

Chapter #8

Operating System Support

Memory Management

- Single process
 - Memory split into two sections
 - Operating System resides in one section
 - User process resides in other section
- Multiple processes
 - Memory split into various sections
 - Operating System resides in one section
 - User processes share remains sections

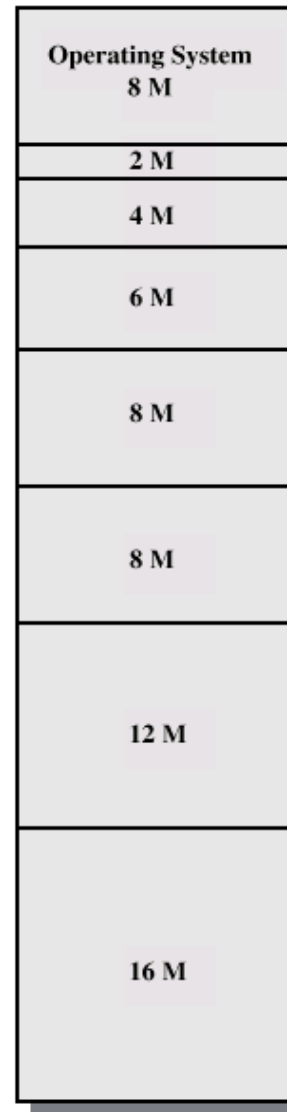
Partitioning

- Splitting memory among processes (including OS)
- Fixed-size partitions
 - Adv:
 - simple (one size fits all)
 - Disadv:
 - Internal fragmentation (process smaller than fixed size hole)
 - Solution:
 - smaller partition size (split process over multiple partitions)
- Variable-size partitions
 - Adv:
 - Partition size matches size of process (initially)
 - Disadv:
 - External fragmentation (new process smaller than original variable-size hole)
 - Solution:
 - compaction (defragmentation)

Fixed-Size Partitioning

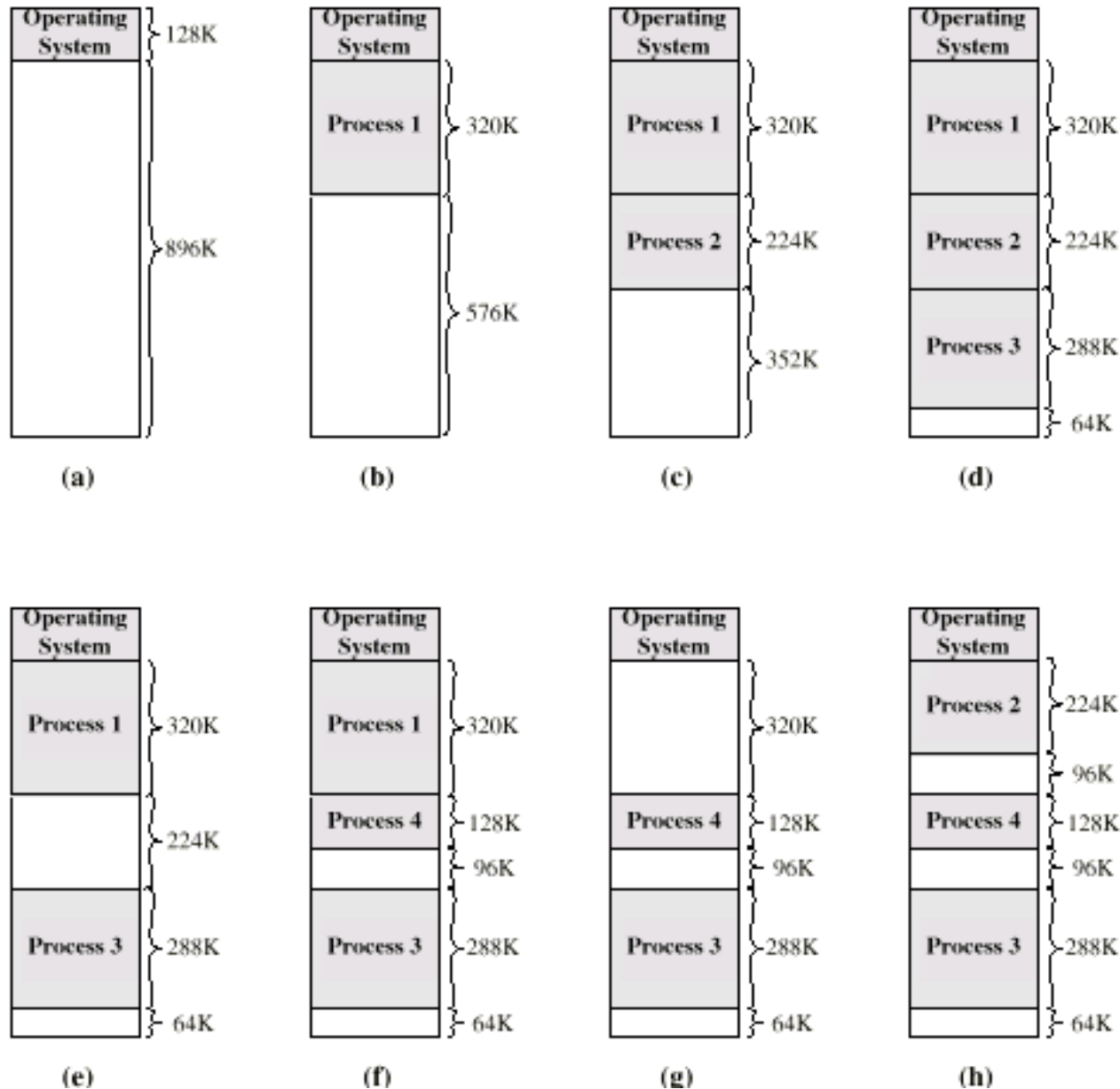


(a) Equal-size partitions



(b) Unequal-size partitions

Variable-Size Partitioning



Paging

- Page frames:
 - Equal-sized sections of Real (Main) Memory
- Pages:
 - Equal-sized sections of a process, each one fits into a page frame
- Page table:
 - Table maintaining frame location for each page

Virtual Memory

- Demand paging
 - Pages are brought into main memory as required
 - Allows handling of process larger than real memory
- Virtual-to-Physical address mapping
 - Virtual (logical)address is mapped to Real (physical) address

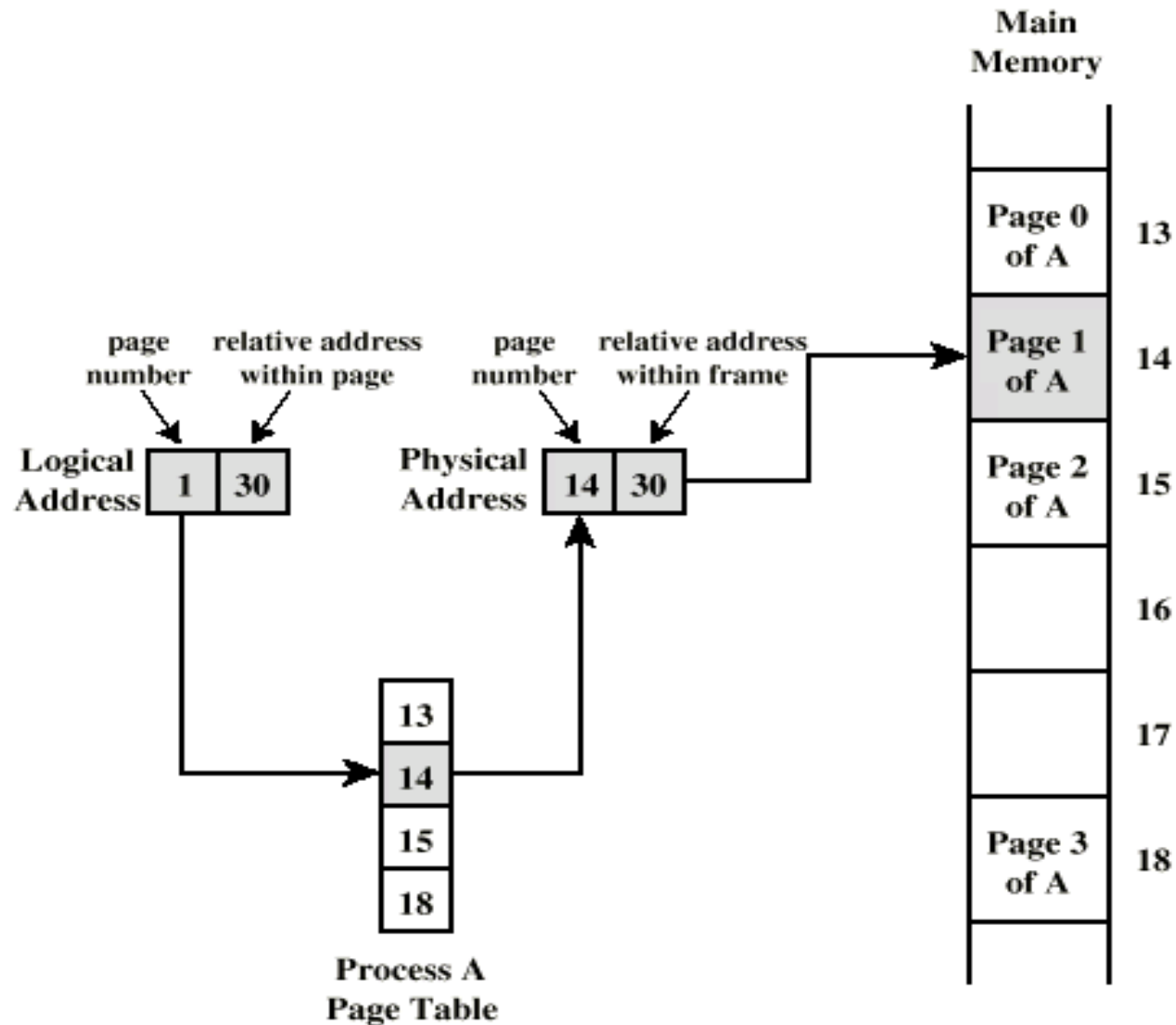
Virtual page = Virtual address DIV Page size

Page frame = Mapping(Virtual Page)

Offset = Virtual address MOD Page size

Real address = Page frame \times Page size + Offset

Virtual-to-Physical Address Mapping



Page Fault

- Problem:
 - Virtual page does not have a corresponding page frame
 - Same page may be visited again, causing more faults
- Solution:
 - Replace a current page from physical memory with faulted page
- Methods:
 - LRU algorithm: replace least recently used page in memory
 - FIFO algorithm: replace oldest page in memory
 - LFU algorithm: replace least frequently used page in memory
 - Random: replace any page in memory