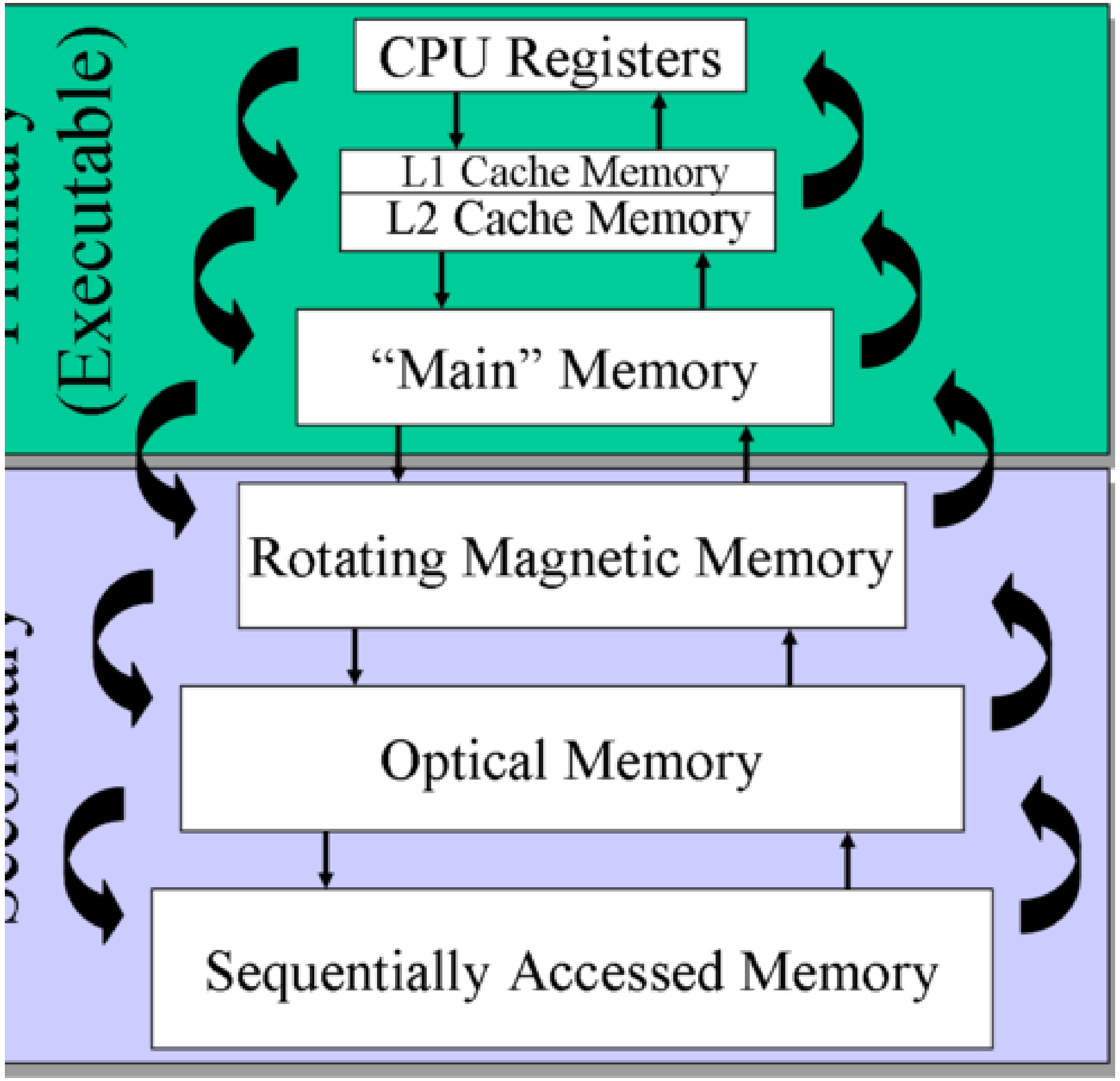
Memory Management Concepts

Bhupendra Pratap Singh

Memory Management – Hierarchal view

- Computer’s main memory – RAM.
- CPU cache is divided into three

main ‘Levels’, L1, L2, and L3.



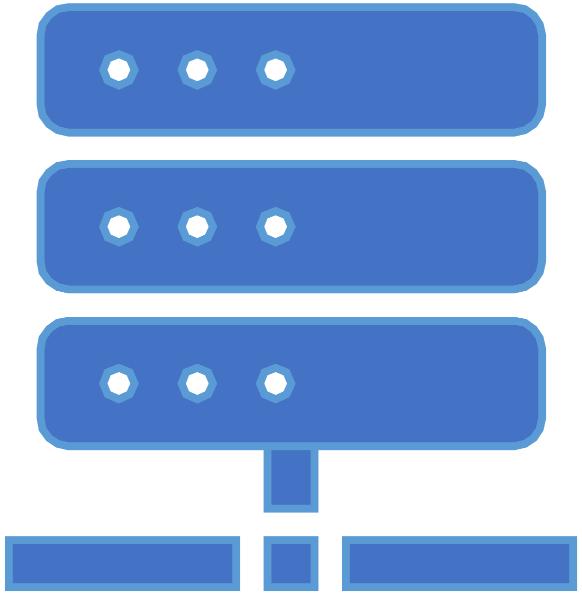
Cache Memory

L1 (Level 1) cache is the fastest memory that is present in a computer system. In terms of priority of access, L1 cache has the data the CPU is most likely to need while completing a certain task.

The L1 cache typically goes up to 256KB. However, some powerful CPUs are now taking it close to 1MB. Some server chipsets (like Intel's top-end Xeon CPUs) now have somewhere between 1-2MB of L1 cache.

Cache Memory – L2

L2 (Level 2) cache is slower than L1 cache, but bigger in size. Its size typically varies between 256KB to 8MB, although the newer, powerful CPUs tend to go past that. L2 cache holds data that is likely to be accessed by the CPU next. In most modern CPUs, the L1 and L2 caches are present on the CPU cores themselves, with each core getting its own cache.



Cache Memory (L3)

L3 (Level 3) cache is the largest cache memory unit, and also the slowest one. It can range between 4MB to upwards of 50MB. Modern CPUs have dedicated space on the CPU die for the L3 cache, and it takes up a large chunk of the space.

Cache Memory

- The data flows from the RAM to the L3 cache, then the L2, and finally L1. When the processor is looking for data to carry out an operation, it first tries to find it in the L1 cache. If the CPU is able to find it, the condition is called a **cache hit**. It then proceeds to find it in L2, and then L3.
- If it doesn't find the data, it tries to access it from the main memory. This is called a cache miss.

Memory Management Unit (MMU)

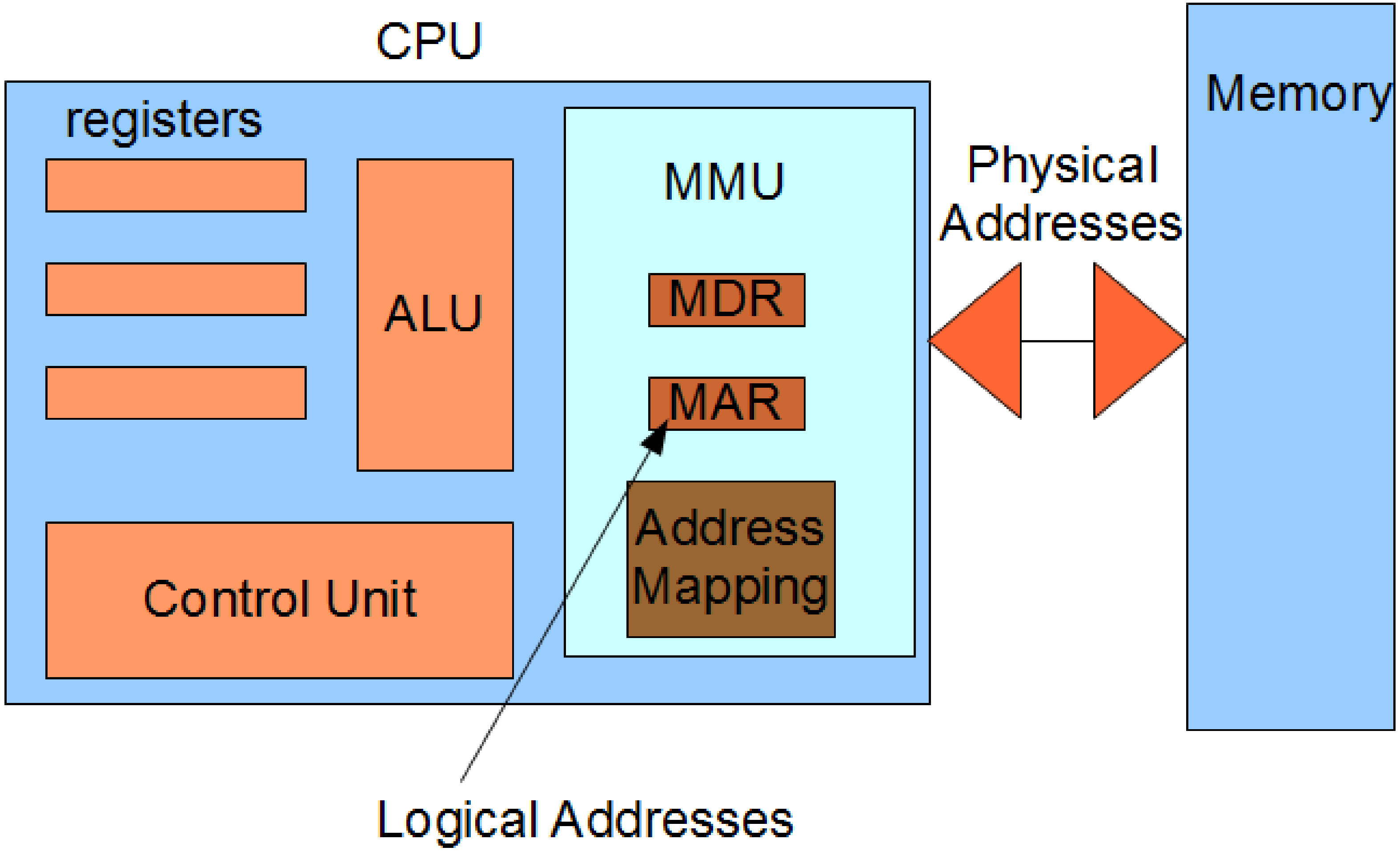
As a program runs, the memory addresses that it uses to reference its data is the logical address. The real time translation to the physical address is performed in hardware by the CPU's Memory Management Unit (MMU).

The MMU has two special registers, accessed by the CPU's control unit.

A data to be sent to main memory or retrieved from memory is stored in the *Memory Data Register* (MDR).

The desired logical memory address is stored in the *Memory Address Register* (MAR). The address translation is also called address binding and uses a memory map that is programmed by the operating **system(mmap-Posix standard).**

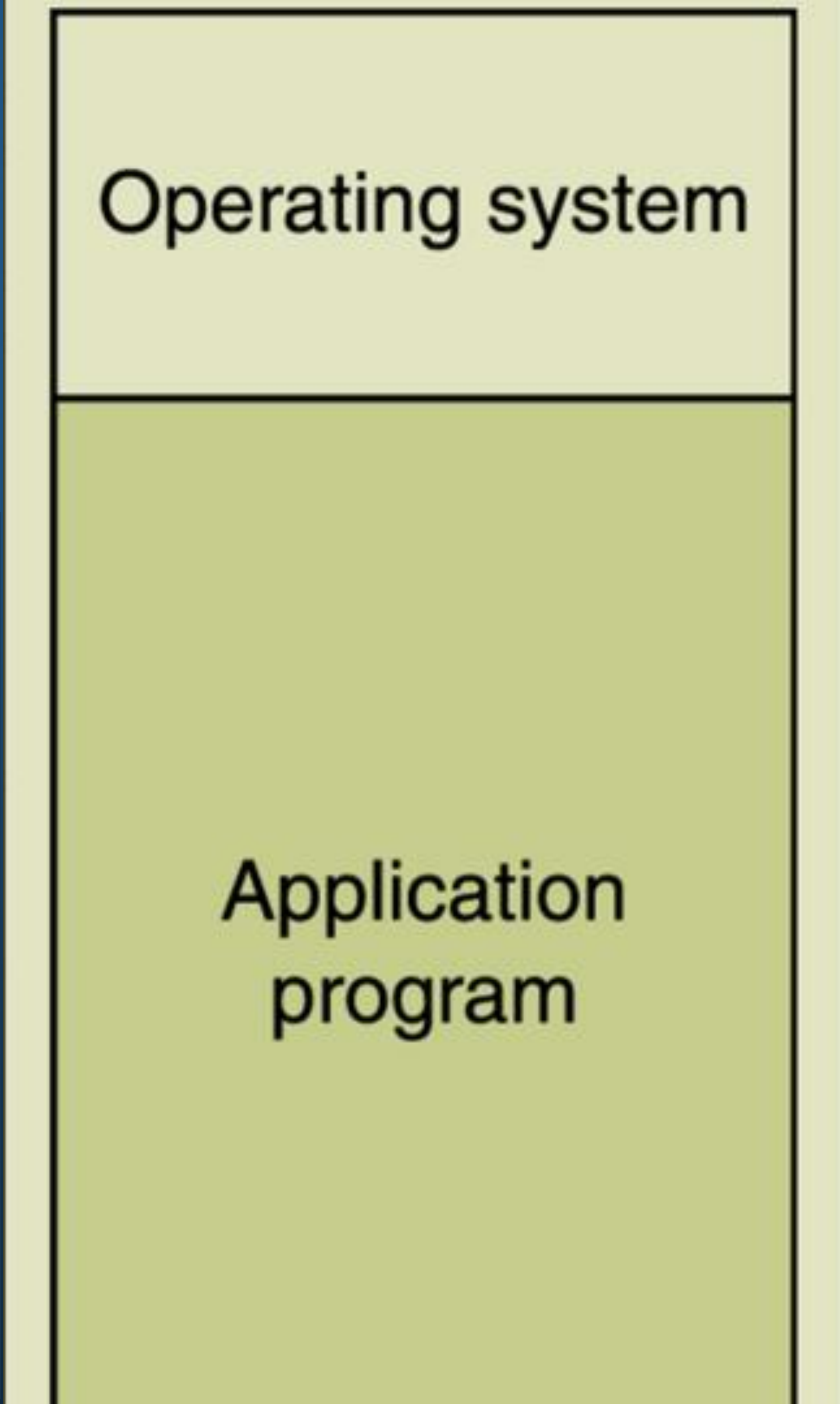
Memory Management Unit (MMU)



Before memory addresses are loaded on to the system bus, they are translated to physical addresses by the MMU.

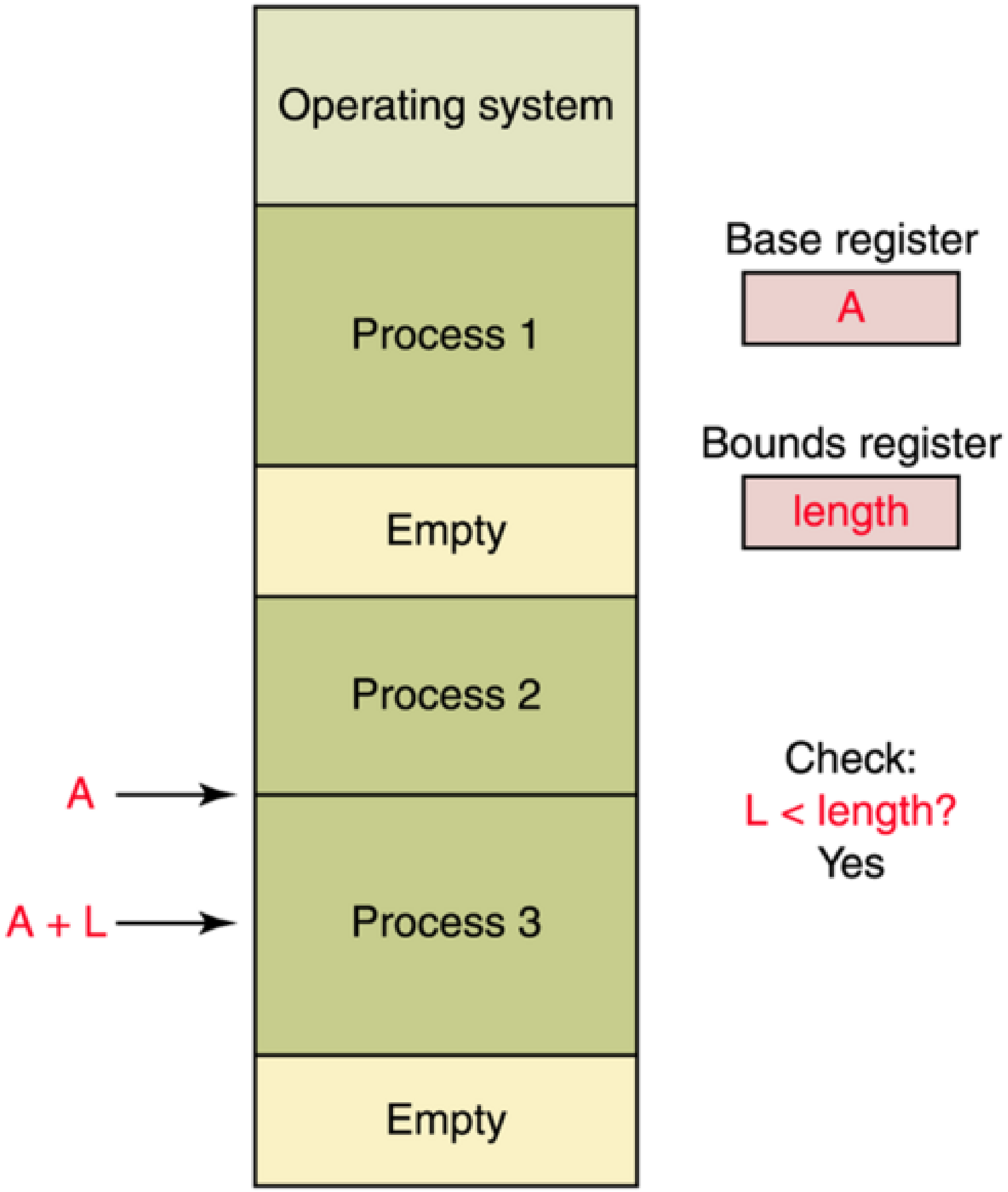
Single Contiguous allocation

Legacy Memory Management Techniques



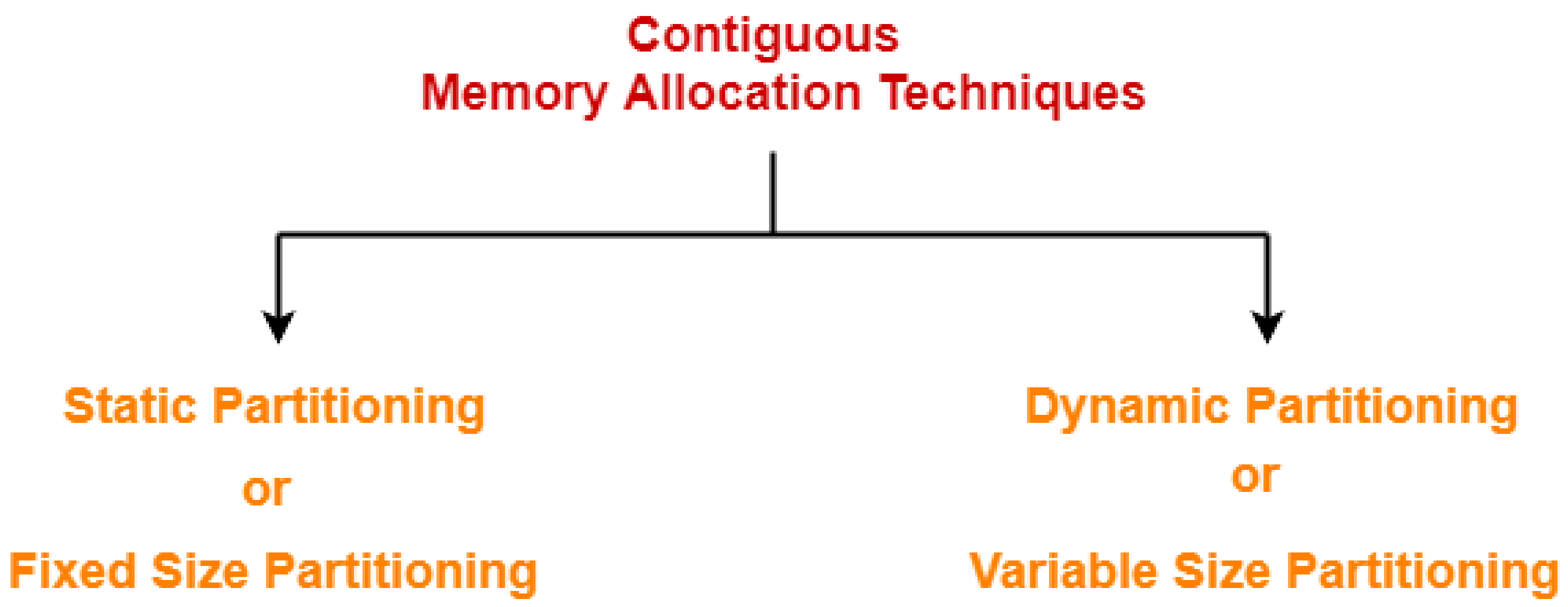
Legacy Memory Management Techniques

- Partitioned Allocation



Contiguous Memory Allocation

There are two popular techniques used for contiguous memory allocation-



Contiguous Memory Allocation Techniques

Static Partitioning
or
Fixed Size Partitioning

Dynamic Partitioning
or
Variable Size Partitioning

Contiguous Memory Allocation

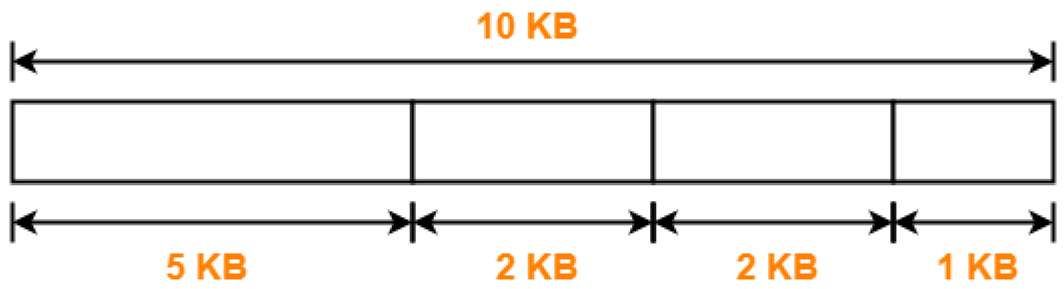
Static Partitioning-

Static partitioning is a fixed size partitioning scheme.

In this technique, main memory is pre-divided into fixed size partitions.

The size of each partition is fixed and can not be changed.

Each partition is allowed to store only one process.



Memory

Contiguous Memory Allocation

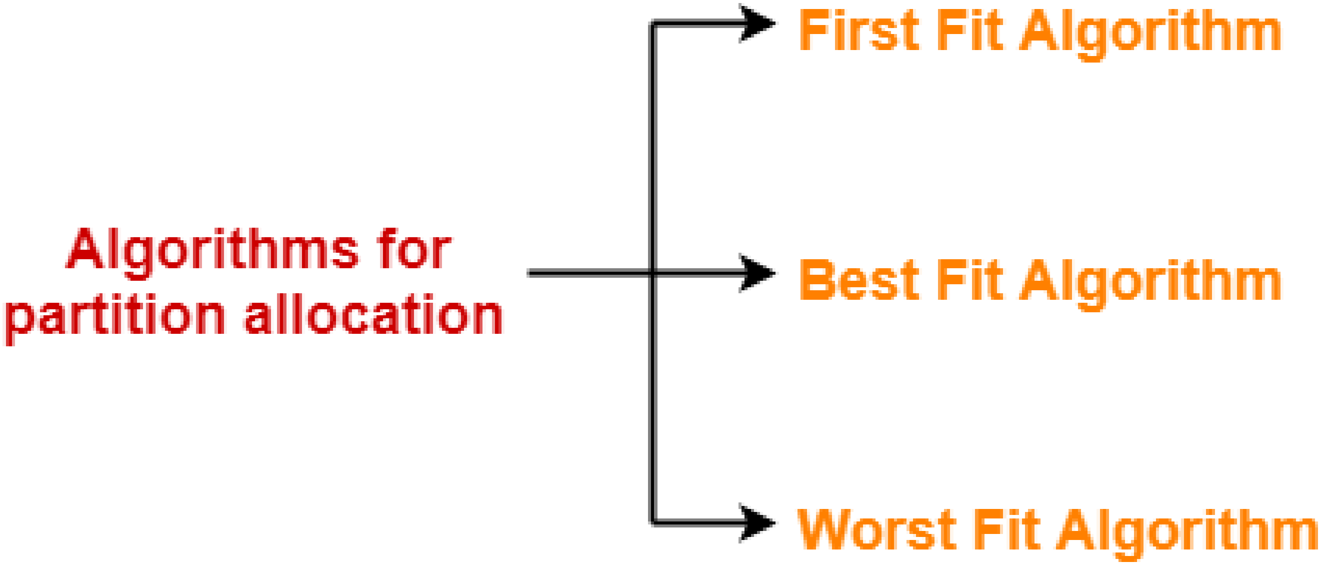
- Dynamic partitioning is a variable size partitioning scheme.
- It performs the allocation dynamically.
- When a process arrives, a partition of size equal to the size of process is created.

Then, that partition is allocated to the process.

Algorithm:

- The processes arrive and leave the main memory.
- As a result, holes of different size are created in the main memory.
- These holes are allocated to the processes that arrive in future.

Algorithms for Partition Allocation

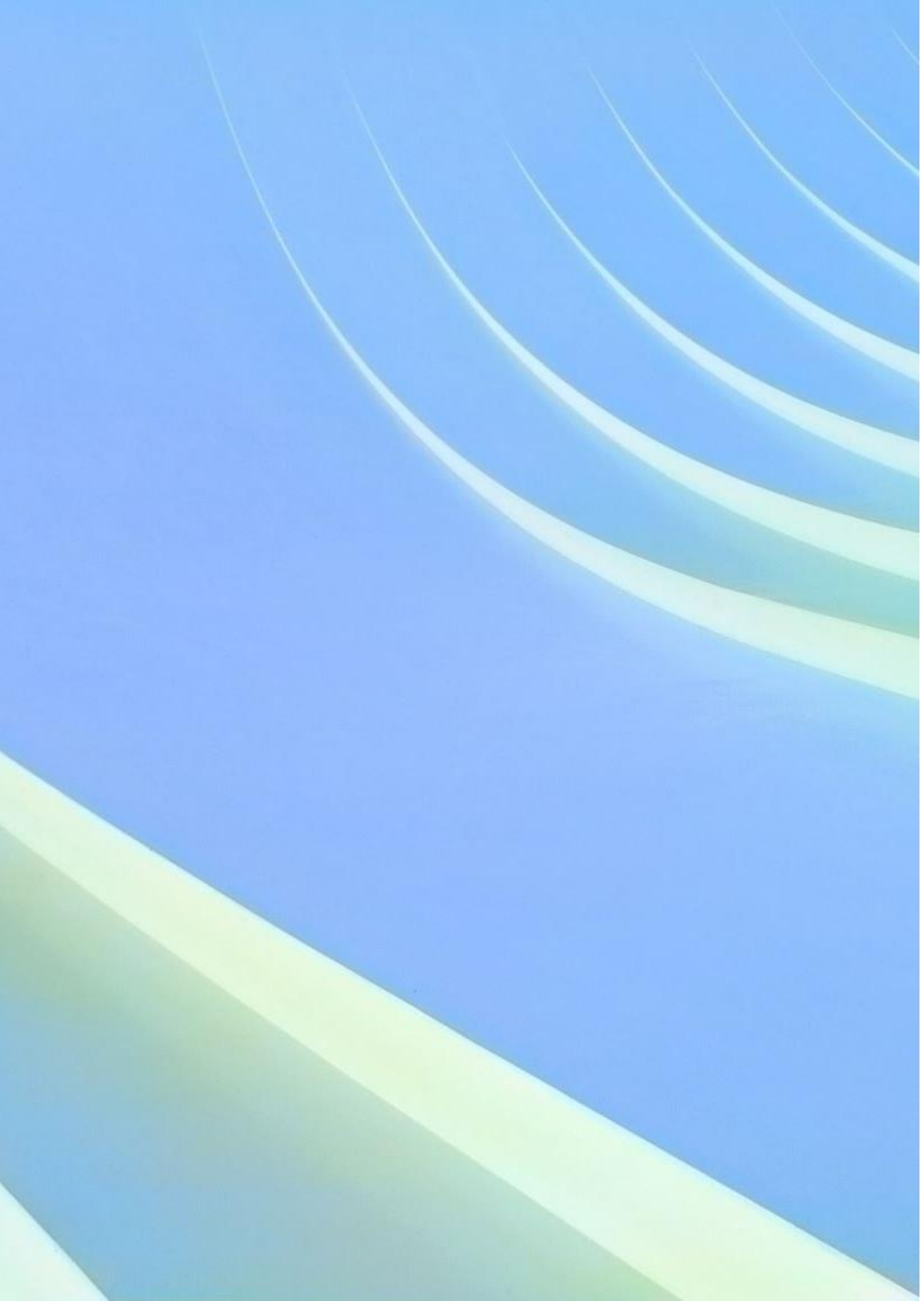


Algorithms for partition allocation

First Fit Algorithm

Best Fit Algorithm

Worst Fit Algorithm



First Fit Algorithm

THIS ALGORITHM STARTS SCANNING THE PARTITIONS SERIALLY FROM THE STARTING.

WHEN AN EMPTY PARTITION THAT IS BIG ENOUGH TO STORE THE PROCESS IS FOUND, IT IS ALLOCATED TO THE PROCESS.

OBVIOUSLY, THE PARTITION SIZE MUST BE GREATER THAN OR AT LEAST EQUAL TO THE PROCESS SIZE.

Best Fit Algorithm & Worst Fit Algorithm

Best Fit Algorithm

- This algorithm first scans all the empty partitions & then allocates the smallest size partition to the process.

Worst Fit Algorithm-

- This algorithm first scans all the empty partitions & then allocates the largest size partition to the process.

Key Points For static partitioning

Best Fit Algorithm works best, because space left after the allocation inside the partition is of very small size, hence internal fragmentation is least.

Worst Fit Algorithm works worst, because space left after the allocation inside the partition is of very large size, hence internal fragmentation is maximum.

Key Points For dynamic partitioning

Worst Fit Algorithm works best.

This is because space left after allocation inside the partition is of large size.

There is a high probability that this space might suit the requirement of arriving processes.

Best Fit Algorithm works worst.

This is because space left after allocation inside the partition is of very small size.

There is a low probability that this space might suit the requirement of arriving processes.

Internal Fragmentation

It occurs when the space is left inside the partition after allocating the partition to a process.

This space is called as internally fragmented space.

This space can not be allocated to any other process.

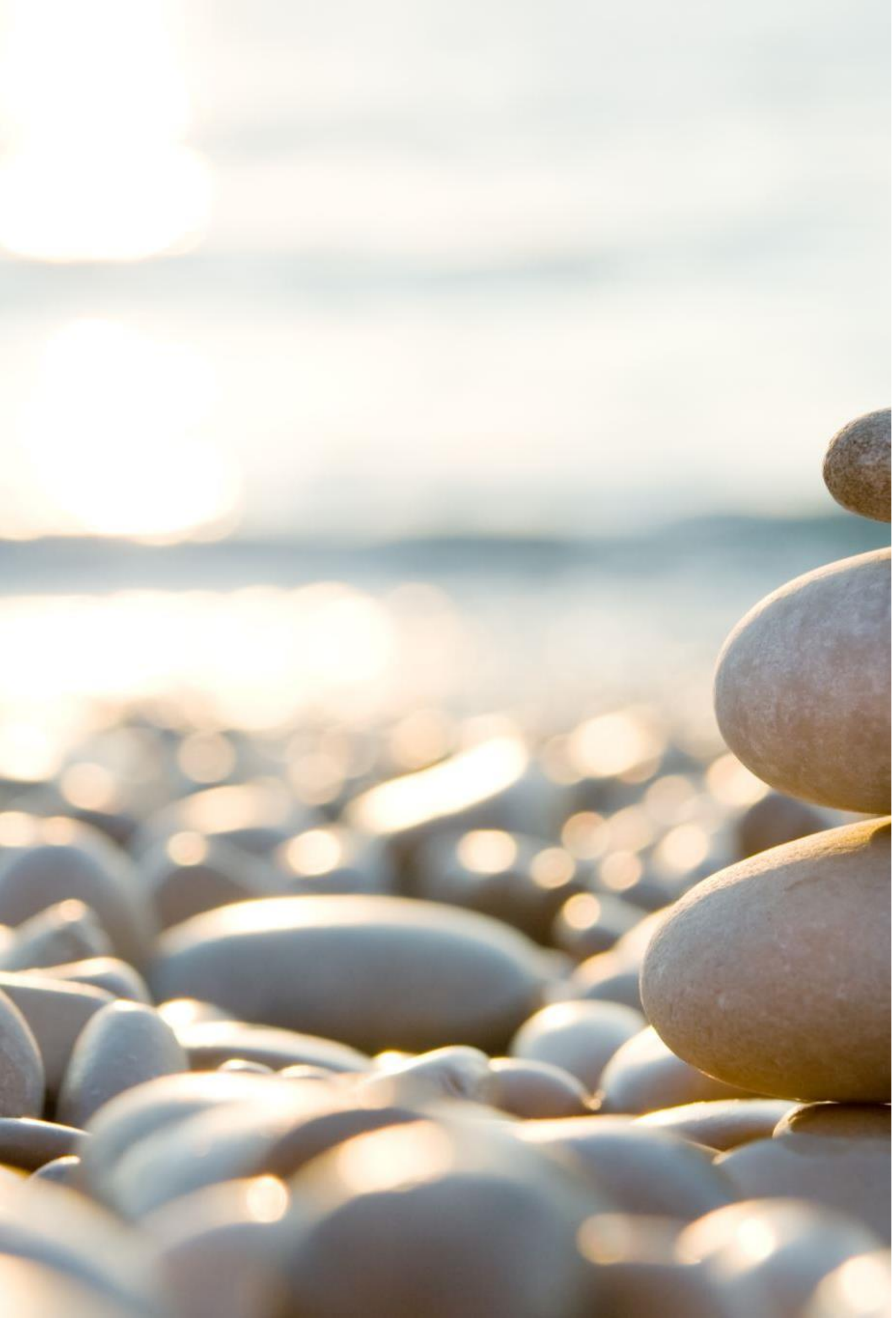
This is because only static partitioning allows to store only one process in each partition.

Internal Fragmentation occurs only in static partitioning.

External Fragmentation

IT OCCURS WHEN THE TOTAL AMOUNT OF EMPTY SPACE REQUIRED TO STORE THE PROCESS IS AVAILABLE IN THE MAIN MEMORY.

BUT BECAUSE THE SPACE IS NOT CONTIGUOUS, SO THE PROCESS CAN NOT BE STORED.



Non-Contiguous Memory allocation

- Non-contiguous memory allocation is a memory allocation technique.
- It allows to store parts of a single process in a non-contiguous fashion.
- Different parts of the same process can be stored at different places in the main memory.
- There are two popular techniques used for non-contiguous memory allocation-

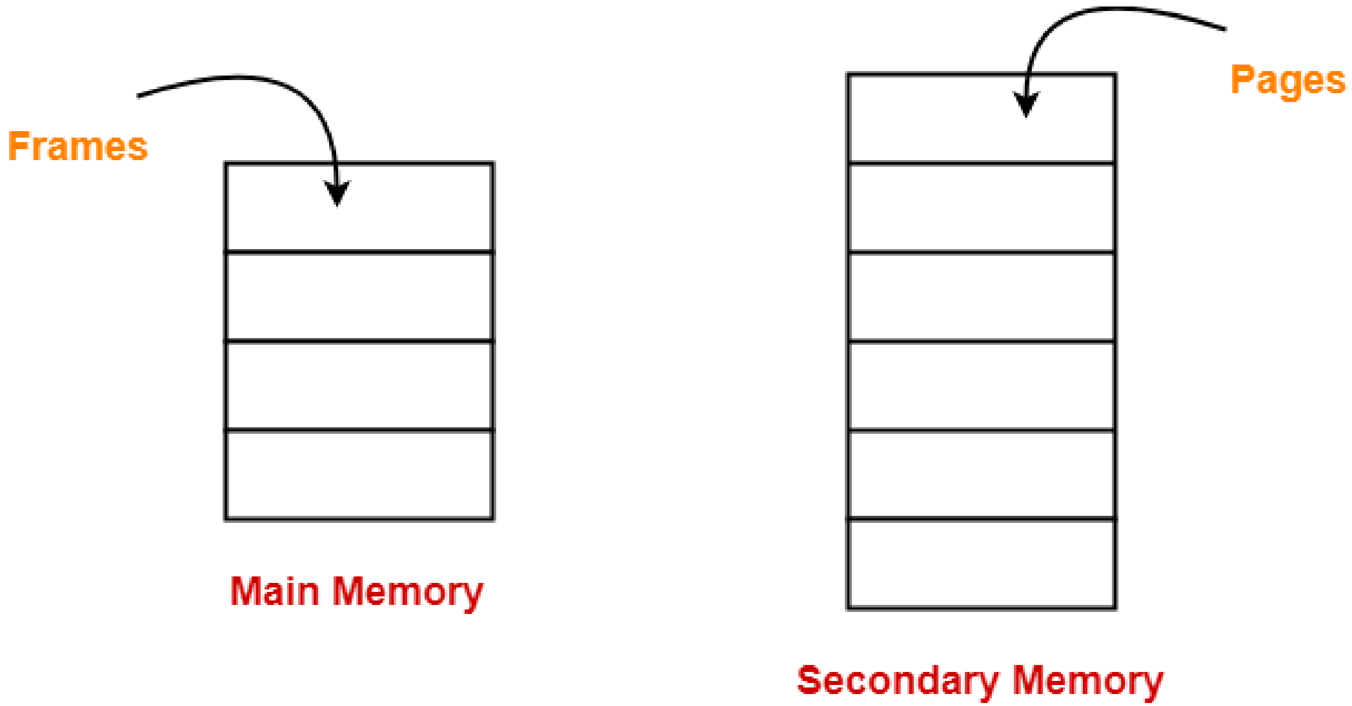
i. Paging

ii. Segmentation

Paging

- Paging is a fixed size partitioning scheme.
- In paging, secondary memory and main memory are divided into equal fixed size partitions.
- The partitions of secondary memory are called as pages.
- The partitions of main memory are called as frames.
- Each process is divided into parts where size of each part is same as page size.

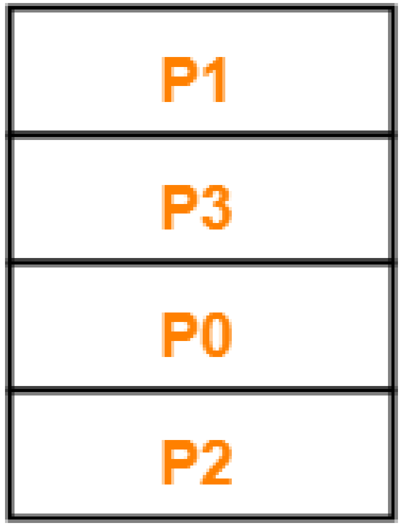
Paging cont. ..



- The size of the last part may be less than the page size.
- The pages of process are stored in the frames of main memory depending upon their availability.

Paging

- Consider a process is divided into 4 pages P0, P1, P2 and P3.
- Depending upon the availability, these pages may be stored in the main memory frames in a non-contiguous fashion as shown-



P1

P3

P0

P2

Main Memory

Translating Logical Address into Physical Address-

CPU always generates a logical address.

A physical address is needed to access the main memory.

Following steps are followed to translate logical address into physical address-

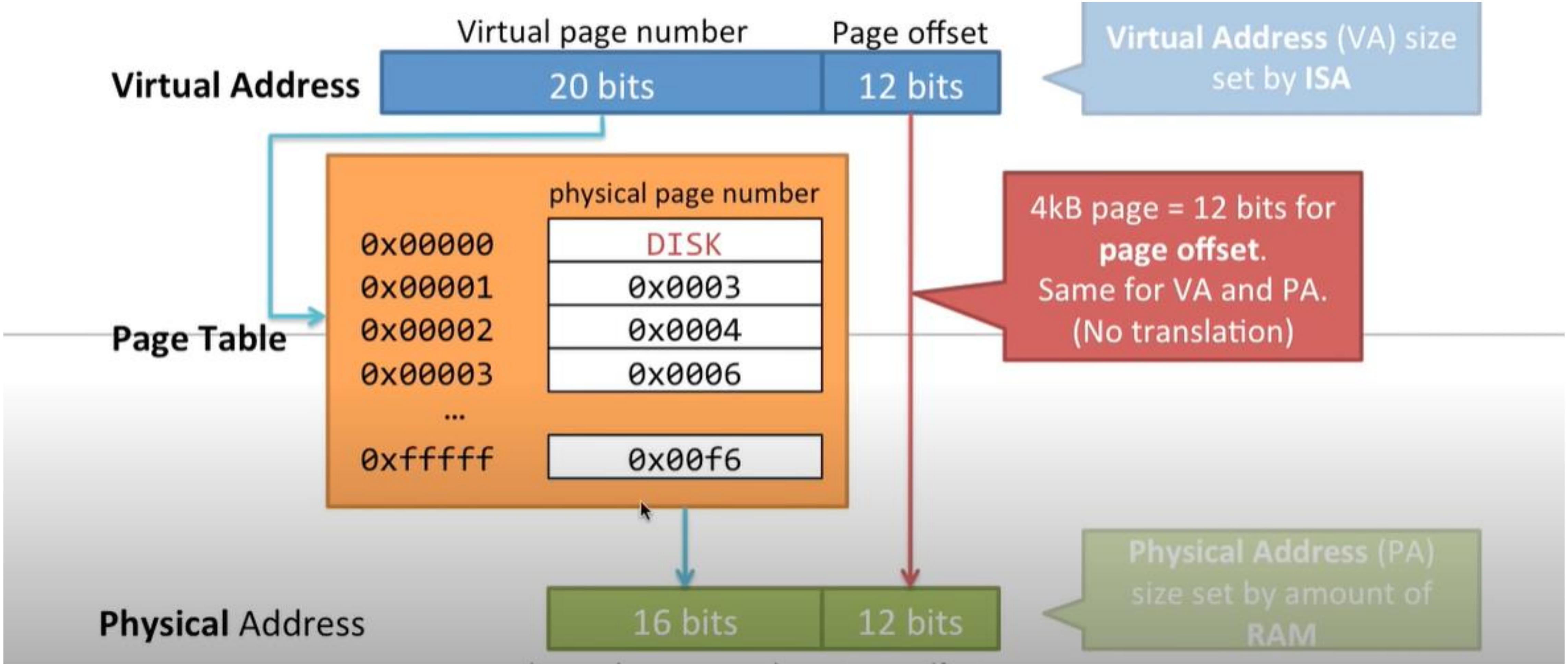
Step-01:

CPU generates a logical address consisting of two parts-

Page Number

Page Offset

How to Do a Page Table Look UP?



	physical page number
0x00000	DISK
0x00001	0x0003
0x00002	0x0004
0x00003	0x0006
…	
0xfffff	0x00f6

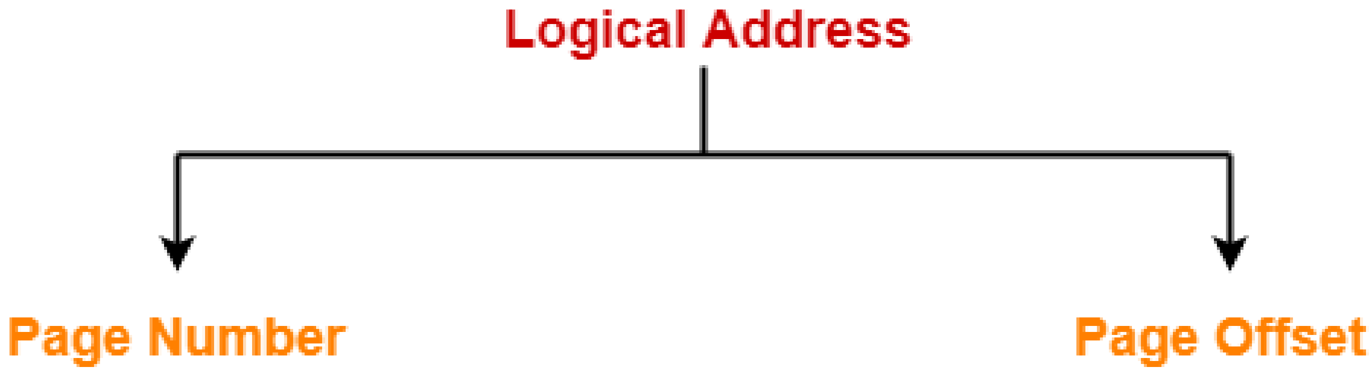
Virtual Address (VA) size set by **ISA**

4kB page = 12 bits for **page offset**. Same for VA and PA. (No translation)

Physical Address (PA) size set by amount of **RAM**

Physical Page number | Page Offset

Translating Logical Address into Physical Address-



- Page Number specifies the specific page of the process from which CPU wants to read the data.
- Page Offset specifies the specific word on the page that CPU wants to read.

Translating Logical Address into Physical Address

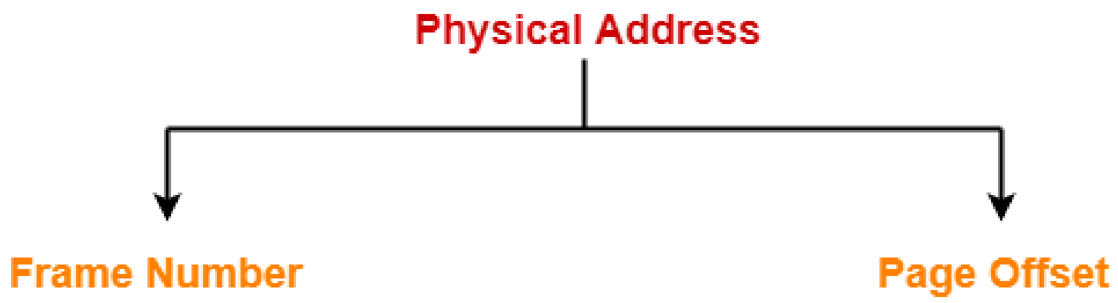
Step-02:

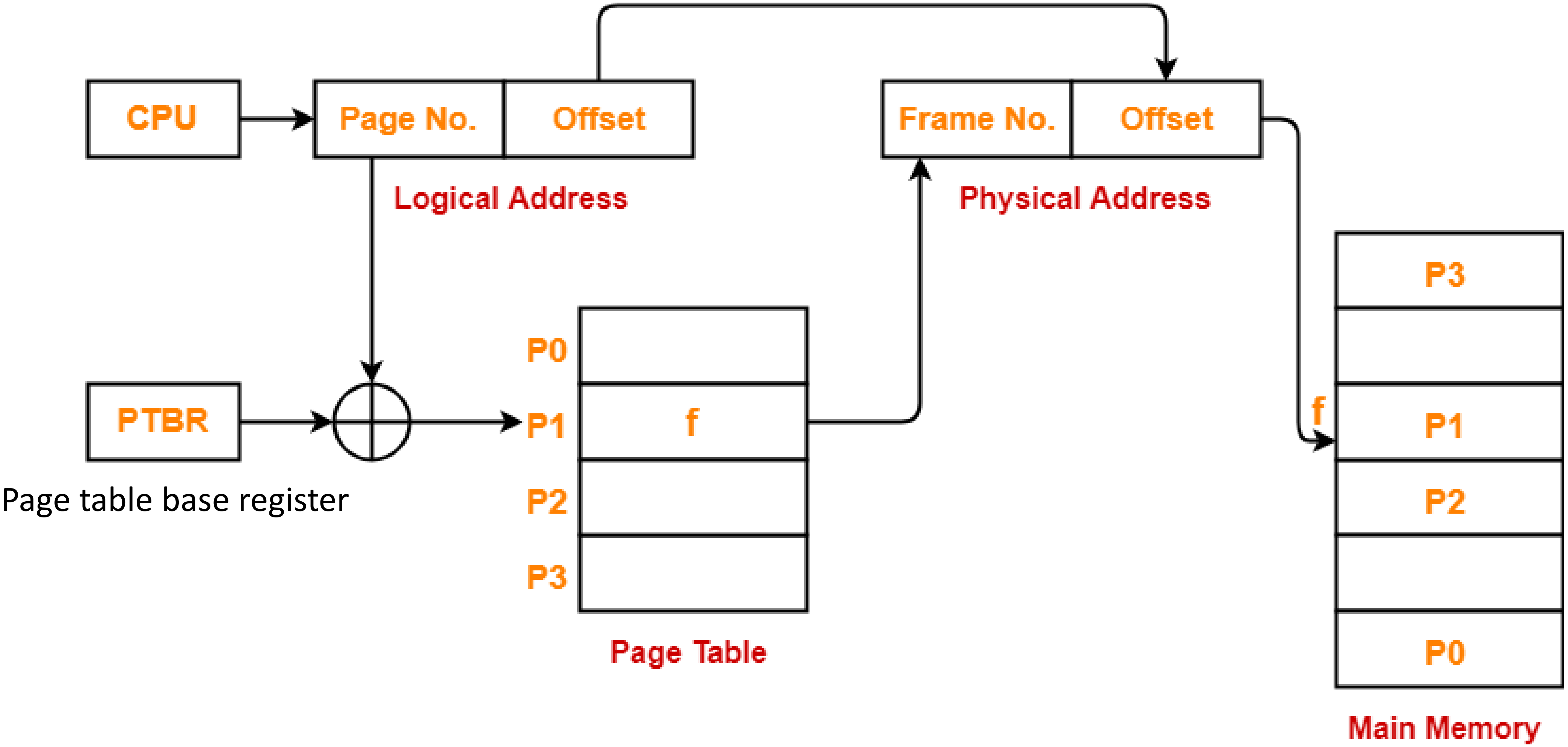
For the page number generated by the CPU:

Page Table provides the corresponding frame number (base address of the frame) where that page is stored in the main memory.

The frame number combined with the page offset forms the required physical address.

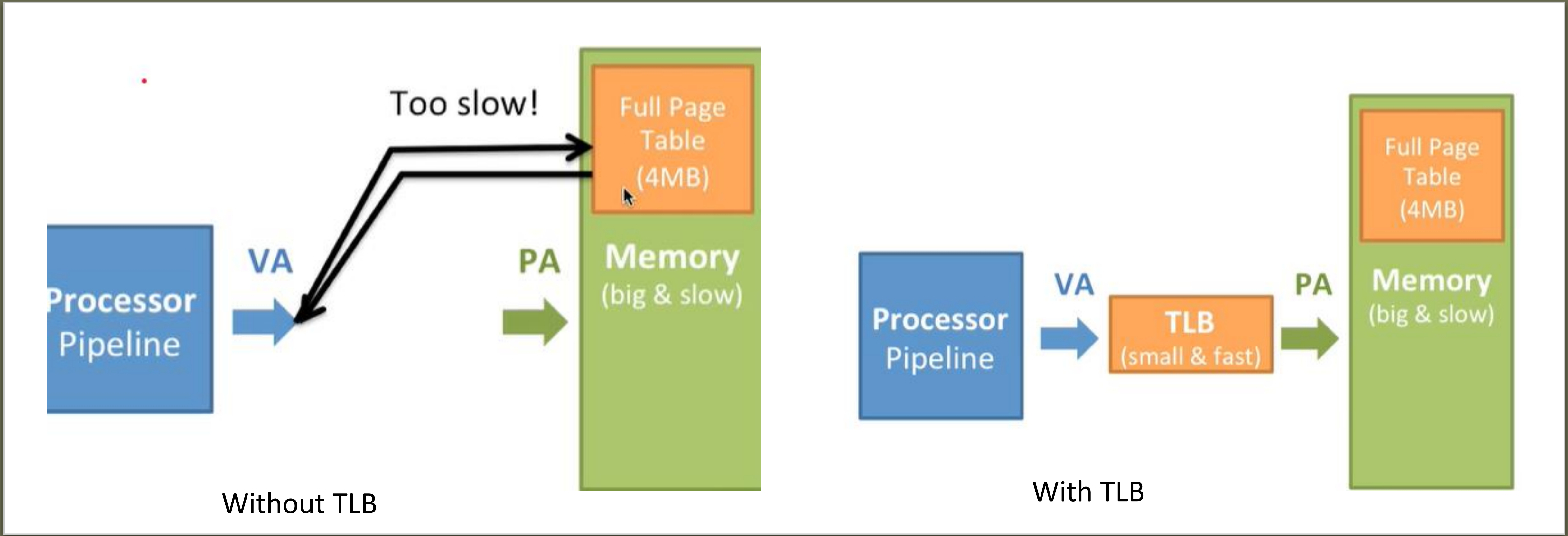
- Frame number specifies the specific frame where the required page is stored.
- Page Offset specifies the specific word that has to be read from that page.





Translating Logical Address into Physical Address

Making Virtual Memory Faster using Translation Look Aside Buffer (TLB) : A special Page Table Cache



Without TLB

With TLB

Advantages and Disadvantages

Advantages:

- It allows to store parts of a single process in a non-contiguous fashion.
- It solves the problem of external fragmentation.

Disadvantages:

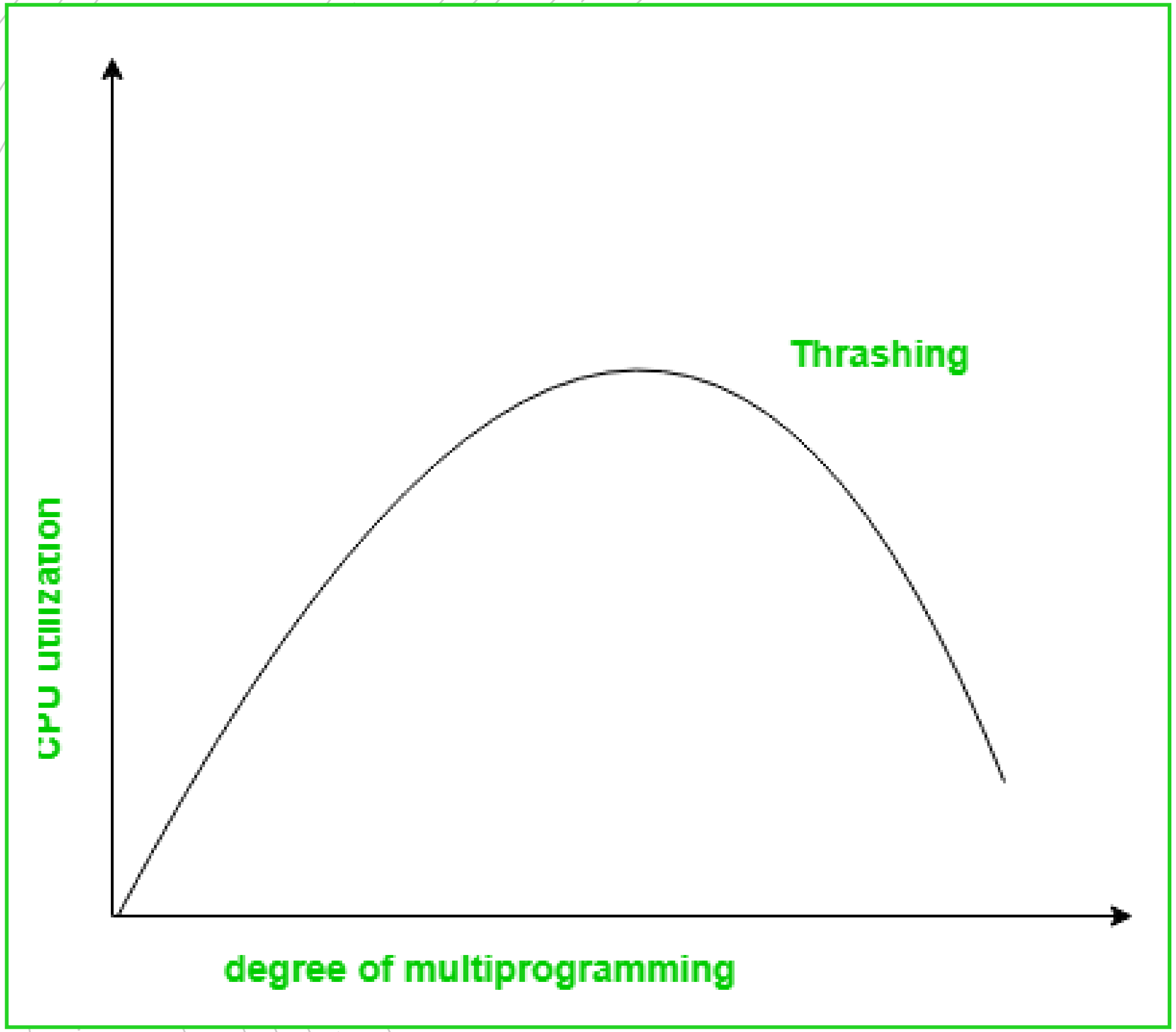
- It suffers from internal fragmentation.
- There is an overhead of maintaining a page table for each process.
- The time taken to fetch the instruction increases since now two memory accesses are required.

Page Fault

When a page referenced by the CPU is not found in the main memory, it is called as a page fault.

When a page fault occurs, the required page has to be fetched from the secondary memory into the main memory.

Thrashing



- Thrashing is a condition or a situation when the system is spending a major portion of its time in servicing the page faults, but the actual processing done is very negligible.

Thank You !! ☺

Q&A