**Operating System 19115045**

**Assignment-2 Kunal Sachdeva**

**5th Semester CSE**

**Sharing in Virtual Memory**

Virtual Memory is a storage allocation system that allows secondary memory to be addressed as if it were main memory. The addresses used by a programme to refer to memory are distinct from the addresses used by the memory system to designate physical storage sites, and program-generated addresses are automatically translated to machine addresses.

It's accomplished by treating a section of secondary memory as primary memory. The user can store processes in virtual memory that are larger than the available main memory.

As a result, rather than loading a single long process into main memory, the OS loads the various components of several processes into main memory. Demand paging and demand segmentation are commonly used to implement virtual memory.

The capacity of virtual storage is limited by the computer system's addressing method, and the amount of secondary memory available is determined by the number of main storage sites available rather than the actual number of main storage locations.

It's a technology that uses both hardware and software to work. It converts virtual addresses, which are used by programmes, into actual locations in computer memory.

1. Within a process, all memory references are logical addresses that are dynamically translated into physical addresses at run time. This means that throughout execution, a process can be swapped in and out of the main memory, occupying different locations in the main memory at different moments.
2. A process can be divided down into several pieces, and these pieces don't have to be kept in the main memory at all times throughout execution. This is possible because to a mix of dynamic run-time address translation and the use of a page or segment table.

The following are some of the benefits of using virtual memory:

* When your computer's physical memory is full, it writes what it needs to remember to the hard disc in the form of a swap file, which serves as virtual memory.
* If a computer running Windows requires additional memory/RAM, the system uses a small piece of the hard drive for this reason.

If these features are present, all of the pages or segments do not need to be present in the main memory during execution. This means that everytime the pages are needed, they must be loaded into memory. Demand Paging or Demand Segmentation are used to implement virtual memory.

With the help of one example, we'll look at virtual memory management. Consider the following scenario:

Assume that an operating system uses 300 MB of RAM to store all of the active programmes. However, there is only 50 MB of actual capacity accessible on the RAM right now.

* The OS will then create 250 MB of virtual memory and manage it with an application called the Virtual Memory Manager (VMM).
* In this situation, the VMM will create a 250 MB file on the hard disc to store the additional memory that is necessary.
* The OS will now proceed to address memory, even when only 50 MB of space is available, because it believes 300 MB of real memory to be stored in the RAM.
* Even if only 50 MB of physical memory space is available, the VMM is responsible for managing 300 MB of memory.

**Demand Paging :**

Demand paging is the process of loading a page into memory on demand (when a page fault occurs).

The steps in the procedure are as follows:

1. When the CPU attempts to refer to a page that is currently unavailable in main memory, an interrupt is generated, signalling a memory access fault.
2. The interrupting process is placed in a blocking state by the operating system. In order for the execution to continue, the OS must first load the appropriate page into memory.
3. The operating system will look in the logical address space for the appropriate page.
4. The required page will be transferred from the logical to the physical address space. Page replacement algorithms are used to make decisions about whether or not to replace a page in physical address space.
5. The page table will be modified to reflect this.
6. The signal will be delivered to the CPU, which will allow the programme to proceed and return the process to the ready state.

**Advantages** :

* More processes can be stored in the main memory: Because we'll only load a portion of each process's pages, there's room for more processes in the main memory. This results in more efficient processor utilisation because at any given time, at least one of the many processes will be ready.
* A process could take up more memory than the entire main memory: One of the most fundamental programming constraints has been lifted. Demand paging allows a process to run that is larger than the primary memory. As needed, the operating system loads pages of a process into main memory.
* It enables for higher levels of multiprogramming because each process uses less of the available (main) memory.

Page Fault Service Time: Page fault service time refers to the time it takes to repair a page fault. The time it takes to complete all six steps in the page fault service is included in the page fault service time.

Let's say the time it takes to reach main memory is m.

The time it takes to resolve a page fault is: s

The rate of page faults is: p

Then (p\*s) + (1-p)\*m is the effective memory access time.

**Swapping:**

Swapping a process out entails deleting all of its pages from memory or designating them for removal using the standard page replacement method. Suspending a process prevents it from running while it is being replaced. The system switches the process from secondary storage to main memory at a later time. Thrashing describes a condition in which a process is busy switching pages in and out.