Paging

Paging

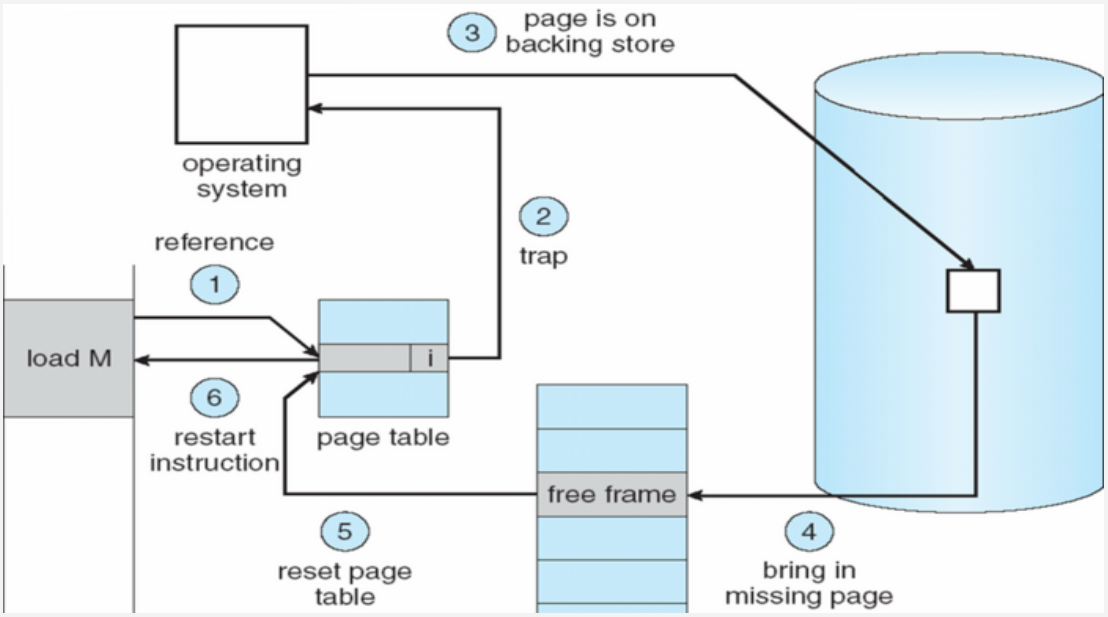
Memory Management technique that permits the physical address space of a process to be non contiguous is known as paging.Paging is used for faster access to data, and it is a logical concept.

Virtual memory

A computer can address more memory than the amount physically installed on the system. This extra memory is actually called virtual memory and it is a section of a hard that's set up to emulate the computer's RAM. Paging technique plays an important role in implementing virtual memory.

Page faults

Page fault dominates like an error. If any program tries to access a piece of memory but which does not exist in physical memory, meaning main memory, then page fault will occur. The fault specifies the O/S that it must trace the all data into virtual memory management, and after that moves it from secondary memory like a hard disk to primary memory of the system.



Page fault

Handling of a Page Fault

1. Check the location of the referenced page in the PMT
2. If a page fault occurred, call on the operating system to fix it
3. Using the frame replacement algorithm, find the frame location
4. Read the data from disk to memory
5. Update the page map table for the process
6. The instruction that caused the page fault is restarted when the process resumes execution.

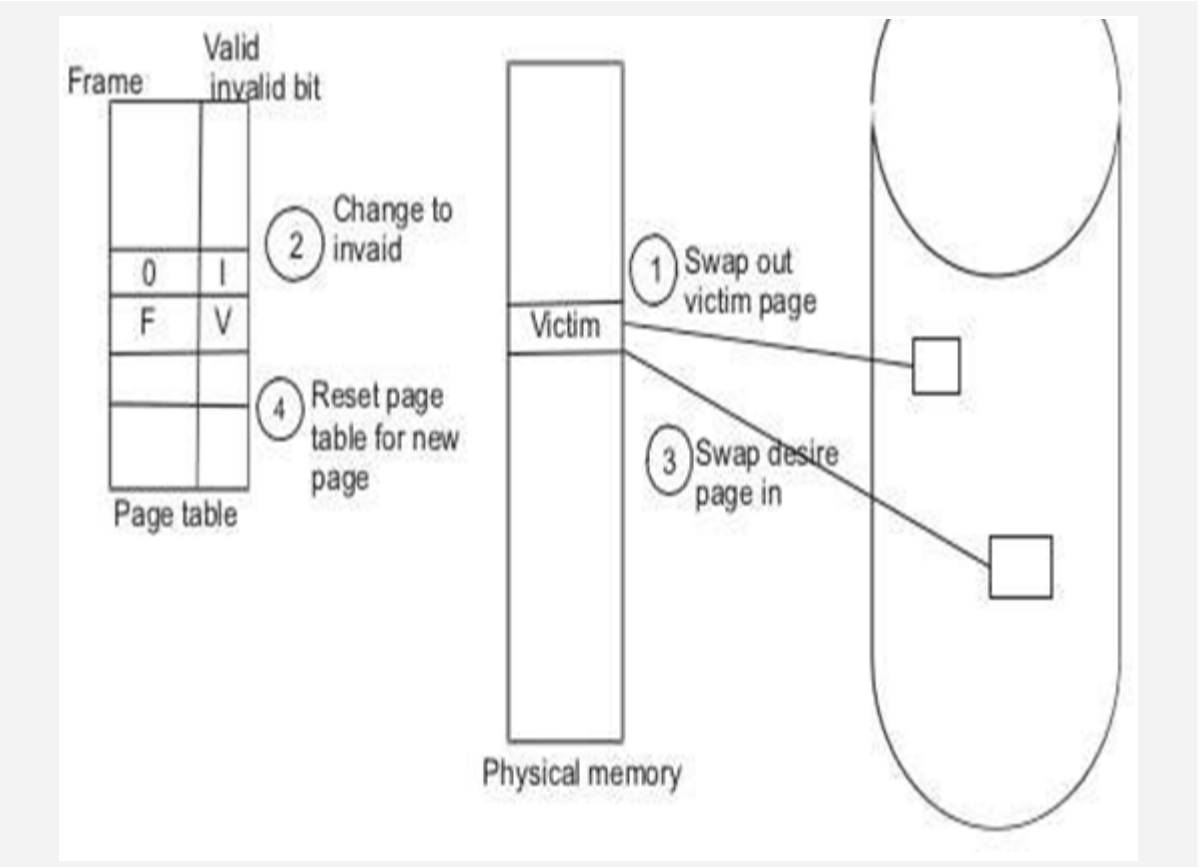
Page Replacement Algorithm

What is page Replacement?

When memory located in secondary memory is needed, it can be retrieved back to main memory.

Process of storing data from main memory to secondary memory ->swapping out

Retrieving data back to main memory ->swapping in



Page Replacement

Why do we need a page replacement algorithm?

The main goal of page replacement algorithms is to provide lowest page fault rate

Algorithms

- First In First Out
- Optimal Replacement
- Not Recently Used
- Second Chance
- CLOCK
- Not Frequently Used
- Least Recently Used
- Random Replacement

First-In First-Out (FIFO)

- Pages in main memory are kept in a list.
- Newest page is in head and the oldest in tail.
- It does not take advantage of page access patterns or frequency.

Optimal Replacement (OPT)

- When the memory is full, evict a page that will be unreferenced for the longest time.
- The OS keeps track of all pages referenced by the program.
- Only if the program's memory reference pattern is relatively consistent.

Not Recently Used (NRU)

- It favours keeping pages in memory that have been recently used.
- The OS divides the pages into four classes based on usage during the last clock tick:
    3. Referenced, modified
    2. Referenced, not modified
    1. Not referenced, modified
    0. Not referenced, not modified
- Pick a random page from the lowest category for removal,i.e. the not referenced, not modified page

Second Chance

- Modified version of FIFO
- Instead of swapping out the last page, the referenced bit is checked
- Gives every page a "second-chance"

Clock

- Modified version of FIFO
- The set of frame candidates for replacement is considered as a circular buffer.

Least Recently Used (LRU)

- It swaps the pages that have been used the least over a period of time.
- It is free from Belady’s anomaly

Not frequently used (NFU)

- This page replacement algorithm requires a counter
- The counters keep track of how frequently a page has been used
- The page with the lowest counter can be swapped out

Random

- This algorithm replaces a random page in memory.
- It fares better than FIFO.