COA18 Computer Organization and Architecture

Lecture 06

Virtual Memory

Tongwei Ren

Software Institute, Nanjing University

Sep. 29, 2018

Review

- Magnetic disk
  - Data organization, formatting, read and write mechanism, timing of a disk transfer, head scan algorithm
- Optical memory
  - CD, CD-ROM, CD-R, CD-RW, DVD, DVD-R, DVD-RW, blue ray
- Magnetic tape

Memory Management

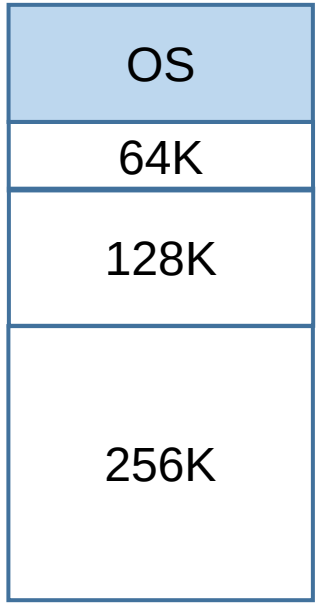
- In the past, only OS and one program are in memory
- Currently, OS and multiple programs are in memory
  - To avoid the idle of processor when the programs wait for I/O, it requires to load more programs in memory
- Memory management
  - In multiprogramming system, the “user part” of memory should be further partitioned to fit multiple programs, which is dynamically carried out by OS
  - Don’t consider “process” in this slide

Solution to Load More Programs

- Enlarge memory
- Using exchange and overlap techniques
  - When no program is ready, OS loads other programs in
  - Partitioning and paging
- Virtual memory
  - Demand paging
  - Virtual address

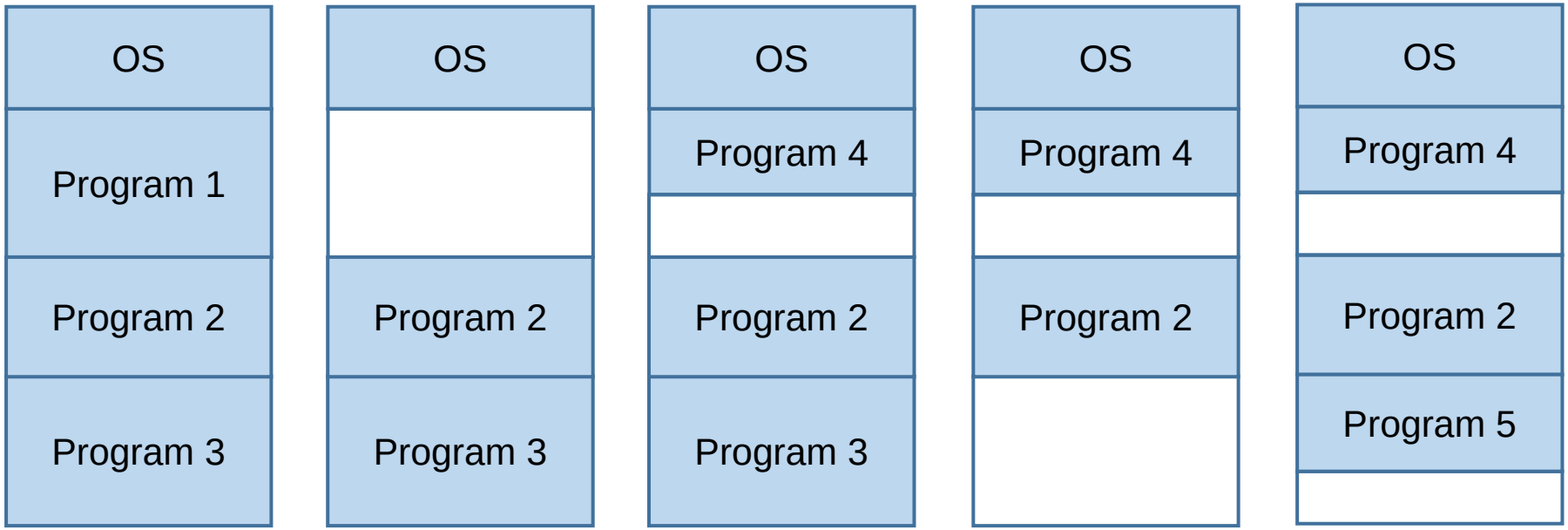
Partitioning

- Fixed-size partition
  - OS: fixed size
  - User program: fixed size with different lengths
    - When load a program, put it in the smallest partition which has enough length
  - Drawback: waste of memory



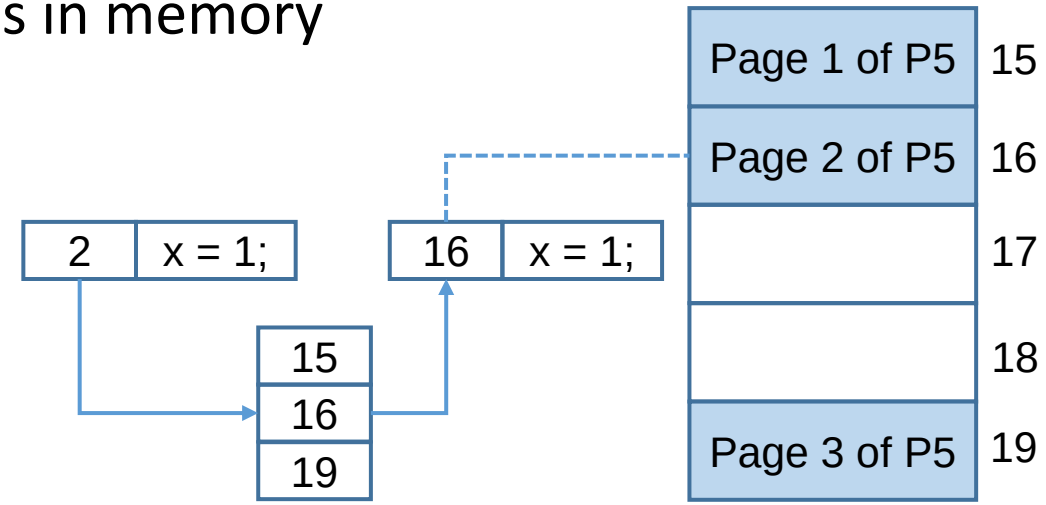
Partitioning (cont.)

- Variable-length partition
  - OS: fixed size
  - User program: as requirement
  - Drawback: the number of fragments increases



Paging

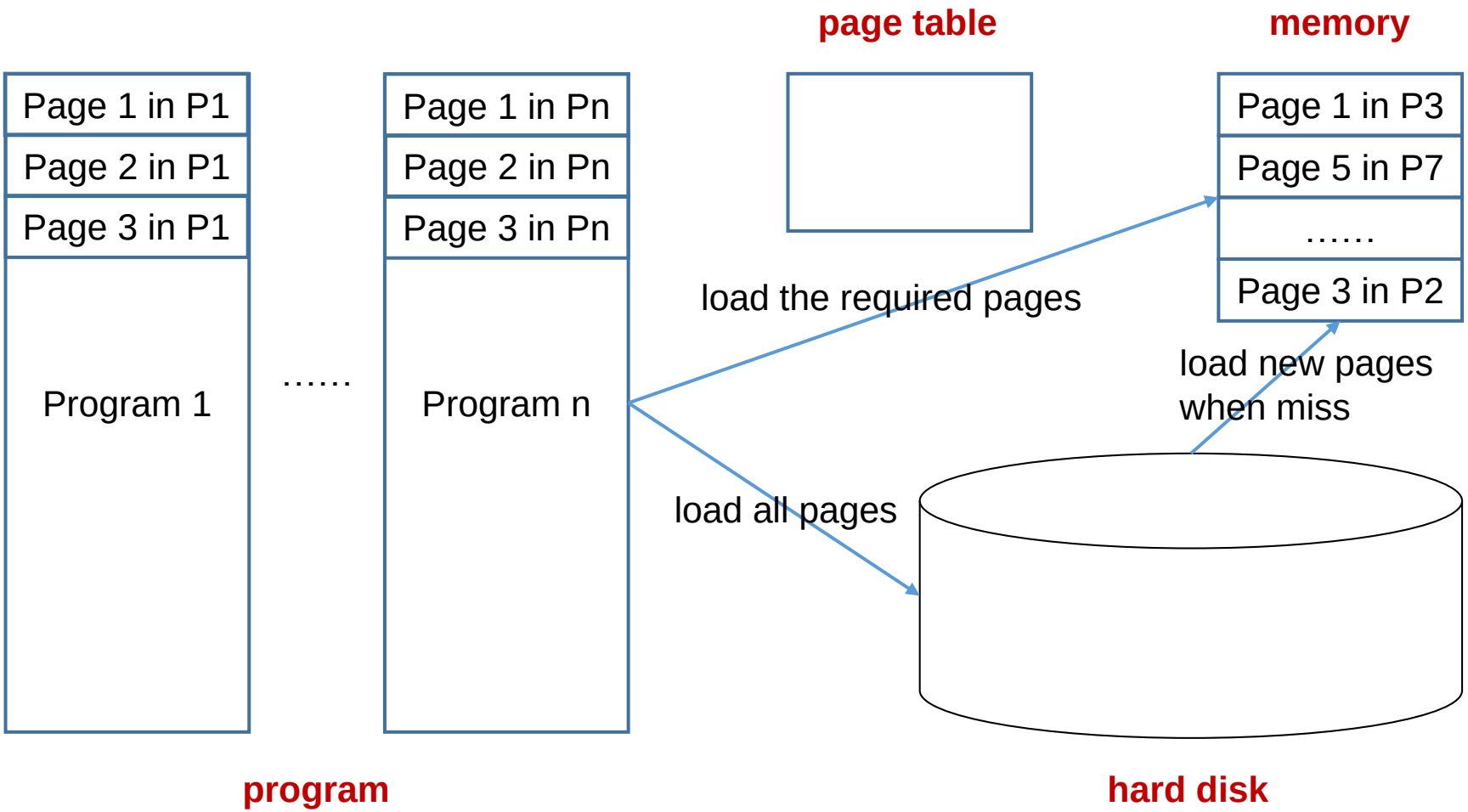
- Basic idea
  - Divide the memory into fixed-size blocks, named **page frame**, and divide the program into fixed size blocks, named **page**
  - Load the pages into page frames
- Logical address: address in instruction
- Physical address: address in memory



Virtual Memory

- Problem
  - The size of memory is limited, but the requirement of memory keeps increasing
- Basic idea
  - Demand paging: only load the active pages in paging
- Essence
  - Programming in a logical address space larger than memory
  - Only load the required programs and data in memory
  - Transfer the logical address to physical address by hardware
  - Exchange information between memory and hard disk when miss

Virtual Memory (cont.)



Virtual Memory (cont.)

- Some problems
  - Size of page
    - 4KB, 8KB, …
  - Mapping function
    - Associative mapping
  - Types
    - Page based virtual memory
    - Segment based virtual memory
    - Segment and page based virtual memory
  - Write policy
    - Write back

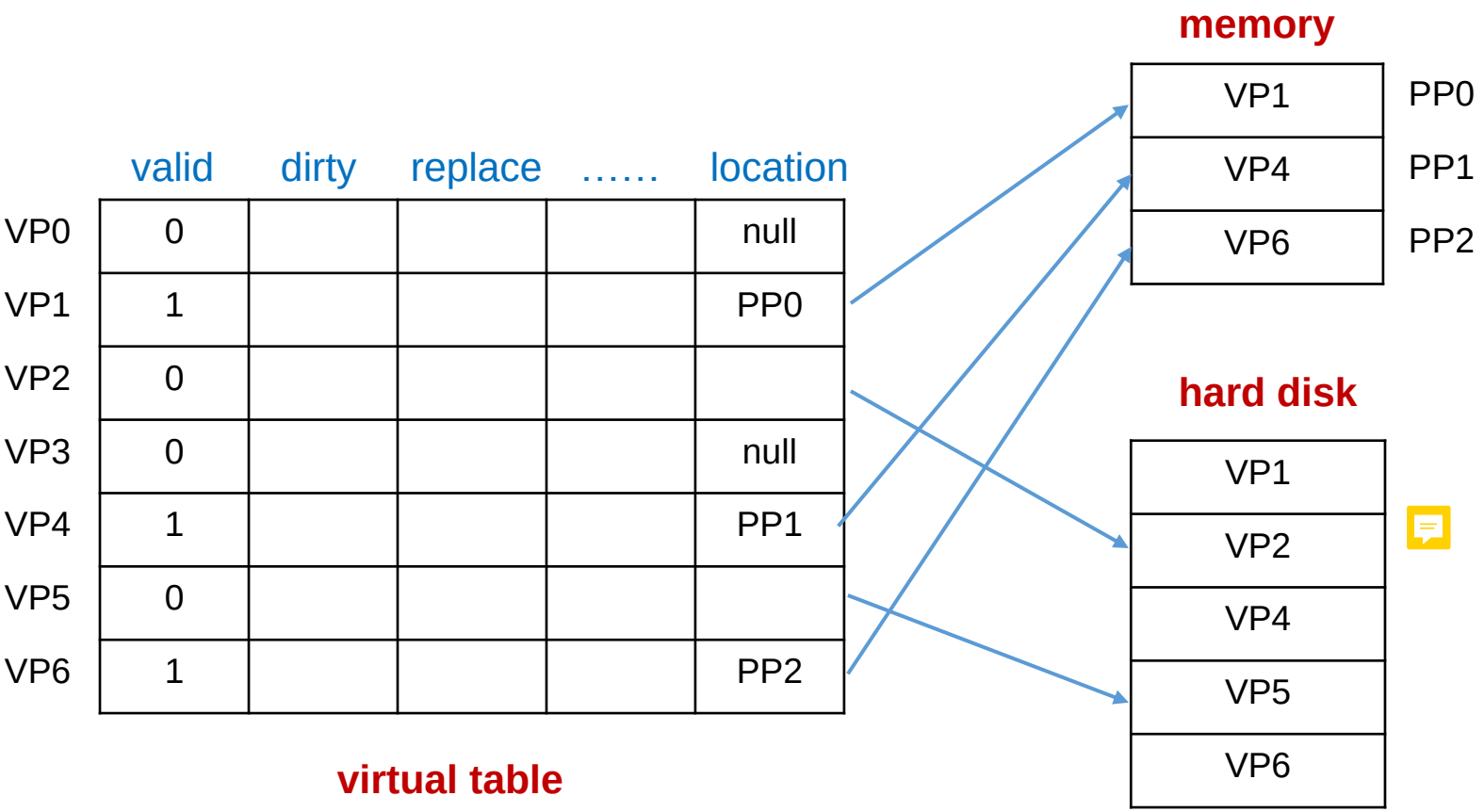
Cache is 10 times faster than memory

Memory is 100000 times faster than hard disk

Page based Virtual Memory

- Divide the main memory and virtual memory into pages with the same size
  - Virtual page (VP) / logical page: page in virtual memory
  - Physical page (PP) / page frame: page in main memory
- Page table
  - A table contains the information of **all VPs**, including location, valid bit, dirty bit, r/w right et al.
  - Store in main memory
  - Virtual address
    - Virtual page number + offset in page

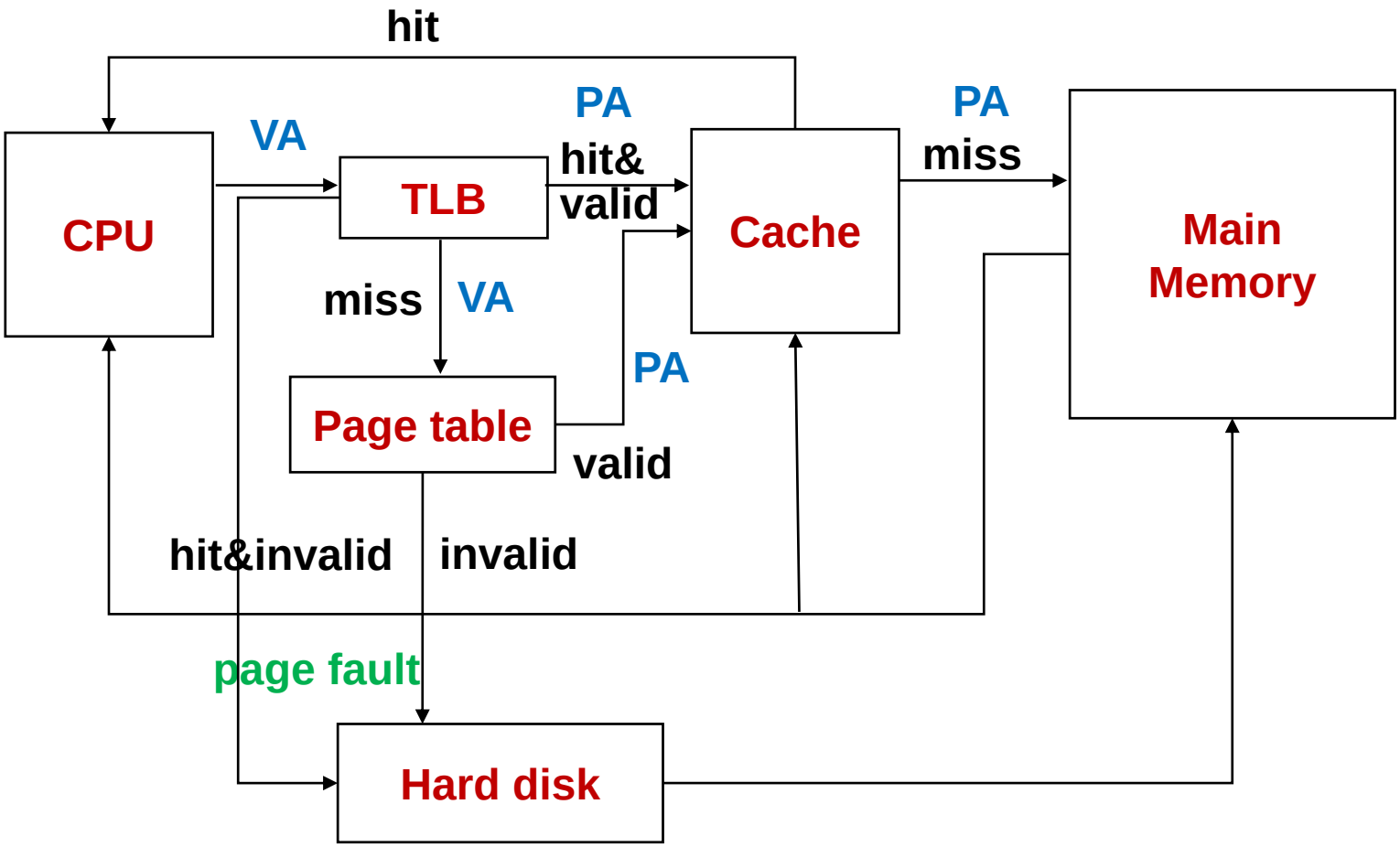
Page based Virtual Memory (cont.)



Translation Lookaside Buffer

- The usage of page table increases the access of main memory
- To reduce memory access, load the frequently used items in page table to cache
- TLB: the page table items loaded in cache
  - Associative mapping, set associative mapping
  - Random replacement

Translation Lookaside Buffer (cont.)



Translation Lookaside Buffer (cont.)

No.	TLB		page table	cache	possibility
1	hit	valid	----	hit	possible, in cache
2	hit	invalid	----	hit	impossible
3	hit	valid	----	miss	possible, in memory but not cache
4	hit	invalid	----	miss	possible, not in memory
5	miss	----	valid	hit	possible, in cache
6	miss	----	invalid	hit	impossible
7	miss	----	valid	miss	possible, in memory but not cache
8	miss	----	invalid	miss	possible, not in memory

Best	**1**	no memory access
Good	**3, 5**	once memory access
Moderate	**7**	twice memory access
Bad	**4**	once memory access and hard disk access
Worst	**8**	twice memory access and hard disk access

Segment based Virtual Memory

- Divide the program and data into segments with different lengths, and load the required segments into memory
- Virtual address
  - Segment number + offset in segment
- Compared to page base virtual memory
  - Page base virtual memory
    - Advantage: simple, low cost
    - Disadvantage: instruction and data may cross pages
  - Segment based virtual memory
    - Advantage: naturally divide program and data
    - Disadvantage: not fixed length

Segment and Page based Virtual Memory

- Divide the program and data into segments, and further divide the segments into pages
  - Each segment has its page table
- Virtual address
  - Segment number + page number + offset in page
- Advantage
  - Program is shared and protected in segment
- Disadvantage
  - Required multiple times table search

Summary

- Memory management
- Exchange and overlap technique
  - Partitioning and paging
- Virtual memory
  - Page based VM, segment based VM, segment and page based VM

Thank You