CSE 120 - Discussion session 8

May 30, 2019

Swapping

- How do we maintain swapped pages on disk?
  - Use a swap file
- One for each process?
  - No. All processes share the swap file
- When and where to create the swap file?
  - Create on VMKernel Initialization
  - Use the underlying file system
  - File = ThreadedKernel.filesystem.open(swapFileName)
  - Choose a unique name

Swapping

- How to perform swap in/out?
  - Simply read and write to the file
  - File.read()/File.write()
- Delete the swap file on VMKernel Termination
  - ThreadedKernel.filesystem.remove(swapFileName)

How to allocate/reclaim swap space?

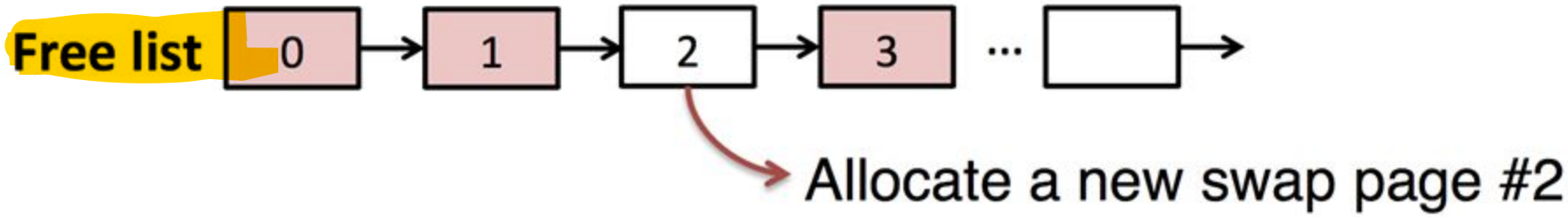
- The swap file can safely grow arbitrarily

- How did we allocate/reclaim physical pages in project 2?
  - Global free pages list
- Can we follow the same idea for the swap file?
  - Yes

- Unit of a swap file is a page
- Somehow maintain information about freeSwapPages

1. Swap out 5 pages
2. Swap in 2 pages
3. Swap out 1 page

So we need to record which places are available in the swap file



- Say we have located the swap page necessary for our operation
- Simply write/read the page to/from the file

Swap file - extension of main memory

- Now, a process's pages are distributed both in main memory and on disk

- Say we look up a vpn in the page table
  - Where can the physical page be?
    - memory or swap file
  - How do we know if the physical page has been swapped out?
    - Valid bit

Updating PTE after Swapping

- How do we know the page index or spn of a page in the swap file?
  - Need to record relation between vpn and spn
    - Why not ppn?
    - Where to record relation?

How to find PTE of victim page?

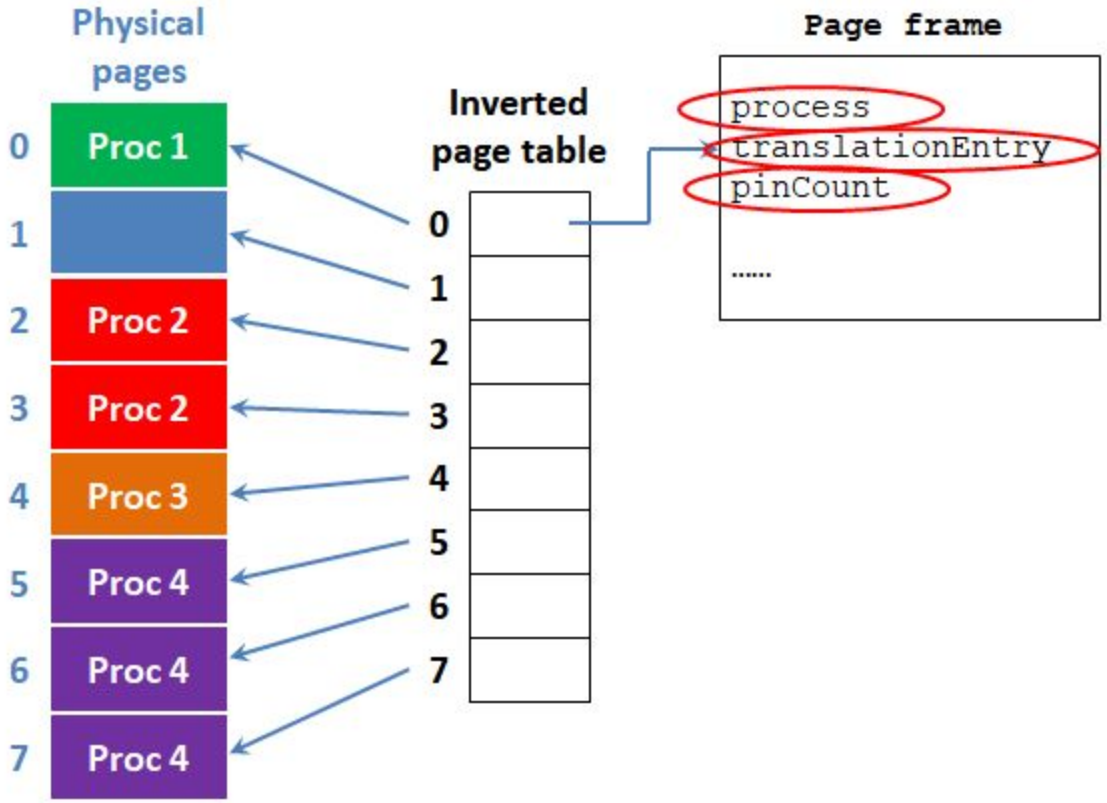
- PTE is indexed by vpn in the page table

- OS picks the victim page from the whole physical page pool
- OS does not know which process the victim page belongs to
- It does not know the corresponding vpn of the victim page

- We thus need a data structure to keep track of the physical pages

Inverted Page Table

- Used to correlate physical page with process and page table entry info
- What should be the index?
  - Physical page number
- Note: When updating a PTE, also update the IPT


Physical pages
0
Proc 1
1
2
Proc 2
3
Proc 2
4
Proc 3
5
Proc 4
6
Proc 4
7
Proc 4
Inverted page table
0
1
2
3
4
5
6
7
Page frame
process
translationEntry
pinCount
.....

Corner case for Page Replacement

- How to prevent the OS from swapping out specific pages?
  - Pinning
- For page replacement, what if all the pages are pinned?
  - Page allocation must wait until some process unpins a page
  - Think about how to implement this

Synchronization

- Identify shared data structures
- Identify relation between shared data structures
  - i.e. independent or correlated?
- Do we need a different lock for each?
- Lock granularity:
  - Coarse grained vs fine grained
  - Pros and cons of both