

Assignment 2 (Operating system)

****Swapping on Mobile system***

SWAPPING

• *Swapping is a mechanism in which a process can be swapped/moved temporarily out of main memory to a backing store , and then brought back into memory for continued execution.*

- For example, assume a multiprogramming environment with a round-robin CPU-scheduling algorithm. When a quantum expires, the memory manager will start to swap out (move) the process that just finished, and to swap in(bring back) another process to the memory space that has been freed.
- A process must be loaded into memory in order to execute.
- If there is not enough memory available to keep all running processes in memory at the same time, then some processes who are not currently using the CPU may have their memory swapped out to a fast local disk called the backing store.

STANDARD SWAPPING

- If compile-time or load-time address binding is used, then processes must be swapped back into the same memory location from which they were swapped out.
- If execution time binding is used, then the processes can be swapped back into any available location.
- Swapping is a very slow process compared to other operations.
- To reduce swapping transfer overhead, it is desired to transfer as little information as possible, which requires that the system know how much memory a process is using, as opposed to how much it might use.
- Programmers can help with this by freeing up dynamic memory that they are no longer using.
- It is important to swap processes out of memory only when they are idle, or more to the point, only when there are no pending I/O operations.
- Swap only totally idle processes, or do I/O operations only into and out of OS buffers, which are then transferred to or from process's main memory as a second step.
- Most modern OSes no longer use swapping, because it is too slow and there are faster alternatives available. (e.g. Paging.)

Assignment 2 (Operating system)

- **Example of OS which use swapping.** Some UNIX systems will still invoke swapping if the system gets extremely full, and then discontinue swapping when the load reduces again.

Windows 3.1 would use a modified version of swapping that was somewhat controlled by the user, swapping process's out if necessary and then only swapping them back in when the user focused on that particular window.

SWAPPING ON MOBILE SYSTEMS

- Swapping is typically not supported on mobile platforms, for several reasons.

- o Mobile devices typically use flash memory in place of more spacious hard drives for persistent storage, so there is not as much space available.

- o Flash memory can only be written to a limited number of times before it becomes unreliable.

- o The bandwidth to flash memory is also lower.

- Apple's IOS asks applications to voluntarily free up memory

- o Read-only data, e.g. code, is simply removed, and reloaded later if needed.

- o Modified data, e.g. the stack, is never removed, but Apps that fail to free up sufficient memory can be removed by the OS

- Android follows a similar strategy.

- o Prior to terminating a process, Android writes its application state to flash memory for quick restarting.