Fragmentation in operating systems

Fragmentation in operating systems refers to the phenomenon where files or free space on a storage device become divided into various non-contiguous segments. It can occur in two forms:

1. External Fragmentation: External fragmentation happens when free space on a storage device becomes scattered throughout the system, resulting in small chunks of free space dispersed between allocated files. As a result, even if sufficient free space exists, it may not be contiguous, making it challenging to allocate large files or new processes that require contiguous blocks of memory.

External fragmentation can occur in both disk storage and main memory. In disk storage, it can arise due to the allocation and deallocation of files of varying sizes over time. In main memory, external fragmentation can occur when processes are loaded and removed from memory, leaving gaps of unused memory between allocated processes.

1. Internal Fragmentation: Internal fragmentation occurs when allocated space within a file or memory block is not fully utilized, resulting in wasted space. It happens when the assigned storage is larger than what is actually required by the file or process. This can occur due to fixed-size allocation units or memory blocks, resulting in inefficient storage utilization.

In the context of file systems, internal fragmentation can occur when the file system allocates fixed-size blocks to store files. If a file's size is not an exact multiple of the block size, the last block allocated to the file may have unused space, leading to internal fragmentation.

Fragmentation can impact system performance and efficiency. External fragmentation can result in slower file access times due to scattered data, increased disk head movement, and decreased disk throughput. In main memory, it can lead to reduced memory utilization and the inability to allocate contiguous memory blocks.

To mitigate fragmentation, several techniques can be employed, such as:

* Defragmentation: This process rearranges files or free space on a disk to consolidate fragmented segments and create larger contiguous blocks of free space.
* Compaction: In memory management, compaction involves rearranging the allocated and free memory blocks to create a single block of free memory, reducing external fragmentation.
* Dynamic Memory Allocation: Using dynamic memory allocation algorithms that attempt to allocate memory blocks that best fit the requested size can help minimize both external and internal fragmentation.

Overall, fragmentation management is an important consideration for operating systems to maintain optimal system performance and efficient resource utilization.

Link : <https://youtu.be/Dv40MQoV__A>