CS/COE1541: Introduction to Computer Architecture

Dept. of Computer Science
University of Pittsburgh

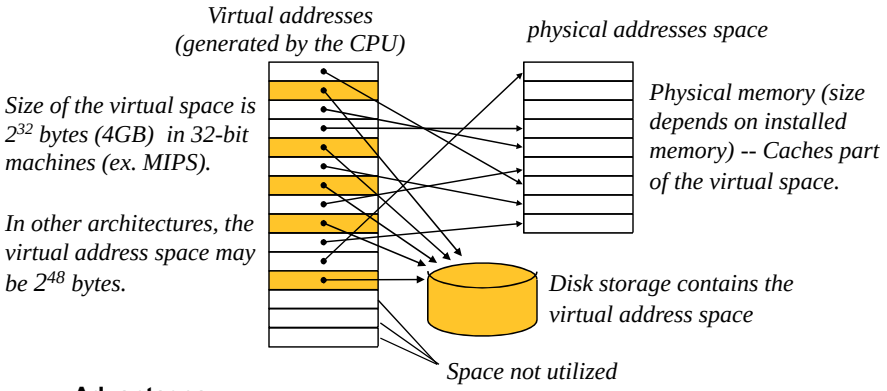
http://www.cs.pitt.edu/~melhem/courses/1541p/index.html

Chapter 5: Exploiting the Memory Hierarchy
Lecture 6: Virtual Memory

Lecturer: Rami Melhem

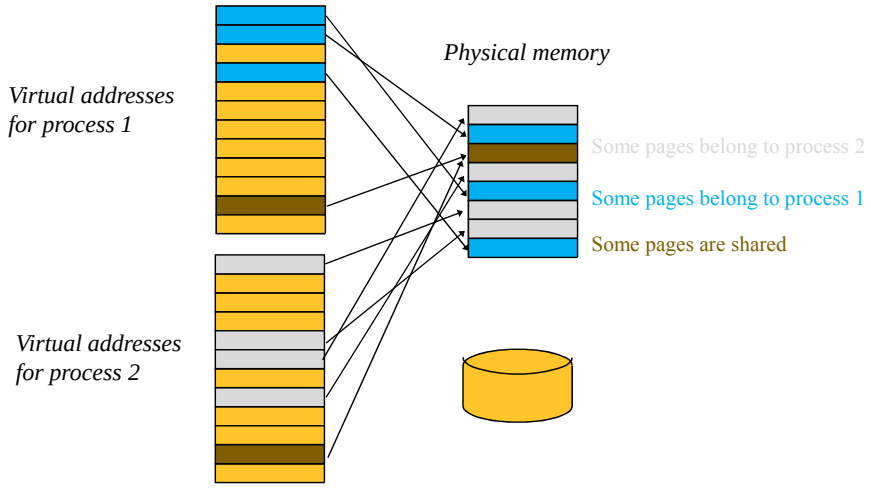
Virtual Memory (section 5.7)

Main memory can act as a cache for the secondary storage (ex. disk)



- **Advantages:**
  1) illusion of having more memory than what can fit in the physical memory

Virtual Memory

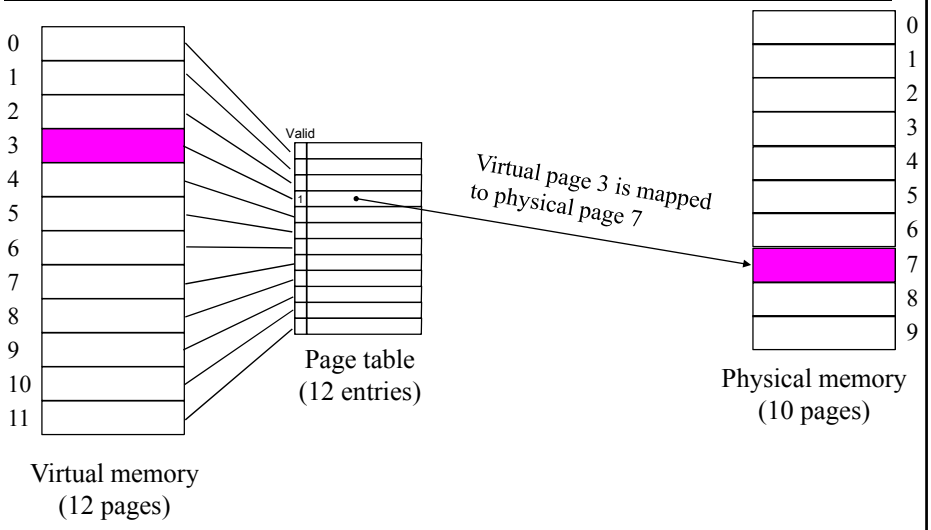


- **Advantages:**
  2) More than one process can share the same physical memory
  3) Allows page sharing among processors

Physical memory as a cache for the virtual memory

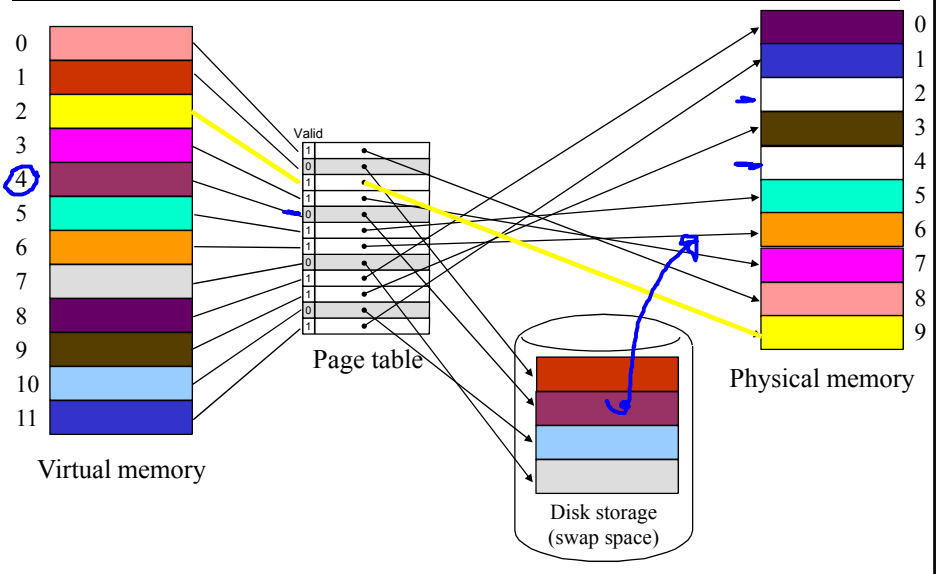
- Block (page) size = 1KB ~ 64KB (large because of huge miss penalty)
  - (as opposed to16B~64B in L1/L2)
- Miss penalty (cost of a page fault) = 1M~10M cycles (to access hard drive)
  - (as opposed to 20~150 cycles (L2/memory)
- Hit time = 50 ~ 150 cycles
  - (as opposed to1~3 in L1 and 6~12 in L2)
- Miss rate = 0.00001~0.001% (called page fault rate)
  - (as opposed to 0.1~10% in L1)
- Always write back (never write through)
  - (always write back in L2, and commonly in L1)
- Fully associative (to maximize hit rate)
  - (as opposed to 4-8 way set associative in L1/L2)
- Replacement implemented in OS
  - (as opposed to hardware implementation for L1/L2)
- Locate pages using page tables
  - (efficient way to implement full associativity)

Address translation through Page Tables



• *Table lookup replaces tag comparison in L1/L2.*

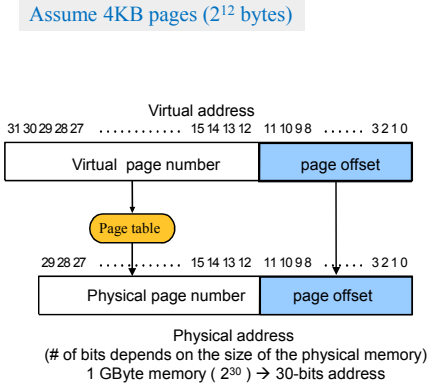
Address translation through Page Tables

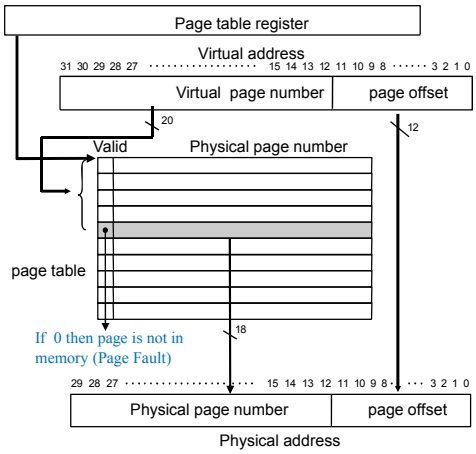


• *Table lookup replaces tag comparison in L1/L2.*

Virtual to physical address translation

Assume 4KB pages (2^{12} bytes)

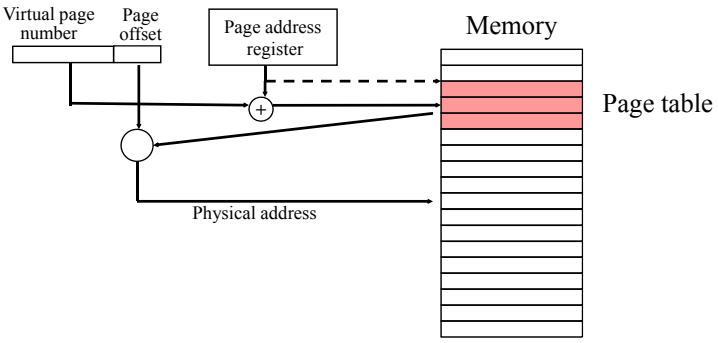




In this example, the physical memory is 1/4 the size of the virtual address space.

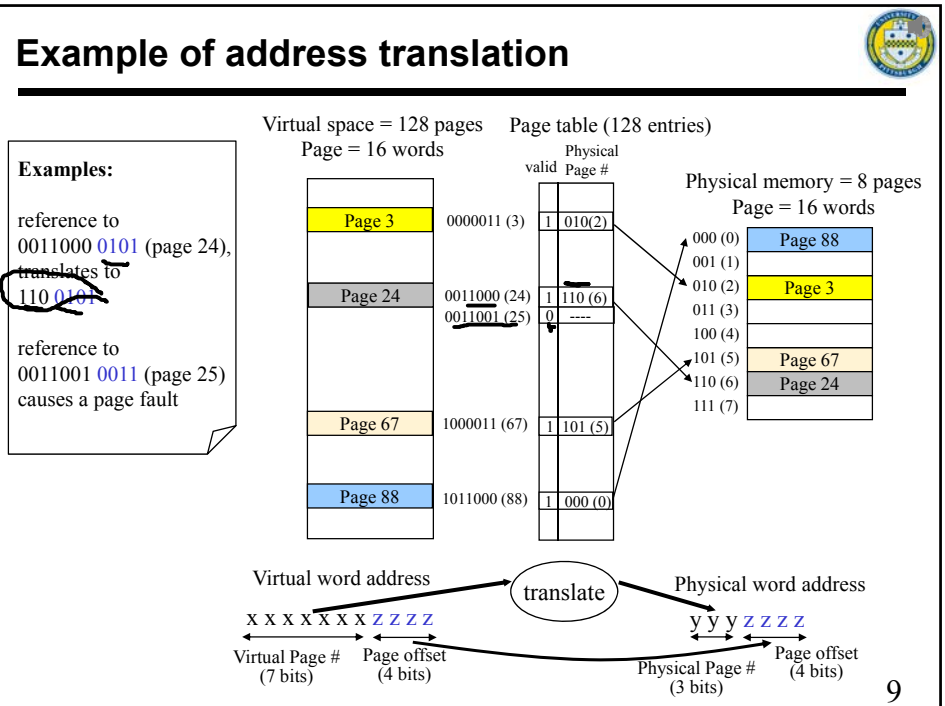
The page table is stored in memory starting at the address stored in the "Page Table Register"

The Page Table is allocated in memory

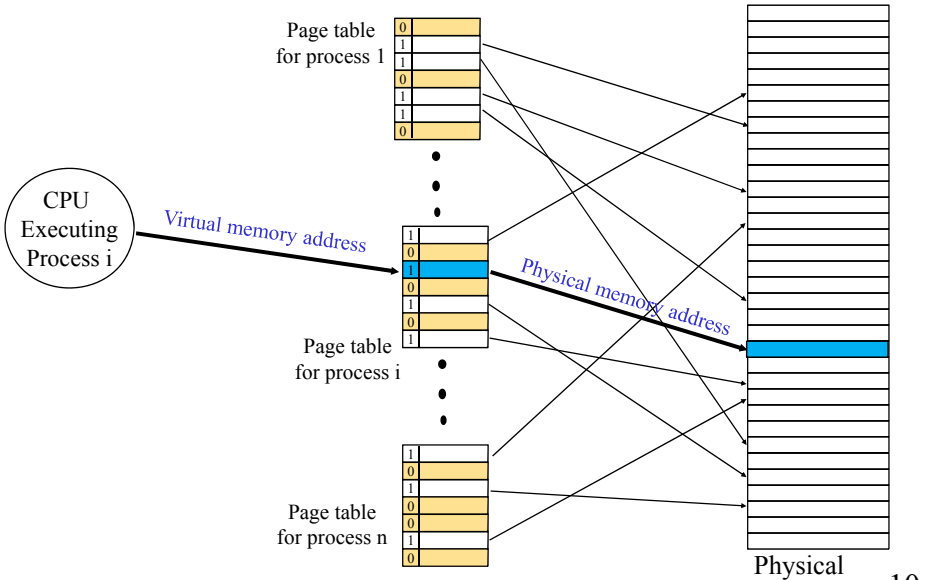


- Each process has its own page table stored in memory starting at a specific address indicated in a *page address register*.
- A memory reference (if hits in main memory) requires two memory operations
- A page fault (main memory miss) results in a disk operation.
- The page table and page address register are part of the process context (along with the PC, stack pointer, registers …)

Example of address translation



Multiple processes share the physical memory



Multiple processes share the physical memory