

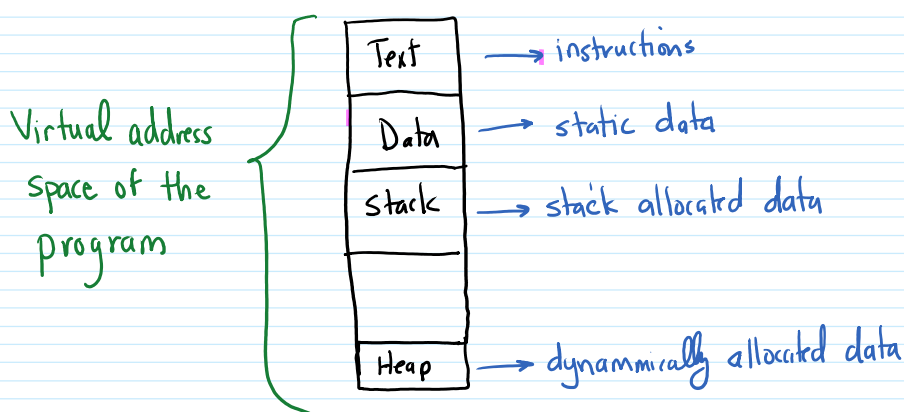
Virtual Memory → Main memory management → OS

Recall: In order to make memory translation table manageable, we needed refer to memory spots in larger granularity

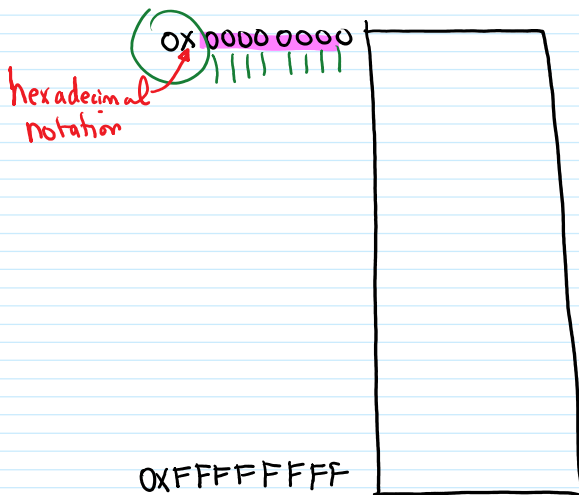
Page: → fixed size unit of memory (contiguous memory locations) for which a single piece of translation information is maintained  
→ "neighbouring"

## Paged Virtual Memory

- When a program is in execution, it can be thought of as a process
- A process has many things which must be present in memory for it to execute.



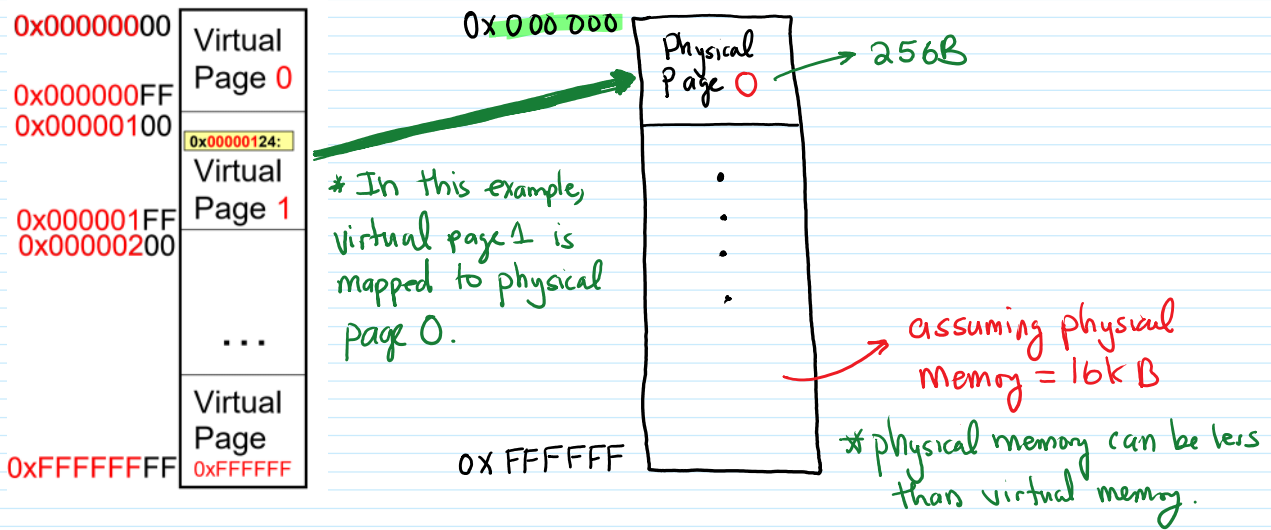
- The virtual address space can also be viewed as addresses 0 to some maximum possible address



- If I'm going to manage the address space in units of 256B, then the actual addresses associated with each unit can be calculated.



- We refer to this as the physical address space.

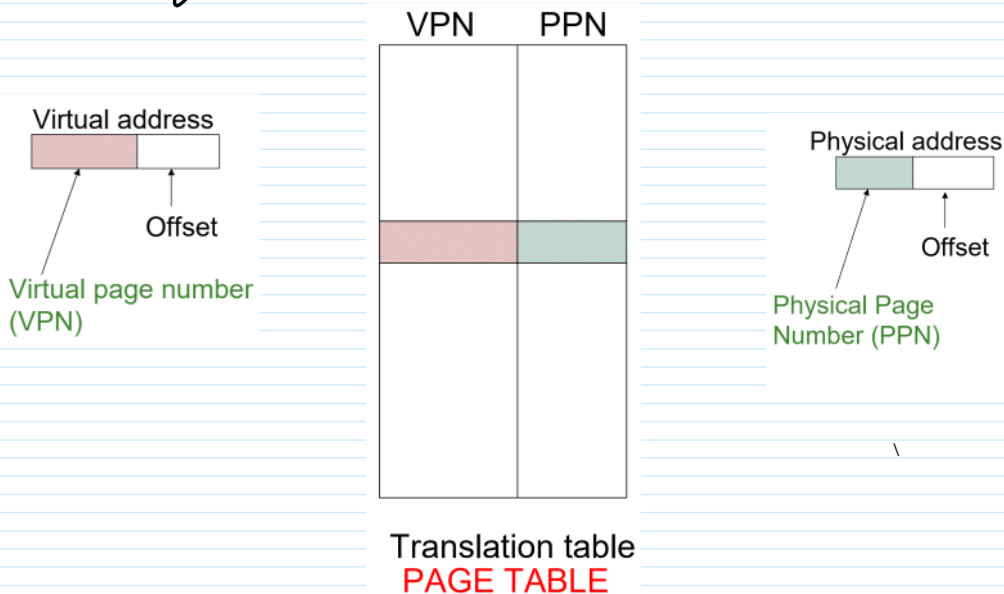


Virtual Address Space

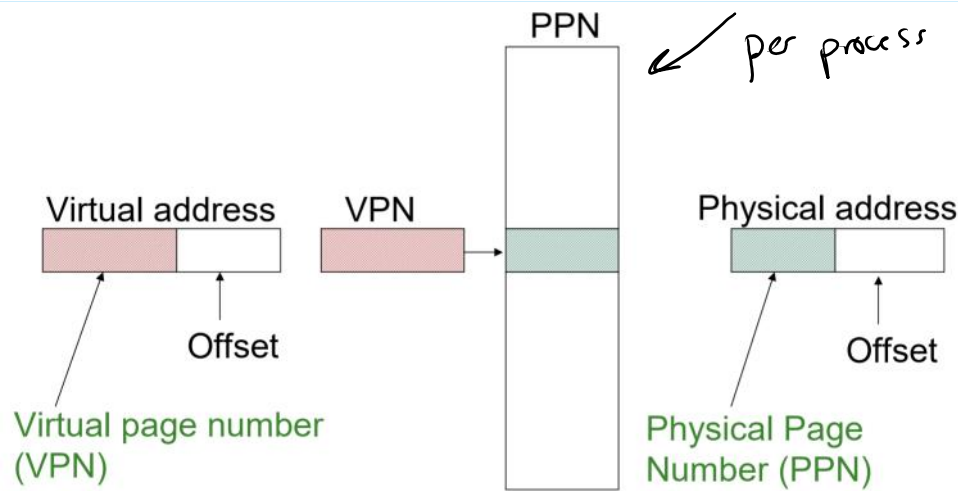
Physical Address Space

## Address Translation

- Purpose is to translate virtual address as generated by the processor into a physical address.
- This virtual address needs to be translated before it can be interpreted or sent to memory for accessing the data or the instruction



## Actual Translation Implementation



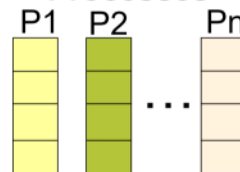
PAGE TABLE \* Incomplete

- We will augment this table as we go along
- We still don't know how many processes will share the same main memory

Main Memory



Processes

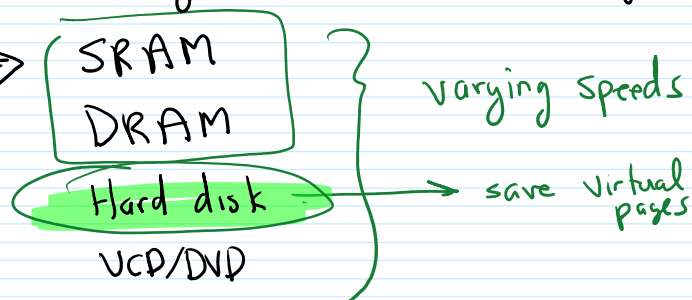


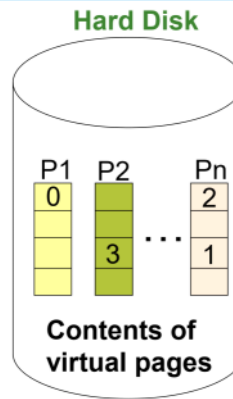
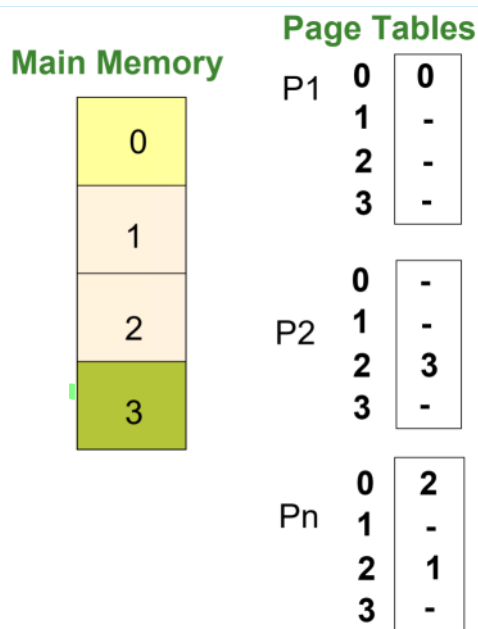
Contents of virtual pages

## Memory

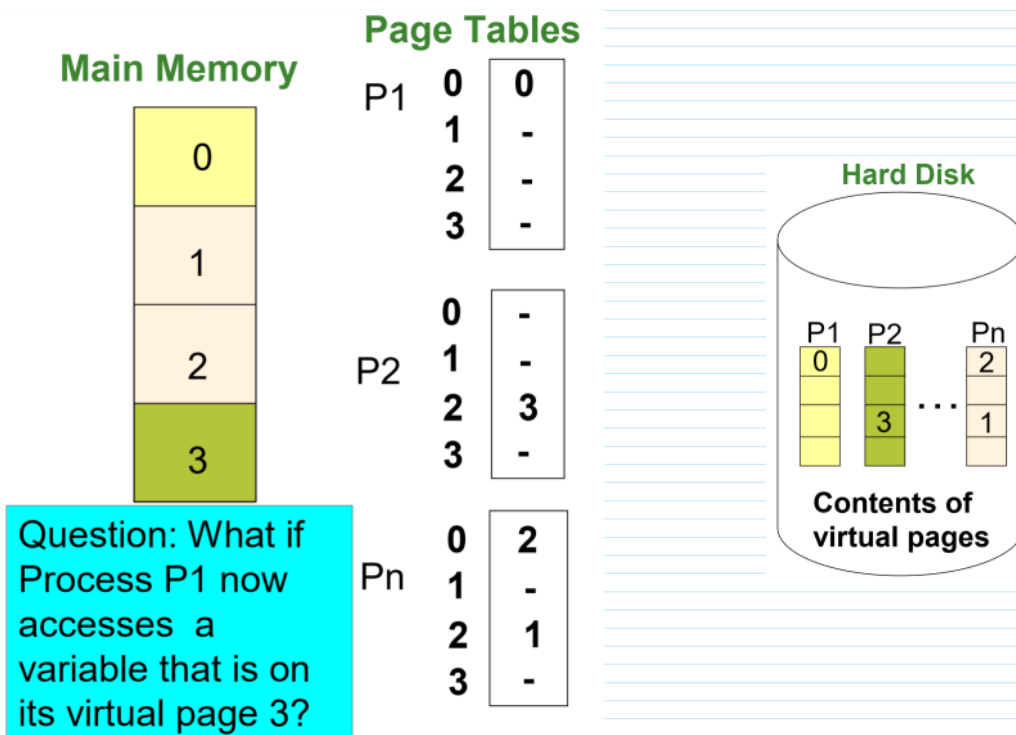
- What is memory?  $\Rightarrow$  something that can remember things

- Different kinds  $\Rightarrow$



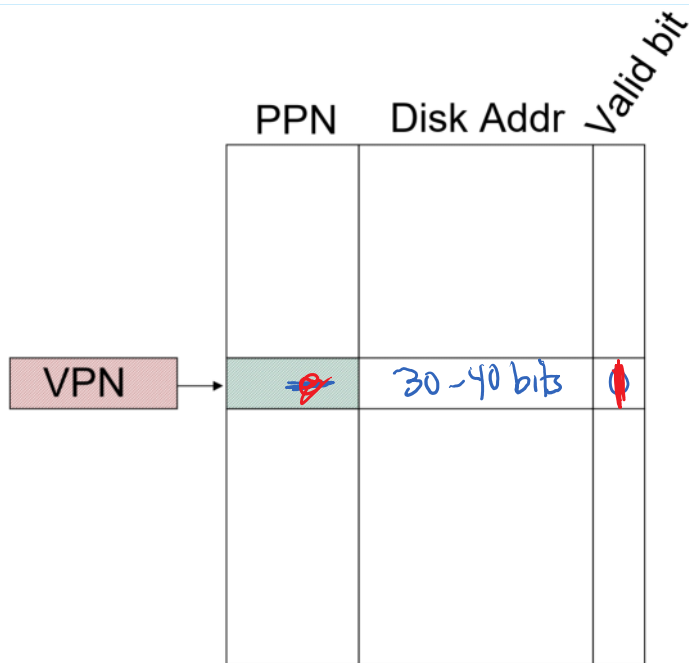


- What if a page is requested that is not in the page table?



**Page Fault**

- Situation where virtual address generated by the processor is not available in the main memory.



- Must be "handled" by the operating system.

↳ identify a slot in main memory \*

↳ get page from disk

↳ copy over

↳ update page table entry.

Empty slot  
link list  
free list

Data can be  
provided to  
the process.