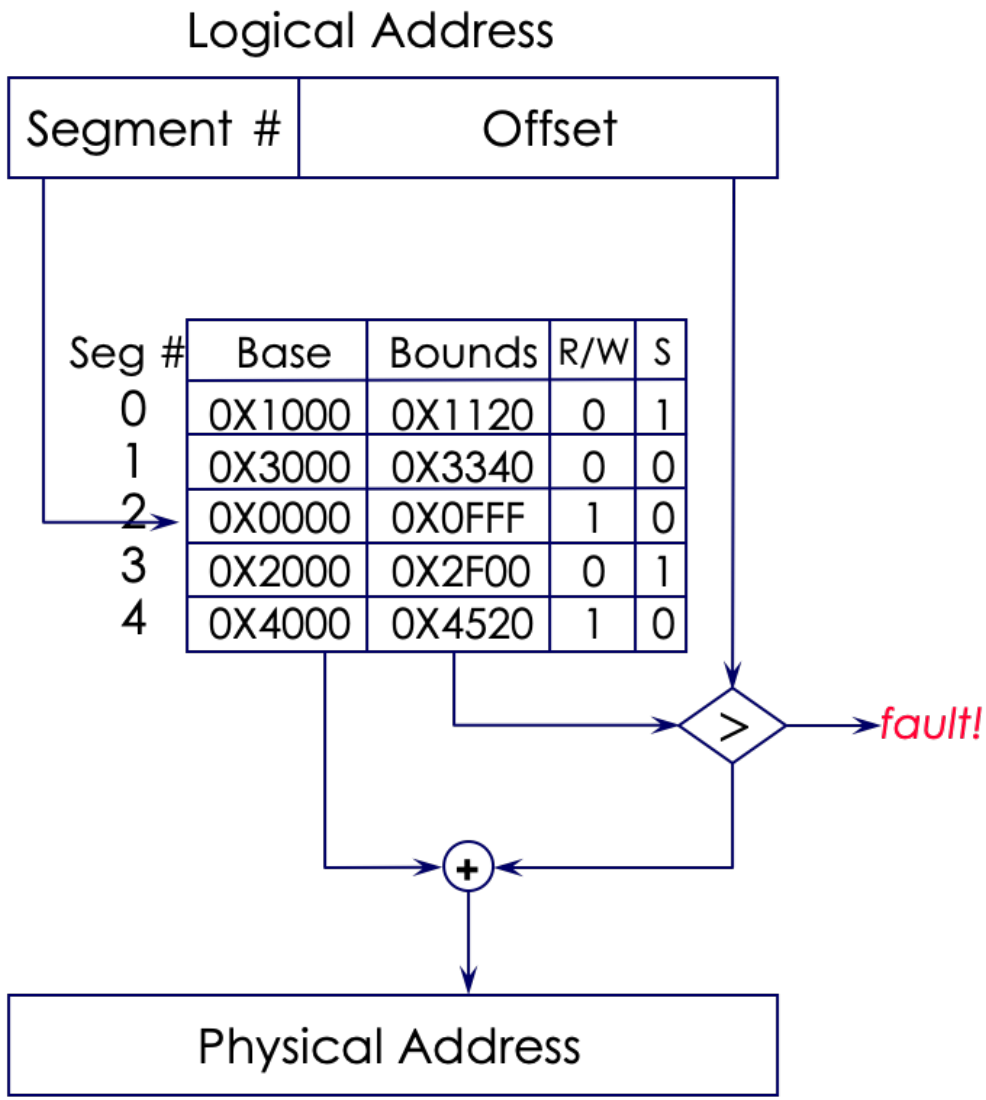
Segmentation

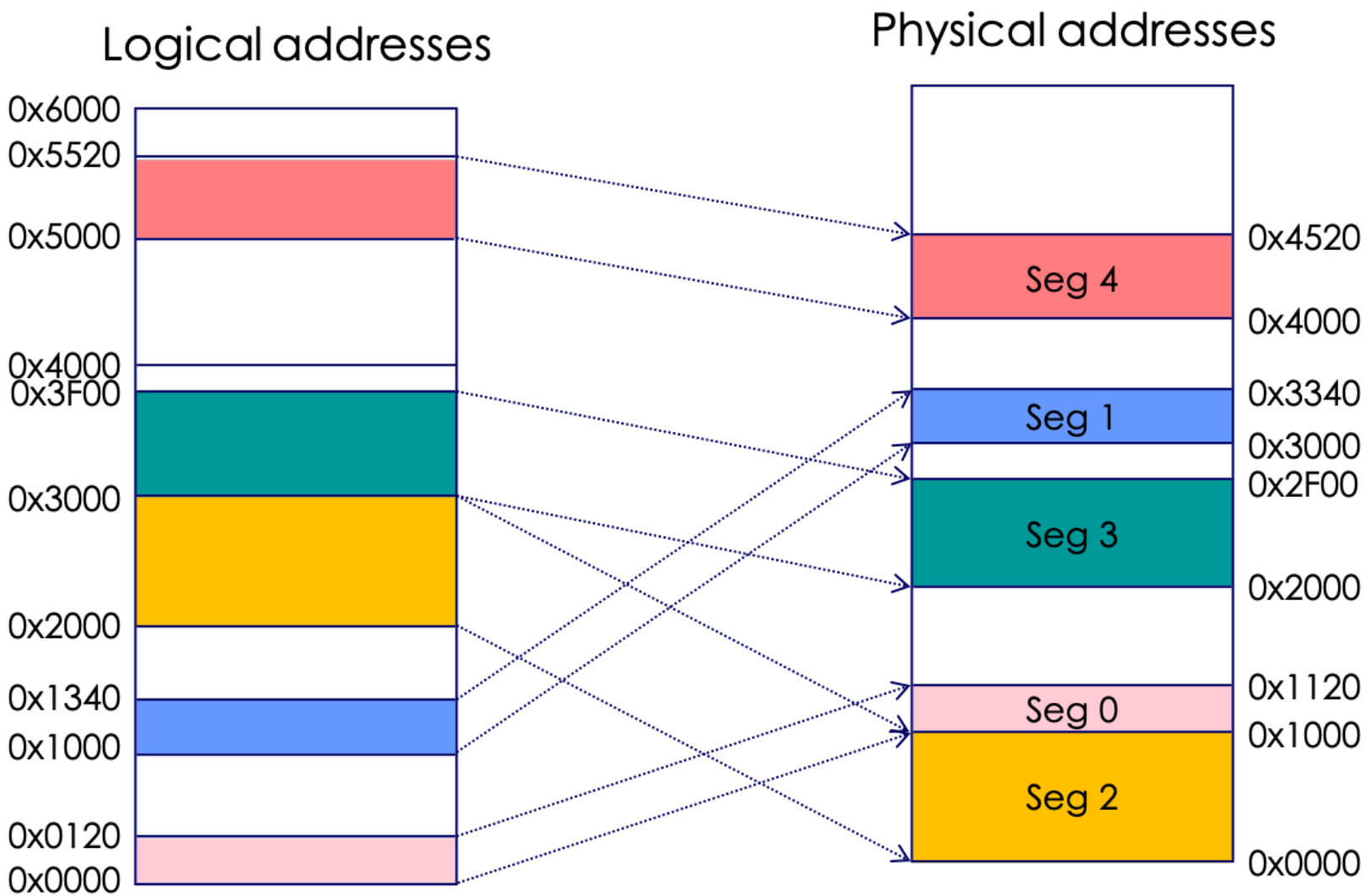
Segmentation...

- A **segment** is a region of contiguous memory. The most important problem with base-and-bounds relocation is that there is only one segment for each process.
- Segmentation generalizes the base-and-bounds technique by allowing each process to be split over several segments. A segment table holds the base and bounds of each segment. Although the segments maybe scattered in memory, each segment is mapped to a contiguous region.
- Additional fields (Read/Write and Shared) in the segment table adds protection and sharing capabilities to segments.

Relocation with segmentation...



An example...


Logical addresses
Physical addresses
0x6000
0x5520
0x5000
0x4000
0x3F00
0x3000
0x2000
0x1340
0x1000
0x0120
0x0000
Seg 4
Seg 1
Seg 3
Seg 0
Seg 2
0x4520
0x4000
0x3340
0x3000
0x2F00
0x2000
0x1120
0x1000
0x0000

Discussion...

- When a process is created, an empty segment table is inserted into the process control block. Table entries are filled as new segments are allocated for the process.
- The segments are returned to the free segment pool when the process terminates. Segmentation, as well as the base and bounds approach, causes external fragmentation and requires memory compaction.
- An advantage of the approach is that only a segment, instead of a whole process, may be swapped to make room for the (new) process.