Memory (Part IV):
Virtual Memory

Professor Travis Peters
CSCI 460 Operating Systems
Fall 2019

Some slides & figures adapted from Stallings instructor resources.
Some slides adapted from Adam Bates’s F’18 CS423 course @ UIUC
https://courses engr.illinois.edu/cs423/sp2018/schedule.html
Today

Announcements
• Project Proposal Due Friday (11/01)!
• HW5 posted — Also Due Friday (11/01)!
• PA2 posted — Due NEXT Friday (11/08)… get started ASAP… ;)

Goals & Learning Objectives
• Wrapping up virtual memory
• Discuss some issues closer to real-world implementations of virtual memory
Address Translation in 2-Level Paging System

Page tables themselves could be paged in/out…but at least part of a page table must be in MM for a process to run…→ 2-Level Paging System

Typically, size of a page table ≤ size of a page i.e., a 2nd-level page table can fit within a single page
Address Translation in 2-Level Paging System (cont.)

Example: 2-level scheme,
- 4 GB ($2^{32}$) address space,
- byte-level addressing,
- 4 kB ($2^{12}$) page size
  $\rightarrow 2^{20}$ total pages

Each 4 MB page table is mapped by root page table $\rightarrow$ 1024 page tables mapped by 4-byte PTE $\rightarrow$ root page table size = 4 kB

Each 4 kB page is mapped by 4-byte PTE $\rightarrow$ user page table requires 4 MB ($2^{22}$) … or according to this scheme, $2^{22} ÷ 2^{12} = 2^{10} = 1K$ pages

4-kbyte root page table

Root Page Table always in Main Memory

4-Mbyte user page table

4-Gbyte user address space

Figure 8.3 A Two-Level Hierarchical Page Table

Figure 8.4 Address Translation in a Two-Level Paging System
Inverted Page Table Structure

Problem
size of 1+ level page tables is that they are proportional to the size of the virtual address space

Idea!
Invert page table (index by physical frames)

• $n >> m$
  → hash page # into smaller page table
  → use chaining for collisions

• Regardless of virtual address space size, the size of an inverted page table remains the same (i.e., inverted page table size is proportional to # of physical frames)
Using a Translation Lookaside Buffer (TLB)

- **Virtual Address**
  - Page #
  - Offset

- **Main Memory**
  - Frame #
  - Offset

- **Secondary Memory**

**Translation Lookaside Buffer**
- TLB hit
- TLB miss

**Page Table**
- Page fault

**Real Address**

**Flowchart**
- **Start**
  - CPU checks the TLB
  - Page Table Entry in TLB?
    - Yes
    - Page Fault Handling Routine
    - OS Instructs CPU to Read the Page from Disk
    - CPU Activates I/O Hardware
    - Page Transferred from Disk to Main Memory
    - Memory Full?
      - Yes
      - Perform Page Replacement
      - Page Tables Updated
    - Update TLB
    - Return to Faulted Instruction
    - Access Page Table
    - Page in Main Memory?
      - Yes
      - Update TLB
      - Return to Faulted Instruction
      - No
      - Access Page Table
      - No
      - Access Page Table
      - Page Table Entry in TLB?
        - Yes
        - Page Fault Handling Routine
        - OS Instructs CPU to Read the Page from Disk
        - CPU Activates I/O Hardware
        - Page Transferred from Disk to Main Memory
        - Memory Full?
          - Yes
          - Perform Page Replacement
          - Page Tables Updated
        - Update TLB
        - Return to Faulted Instruction
        - No
        - Access Page Table
        - No
        - Access Page Table
        - Page Table Entry in TLB?
          - Yes
          - Page Fault Handling Routine
          - OS Instructs CPU to Read the Page from Disk
          - CPU Activates I/O Hardware
          - Page Transferred from Disk to Main Memory
          - Memory Full?
            - Yes
            - Perform Page Replacement
            - Page Tables Updated
          - Update TLB
          - Return to Faulted Instruction
          - No
          - Access Page Table
          - No
          - Access Page Table
          - Page Table Entry in TLB?
            - Yes
            - Page Fault Handling Routine
            - OS Instructs CPU to Read the Page from Disk
            - CPU Activates I/O Hardware
            - Page Transferred from Disk to Main Memory
            - Memory Full?
              - Yes
              - Perform Page Replacement
              - Page Tables Updated
            - Update TLB
            - Return to Faulted Instruction
            - No
            - Access Page Table
            - No
            - Access Page Table
            - Page Table Entry in TLB?
              - Yes
              - Page Fault Handling Routine
              - OS Instructs CPU to Read the Page from Disk
              - CPU Activates I/O Hardware
              - Page Transferred from Disk to Main Memory
              - Memory Full?
                - Yes
                - Perform Page Replacement
                - Page Tables Updated
              - Update TLB
              - Return to Faulted Instruction
              - No
              - Access Page Table
              - No
              - Access Page Table
              - Page Table Entry in TLB?
                - Yes
                - Page Fault Handling Routine
                - OS Instructs CPU to Read the Page from Disk
                - CPU Activates I/O Hardware
                - Page Transferred from Disk to Main Memory
                - Memory Full?
                  - Yes
                  - Perform Page Replacement
                  - Page Tables Updated
                - Update TLB
                - Return to Faulted Instruction
                - No
                - Access Page Table
                - No
                - Access Page Table
                - Page Table Entry in TLB?
                  - Yes
                  - Page Fault Handling Routine
                  - OS Instructs CPU to Read the Page from Disk
                  - CPU Activates I/O Hardware
                  - Page Transferred from Disk to Main Memory
                  - Memory Full?
                    - Yes
                    - Perform Page Replacement
                    - Page Tables Updated
                  - Update TLB
                  - Return to Faulted Instruction
                  - No
                  - Access Page Table
                  - No
                  - Access Page Table
                  - Page Table Entry in TLB?
                    - Yes
                    - Page Fault Handling Routine
                    - OS Instructs CPU to Read the Page from Disk
                    - CPU Activates I/O Hardware
                    - Page Transferred from Disk to Main Memory
                    - Memory Full?
                      - Yes
                      - Perform Page Replacement
                      - Page Tables Updated
                    - Update TLB
                    - Return to Faulted Instruction
                    - No
                    - Access Page Table
                    - No
                    - Access Page Table
                    - Page Table Entry in TLB?
                      - Yes
                      - Page Fault Handling Routine
                      - OS Instructs CPU to Read the Page from Disk
                      - CPU Activates I/O Hardware
                      - Page Transferred from Disk to Main Memory
                      - Memory Full?
                        - Yes
                        - Perform Page Replacement
                        - Page Tables Updated
                      - Update TLB
                      - Return to Faulted Instruction
                      - No
                      - Access Page Table
                      - No
                      - Access Page Table
                      - Page Table Entry in TLB?
                        - Yes
                        - Page Fault Handling Routine
                        - OS Instructs CPU to Read the Page from Disk
                        - CPU Activates I/O Hardware
                        - Page Transferred from Disk to Main Memory
                        - Memory Full?
                          - Yes
                          - Perform Page Replacement
                          - Page Tables Updated
                        - Update TLB
                        - Return to Faulted Instruction
                        - No
                        - Access Page Table
                        - No
                        - Access Page Table
                        - Page Table Entry in TLB?
                          - Yes
                          - Page Fault Handling Routine
                          - OS Instructs CPU to Read the Page from Disk
                          - CPU Activates I/O Hardware
                          - Page Transferred from Disk to Main Memory
                          - Memory Full?
                            - Yes
                            - Perform Page Replacement
                            - Page Tables Updated
                          - Update TLB
                          - Return to Faulted Instruction
                          - No
                          - Access Page Table
                          - No
                          - Access Page Table
                          - Page Table Entry in TLB?
                            - Yes
                            - Page Fault Handling Routine
                            - OS Instructs CPU to Read the Page from Disk
                            - CPU Activates I/O Hardware
                            - Page Transferred from Disk to Main Memory
                            - Memory Full?
                              - Yes
                              - Perform Page Replacement
                              - Page Tables Updated
                            - Update TLB
                            - Return to Faulted Instruction
                            - No
                            - Access Page Table
                            - No
                            - Access Page Table
                            - Page Table Entry in TLB?
                              - Yes
                              - Page Fault Handling Routine
                              - OS Instructs CPU to Read the Page from Disk
                              - CPU Activates I/O Hardware
                              - Page Transferred from Disk to Main Memory
                              - Memory Full?
                                - Yes
                                - Perform Page Replacement
                                - Page Tables Updated
                              - Update TLB
                              - Return to Faulted Instruction
                              - No
                              - Access Page Table
                              - No
                              - Access Page Table
                              - Page Table Entry in TLB?
                                - Yes
                                - Page Fault Handling Routine
                                - OS Instructs CPU to Read the Page from Disk
                                - CPU Activates I/O Hardware
                                - Page Transferred from Disk to Main Memory
                                - Memory Full?
                                  - Yes
                                  - Perform Page Replacement
                                  - Page Tables Updated
                                - Update TLB
                                - Return to Faulted Instruction
                                - No
                                - Access Page Table
                                - No
                                - Access Page Table
                                - Page Table Entry in TLB?
                                  - Yes
                                  - Page Fault Handling Routine
                                  - OS Instructs CPU to Read the Page from Disk
                                  - CPU Activates I/O Hardware
                                  - Page Transferred from Disk to Main Memory
                                  - Memory Full?
                                    - Yes
                                    - Perform Page Replacement
                                    - Page Tables Updated
                                  - Update TLB
                                  - Return to Faulted Instruction
                                  - No
                                  - Access Page Table
                                  - No
                                  - Access Page Table
                                  - Page Table Entry in TLB?
                                    - Yes
                                    - Page Fault Handling Routine
                                    - OS Instructs CPU to Read the Page from Disk
                                    - CPU Activates I/O Hardware
                                    - Page Transferred from Disk to Main Memory
                                    - Memory Full?
                                      - Yes
                                      - Perform Page Replacement
                                      - Page Tables Updated
                                    - Update TLB
                                    - Return to Faulted Instruction
                                    - No
                                    - Access Page Table
                                    - No
                                    - Access Page Table
                                    - Page Table Entry in TLB?
                                      - Yes
                                      - Page Fault Handling Routine
                                      - OS Instructs CPU to Read the Page from Disk
                                      - CPU Activates I/O Hardware
                                      - Page Transferred from Disk to Main Memory
                                      - Memory Full?
                                        - Yes
                                        - Perform Page Replacement
                                        - Page Tables Updated
                                      - Update TLB
                                      - Return to Faulted Instruction
                                      - No
                                      - Access Page Table
                                      - No
                                      - Access Page Table
                                      - Page Table Entry in TLB?
                                        - Yes
                                        - Page Fault Handling Routine
                                        - OS Instructs CPU to Read the Page from Disk
                                        - CPU Activates I/O Hardware
                                        - Page Transferred from Disk to Main Memory
                                        - Memory Full?
                                          - Yes
                                          - Perform Page Replacement
                                          - Page Tables Updated
                                        - Update TLB
                                        - Return to Faulted Instruction
                                        - No
                                        - Access Page Table
                                        - No
                                        - Access Page Table
                                        - Page Table Entry in TLB?
                                          - Yes
                                          - Page Fault Handling Routine
                                          - OS Instructs CPU to Read the Page from Disk
                                          - CPU Activates I/O Hardware
                                          - Page Transferred from Disk to Main Memory
                                          - Memory Full?
                                            - Yes
                                            - Perform Page Replacement
                                            - Page Tables Updated
                                          - Update TLB
                                          - Return to Faulted Instruction
                                          - No
                                          - Access Page Table
                                          - No
                                          - Access Page Table
                                          - Page Table Entry in TLB?
Using a Translation Lookaside Buffer (TLB) — another look

HW Support for Virtual Memory — A Special Cache for PTEs

1. Virtual address
2. TLB hit
3. Page hit
4. Page Table Update
5. Page Table Snoop

Secondary Storage

CPU

TLB

Main Memory

Page Table

0 1 2

123456

987654

3.b. Invalid Page Request

3.c. Page Fault

3.a. Page hit

2.b. TLB miss

4. Page Table Update

5. Page Table Snoop

https://www.traviswpeters.com/cs460/
Address Translation in a Segmentation/Paging System

Program

Segmentation Mechanism

Page Table

Paging Mechanism

Main Memory

Virtual Address

Segment Table

Segment Table Entry

Page Table Entry

Page Frame

(a) Paging only

(b) Segmentation only

(c) Combined segmentation and paging

Figure 8.12  Address Translation in a Segmentation/Paging System

Figure 8.1 Typical Memory Management Formats
Memory Management Formats

<table>
<thead>
<tr>
<th>Paging</th>
<th>Segmentation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Virtual Address</td>
<td>Virtual Address</td>
</tr>
<tr>
<td>Page Number</td>
<td>Segment Number</td>
</tr>
<tr>
<td>Offset</td>
<td>Offset</td>
</tr>
<tr>
<td>Page Table Entry</td>
<td>Segment Table Entry</td>
</tr>
<tr>
<td>PM Other Control Bits</td>
<td>PM Other Control Bits</td>
</tr>
<tr>
<td>Frame Number</td>
<td>Length</td>
</tr>
<tr>
<td></td>
<td>Segment Base</td>
</tr>
</tbody>
</table>

**If Page is Present (P)**

→ *use frame number*

**If Page Modified (M)**

→ *write out*

P= present bit
M = Modified bit
Protection Between Segments

- Configurable settings for segments
  - readable
  - writable
  - executable

- This is possible on a per-page level, but it is less intuitive/kind of awkward since paging structure is not visible to programmers.
Designing an OS with Virtual Memory

**Goal:** Reduce Frequency of Page Faults
Designing an OS with Virtual Memory

**Goal:** Reduce Frequency of Page Faults

**Fetch Policy** — when should a page be brought into main memory?

- **Demand Paging**
  
  Bring pages in when a reference is made (as needed / on demand)

- **Prepaging**
  
  Bring in a number of pages at once (e.g., other nearby pages on disk)
Designing an OS with Virtual Memory

Goal: Reduce Frequency of Page Faults

Placement Policy — where should a page be put in main memory?
(Re: Chapter 7)

• Segmentation
  → Best-Fit, First-Fit, Etc.

• Paging
  → Not an issue (page-frame mappings are equally efficient)
Designing an OS with Virtual Memory

**Goal:** Reduce Frequency of Page Faults

**Replacement Policy** — which page* should be replaced when a new page must be brought into main memory?**

- **Random Page Replacement**
  Choose a page randomly

- **Optimal (OPT)**
  Replace the page that will not be used the most time in the future

- **LRU**
  Replace the page that has not been used for the longest time

- **FIFO**
  Replace the page that has been in primary memory the longest; use a circular buffer

- **Clock** *(approximation of LRU)*
  Basically, FIFO w/ a “use” bit; cycle through pages, replace pages not used recently

- **LFO**
  Replace the page that is used least often

- **2nd Chance**
  Choose a victim to evict; “mark” it; spare it until later; if encountered again, then page out

*Benchmarks*

- Nearly optimal
- Complex

**Goal:** Reduce Frequency of Page Faults

Want to replace an unlocked page that is least likely to be referenced in the near future.

**Most interesting if we assume that all frames in main memory are occupied — then a page fault occurs.**
**Example: Page-Replacement Algorithms**

Order of page references: 2 3 2 1 5 2 4 5 3 2 5 2

<table>
<thead>
<tr>
<th>Page address stream</th>
<th>2</th>
<th>3</th>
<th>2</th>
<th>1</th>
<th>5</th>
<th>2</th>
<th>4</th>
<th>5</th>
<th>3</th>
<th>2</th>
<th>5</th>
<th>2</th>
</tr>
</thead>
<tbody>
<tr>
<td>OPT</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>LRU</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>FIFO</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>CLOCK</td>
<td>2*</td>
<td>2*</td>
<td>2*</td>
<td>2*</td>
<td>2*</td>
<td>5*</td>
<td>5*</td>
<td>5*</td>
<td>3*</td>
<td>3*</td>
<td>3*</td>
<td>3*</td>
</tr>
<tr>
<td></td>
<td>3*</td>
<td>3*</td>
<td>3*</td>
<td>3*</td>
<td>3*</td>
<td>3*</td>
<td>3*</td>
<td>3*</td>
<td>3*</td>
<td>3*</td>
<td>3*</td>
<td>3*</td>
</tr>
</tbody>
</table>

*F = page fault occurring after the frame allocation is initially filled*
Designing an OS with Virtual Memory

**Goal:** Reduce Frequency of Page Faults

**Resident Set Management** — *how many pages should be in main memory?*
- Resident Set Size (Fixed vs. Variable)
- Replacement Scope (Local [current process] vs. Global [everything])

**Cleaning Policy** — *when should a page be written out from main memory to secondary memory?* (opposite of “Fetch Policy”)

**Load Control** — *how many processes should be in main memory?* (i.e., the level/degree of multiprogramming)