



Page Replacement

What happens if there is no free frame?

- While a process is executing, a page fault occurs.
- The operating system determines where the desired page is residing on secondary storage
- Then finds that there are ***no*** free frames on the free-frame list; all memory is in use.
 - Terminate the process – Not a good option
 - Use standard swap in swap out procedure – Not efficient, not used
 - Swapping pages with page replacement

Basic Page Replacement

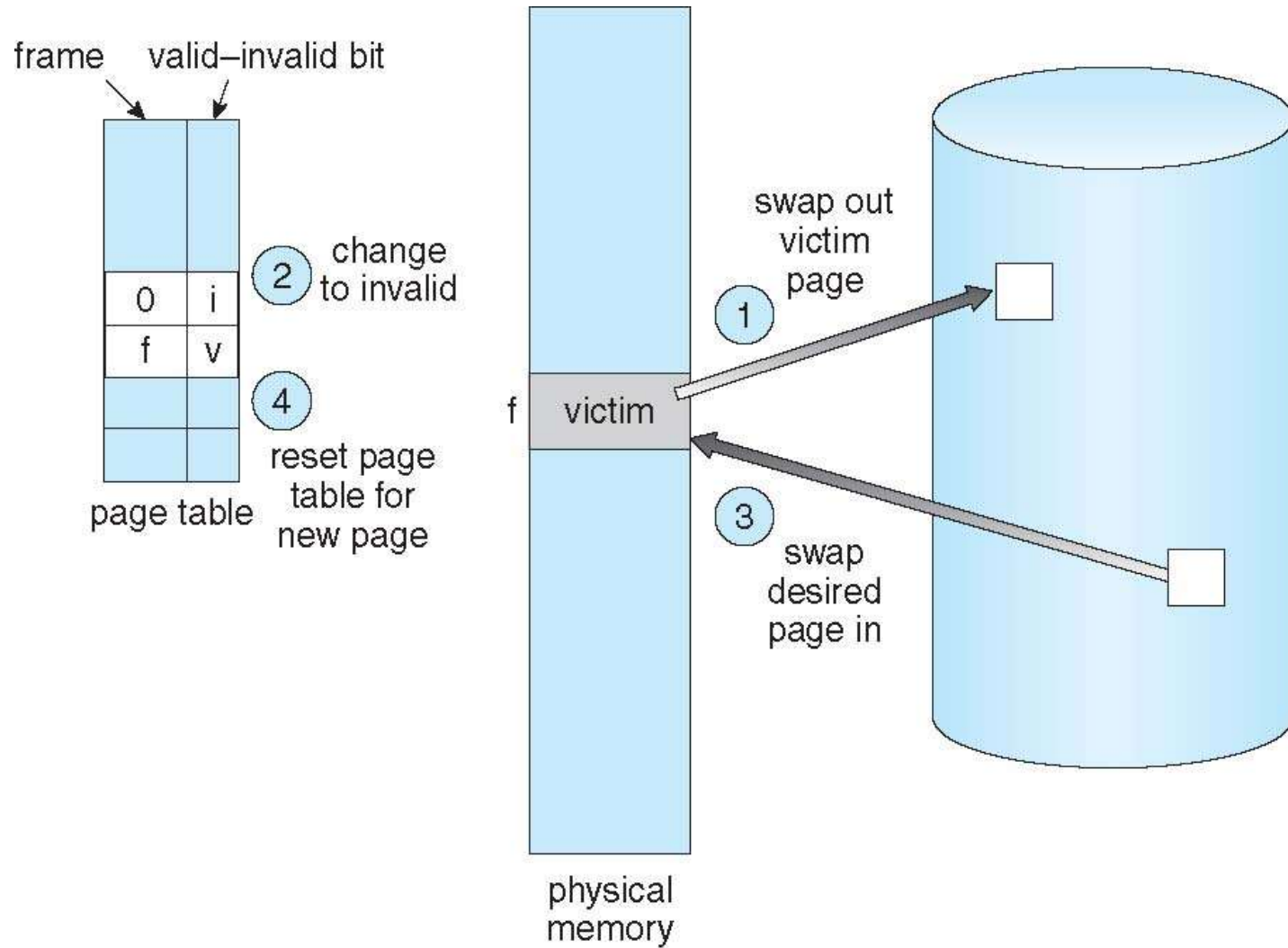
- If no frame is free, find one that is not currently being used and free it.
- We can free a frame by writing its contents to swap space and changing the page table (and all other tables) to indicate that the page is no longer in memory
- We can now use the freed frame to hold the page for which the process

Basic Page Replacement

1. Find the location of the desired page on disk
2. Find a free frame:
 - If there is a free frame, use it
 - If there is no free frame, use a page replacement algorithm to select a **victim frame**
 - Write victim frame to disk if dirty
3. Bring the desired page into the (newly) free frame; update the page and frame tables
4. Continue the process by restarting the instruction that caused the trap

Note now potentially 2 page transfers for page fault – increasing EAT

Page Replacement



Modify/Dirty Bit

- Each page or frame has a *Modify/Dirty* bit associated with it in the hardware.
- The *Dirty* bit = 1 \rightarrow indicates the page has been modified.
 - Should write back to disk before replacement
- The *Dirty* bit = 0 \rightarrow indicates the page has not been modified.
 - No need to write back to disk before replacement

Page and Frame Replacement Algorithms

- Two major problems
 - Frame allocation algorithm
 - How many frames to give each process
 - Which frames to replace
 - Page replacement algorithm
 - Want lowest page-fault rate on both first access and re-access

Page and Frame Replacement Algorithms

- Evaluate algorithm by running it on a particular string of memory references (reference string) and computing the number of page faults on that string
 - String is just page numbers, not full addresses
 - Repeated access to the same page does not cause a page fault
 - Results depend on number of frames available

Graph of Page Faults Vs Number of Frames

