CS 350 Notes

Matthew Visser

Nov 1, 2011

1 Exploiting Secondary Storage

We have memory and we also have a hard disk that we can utilize. We can write memory pages to disk instead of RAM to increase our virtual memory size. This is called *paging*.

We utilize a *two-level page table* to keep track of where the memory is. See slide 46 of `vm.pdf`.

We can also use an *inverted page table*.

- We have one inverted table per frame of physical memory.
- Because page tables are expensive, we don't want to go to them for every address translation.
- If we have a TLB miss, we will first look in the inverted page table.
- It is highly likely that the memory we want is already in memory.

Once we have paging set up, how do we determine which pages we page in and out? We could use a

- FIFO approach — pages that were loaded first get un-loaded first.
- Optimal — predict the future and keep the pages we need (unrealistic, but would be nice).
- Random replacement — just pick a random one.

The *page reference string* tells us what pages we have.