23CST202 – Operating Systems

Paging

Structure of Page Table

Dr.V.Savitha
Associate Professor
SNS College of Technology

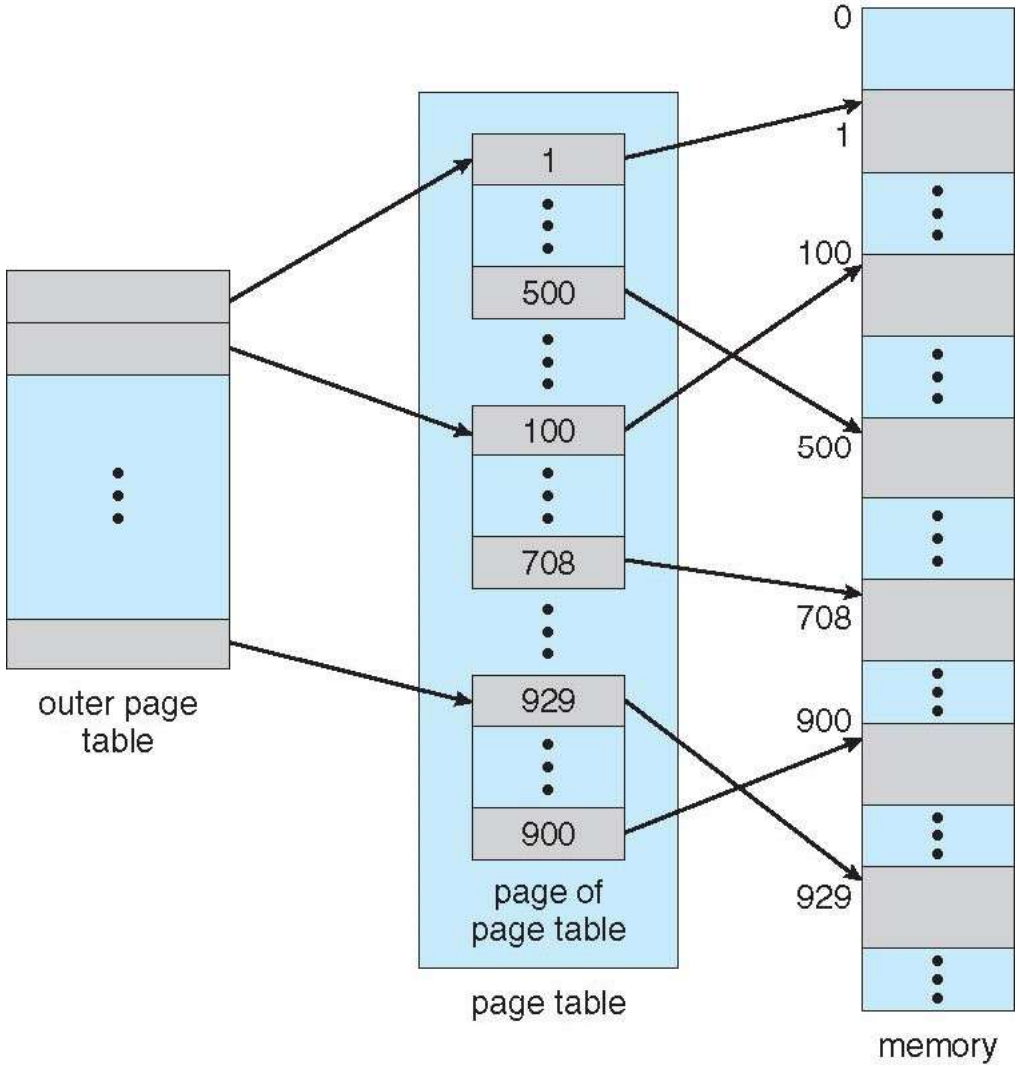
Structure of the Page Table

- Hierarchical Paging
- Hashed Page Tables
- Inverted Page Tables

Hierarchical Page Tables

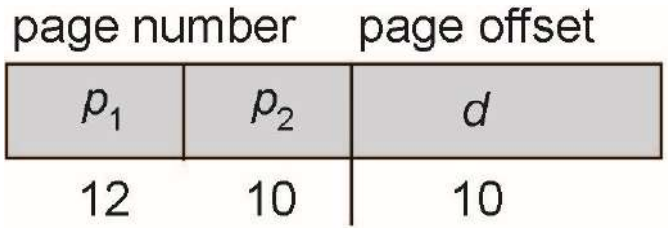
- Break up the logical address space into multiple page tables
- A simple technique is a two-level page table

Two-Level Page-Table Scheme



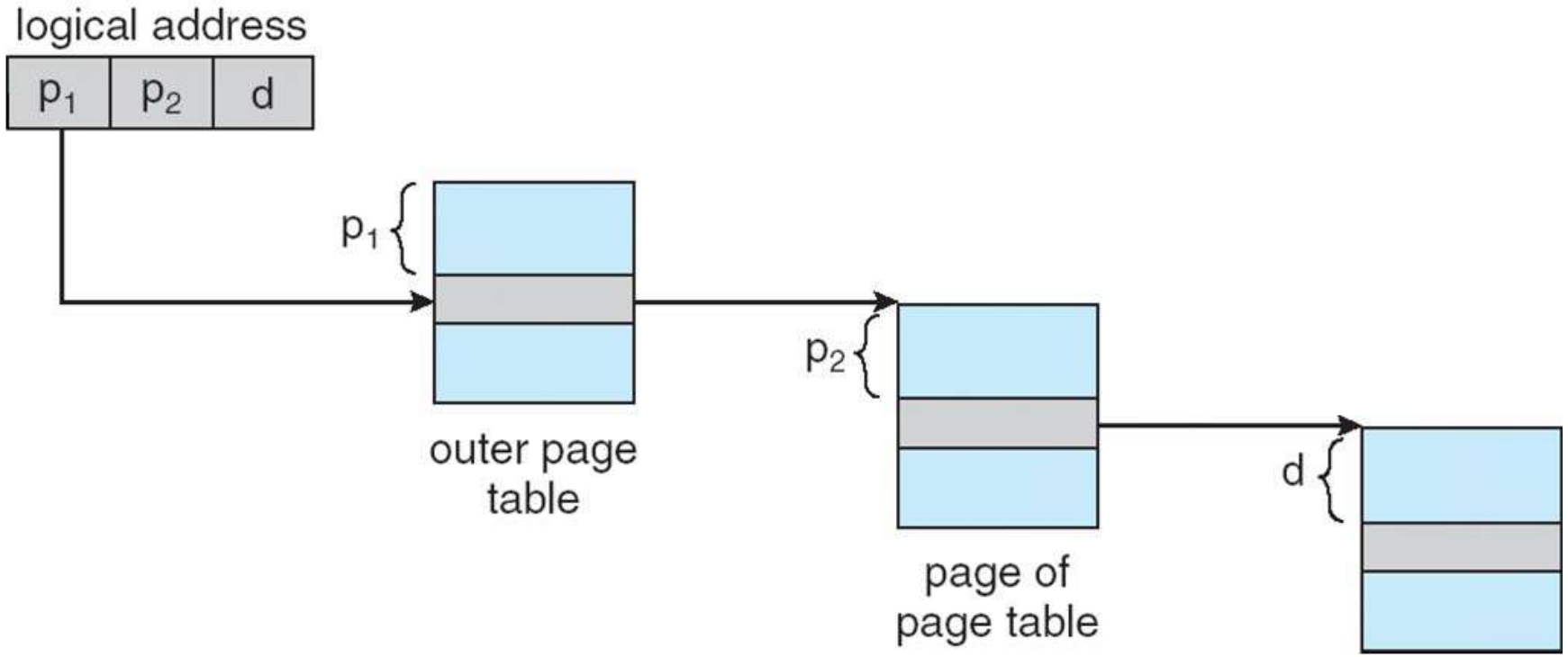
- A logical address (on 32-bit machine with 1K page size) is divided into:
  - a page number consisting of 22 bits
  - a page offset consisting of 10 bits

- A logical address (on 32-bit machine with 1K page size) is divided into:
  - a page number consisting of 22 bits
  - a page offset consisting of 10 bits



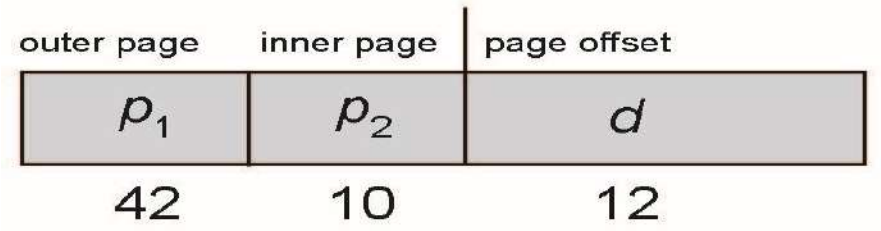
- p_1 is an index into the outer page table, and p_2 is the displacement within the page of the inner page table known as **forward-mapped page table**

Address-Translation Scheme

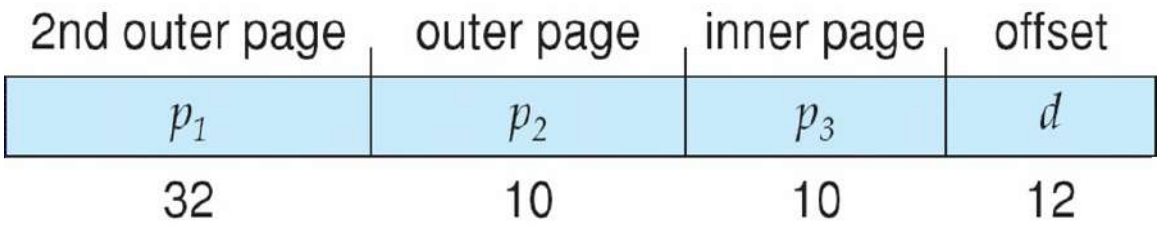


64-bit Logical Address Space

- Two-level paging scheme not sufficient
- Outer page table has 2^{42} entries or 2^{44} bytes



- Three-level Paging Scheme



Hashed Page Tables

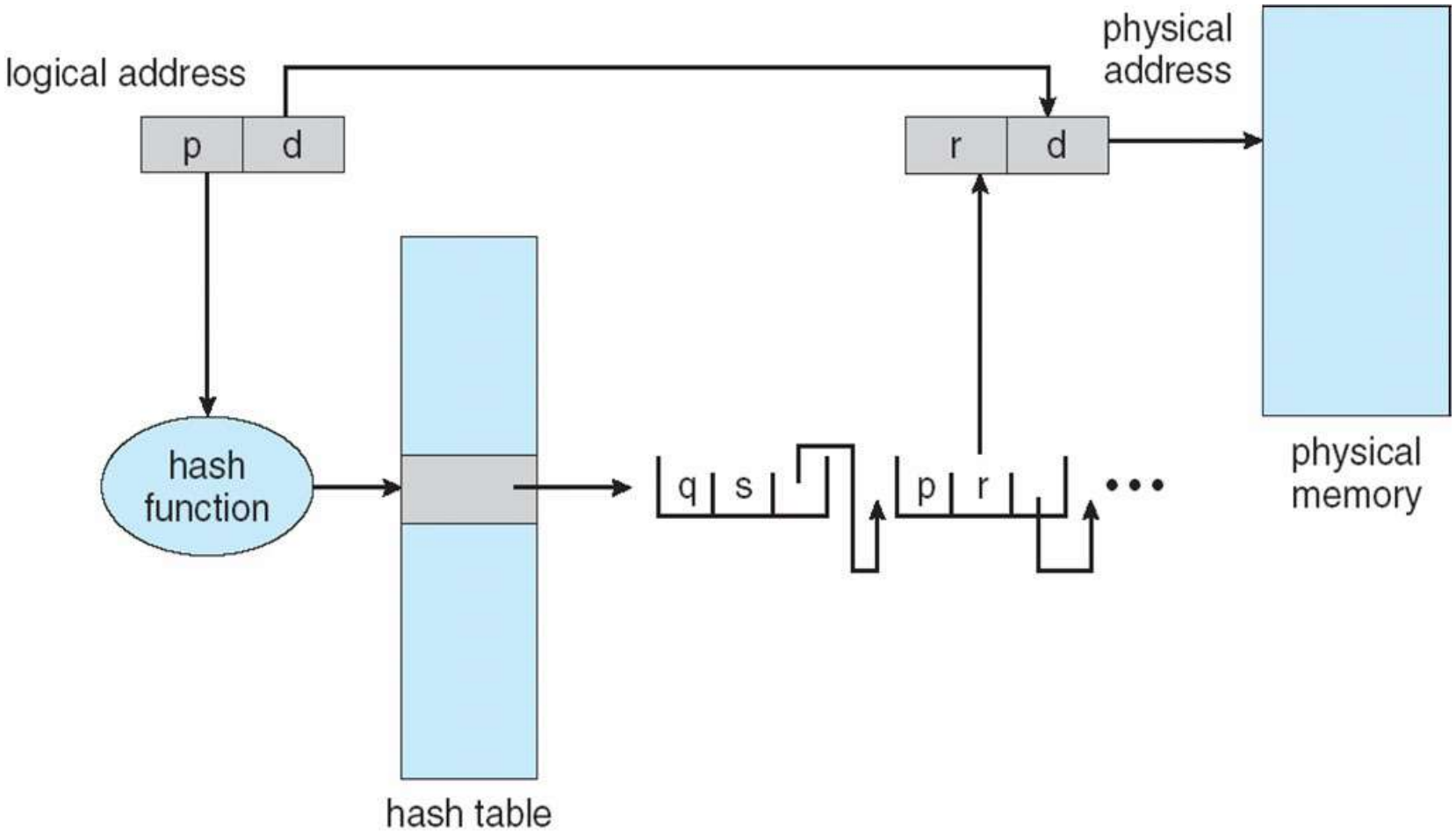
- Virtual page number is **hashed** into a page table
- Each element contains

  (1) the virtual page number

  (2) the value of the mapped page frame

  (3) a pointer to the next element
- Virtual page numbers are compared in this chain searching for a match
  - If a match is found, the corresponding physical frame is extracted

Hashed Page Table



Inverted Page Table

- Rather than each process having a page table and keeping track of all possible logical pages, track all physical pages