Lab3 (how to approach)

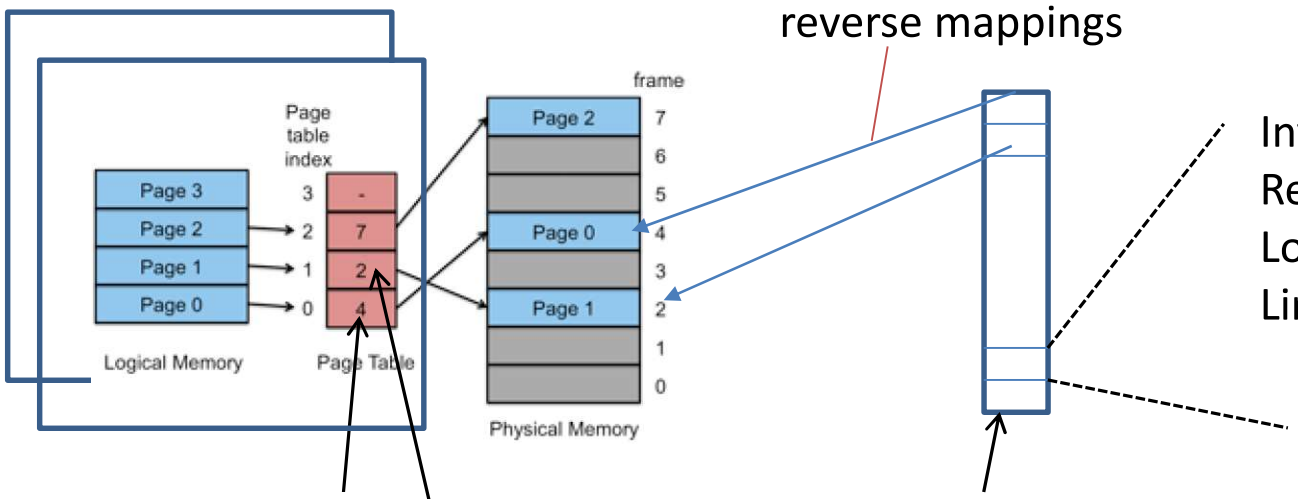
- You are emulating HW / SW behavior:
- For every instruction simulated
  - Check whether PTE is present
  - If not present → pagefault

    -> OS must resolve:
    - select a frame to replace (make pluggable with different replacement algorithms)
    - Unmap its current user (UNMAP)
    - Save frame to disk if necessary (OUT / FOUT)
    - Fill frame with proper content of current instruction's address space (IN, FIN,ZERO)
    - Map its new user (MAP)
    - Its now ready for use and instruction
  - Mark reference/modified bit as indicated by instruction

Lab3

- Create your frame table
- Make all algorithms run through frame table
- Make all algorithms maintains a "hand" from where to start the next "selection"
- On process exit() return all used frames to a free pool and start getting a free frame from there again till free pool is empty; then continue algorithm at hand.
  (this can at times select a frame that was just used .. C'est la vie .. You can make algorithms to complicated, rather make simple and occasionally not make an optimal decision)

Data Structures (also lab3)



PageTable (one per process)

FrameTable:

- This is a data structure the OS maintains to track the usage of each frame by a pagetable (speak reverse mapping).

- one entry related to each physical frame
- Used by OS to keep state for each frame

PageTable Entry (one per virtual page)