

LEC-25: Free Space Management



1. Defragmentation/Compaction

- a. Dynamic partitioning suffers from external fragmentation.
- b. Compaction to minimize the probability of external fragmentation.
- c. All the free partitions are made contiguous, and all the loaded partitions are brought together.
- d. By applying this technique, we can store the bigger processes in the memory. The free partitions are merged which can now be allocated according to the needs of new processes. This technique is also called **defragmentation**.
- e. The efficiency of the system is decreased in the case of compaction since all the free spaces will be transferred from several places to a single place.

2. How free space is stored/represented in OS?

- a. Free holes in the memory are represented by a free list (Linked-List data structure).

3. How to satisfy a request of a of n size from a list of free holes?

- a. Various algorithms which are implemented by the Operating System in order to find out the holes in the linked list and allocate them to the processes.

b. First Fit

- i. Allocate the first hole that is big enough.
- ii. Simple and easy to implement.
- iii. Fast/Less time complexity

c. Next Fit

- i. Enhancement on First fit but starts search always from last allocated hole.
- ii. Same advantages of First Fit.

d. Best Fit

- i. Allocate smallest hole that is big enough.
- ii. Lesser internal fragmentation.
- iii. May create many small holes and cause major external fragmentation.
- iv. Slow, as required to iterate whole free holes list.

e. Worst Fit

- i. Allocate the largest hole that is big enough.
- ii. Slow, as required to iterate whole free holes list.
- iii. Leaves larger holes that may accommodate other processes.