Kernolology

Memory Management Process (Part 1)

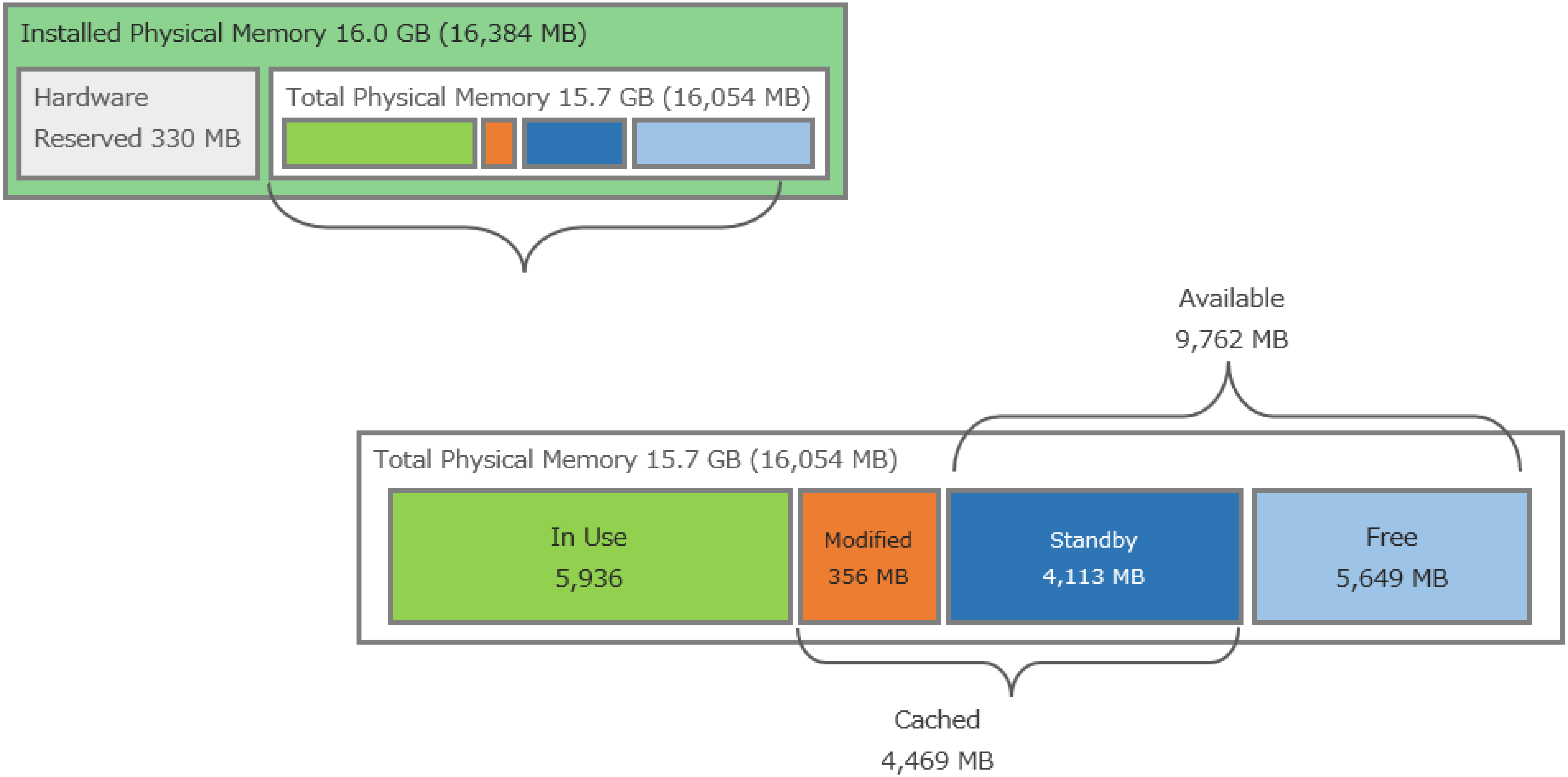
M. Moslemi AbarGhan


Physical Memory

Concept: What is the Physical Memory?

Physical Memory in the Linux Kernel refers to the **Actual Hardware Memory** (**RAM**). It is a **Finite Resource** that the operating system manages directly. Unlike virtual memory, which provides an abstraction to processes, Physical Memory represents the real storage locations where data resides. It refers to the **Management** and **Utilization** of the **actual RAM hardware**.

Concept: What is the Physical Memory?

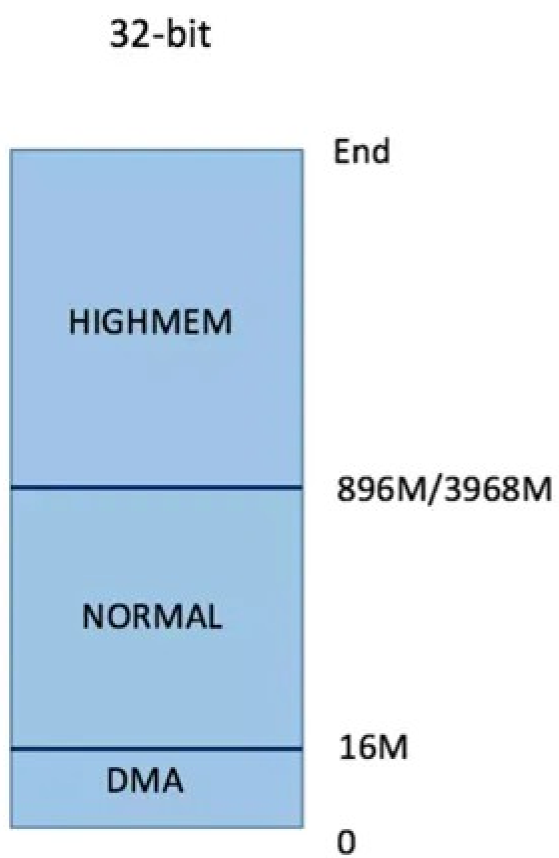


Structure: Memory Zones of Physical

- **Physical Memory** is divided into different zones based on **Accessibility** and **Usage**:

1) **ZONE_DMA:** Memory within this zone can be used for **Direct Memory Access** (**DMA**) operations by devices that have restrictions on the physical addresses they can use.
2) **ZONE_NORMAL:** This zone is where most of the **Kernel's Regular Memory Allocations** occur.
3) **ZONE_HIGHMEM:** Present in 32-Bit Systems, it represents memory that is not permanently mapped into the kernel's address space due to limited addressable space. The kernel must temporarily map high memory to access it.
4) **ZONE_MOVABLE:** Used for memory that **can be migrated to other physical locations**, aiding in defragmentation.

Structure: Memory Zones of Physical



Function：Management and Usage

1) **Kernel vs. User Space：** The Kernel Manages Physical Memory Allocations for both the OS and User Applications. The memory is allocated to user processes as needed and protected to **ensure** that processes **do not interfere with each other**.

2) **Efficiency and Performance：Efficient Management** of Physical Memory is **Crucial** for **System Performance**. The kernel uses various Algorithms and Data Structures to keep track of **Free** & **Used** memory, optimize **Memory Allocation**, and manage **the Swapping of Pages** to and from **Disk**.

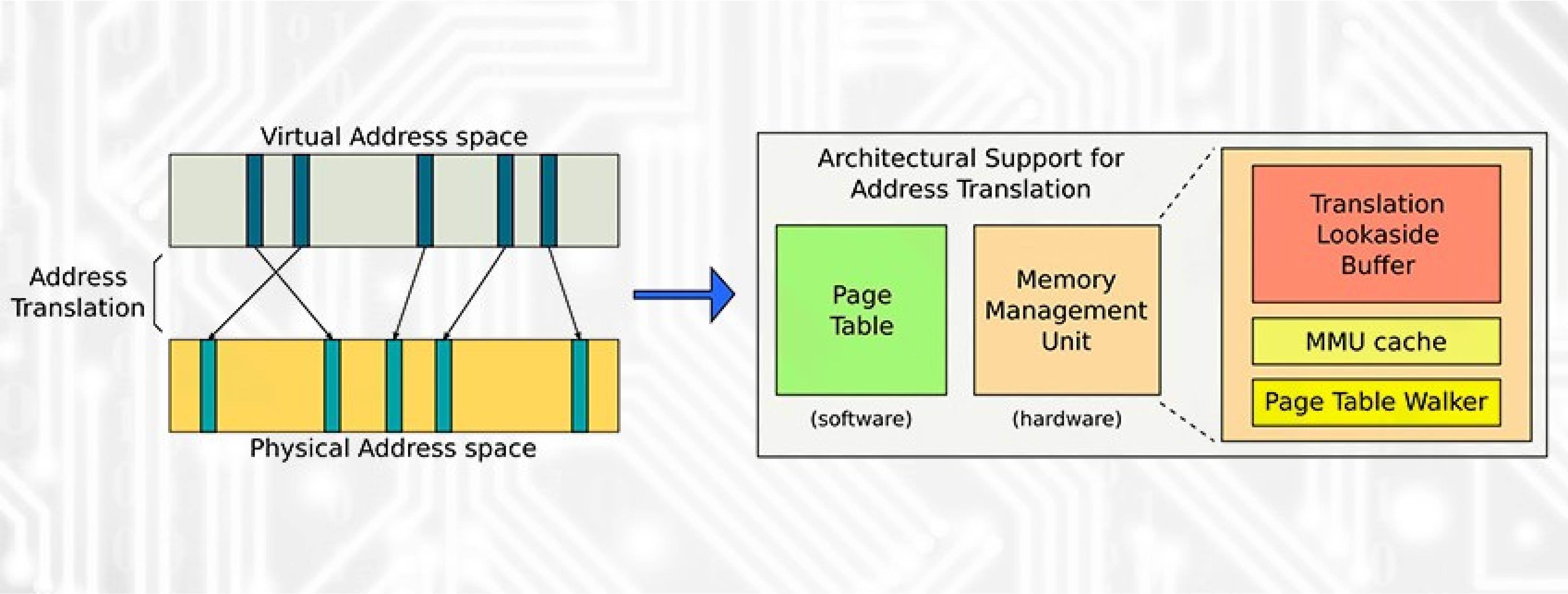
3) **Security：** By **Controlling Access** to Physical Memory, the kernel helps **maintain System Security**, ensuring that **Sensitive Data** is protected from **Unauthorized Access.**

Virtual Memory

Concept: What is the Virtual Memory?

The Linux Kernel's implementation of **Virtual Memory** involves a complex interplay of **Data Structures**, **Hardware Support**, and **Sophisticated Algorithms.** These components work together to provide a **robust**, **efficient**, and **Secure Memory Management System**. all working together to manage the translation between **Virtual and Physical Addresses**, ensure **Memory Protection**, and optimize **Memory Usage**.

Concept: What is the Virtual Memory?



Structure: Virtual Address Space

- This space is divided into **User Space** (where user applications run) and **Kernel Space** (used by the Kernel). The **Separation** ensures process **Isolation and Protection**.

1) **User Space** consists of:

A) **Text Segmen**t**:** Contains the **executable code of a process**.

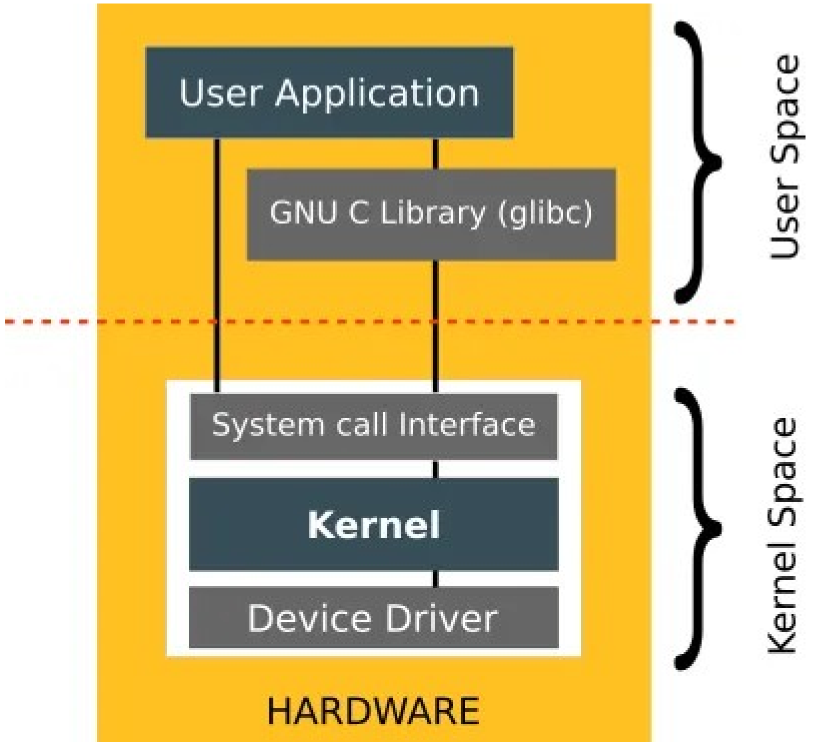
B) **Data Segment:** Contains **global and static variables**.

C) **Heap: Dynamic Memory** allocated during program execution (e.g., via malloc).

D) **Stack:** Memory used for **function call management**, including **local variables** and **return addresses**.

E) **Memory Mapped Regions:** Used for memory-mapped files and shared libraries.

Structure: Virtual Address Space



Structure: Virtual Address Space

- This space is divided into **User Space** (where user applications run) and **Kernel Space** (used by the kernel). The **Separation** ensures process **Isolation and Protection**.

2) **Kernel Space** consists of:

A) **Kernel Code and Data**: The Kernel's own **Code and Global Data Structures**.

B) **Kernel Heap**: Dynamically allocated memory for **Kernel Operations**.

C) **Memory Mapped I/O**: Regions mapped to **device Memory or Hardware Registers**.

Function：Wide range of benefits!

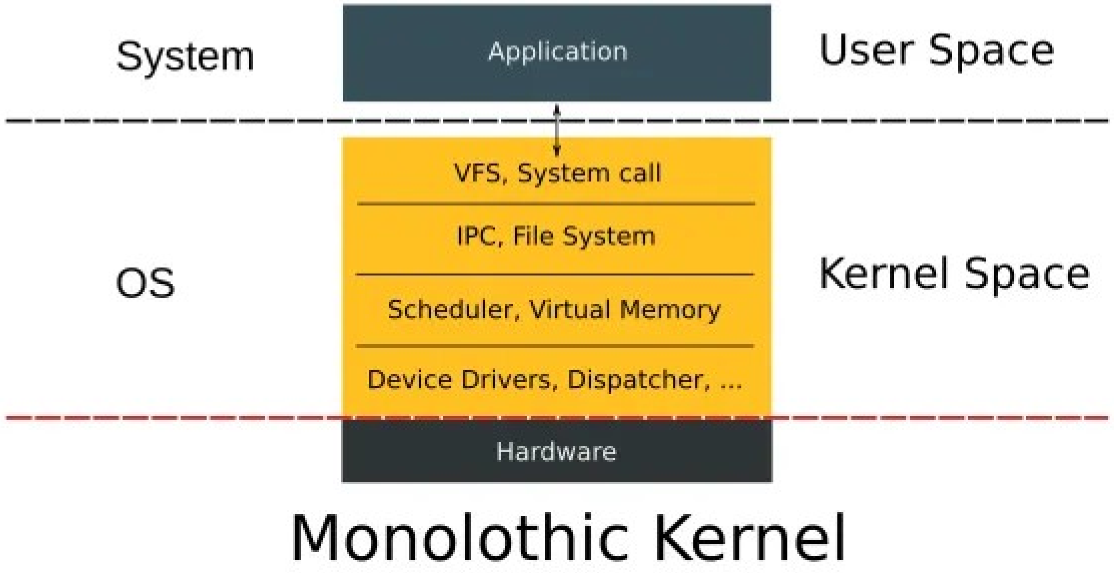
- Virtual Memory in the Linux Kernel serves several **Critical Functions**, providing a wide range of benefits that enhance system **Performance**, **Stability**, **Security**：

1) **Process Isolation and Security：** Virtual Memory ensures that each process operates in its own **Isolated Address Space.**

2) **Abstraction and Simplification：** Virtual Memory provides an **Abstraction Layer** that simplifies **Programming** and **System Design**.

3) **Memory Protection and Access Control：** Virtual Memory allows the Kernel to enforce **Access Control Policies** on **Memory Regions**.

Structure: Virtual Address Space



Function: Wide range of benefits!

- Virtual Memory in the Linux Kernel serves several Critical Functions, providing a wide range of benefits that enhance system Performance, Stability, Security:

4) **Address Space Layout Randomization (ASLR):** Virtual Memory supports security features like **ASLR**, which randomizes the Starting **Addresses of key Data Areas** (such as the Stack, Heap, and Loaded Libraries) for each process.

5) **Efficient Use of Physical Memory:** The virtual memory system enables more efficient use of Physical Memory (RAM) through mechanisms like **Paging** and **Swapping**.

References

- https://www.wikipedia.org/
- https://developer.ibm.com/articles/l-kernel-memory-access/
- https://codeahoy.com/learn/computersos/ch4/
- https://users.cs.fiu.edu/~cpoellab/teaching/cop4610_fall22/project4.html
- https://www.spiceworks.com/tech/devops/articles/what-is-virtual-memory/