Virtual Memory - Introduction

By
Dr. Upasana Pandey
Department of Computer Science & Engineering
IMS Engineering College (College Code:143)

VM - Introduction

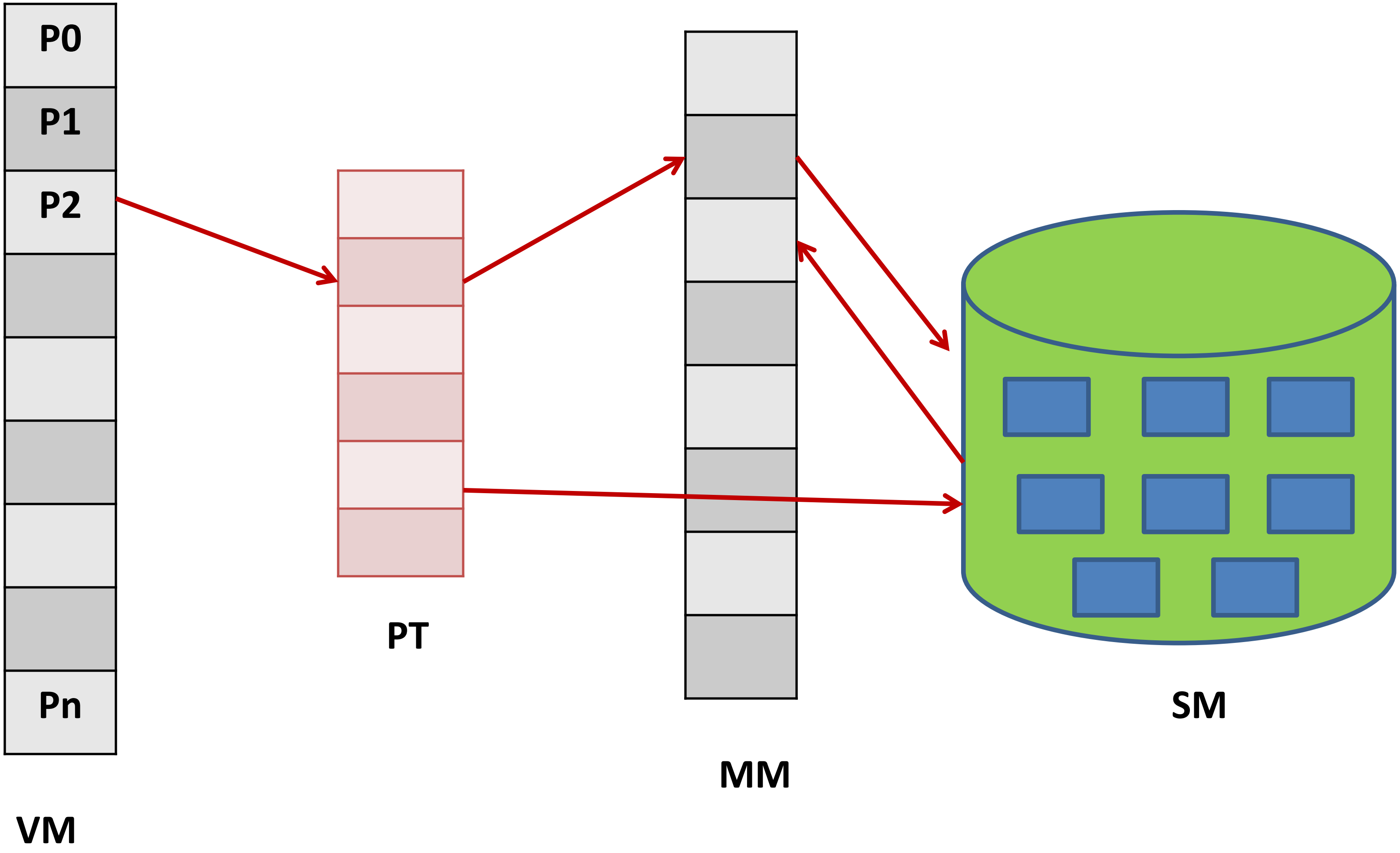
- A computer can address more memory than the amount physically installed on the system. This extra memory is actually called **virtual memory** .
- It is a section of a hard disk that's set up to emulate the **main memory**.
- It is a technique that allows the execution of program that are not completely in memory.
- The main visible advantage of this scheme is that programs can be larger than physical memory.
- It allows us to extend the use of physical memory by using disk.

VM - Introduction

Following are the situations, when entire program is not required to be loaded fully in main memory:

- User written **error handling routines** are used only when an error occurred in the data or computation.
- Certain **options and features** of a program may be used rarely.
- **Many tables** are assigned **a fixed amount of address space** even though only a small amount of the table is actually used.
- **Many routines** are commonly used at **mutually exclusive** times during a run.
- A program would **no longer be constrained by the amount of physical memory** that is available.
- Each user program **could take less physical memory**, more programs could be run the same time, with a corresponding increase in CPU utilization and throughput.

VM- Introduction



VM - Introduction

Advantages:

- Programmer should not worry about the size of available main memory.
- Bigger size program can easily be executed.
- Reduces the external fragmentation.
- The amount of space in used by a process may varied during its memory residence.
- As result execution of important processes may be speed up by allowing them more real memory.
- Degree of multiprogramming increased.
- It is implemented by Demand Paging.
- It can be implemented in segmented system and paged segmentation.

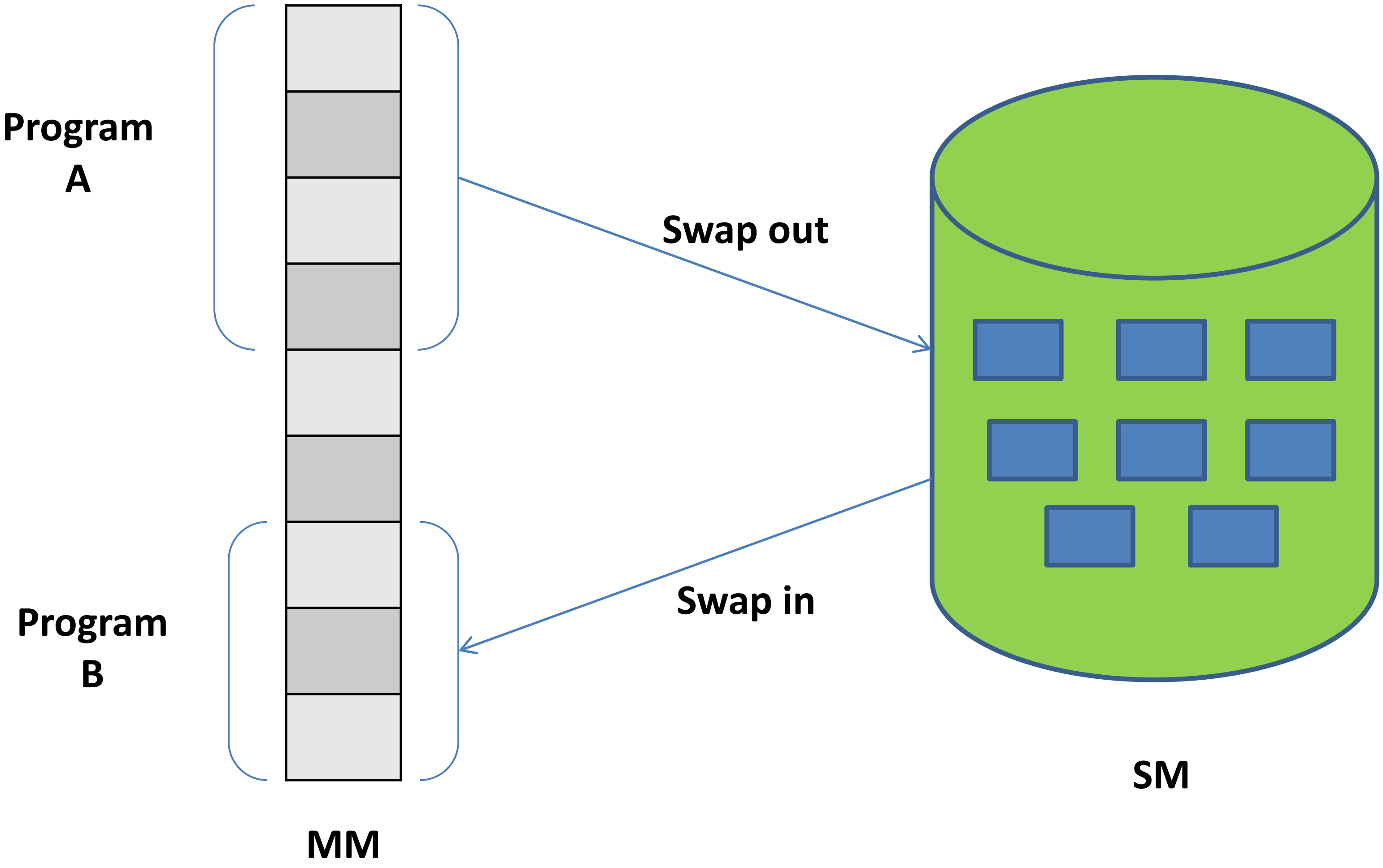
Disadvantage:

- Quite Complex.

Demand Paging

- Demand Paging: It is combination of “Paging with Swapping” – Lazy Swapper.
- A page is brought into main memory only when a reference is made to a location on that page.
- A lazy swapper never swaps a page into a memory unless that page will be needed.
- Each page of a program is stored contiguously in the paging swap space on a secondary storage.
- As locations in pages are referenced, the pages are copied into memory page frame.
- Once the page is in the memory, it is accessed as in simple paging.

Demand Paging

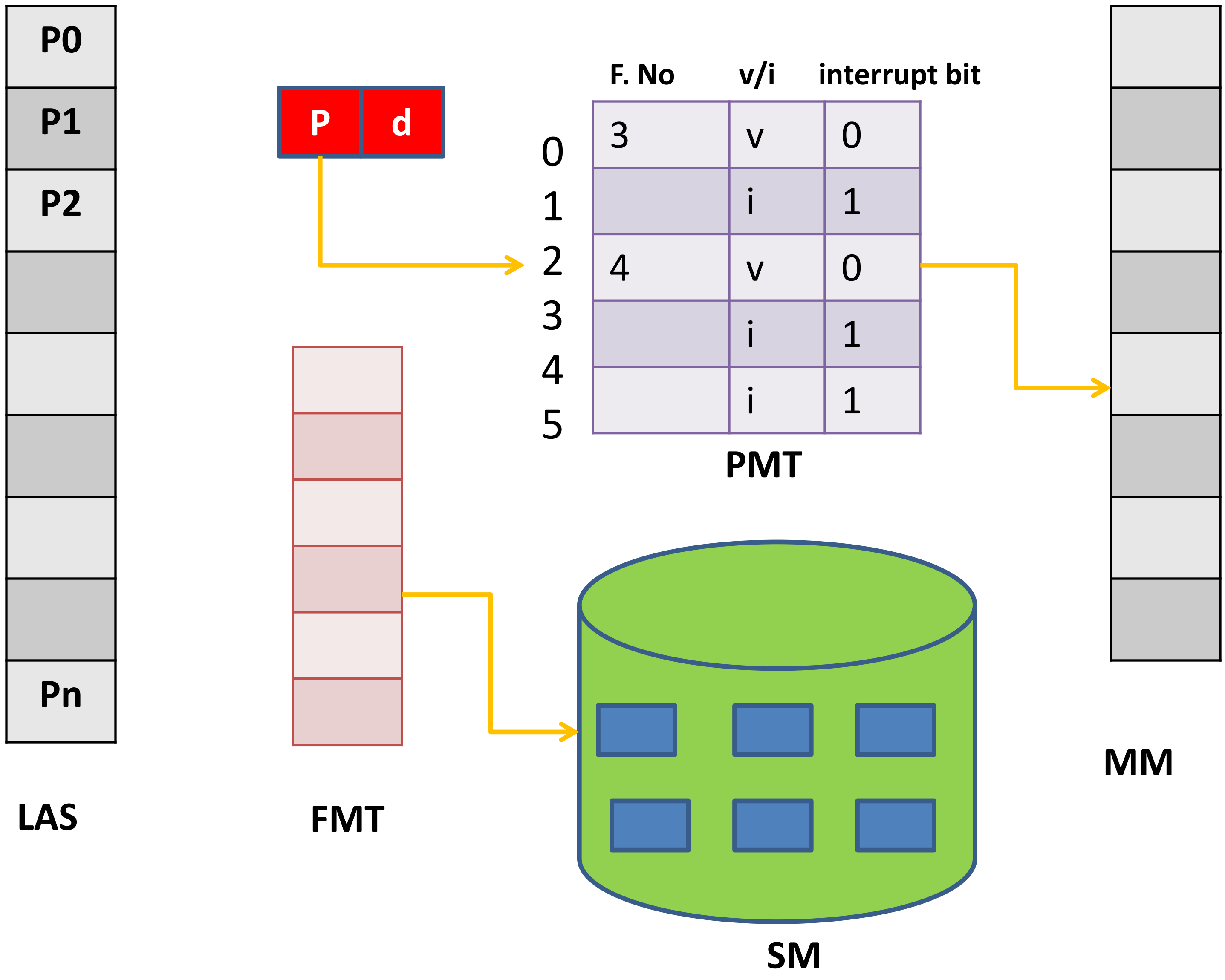


Swapping

H/W Support for Demand Paging

- Page Map Table (PMT) will have entries of those pages which are in main memory.
- File Map Table (FMT) gives address of secondary storage contains pages of logical address space.
- In PMT, there is a bit called "Interrupt bit", if it is 0 means page is in the main memory otherwise interrupt bit will be 1 means page is not in the main memory. It is called **"Page Fault"**.
- Now OS goes to the FMT and get address of that page and loaded into main memory.
- May be frame is not free then one frame has to be freed by using **"Page Replacement Algorithm"**.

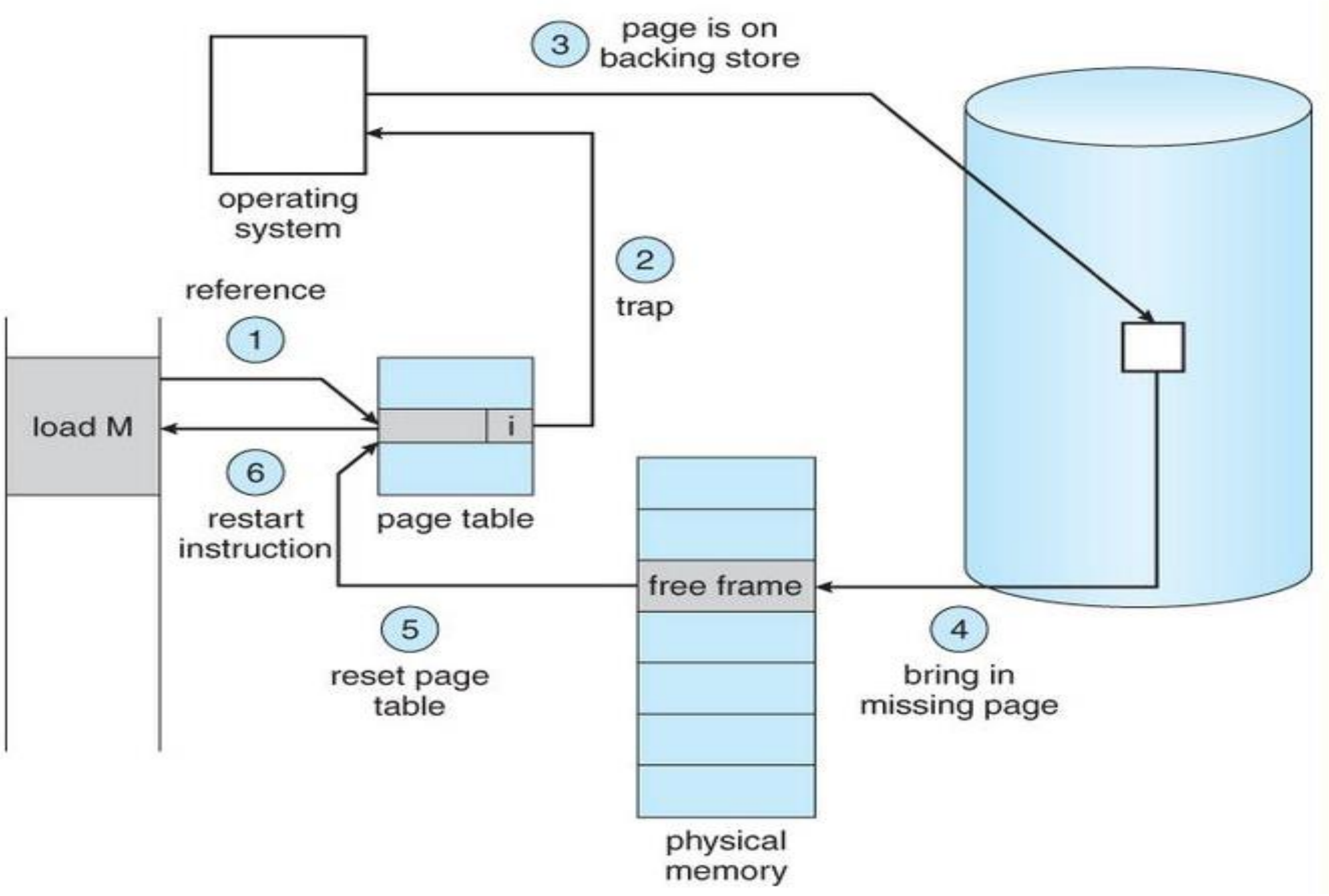
H/W Support for Demand Paging



F. No	v/i	interrupt bit
3	v	0
	i	1
4	v	0
	i	1
	i	1

PMT

Steps for handling page fault



Performance of Demand Paging

Demand paging can have a significant effect on the performance of a computer system. Let p be the probability of a page fault (0<=p<=1).

Effective access time=(1-p)*ma + p*page fault time

where p=page fault, Ma=memory access time

It is important to keep the page fault rate low in demand paging. Otherwise, the effective access time increases.

Demand Paging

Advantages:

- Large virtual memory.
- More efficient use of memory.
- There is no limit on degree of multiprogramming.

Disadvantages:

- Number of tables and amount of processor over heads for handling page interrupts are greater

Thank You