Memory Hierarchy:
Locality and Cache Implementation

CptS 260
Introduction to Computer Architecture
Week 5.3
Wed 2014/07/09
Reading Assignment

• [P&H14] Chapter 5 Large and Fast: Exploiting Memory Hierarchy
  – §5.1 Introduction
  – §5.2 Memory Technologies
  – §5.3 The Basics of Caches
  – §5.4 Measuring and Improving Cache Performance
  – §5.5 Dependable Memory Hierarchy
  – §5.6 Virtual Machines
  – §5.9 Using a Finite-State Machine to Control a Simple Cache
  – §5.8 A Common Framework for Memory Hierarchies
  – §5.7 Virtual Memory
  – §5.10 Parallelism and Memory Hierarchy: Cache Coherence
## Cache and Memory Hierarchy

<table>
<thead>
<tr>
<th>Type</th>
<th>How</th>
<th>Access (ns)</th>
<th>$/GB 2008</th>
<th>2014</th>
</tr>
</thead>
<tbody>
<tr>
<td>SRAM</td>
<td>bilatched 4-6 trans.</td>
<td>0.5 – 2.5</td>
<td>$2K–$5K</td>
<td>$1K–$3K</td>
</tr>
<tr>
<td>DRAM</td>
<td>1 transistor + capacitor</td>
<td>50 – 70</td>
<td>$20–$75</td>
<td>$10–$66</td>
</tr>
<tr>
<td>magnetic</td>
<td></td>
<td>2,900,000</td>
<td>$0.20–$2</td>
<td>$0.05–$1</td>
</tr>
<tr>
<td>solid state</td>
<td>NAND flash</td>
<td>&lt; 100,000</td>
<td>$0.45–$3</td>
<td></td>
</tr>
</tbody>
</table>
The Principle of Locality

§5.1 Introduction

• “At any instant”, program uses subset of address space ➔ the “working set”
  – Worker needs subset of tools
  – Speaker uses subset of vocabulary

• Temporal locality
  – Accessed recently ➔ likely accessed again soon
    • loop variables
    • fork, spoon

• Spatial locality
  – Accessed nearby ➔ likely accessed soon
    • sequential instructions
    • arrays, data structures
    • local variables in same block / function
Exploiting Locality

(Use registers: optimization)

Save “working set” in cache: fast!!

Swap recent/nearby into memory: lots!

Store on disk: cheap! non-volatile!

Store on Redundant Array of Independent Disks (RAID): robust!
Paradigm: Cach’e is like Hawai’i

CPU request size: lw, sb, etc.

Cache block size: 1–16 words
Cache Implementation

• Data
  – Block size at design time
    Bigger: spatial locality
    DMA, burst transfers
    wasted effort

• Tag
  – A portion of the address

• Valid bit
  – Cold start: false
  – After load: true

• Dirty bit
  – true = consistent with memory
  – false = inconsistent