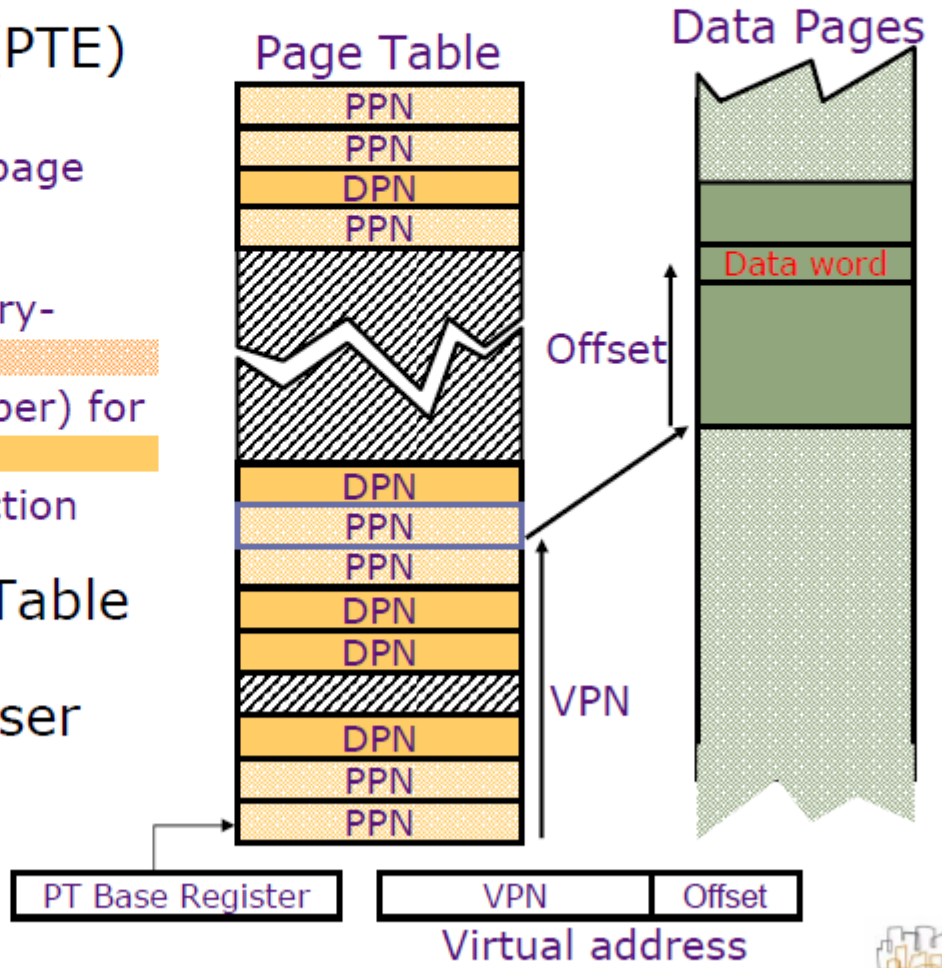
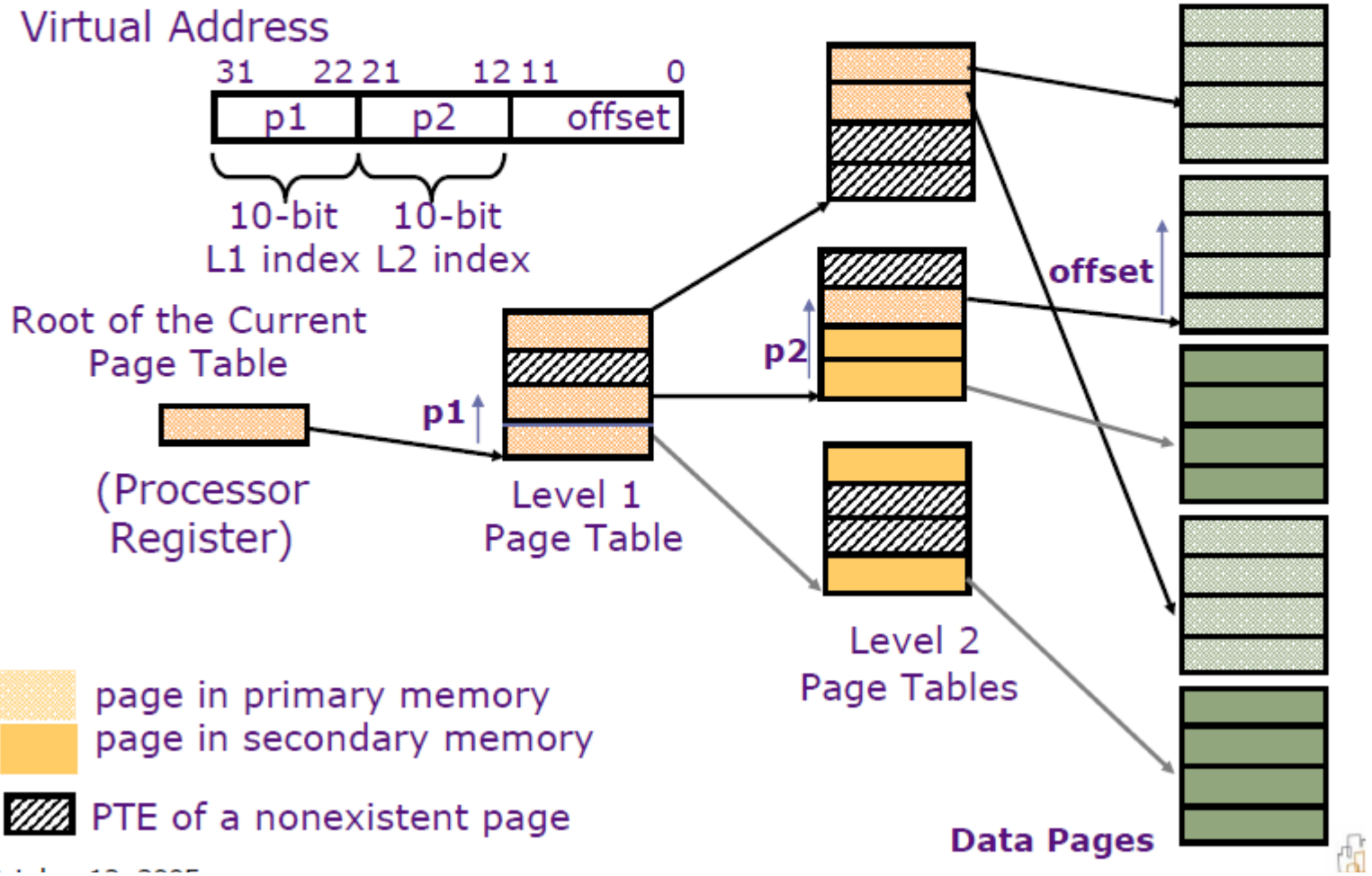
Virtual Memory

Linear Page Table

- Page Table Entry (PTE) contains:
  - A bit to indicate if a page exists
  - PPN (physical page number) for a memory-resident page
  - DPN (disk page number) for a page on the disk
  - Status bits for protection and usage
- OS sets the Page Table Base Register whenever active user process changes



Hierarchical Page Table



Translation Lookaside Buffers

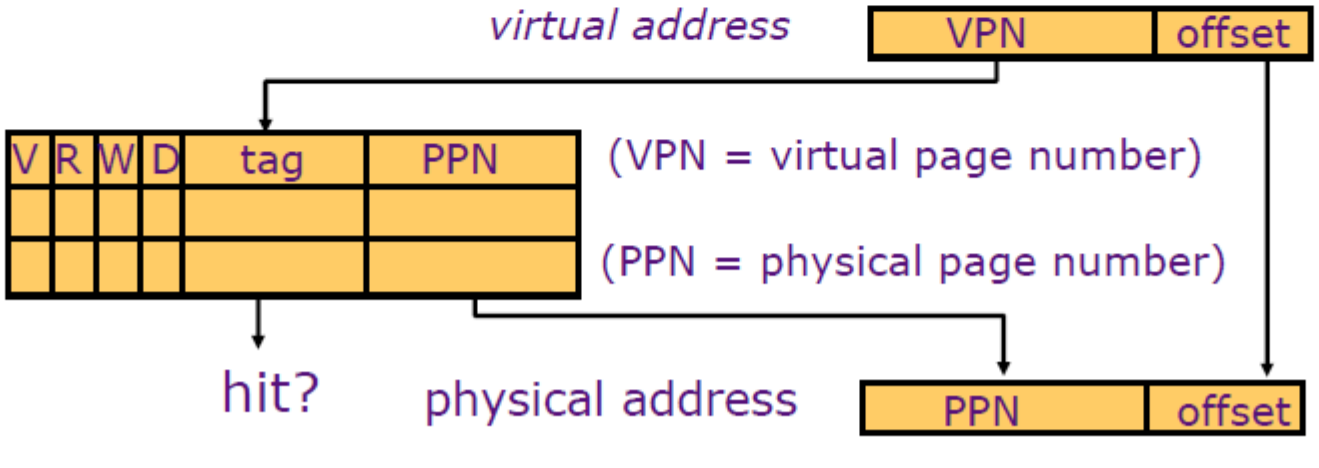
Address translation is very expensive!

In a two-level page table, each reference becomes several memory accesses

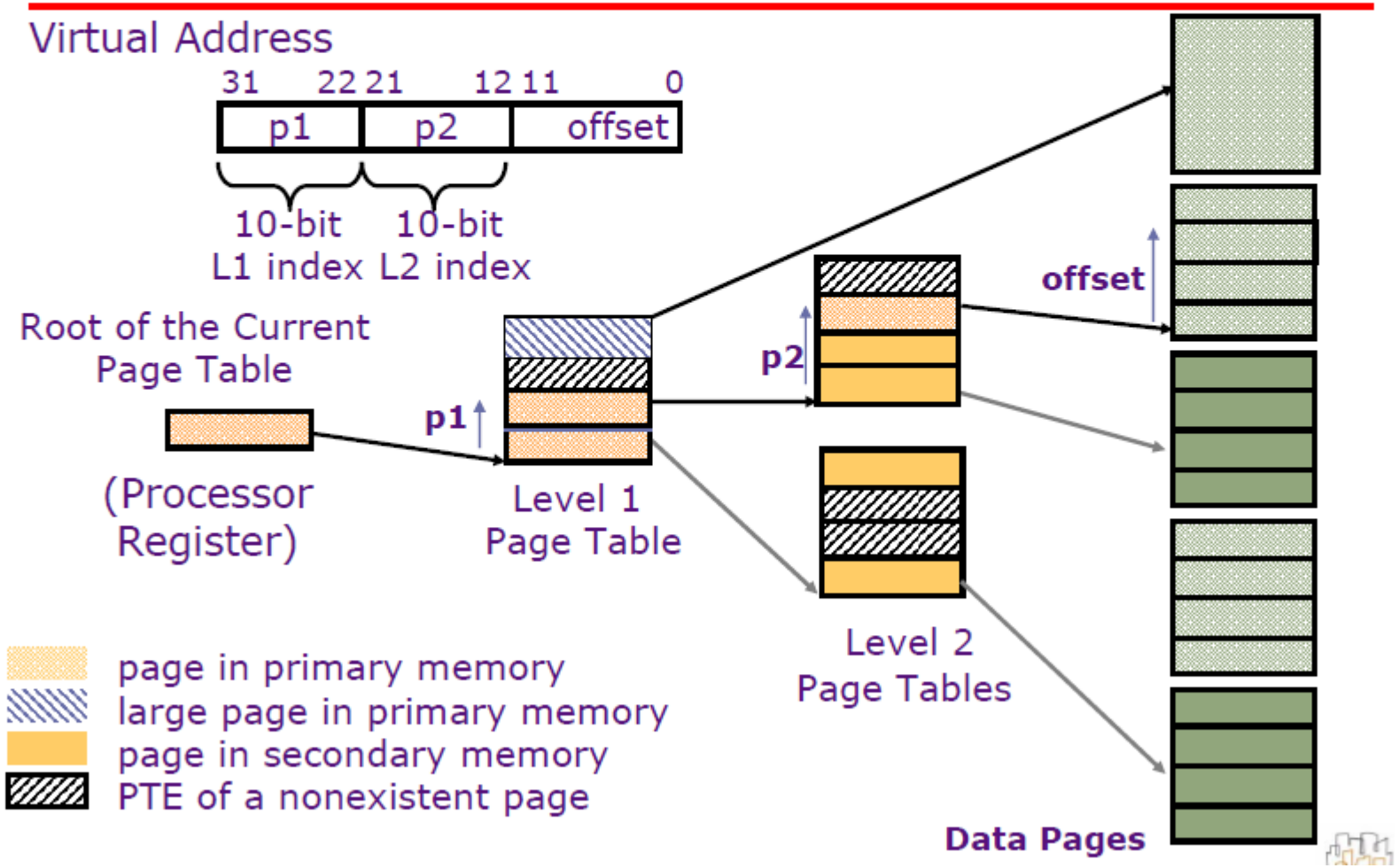
Solution: *Cache translations in TLB*

TLB hit ⇒ *Single Cycle Translation*

TLB miss ⇒ *Page Table Walk to refill*

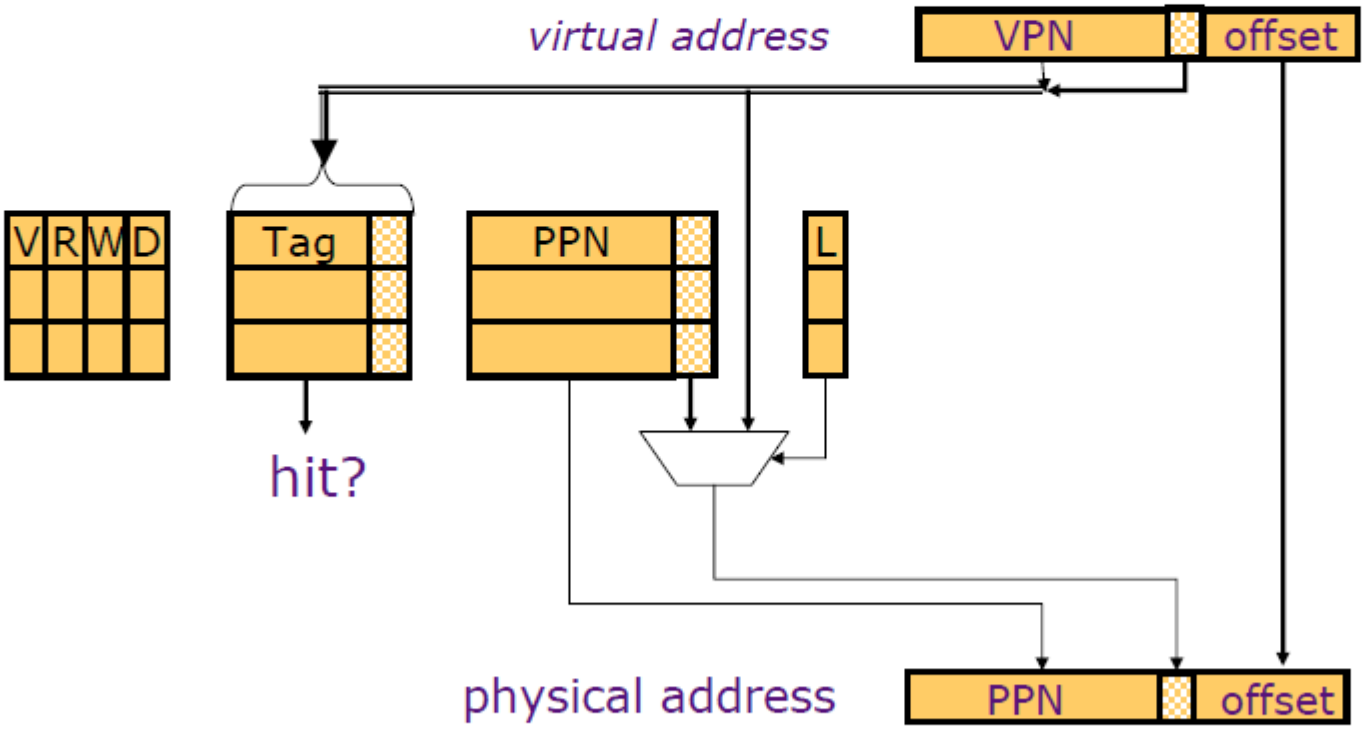


Variable Sized Page Support

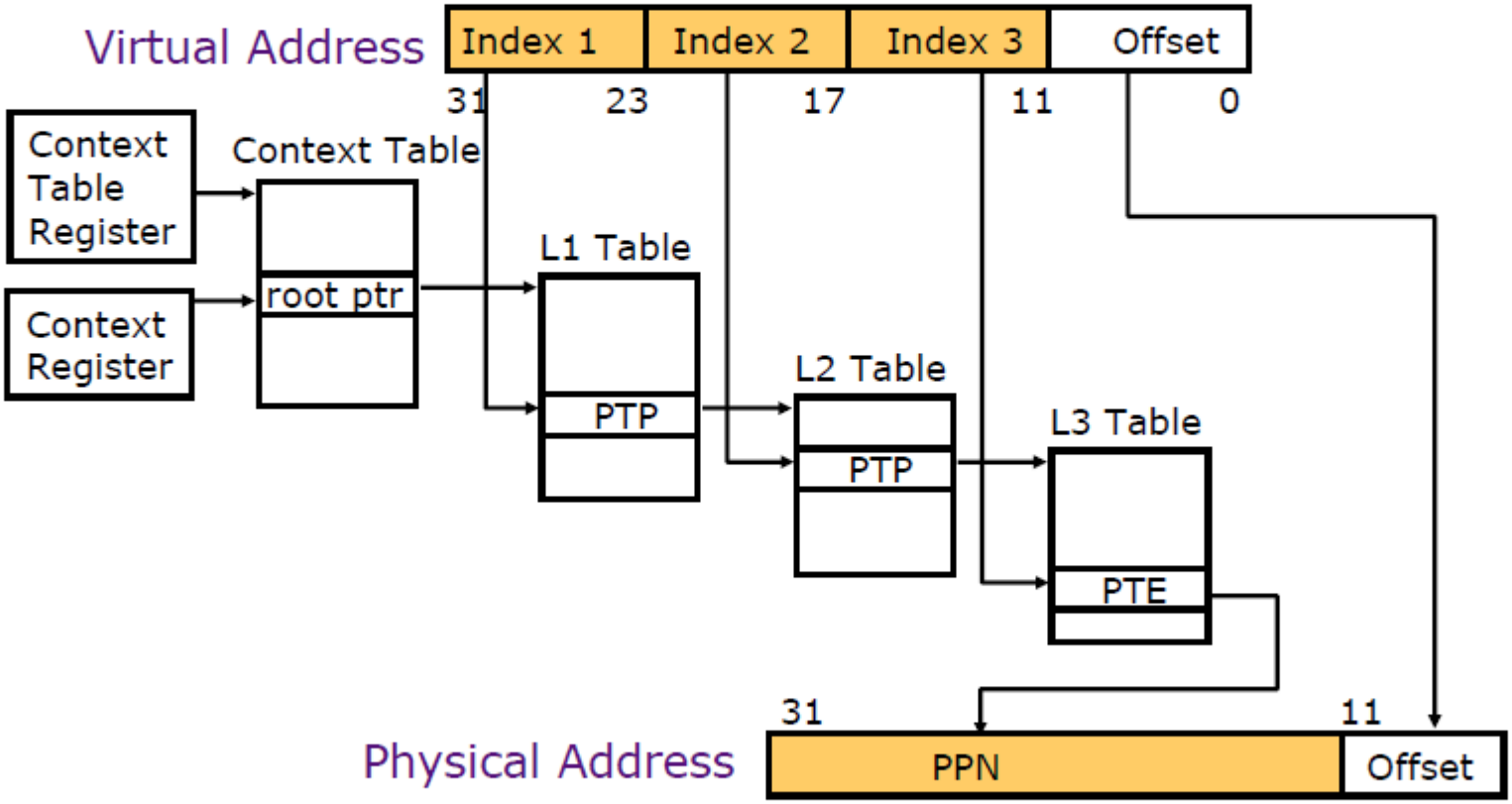


Variable Size Page TLB

Some systems support multiple page sizes.



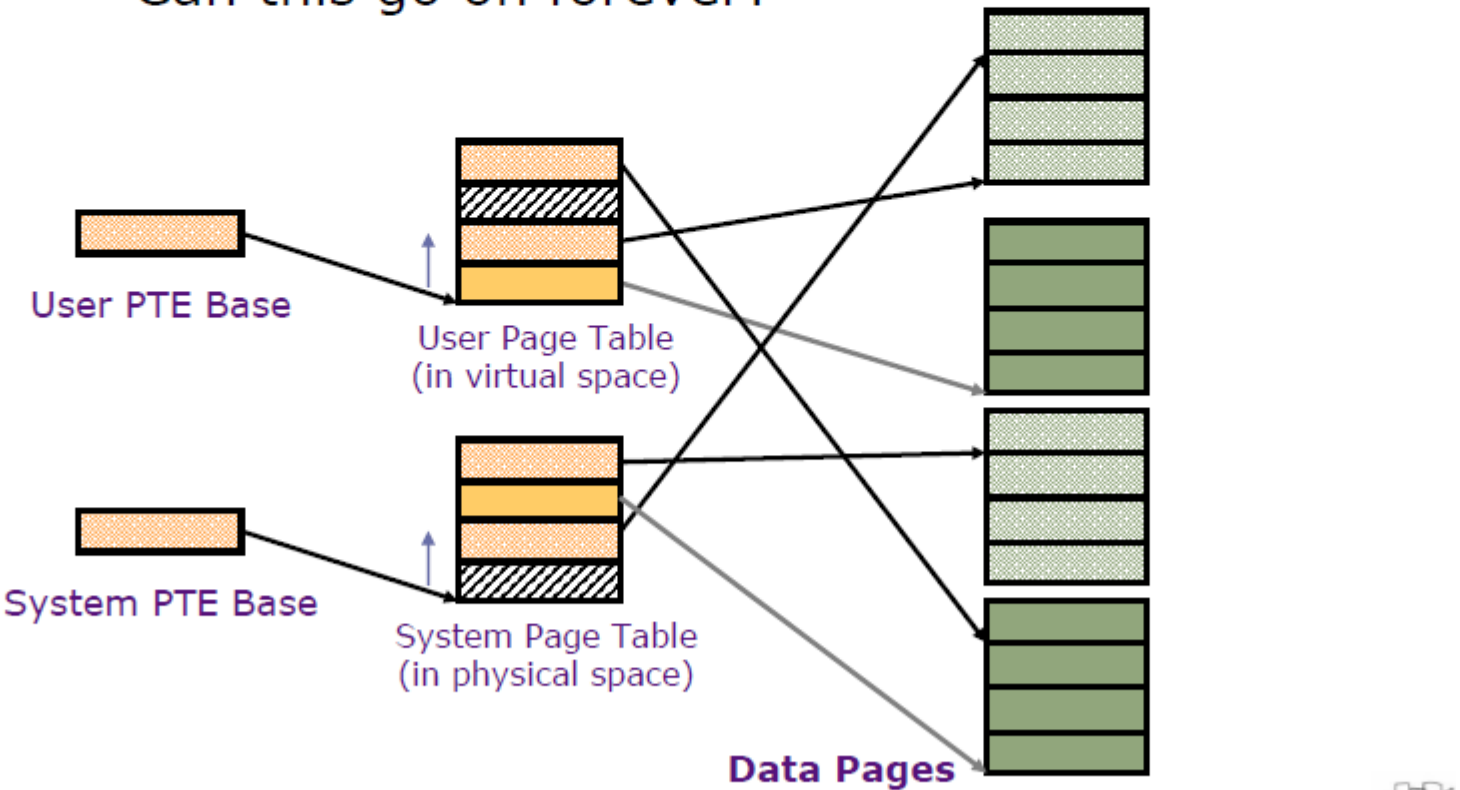
Hierarchical Page Table Walk: SPARC v8



MMU does this table walk in hardware on a TLB miss

Translation for Page Tables

- Can references to page tables TLB miss
- Can this go on forever?



Address Translation:
putting it all together