Virtual Memory

Virtual Memory is the process of mapping a logical address space numbered from 0 to the physical address space of the computer so that the RAM serves as a cache for the program's memory stored on the disk. The mapping is performed on a page basis.

Virtual Memory

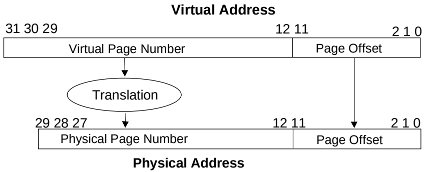
Virtual memory solves several critical problems:

- Simplifying RAM memory management by separating the address of memory from its physical location
- Providing protection for users by giving them their own address spaces
- Simplifying compilation and usage of libraries by allowing all programs to begin addressing memory from zero

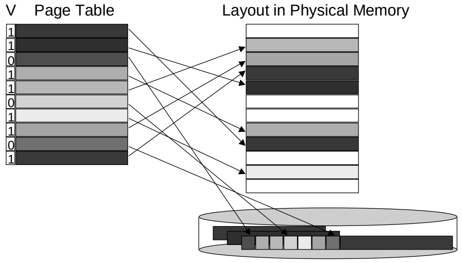
Virtual to Physical Translation

Physical memory (RAM) is divided into pages -- contiguous sequences of memory typically in the 4KB -- 16KB range

The pages are blocks of a fully associative cache for the memory of the program which is stored in the next lower level of the memory hierarchy (disk)



Page Table

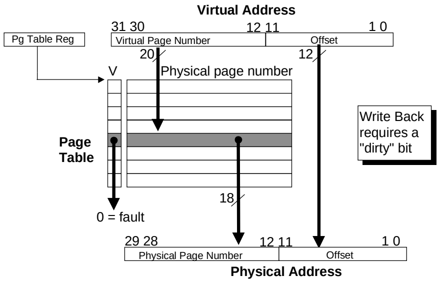


Page table and disk addresses may be kept in separate tables

VM Considerations

- Large page sizes allow the huge miss penalties to be amortized over many references
- Avoiding premature page replacement, e.g. by address collisions, extends page life and reduces fault rate ... thus full associativity
- Page faults can be handled in software, which can use cleverness to reduce fault rate
- Write back is needed because of the high cost of writes

Implementing Address Translation



Write Back requires a "dirty" bit

Page Replacement Strategy

When the Valid = 0, a page fault is signaled.

Some page must be replaced -- pick the page that will be used furthest into the future: Opt

Replace the least recently used (LRU) page.

LRU strategies are effective, but expensive.

"Use" bits can be a decent approximation.

Space required for page tables can be substantial:

4K pages imply 20 bits of virtual address.

At x bytes per entry implies xMB for full page table.

Represent only the prefix of the table using base + extent.

Grow low addresses and high addresses separately by using the msb to indicate which part of the VM space is being used.