Operating Systems

BITS Pilani
K K Birla Goa Campus

Dr. Lucy J. Gudino
Dept. of CS and IS

Last Class

- Single – Partition allocation
  - Hardware support for relocation and limit registers
- Memory Management Techniques
  - Fixed Partitioning
    - Fixed size
    - Variable size
      - One process queue per partition : lesser internal fragmentation and not an optimal solution
      - Single queue : optimum solution

Contd…

- Dynamic Partitioning : external fragmentation ➔ compaction
  - Four allocation schemes :
    - First fit
    - Best fit
    - Next fit
    - worst fit

Paging

- Fixed size and dynamic partitions are inefficient
  - Fixed size ➔ internal fragmentation
  - Dynamic➔ external fragmentation
- Paging helps to reduce internal fragmentation and completely eliminates external fragmentation

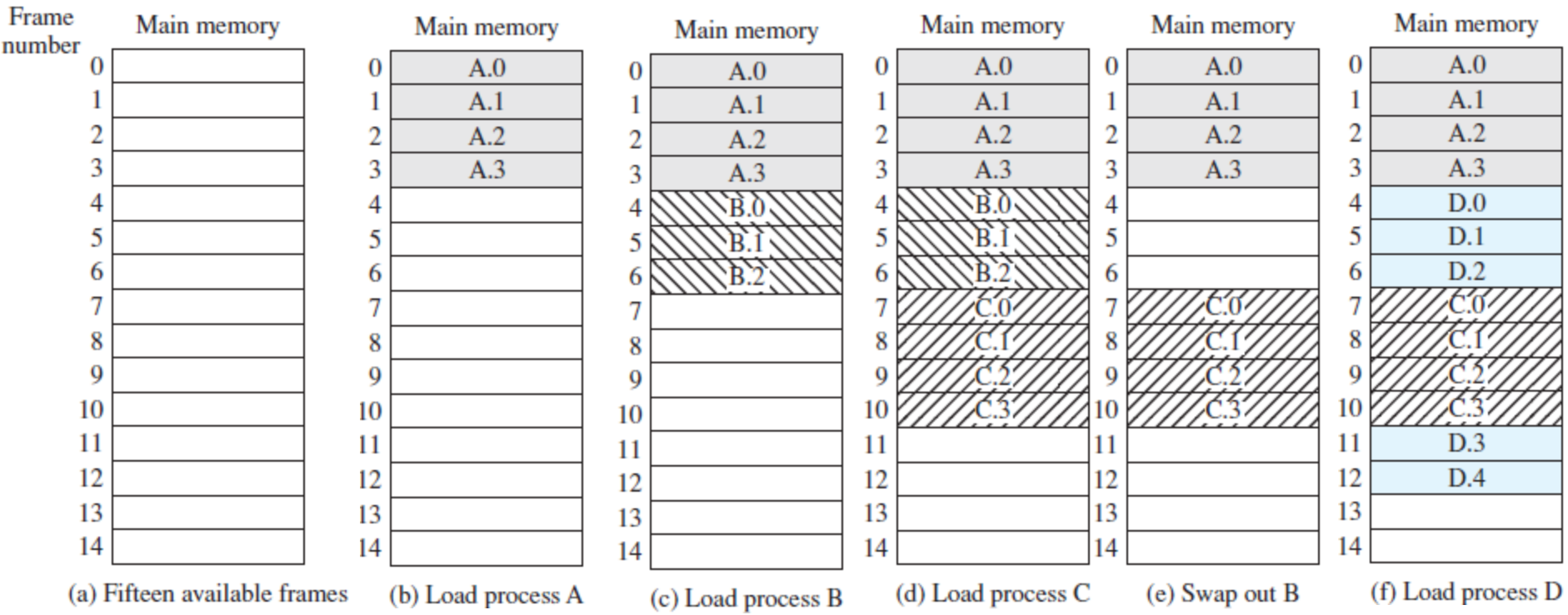
Paging

- Paging permits the physical address space of a process to be non- contiguous.
- Fragmentation problems in main memory and backing store
- Divide physical memory into fixed-sized blocks called **frames** (size is power of 2).
- Divide logical memory into blocks of same size called **pages**.
- Keep track of all free frames.
- To run a program of size *n* pages, need to find *n* free frames and load program.
- Set up a page table to translate logical to physical addresses.

Example

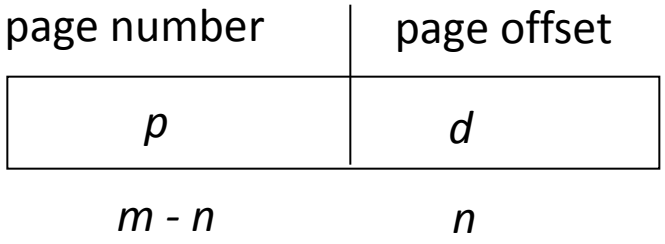
Process A : 4 pages Process B: 3 pages Process C: 4 Pages

Process D: 5 Pages Main Memory : 14 frames

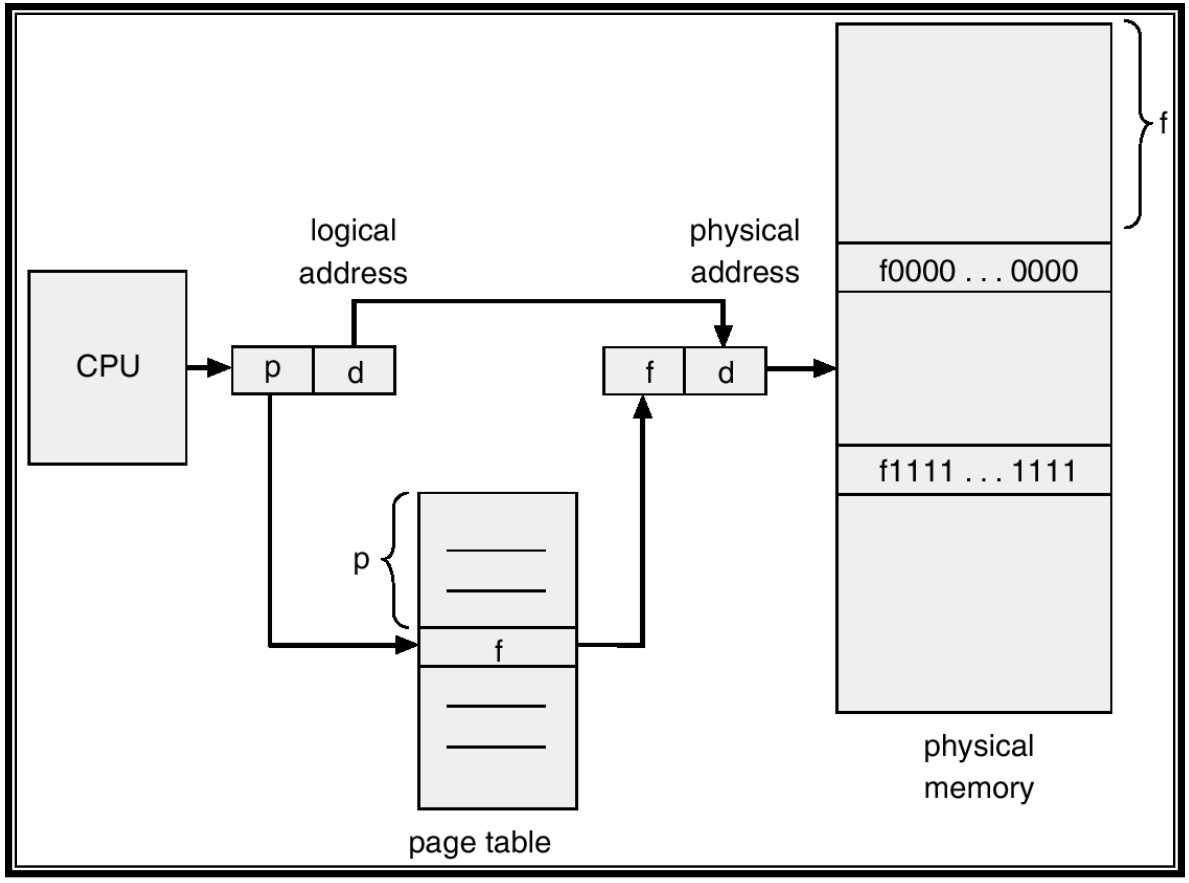


Address Translation Scheme

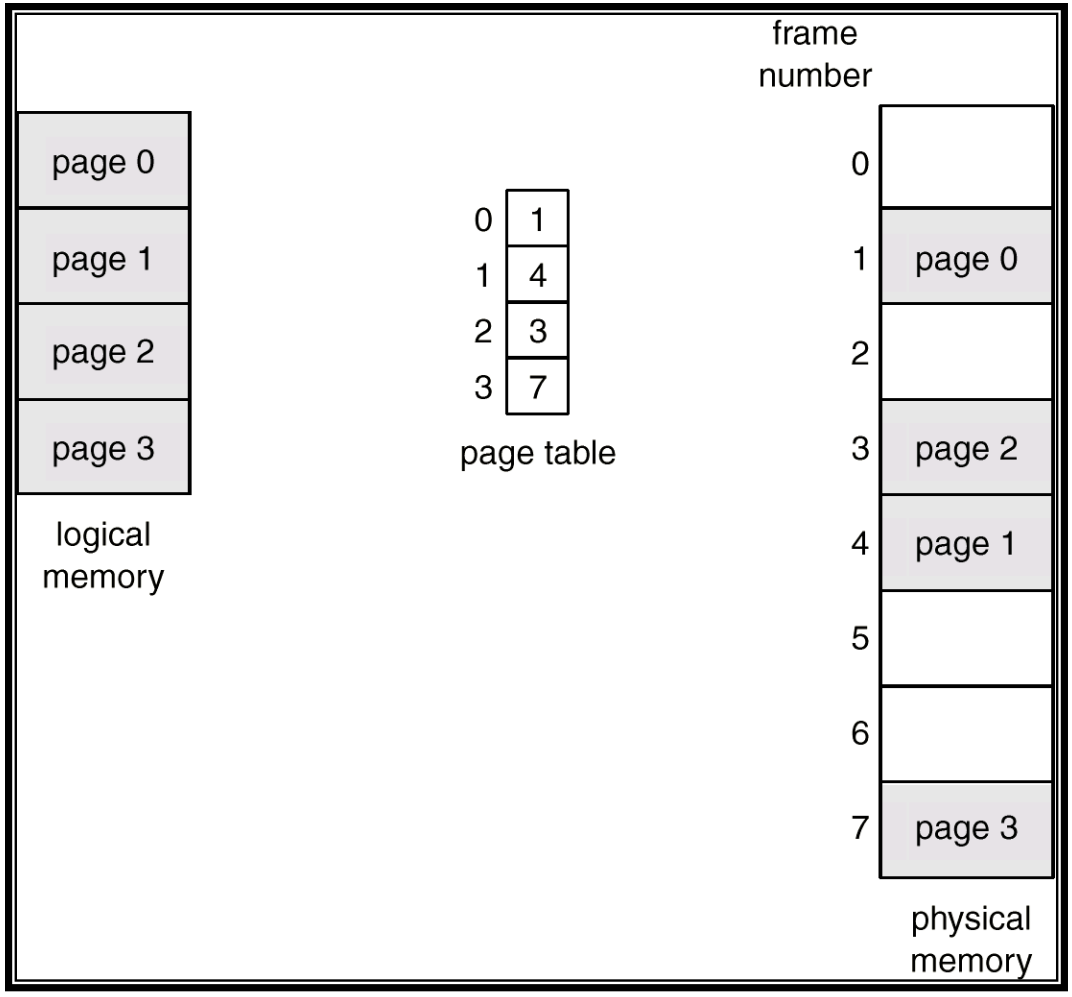
- Address generated by CPU is divided into:
  - *Page number (p)* – used as an index into a *page table* – Page table contains base address of each page in physical memory.
  - *Page offset (d)* – combined with base address to define the physical memory address that is sent to the memory unit.
  - For given logical address space 2^m *and page size* 2^n



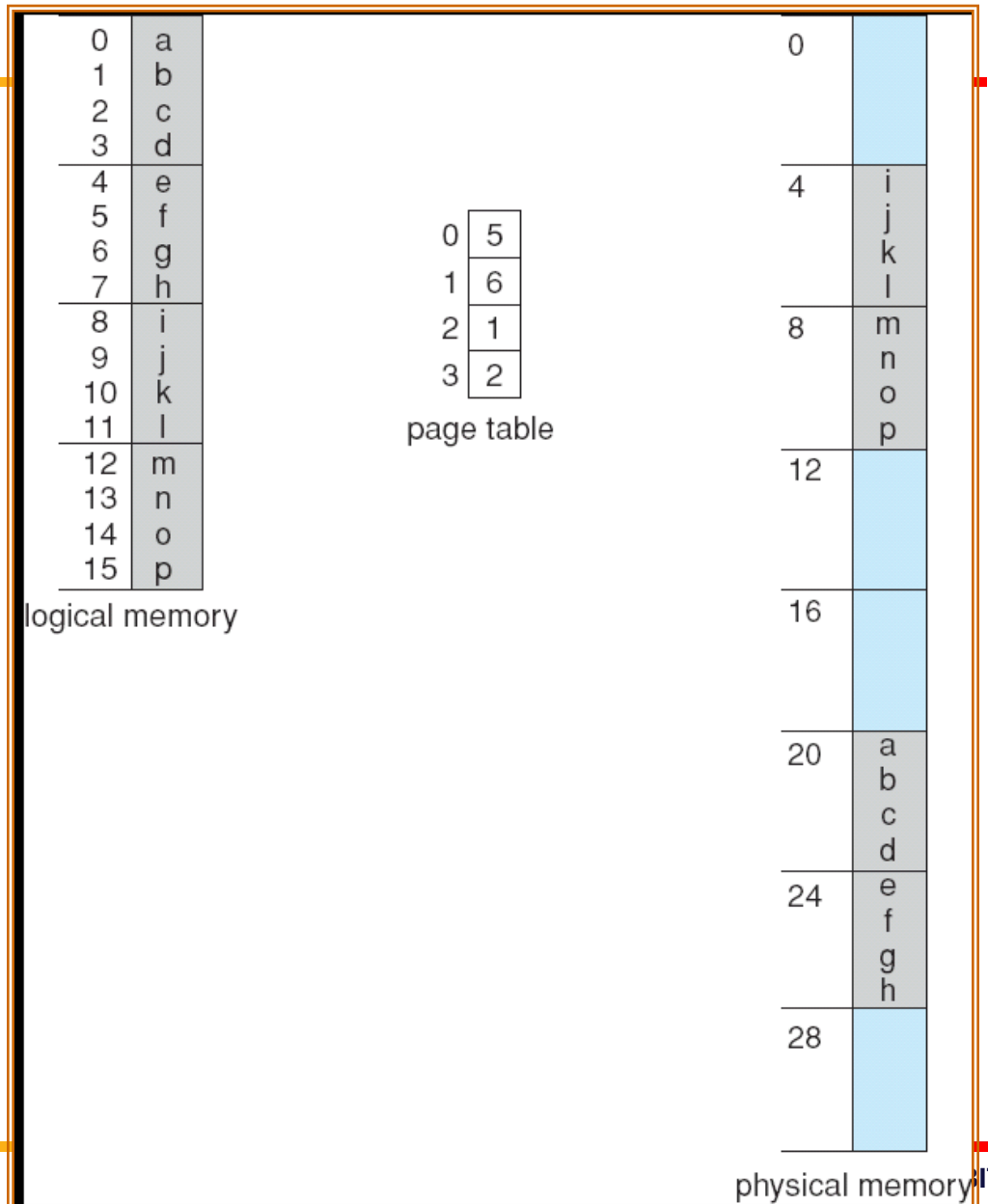
Address Translation Architecture



Paging Example



Paging Example



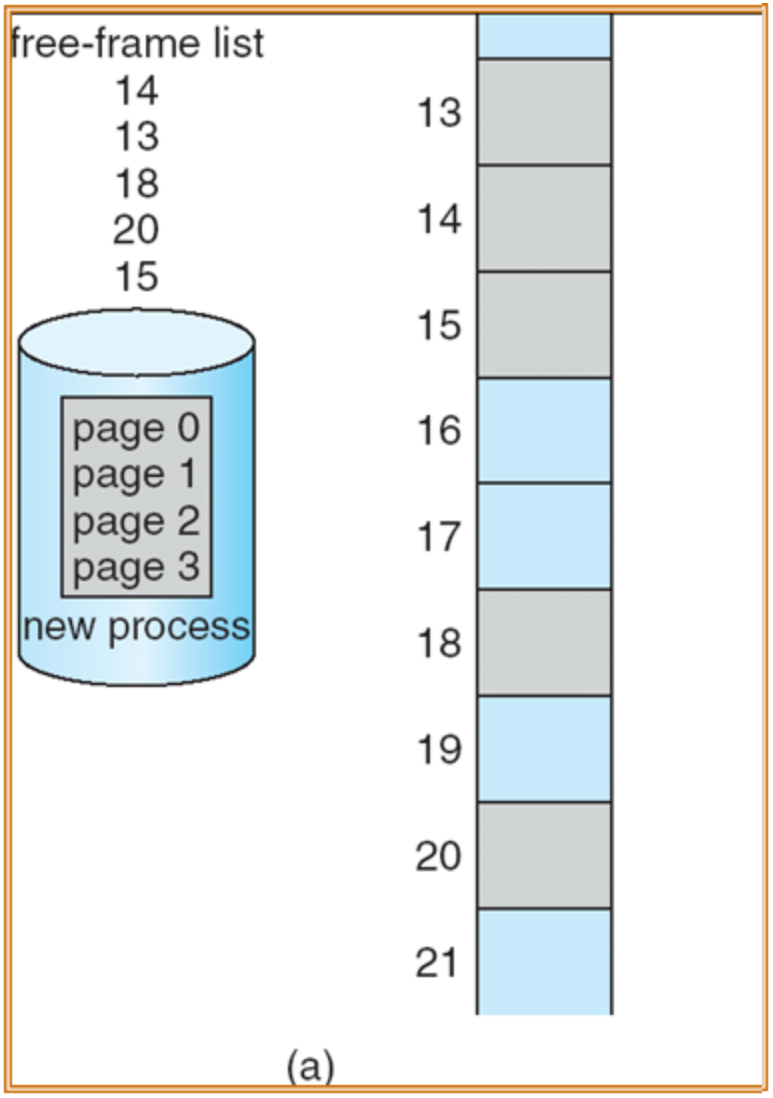
Important points….

- Paging is a form of dynamic relocation
- No external fragmentation but may have some internal fragmentation
- Small or large page size ?
- Page size determines
  - Memory wastage due to internal fragmentation
  - Size of the page table for a process
  - disk I/O data transfer
- small page : increases paging table overhead !!!!, internal fragmentation decreases
- large page : Decreases paging table overhead, internal fragmentation increases, disk I/O is more efficient when the number of data being transferred is larger

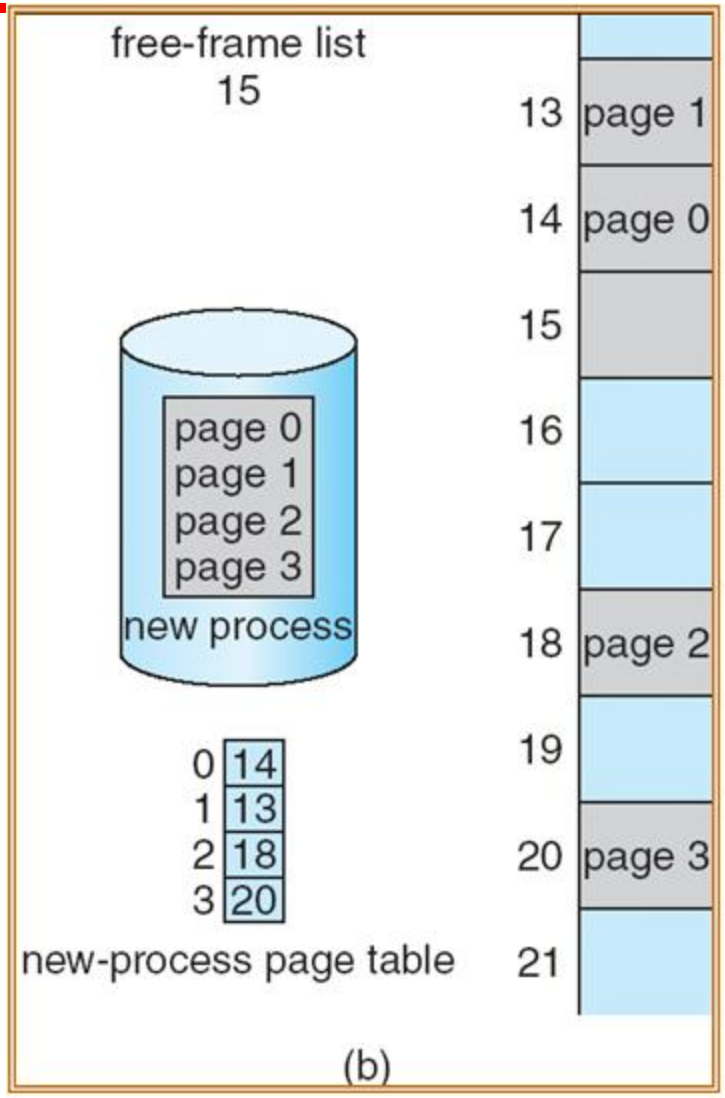
Contd…

- page size is usually 4 KB to 8 KB
- Needs to maintain the allocation details of main memory
  ➔ frame table

Free Frames



Before allocation



After allocation

Example

Consider a simple paging system with the following parameters: 2^{32} bytes of physical memory; page size of 2^{10} bytes; 2^{16} pages of logical address space.

a. How many bits are in a logical address?

b. How many bytes in a frame?

c. How many bits in the physical address specify the frame?

d. How many entries in the page table?

e. How many bits in each page table entry?

Hardware implementation of page table

- page table for each process
- Implementation of page table
  - Registers :
    - efficient paging – address translation
    - privileged instructions to load or modify page table
    - Ex: DEC PDP-11 : address of 16 bits, page size of 8 KB, number of entries in page table : 8
  - Page table in main memory
    - **Page table base register(PTBR)** points to the page table and **Page-table length register (PRLR)** indicates size of the page table
    - Changing page tables requires changing only PTBR register, substantially reducing context-switch time.
    - time consuming : 2 memory accesses

Contd…

- Solution : use a special, small, fast lookup hardware cache, called a translation look-aside buffer (TLB)
- Associative memory