

Segmentation

- Programs are broken into segments of variable length.
- The virtual address is broken into **Segment** and **Offset** fields:

$$\begin{array}{|c|c|} \hline \text{Segment} & \text{Offset} \\ \hline \end{array}$$

$n-1$ 0
- A segment table is used to map virtual addresses to physical addresses.
 - resides in physical memory.
 - constructed and managed by the operating system.
- Each process has its own segment table

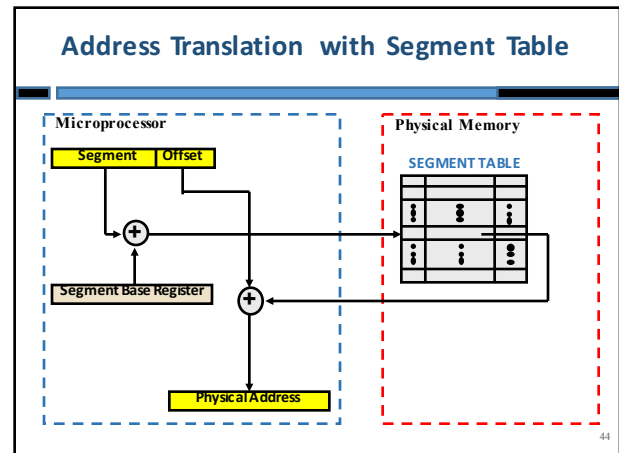
Segment Table

- An array with an entry for each segment of the virtual space:

SEGMENT TABLE		
0		
1		
2		
⋮	⋮	⋮
2^i-1		

Control bits
Base Address
Length

- Page fault occurs under any of the following circumstances:
 - segment is not valid
 - violation of access permission
 - offset larger than the segment length



Fragmentation

- As segments are allocated and replaced into physical memory, blocks of free memory locations of varied sizes are created.
- New blocks that do not fit into a free block end up replacing an existing block.
- New blocks that are allocated in a free block that is larger end up increasing fragmentation.
- On physical memory, fragmentation causes an inefficient use of the memory space.
- On a secondary memory like a hard disk, fragmentation increases access time due to an increase in travel time of the disc reader mechanism as it moves from track to track.

Physical Memory

ProcessD Seg12
ProcessC Seg0
ProcessE Seg20
ProcessD Seg5
ProcessE Seg2
ProcessC Seg7