Memory Hardware

Lecture 30

Klara Nahrstedt

CS241 Administrative

- Read Stallings Chapter 7 and 8 about Memory Management and VM
- LMP2 is on (Deadline April 16, but Part I has deadline April 9)

Vinton Cerf Lecture, April 11, 11am, 1404 SC

Distinguished Lecture Series in Computer Science

Wednesday, April 11

11:00am, 1404 Siebel

Dr. Vinton Cerf, Chief Internet Evangelist and VP at Google

Dr. Cerf is widely known as one of the founders of the Internet and as the co-designer of the TCP/IP protocol, the communications protocol that gave birth to the Internet and that is commonly used today.

This talk will be more technical in nature than the Arnold O. Beckman Lecture that Dr. Cerf will deliver on Tuesday, April 10th, at 4pm in Foellinger. Computer Science faculty and students are encouraged to attend both talks.

Concepts this Lecture

Paging Hardware

- TLB

Multi-Level Paging

- Two-Level Paging Example

Paging Hardware - Registers

Can put page table in registers (small page table < 256).

Registers must be changed at context switch.

Paging - Caching the Page Table

Can cache page table in registers and keep page table in memory at location given by a page table base register.

Page table base register changed at context switch time.

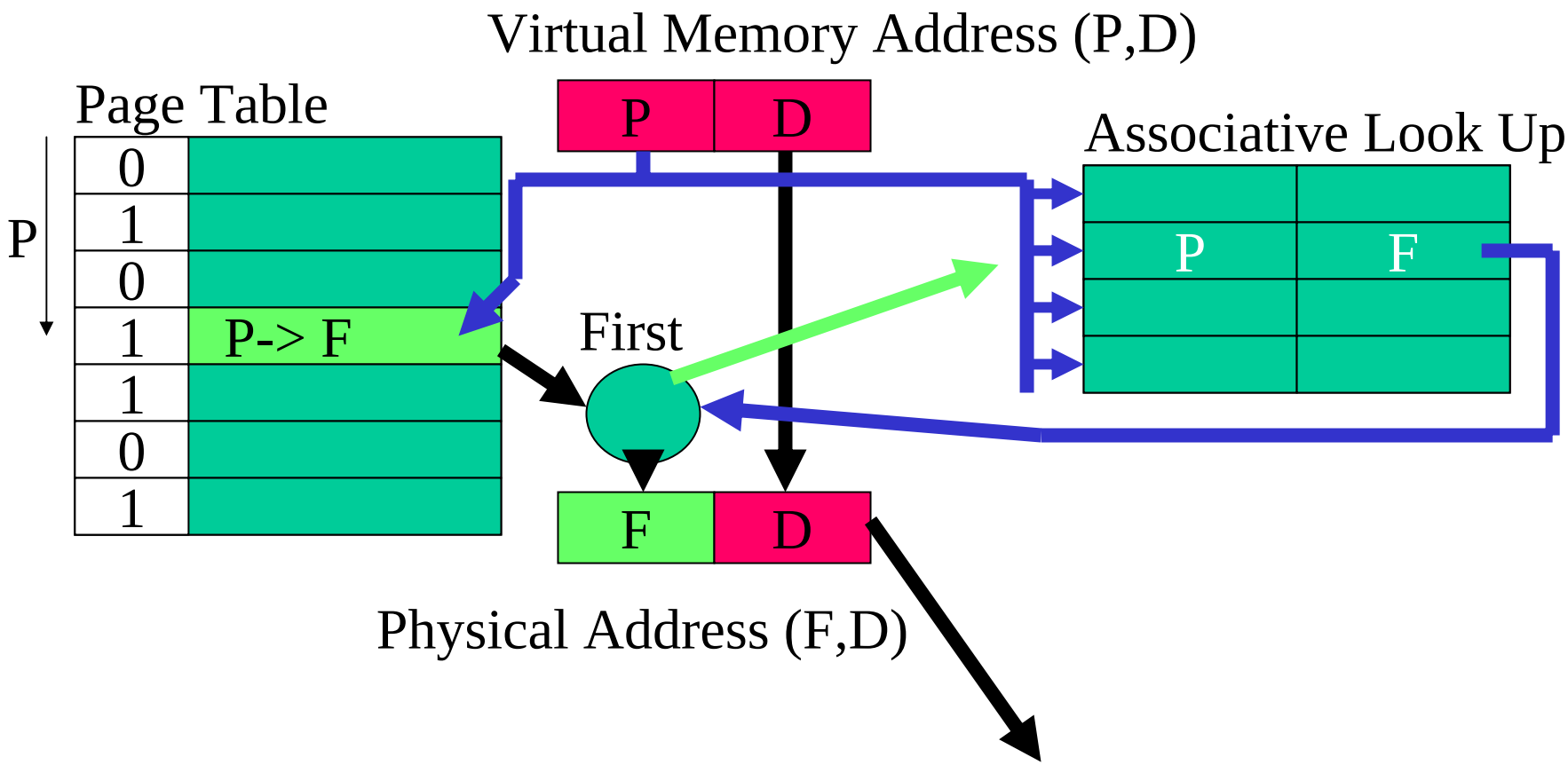
Paging Implementation Issues

Caching scheme can use associative registers, look-aside memory or content-addressable memory- TLB.

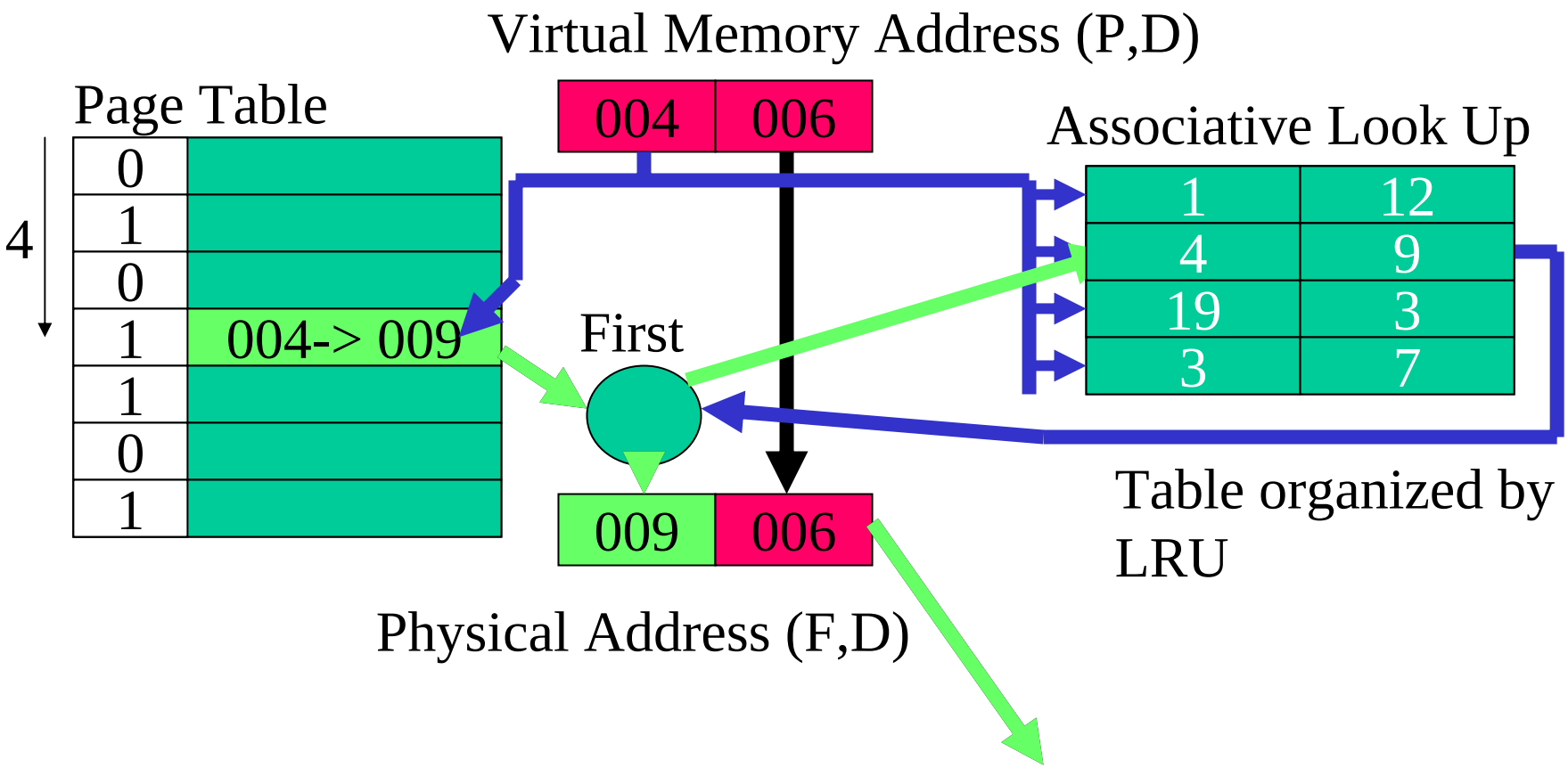
Page address cache (TLB) hit ratio: percentage of time page found in associative memory.

If not found in associative memory, must load from page tables: requires additional memory reference.

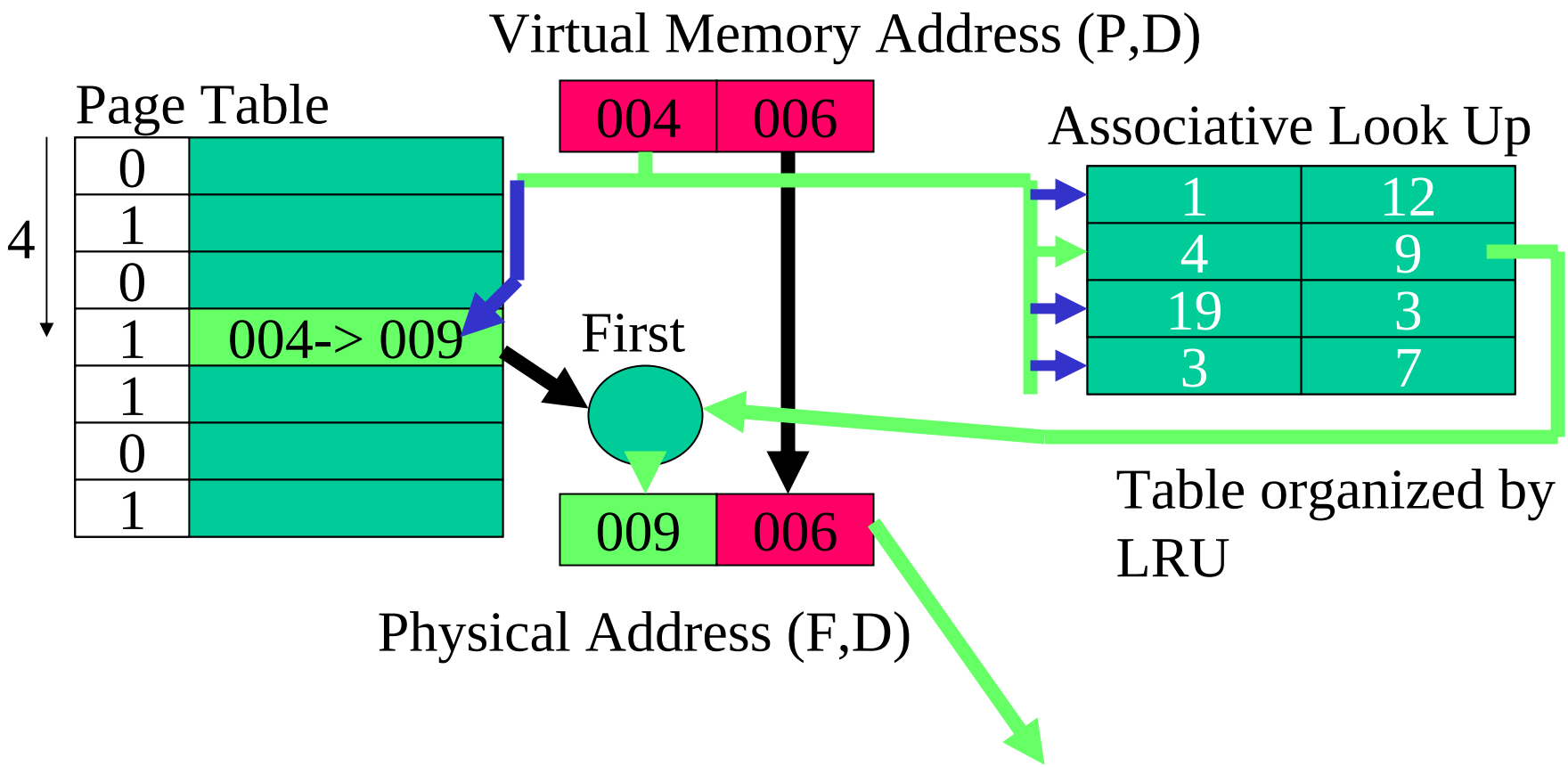
Page Mapping Hardware



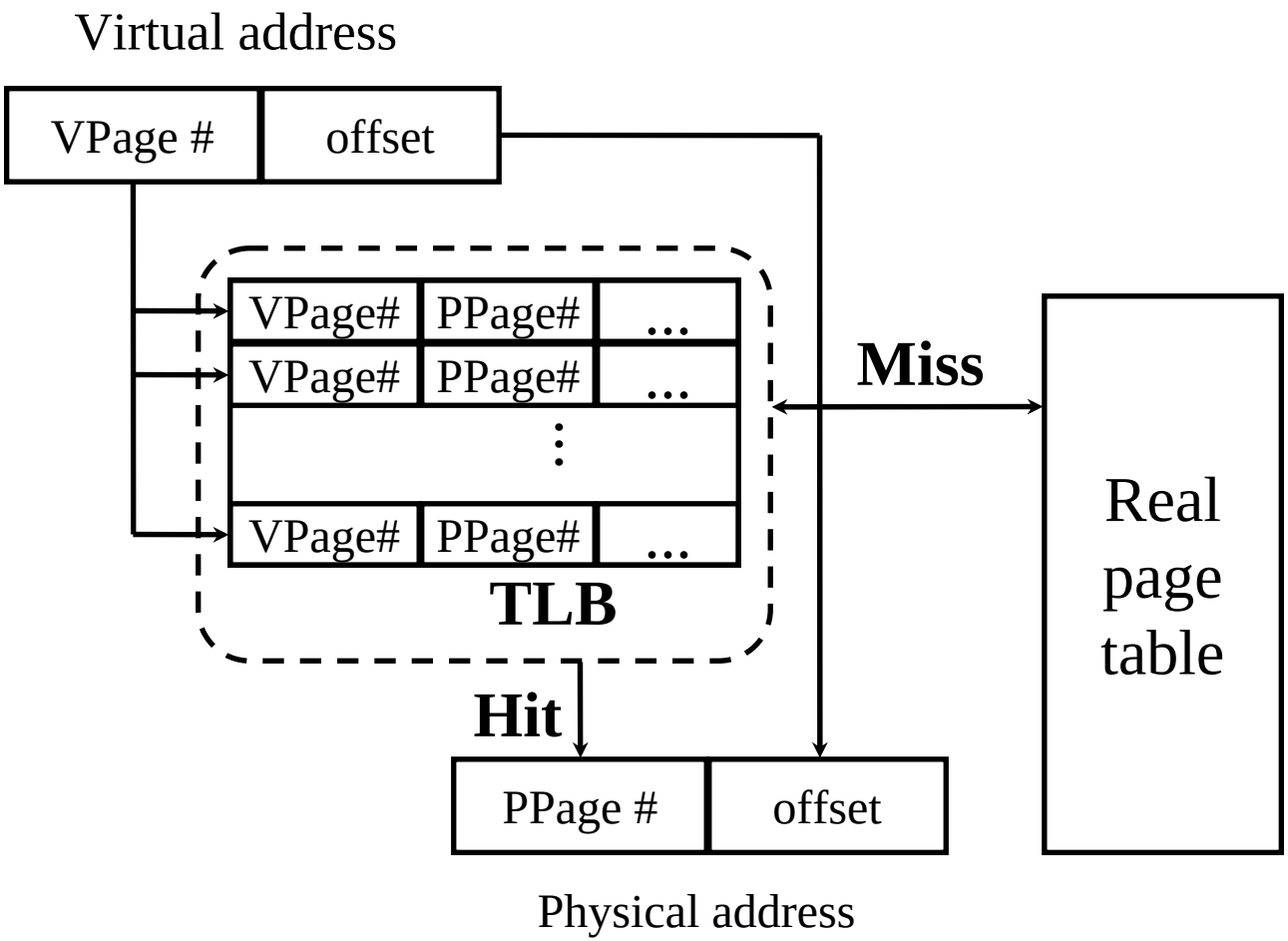
Page Mapping Hardware



Page Mapping Hardware



Translation Lookaside Buffer (TLB)



TLB Function

If a virtual address is presented to MMU, the hardware checks TLB by comparing all entries simultaneously (in parallel).

If match is valid, the page is taken from TLB without going through page table.

If match is not valid

- MMU detects miss and does an ordinary page table lookup.
- It then evicts one page out of TLB and replaces it with the new entry, so that next time that page is found in TLB.

Effective Access Time

Associative lookup time = ε time unit.

Memory cycle -- m microsecond.

Hit ratio -- α.

Effective access time

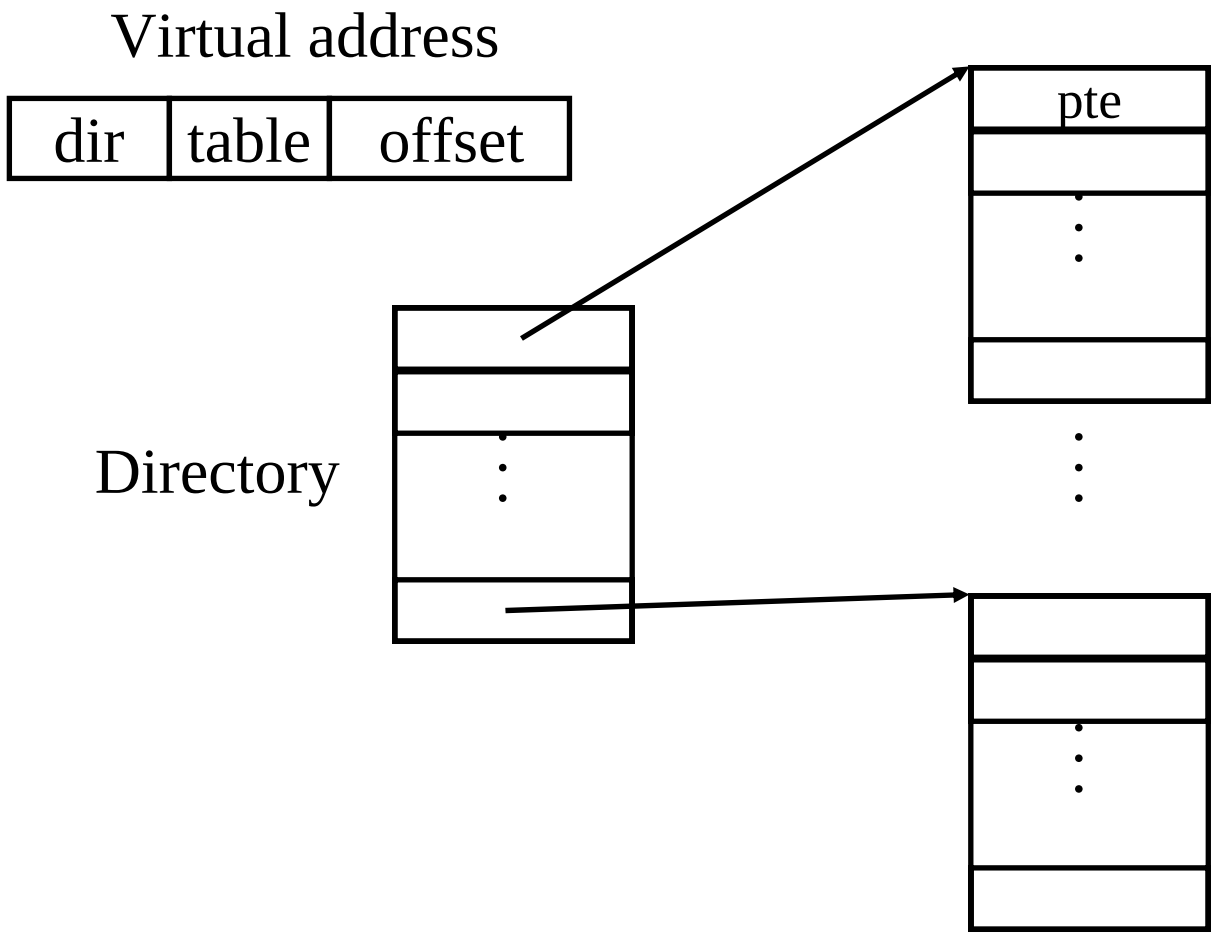
$$Eat = (1m+\varepsilon)\alpha+(2m+\varepsilon)(1-\alpha)$$

$$Eat = 2m+\varepsilon-\alpha.$$

Multilevel Page Tables

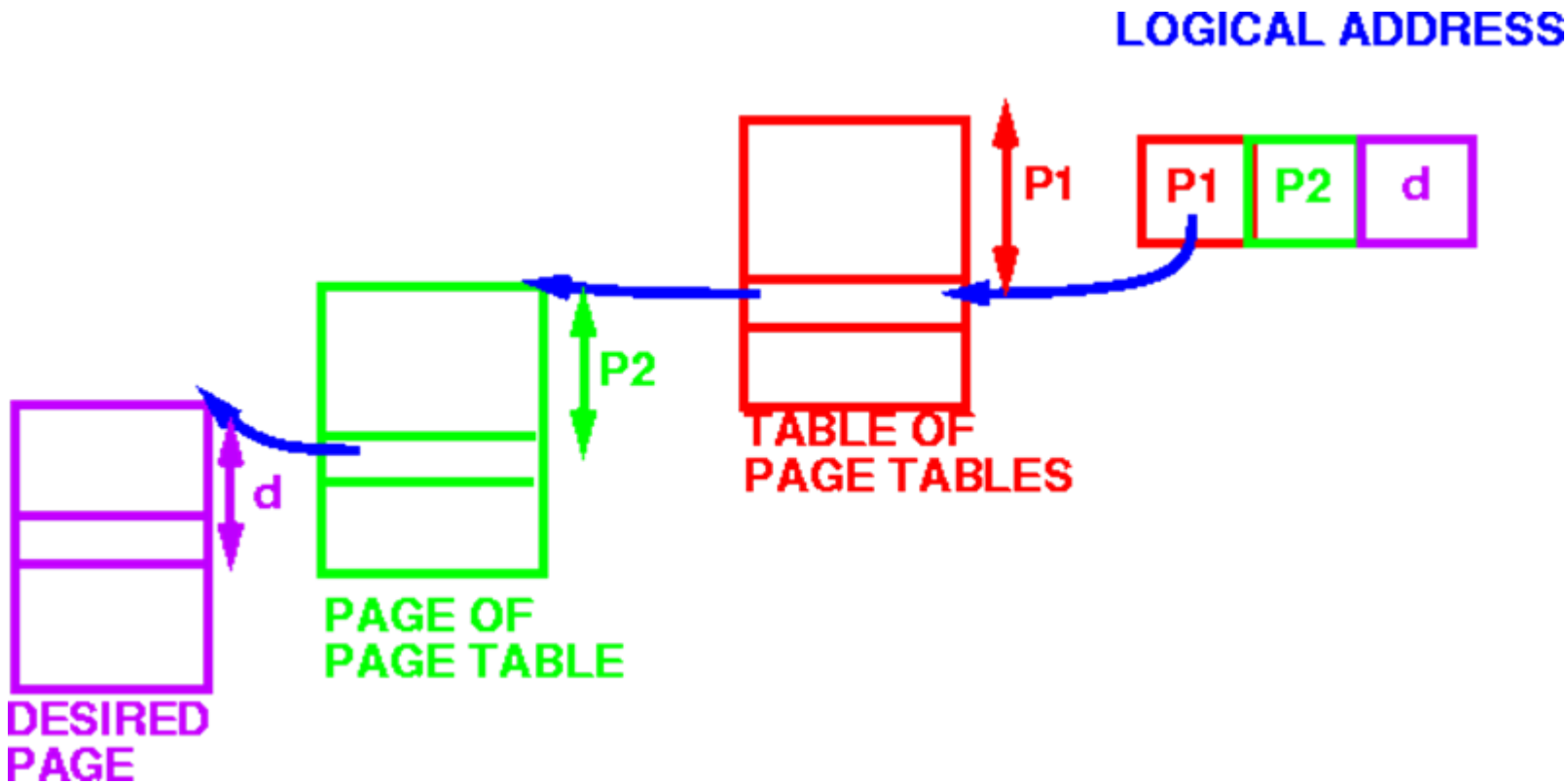
- Since the page table can be very large, one solution is to page the page table
- Divide the page number into
  - An index into a page table of second level page tables
  - A page within a second level page table
- Advantage
  - No need to keeping all the page tables in memory all the time
  - Only recently accessed memory's mapping need to be kept in memory, the rest can be fetched on demand

Multilevel Page Tables



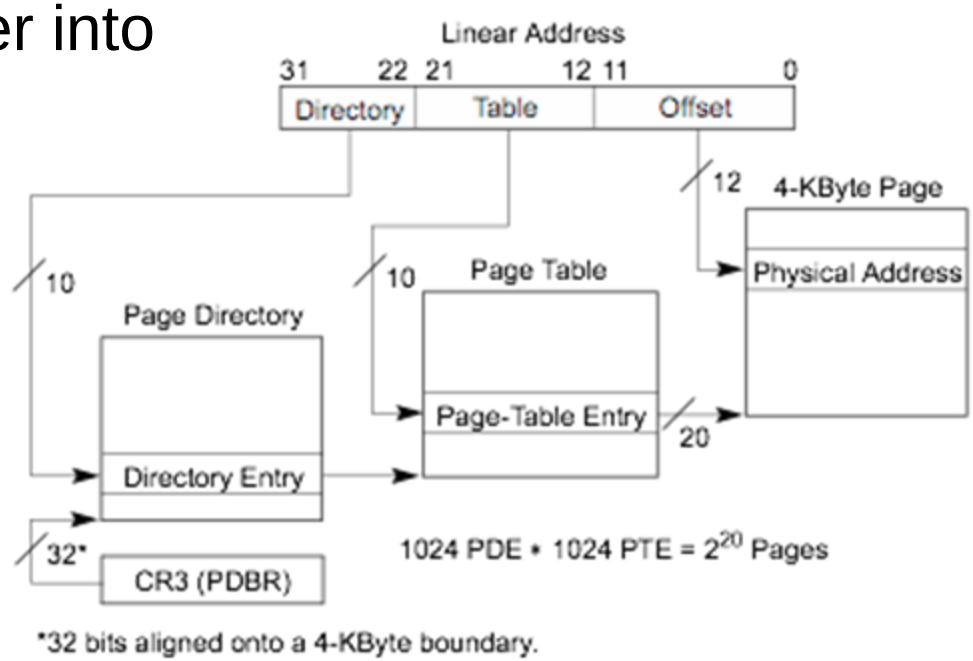
What does this buy us? Sparse address spaces and easier paging

Multilevel Paging

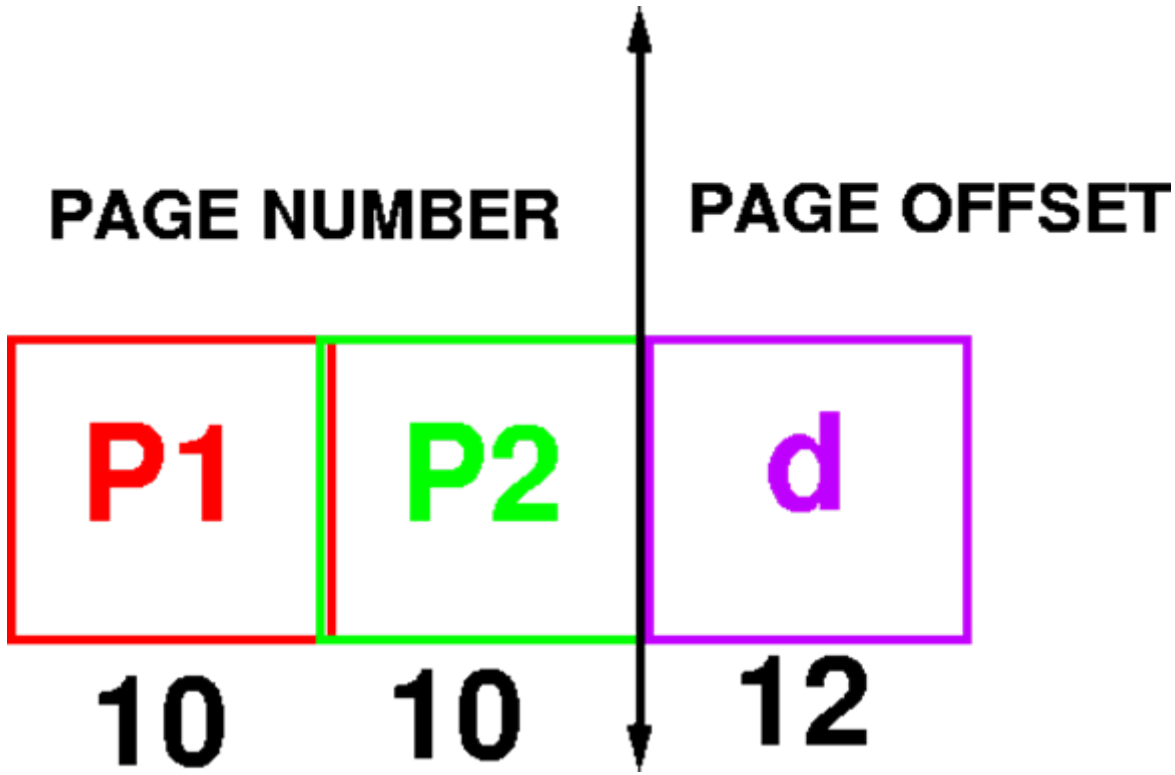


Example Addressing on a Multilevel Page Table System

- A logical address (on 32-bit x86 with 4k page size) is divided into
  - A page number consisting of 20 bits
  - A page offset consisting of 12 bits
- Divide the page number into
  - A 10-bit page number
  - A 10-bit page offset



Addressing on Multilevel Page Table



Summary

Paging Hardware

- TLB

Multi-level Paging

- Two Level Paging