Chapter-5
Memory- Management

- Memory is one of the most important resources of the computer that is used to store data and programs temporarily/permanently.
- Part of an operating system that manages a memory is called the Memory Manager (MM).
- The main functions of MM are:
  - Keeping track of which part of memory is in use and which parts are free
  - Allocating and de-allocating memory to processes.
  - Managing swapping between memory and disk when memory is not big enough to hold all the processes.

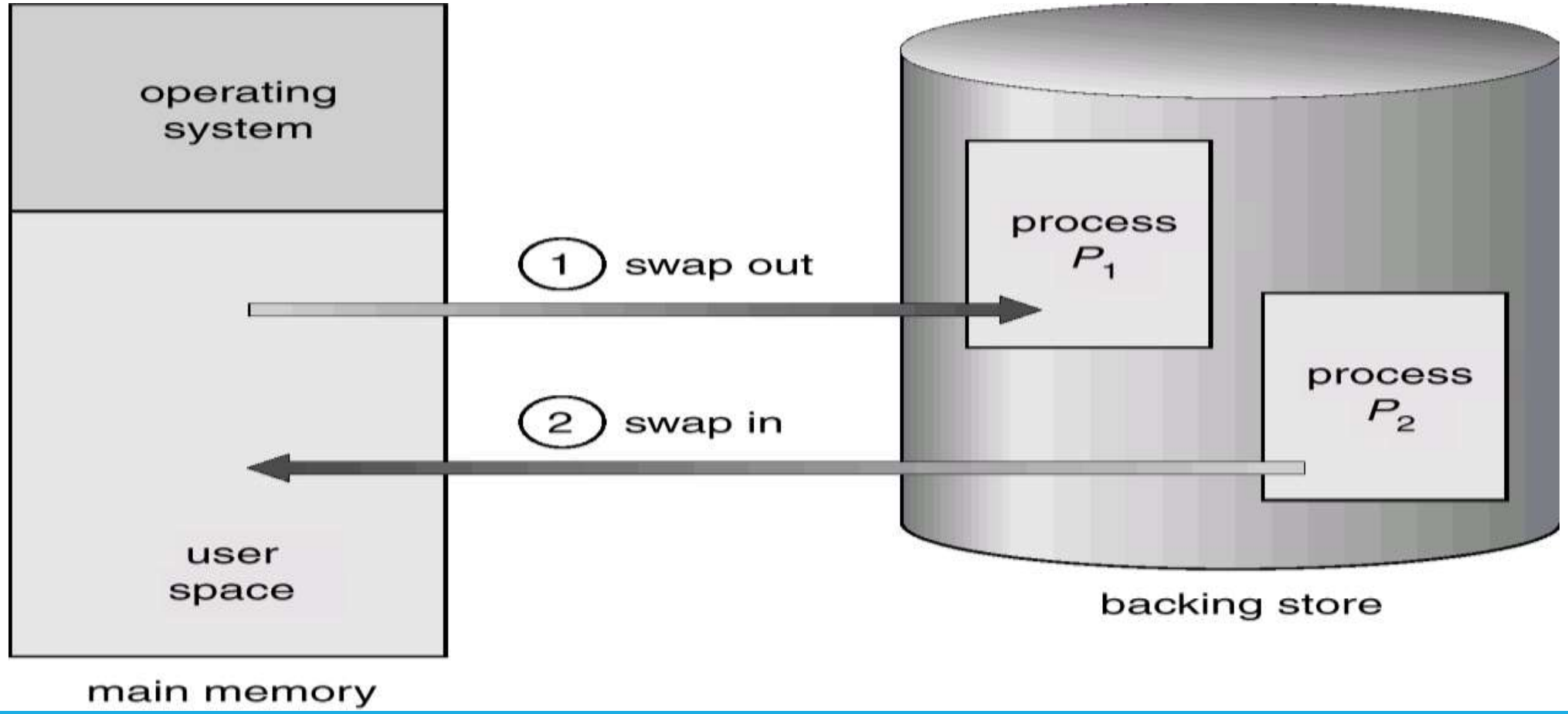
Swapping

- Swapping is a technique of temporarily removing inactive programs from the memory of the system.
- The programs resides in the secondary memory (the disks) must be brought into the main memory from the secondary memory because CPU has the direct access to the main memory.
- But the size of the main memory is limited.
- To maintain the degree of multiprogramming ,a process can be swapped temporarily out of memory to a backing store, and then **brought back** into the memory for continuing execution.

Swapping Con't…

- There are various scenarios where the OS swaps a process from a main memory to secondary memory. Some of these are:
  - If a process completes its execution
  - If a process is waiting for some I/O operations
  - If a process is trying to access some OS area of some other processes
  - If a multi-programming environment with a round robin CPU scheduling is used
  - If a multi-programming environment with a priority CPU scheduling is used

Swapping Con't…



Memory Allocation Techniques: Contiguous and Non –Contiguous

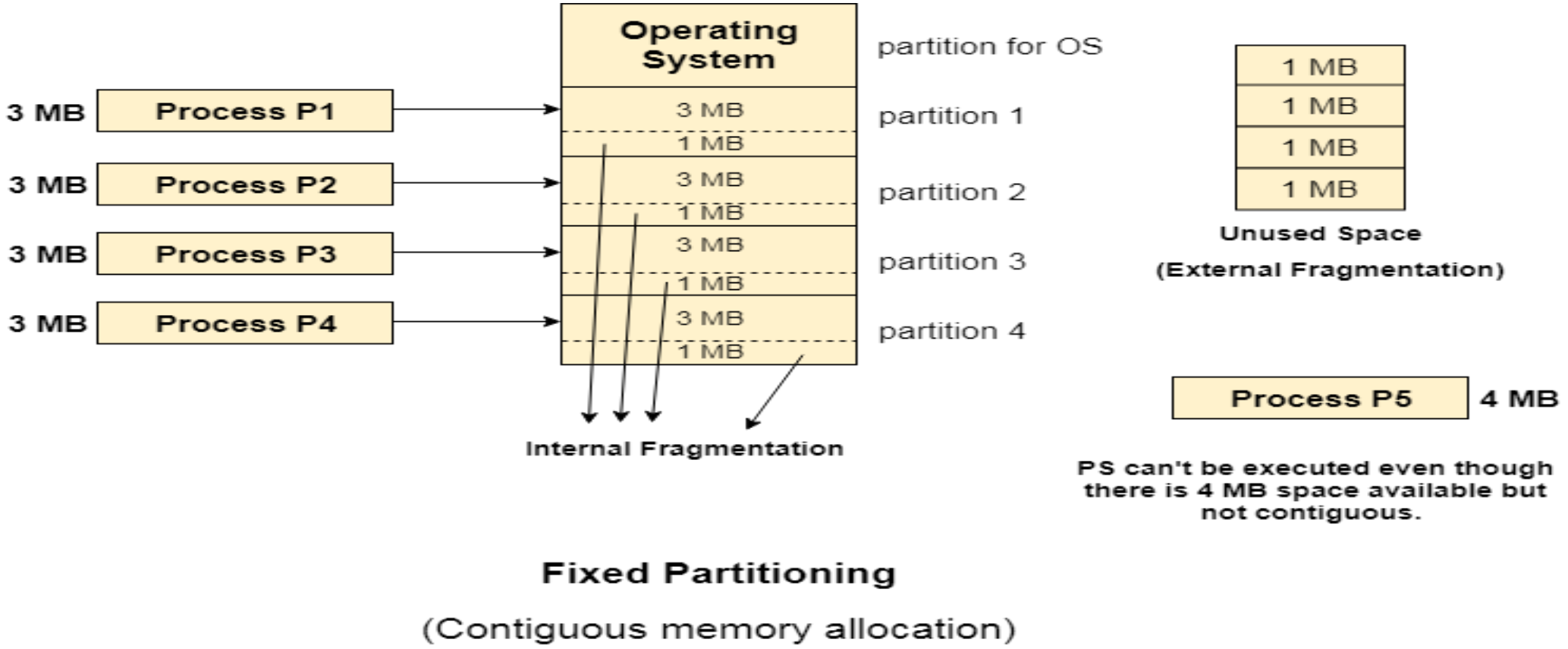
❑Contiguous Memory Allocation:

- ➢In order to run multiple programs in parallel a memory should be partitioned.
- ➢There are two partitioning approaches :
  - ➢**Fixed partitioning** – Partitioning is done before the processes comes to memory.
  - ➢**Dynamic partitioning** – Partitioning is done when processes request memory space.

Fixed partitioning

- It is the simplest and oldest partitioning technique
- Main Memory is divided into partitions of fixed size which cannot be changed.
- The memory is assigned to the processes in contiguous way and each partition can only contain one process.
- The operating system always resides in the first partition while the other partitions can be used to store user processes.
- Suffers from internal fragmentation
- To avoid these two problems we move to dynamic or variable size partition scheme

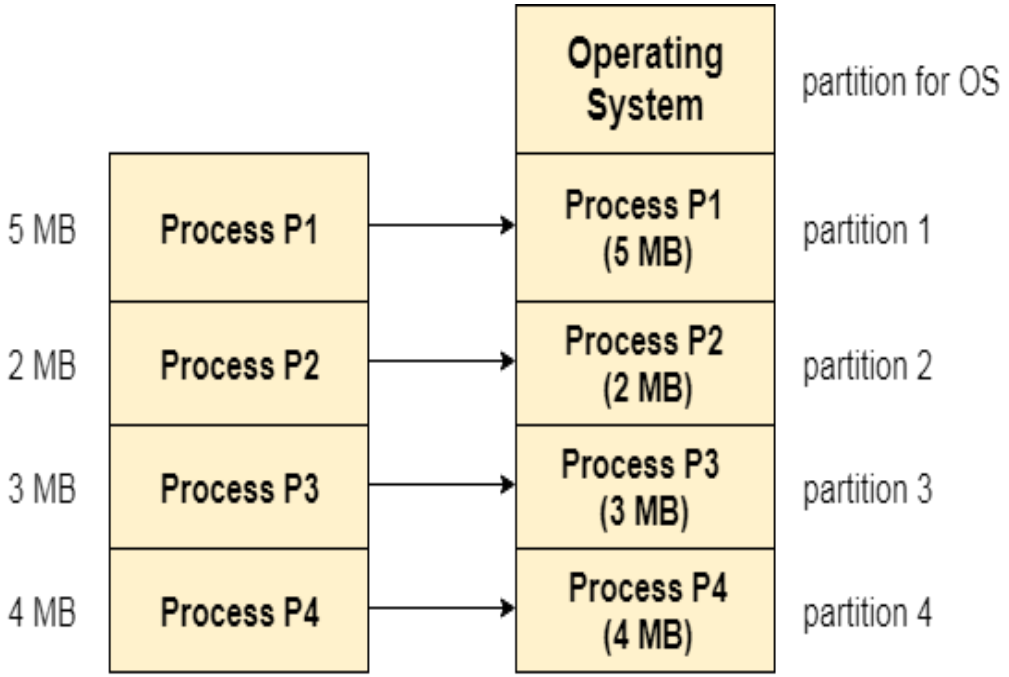
Fixed partitioning Con't…



Dynamic (Variable) partitioning

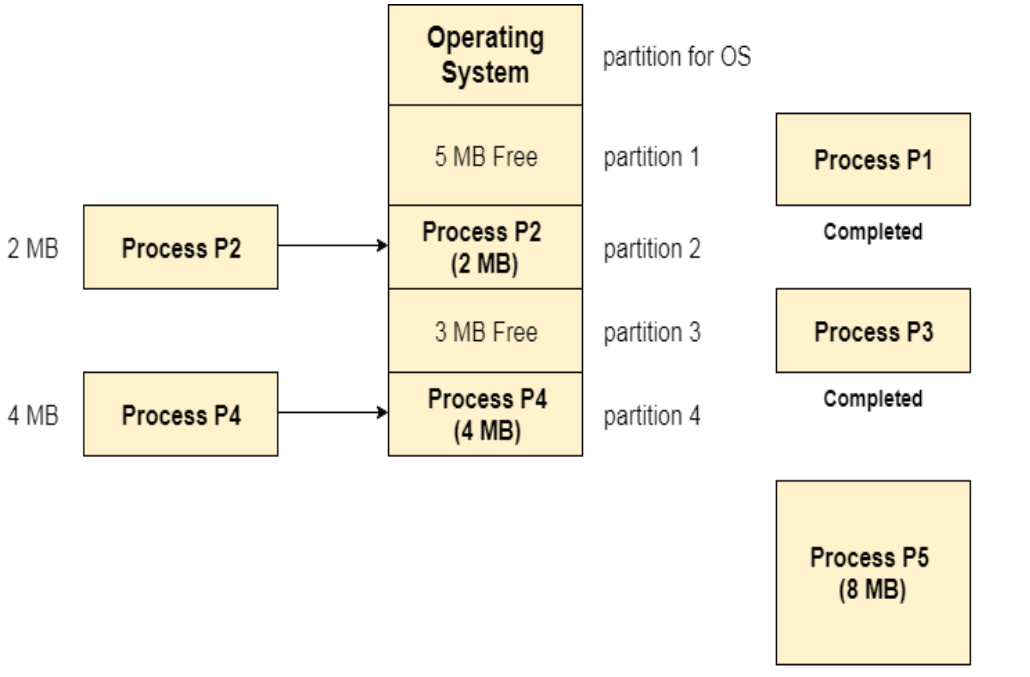
- Is done whenever the request of the process arrives accordingly partition is made in memory.
- Partitions are created dynamically as processes are loaded.
- For example, if a process of size 50kB arrives, a partition of 50kB is made in a memory.
- **Disadvantage:** suffers from external fragmentation
- To avoid this problem we move to Compaction or defragmentation scheme
- **Compaction or Defragmentation:** All the free partitions are merged which can now be allocated according to the needs of new processes.
- By applying this technique, we can store the bigger processes in the memory.

Dynamic (Variable) partitioning Con't…



Dynamic Partitioning

(Process Size = Partition Size)

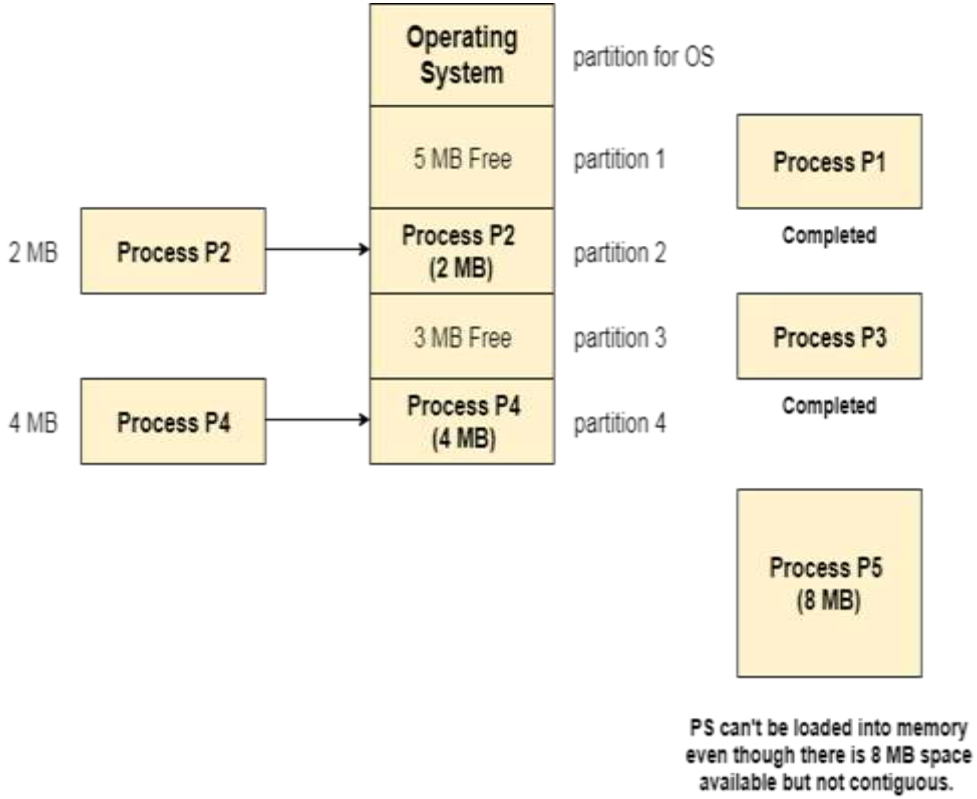


PS can't be loaded into memory even though there is 8 MB space available but not contiguous.

External Fragmentation in Dynamic Partitioning

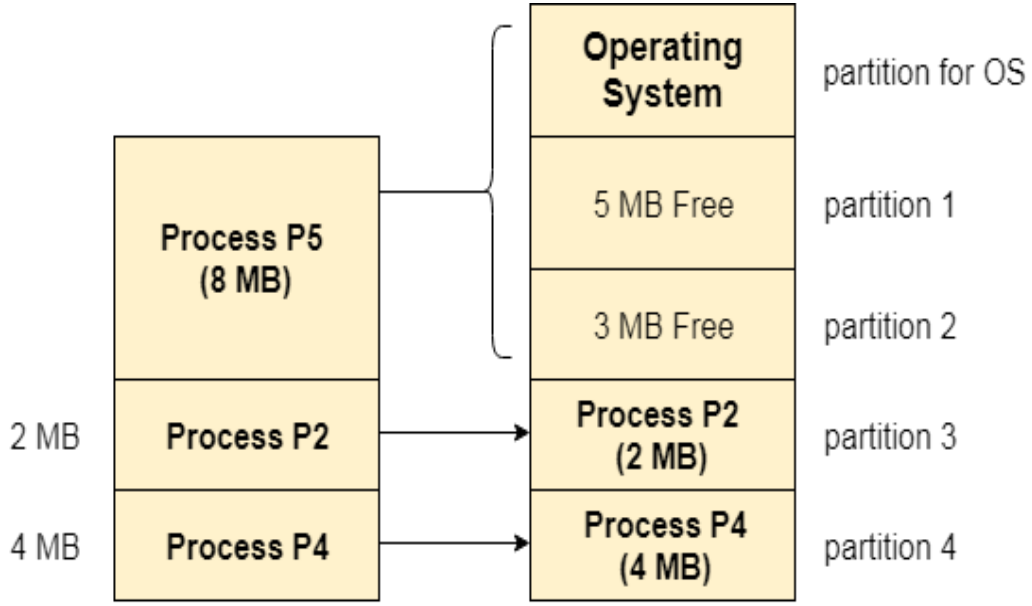
X

Dynamic (Variable) partitioning Con't…



External Fragmentation in Dynamic Partitioning

X



Compaction

Partition Allocation techniques

- How to satisfy a request of size **n** from a list of free holes ? Using 3 placement algorithms

- First fit: allocate the process in a partition which is first sufficient partition from the top of the memory.

- Best fit : allocate the process in a partition which is the smallest sufficient partition among the free available partition .
  - ✓To find the smallest sufficient partition it requires to search all the free partitions in the memory.

- Worst fit: allocate the process in a partition which is the largest sufficient partition among the free available partition .
  - ✓To find the larges sufficient partition it requires to search all the free partitions in the memory.

Partition Allocation Example

- Given 5 memory partitions of 100KB, 500KB, 200KB,300KB and 600 KB in order. How would each of the first-fit, best-fit and worst-fit algorithms place processes of 212KB,417 KB , 112KB, and 426 KB in order for P1,p2,p3 and p4 respectively?

- **First-fit**

Process	Partition size
	100KB
P1 (212)	500KB
P3 (112)	200KB
	300KB
P2 (417)	600KB

Best-fit

Process	Partition size
	100KB
P2 (417)	500KB
P3 (112)	200KB
P1 (212)	300KB
P4 (426)	600KB

Worst-fit

Process	Partition size
	100KB
P2 (417)	500KB
	200KB
P3 (112)	300KB
P1 (212)	600KB

Fragmentation

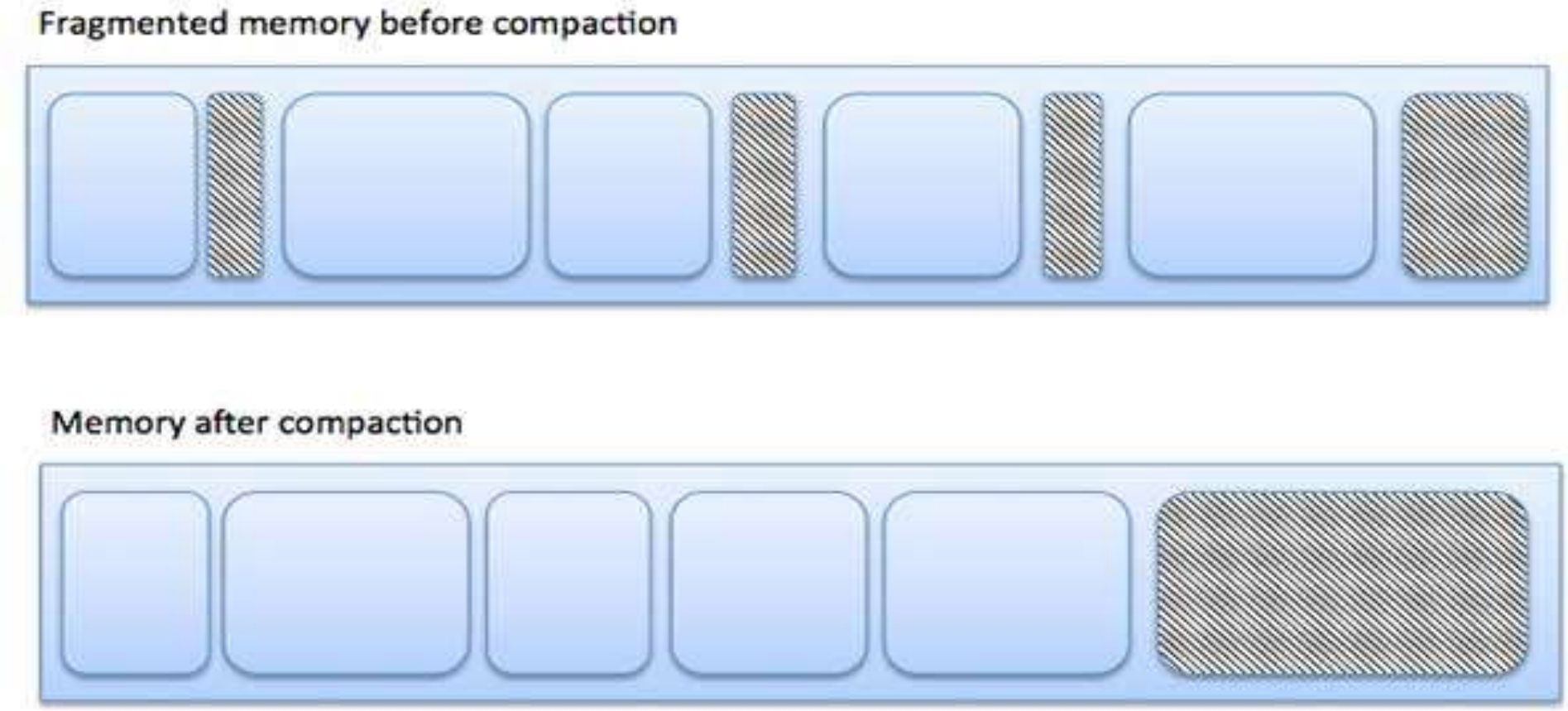
- **Internal Fragmentation** – allocated memory may be slightly larger than requested memory. This size difference is memory internal to a partition, but not being used.

  - For example , If there is a block of 50kb and if the process requests 40kb and if the block is allocated to the process then there will be 10kb of memory left.

- **External Fragmentation** – total memory space exists to satisfy a request, but it is not contiguous. storage is fragmented into a large number of small holes.

  - For example, There is a hole of 200 KB and 500 KB in multiple partition allocation schemes.
  - Next process request for 700 KB of memory. Actually 700 KB of memory is free which satisfy the request but the hole is not contiguous.

Compaction or Defragmentation



Paging and Segmentation

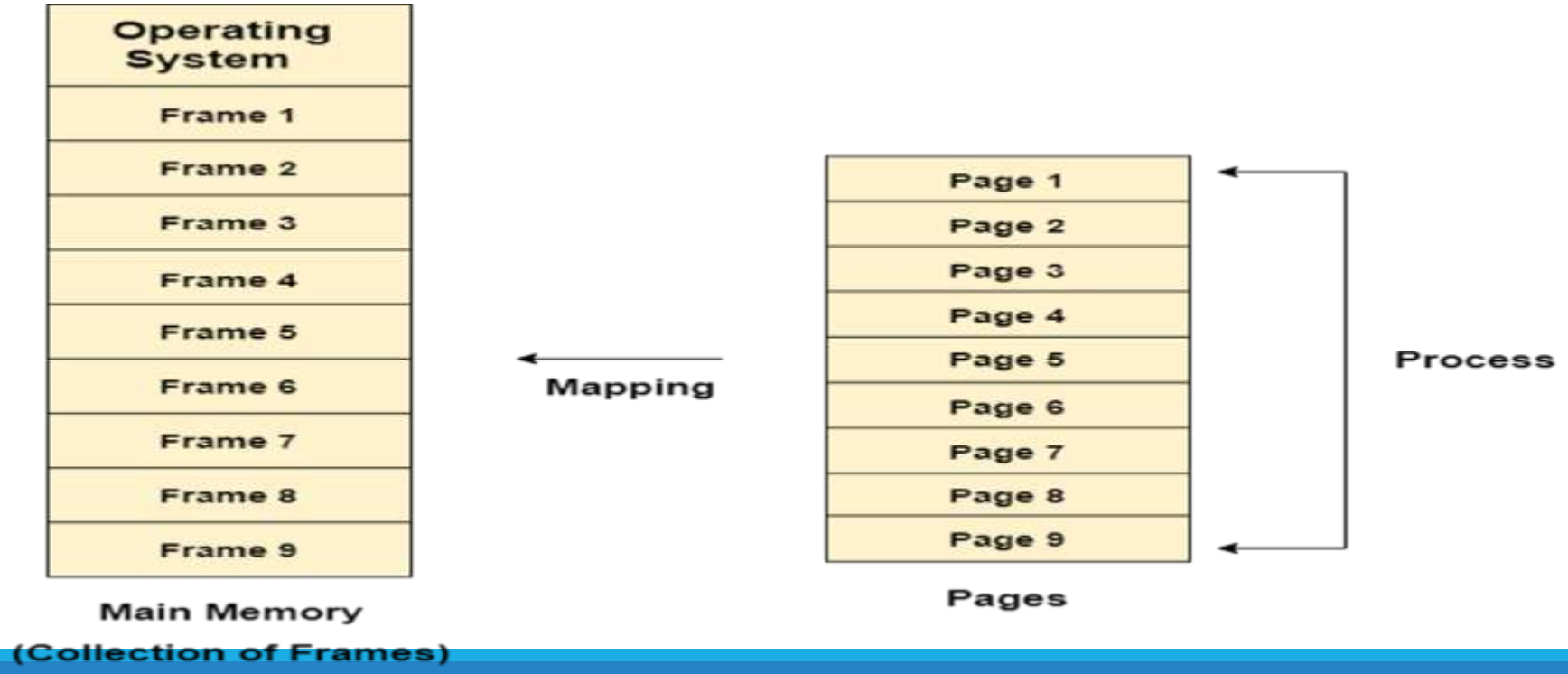
- Are the two ways which allow a process's physical address space to be non-contiguous.
- The mapping from virtual to physical address is done by the memory management unit (MMU) which is a hardware device and this mapping is known as paging technique.
- Here, a process can be spanned across different spaces in main memory in non-contiguous manner.

Paging

- In Operating Systems, Paging is a storage mechanism used to retrieve processes from the secondary storage into the main memory in the form of pages.
- The main idea behind the paging is to divide the process in pages. So, that, we can store them in the memory at different holes.
- One page of the process is to be stored in one of the frames of the memory.
- The pages can be stored at the different locations of the memory but the priority is always to find the contiguous frames or holes.
- Pages of the process are brought into the main memory only when they are required otherwise they reside in the secondary storage.

Paging Con't…

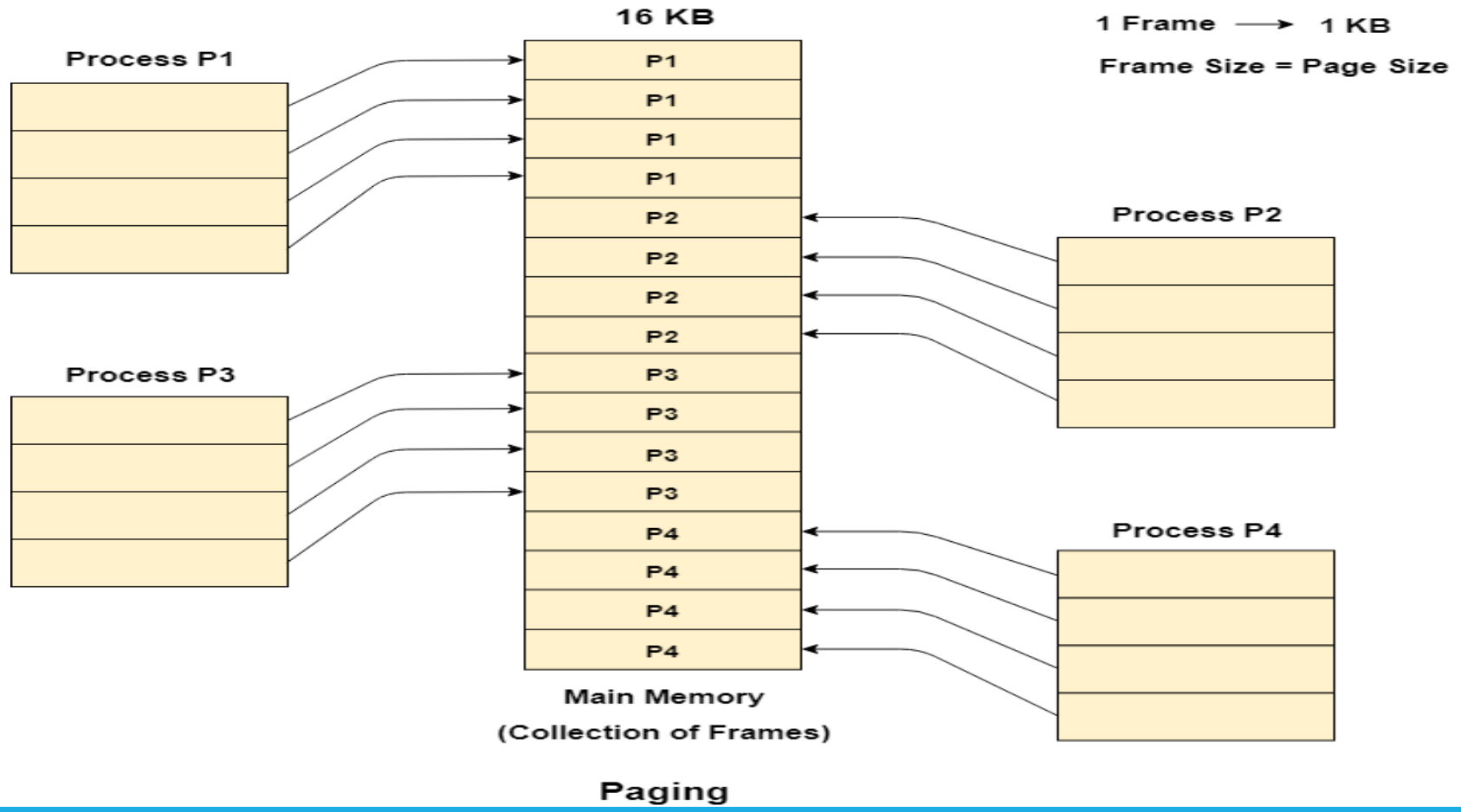
- Different operating system defines different frame sizes. The sizes of each frame must be equal. Considering the fact that the pages are mapped to the frames in Paging, page size needs to be as same as frame size.



Paging Example

- Let us consider the main memory size 16 KB and Frame size is 1 KB therefore the main memory will be divided into the collection of 16 frames of 1 KB each.
- There are 4 processes in the system that is P1, P2, P3 and P4 of 4 KB each
- Each process is divided into pages of 1 KB each. So, that one page can be stored in one frame.
- Initially, all the frames are empty. Therefore, pages of the processes will get stored in the contiguous way.
- Frames, pages and the mapping between the two is shown in the image below.

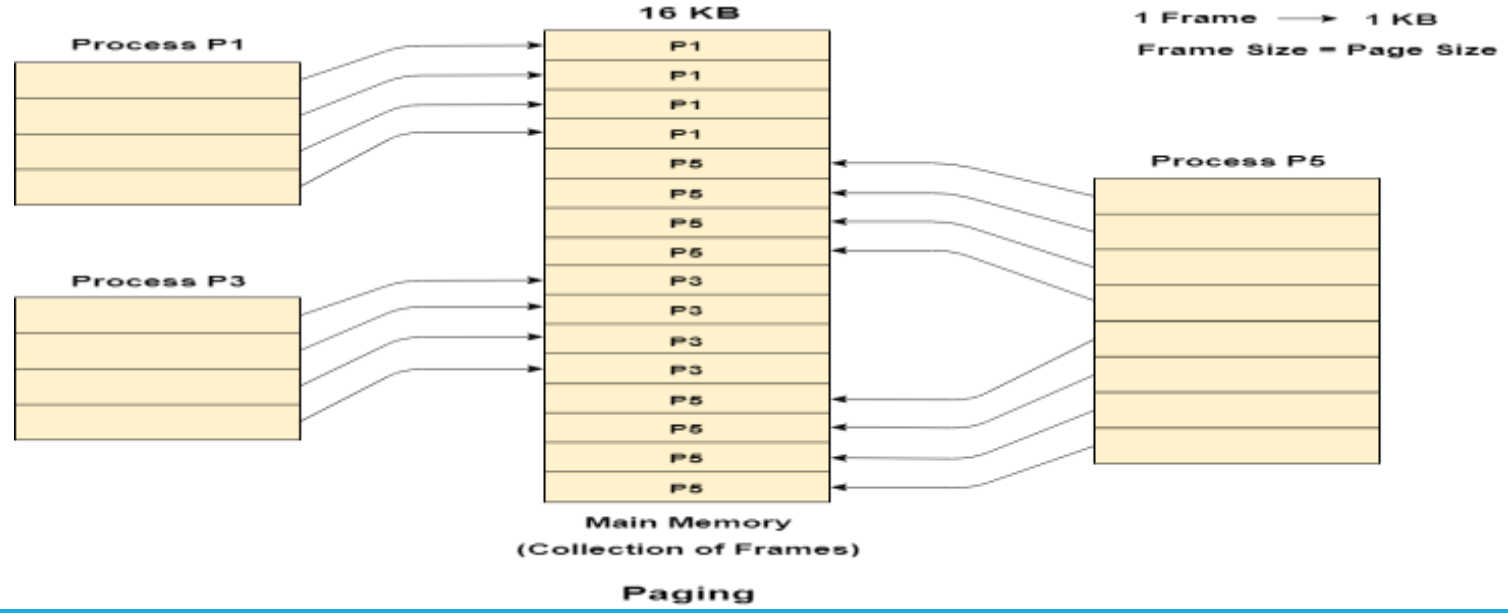
Paging Example Con't…



Paging

Paging Example Con't…

- Let us consider that, P2 and P4 are moved to waiting state after some time. Now, 8 frames become empty and therefore other pages can be loaded in that empty place. The process P5 of size 8 KB (8 pages) is waiting inside the ready queue.
- Given the fact that, we have 8 non contiguous frames available in the memory and paging provides the flexibility of storing the process at the different places. Therefore, we can load the pages of process P5 in the place of P2 and P4.



Logical address vs Physical address

- The purpose of Memory Management Unit (MMU) is to convert the logical address into the physical address.
- The logical address is the address generated by the CPU for every page while the physical address is the actual address of the frame where each page will be stored.
- When a page is to be accessed by the CPU using the logical address, the operating system needs to obtain the physical address to access that page physically.
- Page table : is the data structure used by a virtual memory system in a computer operating system to store the mapping between virtual address and physical addresses.

Segmentation

- Paging is simple to implement and assumed as an efficient memory management technique. but still suffer from internal fragmentation.
- However, Page table requires extra memory space and it doesn't bother where are program starts and ends. So, may not be good for a system having small RAM.
- To avoid this problem, we move to segmentation technique.
- Segmentation is a memory management technique in which each process is divided into variable size parts called segments.
- Allows breaking of the virtual address space of a single process into segments that may be placed in non-contiguous areas of physical memory.

Segmentation Con't…

- Advantages: no internal fragmentation and segment space consumes less space as compared to page table.
- Disadvantages: external fragmentation.
- A program segment contains the program's main function, utility functions, data structures, and so on.
- The operating system maintains a **segment map table** for every process and a list of free memory blocks along with segment numbers, their size and corresponding memory locations in main memory.
- For each segment, the table stores the starting address of the segment and the length of the segment.
- A reference to a memory location includes a value that identifies a segment and an offset.

Segmentation vs Paging

Segmentation	Paging
Program is divided into variable size segments.	Program is divided into fixed size pages.
User or compiler is responsible for dividing the program into segments.	Division into pages is performed by the Operating System.
Segmentation is slower than paging.	Paging is faster than segmentation.
Segmentation is visible to the user.	Paging is invisible to the user.
Segmentation eliminates internal fragmentation.	Paging suffers from internal fragmentation.
Segmentation suffers from external fragmentation.	There is no external fragmentation.
Processor uses page number, offset to calculate absolute address.	Processor uses segment number, offset to calculate absolute address.
Operating System maintains a list of free holes in main memory.	Operating System maintains a free frame list.

END OF CHAPTER -FIVE ON MEMORY MANAGEMENT