver Software & Application Software.* [Online]   
Available at: https://jackdenverblog.medium.com/difference-between-driver-software-application-software-3463fe321d8e#:~:text=Application%20software%20is%20a%20software,different%20hardware%20of%20the%20computer.  
[Accessed 03 02 2024].

Hudek, T., Graff, E. & Sherer, T., 2024. *Kernel-Mode Driver Architecture Design Guide.* [Online]   
Available at: https://learn.microsoft.com/en-us/windows-hardware/drivers/kernel/  
[Accessed 03 02 2024].

Javatpoint, 2024. *What is Kernel?.* [Online]   
Available at: https://www.javatpoint.com/what-is-kernel  
[Accessed 03 02 2024].

SavvyNik, 2022. *What is a Kernel and what does it do? Explore the Kernels of Linux, Windows, and MacOS..* [Online]   
Available at: https://youtu.be/IvGdY6luTtU  
[Accessed 03 02 2024].