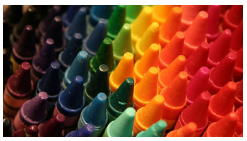
Virtual memory

A new level of cache

CS240 Computer Organization
Department of Computer Science
Wellesley College

Crowded memory

- Total memory required by a collection of programs running at once on a computer may be larger than main memory.
- However, only a fraction of this memory is actively used at any given moment.
- Virtual memory allows programs to share memory by using main memory as a cache for secondary storage.



User protection

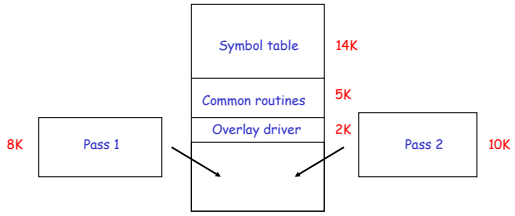
- Each program is compiled into its own private address space, a separate range of memory accessible to only this program.
- Virtual memory translates the program's address space into physical addresses.
- It also enforces protection of a program's address space from other programs.

No room at the Inn

- In the late 1950s the leading scientific computer, the IBM 650, had only 2000 words of memory. Memory was precious.
- Programmers spent spent a lot of time trying to squeeze programs into tiny memory
- Fortunately, the entire program need not be in memory throughout its execution.

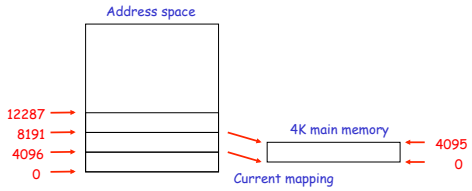
Overlays

- A technique called overlays was used to allow a program to be larger than the amount of memory allocated to it.



Problems with overlays

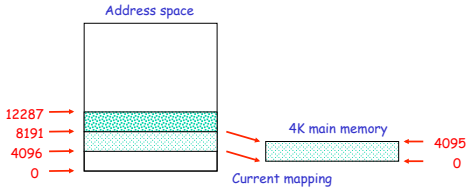
- The programmer was responsible for breaking her program into pieces, deciding where in secondary memory each piece goes . . .



- . . . and arranging for the transport between main memory and secondary storage.

Virtual memory

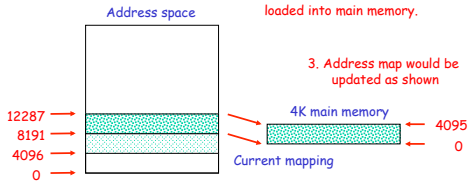
- In 1961 a group at Manchester, England proposed a method for performing the overlay process automatically.



- A jump from location 5012 to location 9140 causes an automatic page fault. The operating system is called without the programmer being aware.

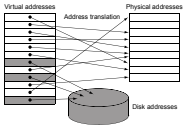
Virtual memory in action

1. Contents of main memory would be saved on disk.
2. Words 8192-12287 would be loaded into main memory.
3. Address map would be updated as shown
4. Program execution resumes as though nothing happened.



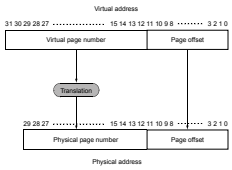
Virtual memory

- Virtual address space is broken into a number of equal sized pages.
- Memory is broken into the same size chunks known as page frames.
- Some pages in virtual memory are in physical memory, some are on disk.
- A virtual memory miss is called a page fault.



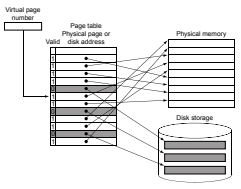
Mapping from virtual to physical address

- The address is broken into a virtual page number and a page offset.
- The page size in this example is 2^{12} = 4 KB.
- The size of physical pages is 1 GB, while virtual memory is 4 GB. Why?

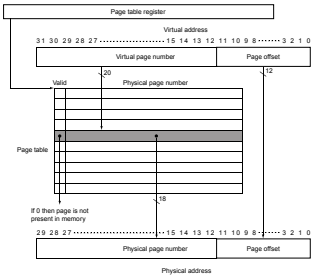


Page tables

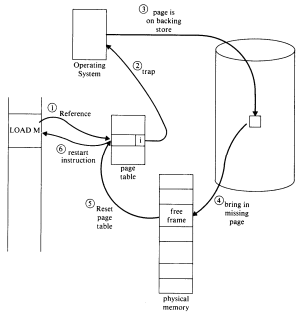
- Each program has its own page table that resides in memory.
- The page table is indexed with the virtual page number and points to the actual page in physical memory (either main memory or backing store).



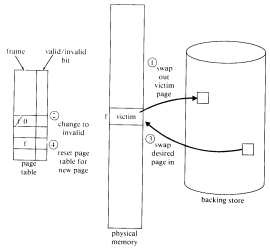
Page table register points to page table



Page faults



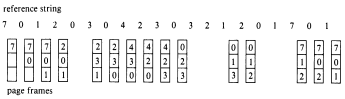
Page replacement algorithms*



*When physical memory fills up and a page fault occurs, somebody's got to go.

FIFO*

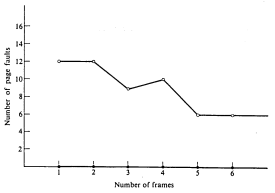
- When a page must be replaced, the oldest page in memory sleeps with the fishes.



*This is the simplest, but requires some form of time stamp.

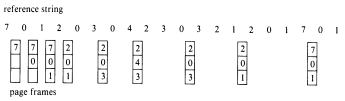
Belady's anomaly

- FIFO's performance isn't always optimal.
- It also suffers from an odd little anomaly. Consider the reference string: 1, 2, 3, 4, 1, 2, 5, 1, 2, 3, 4, 5.



Optimal replacement

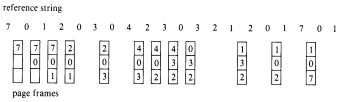
- Replace the page which will not be used for the longest period of time.



*Unfortunately, the optimal page replace algorithm is difficult to implement since it requires predicting the future.

Least recently used*

- Replace the page that has not been used for the longest period of time.



*We try to predict the future, by observing the past.

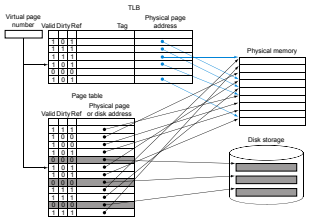
Reference bits

- Implementing an accurate LRU is too expensive, since it requires updating a data structure on every memory reference.
- Most systems provide a use or reference bit, which is set whenever a page is accessed.
- The OS periodically clears the reference bits, so it can determine which pages have been touched during a time period.

What about writes?

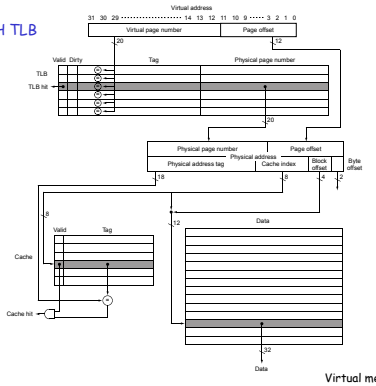
- Writes back to the disk take millions of processor cycles, so building a write buffer is not practical.
- Instead, virtual memory systems must use write-back, copying the page back to disk when it is replaced.

The translation-lookaside buffer*

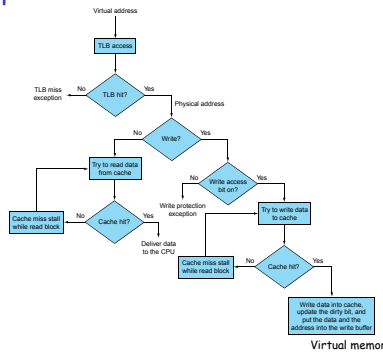


*Typical values: 16-512 entries; miss-rate = .01% - 1%; miss-penalty = 10 – 100 cycles.

Intrinsity FastMATH TLB

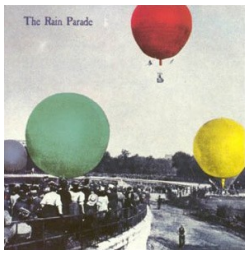


TBL hit parade



Raining on our parade

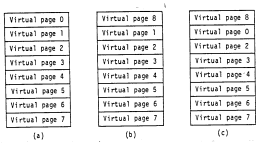
- When a TLB miss occurs, MIPS hardware saves the page number in a special register and generates an exception.
- The OS handles the miss in software by indexing the page table through the page table register.
- The system places the physical address from the page table into the TBL.*



*Takes about 13 clock cycles, assuming the code and page table entry are in the instruction and data cache respectively.

Thrashing

- If the working set is larger than the number of available page frames, no algorithm short of OPT will give good results.



*Assume a LRU algorithm.