Lecture 11: Memory Hierarchy—Reducing Hit Time, Main Memory, and Examples

Professor David A. Patterson

Computer Science 252

Spring 1998

Review: Reducing Misses

$$CPUtime = IC \cdot \left(CPI_{\text{Execution}} + \frac{Memory\ \ accesses}{Instruction} \cdot \text{Miss rate} \cdot Miss\ \ penalty \right) \cdot Clock\ \ cycle\ \ time$$

- 3 Cs: Compulsory, Capacity, Conflict Misses
- Reducing Miss Rate
  1. Reduce Misses via Larger Block Size
  2. Reduce Misses via Higher Associativity
  3. Reducing Misses via Victim Cache
  4. Reducing Misses via Pseudo-Associativity
  5. Reducing Misses by HW Prefetching Instr, Data
  6. Reducing Misses by SW Prefetching Data
  7. Reducing Misses by Compiler Optimizations
- Remember danger of concentrating on just one parameter when evaluating performance

Reducing Miss Penalty Summary

$$CPUtime = IC \cdot \left(CPI_{Execution} + \frac{Memory\ \ accesses}{Instruction} \cdot \textbf{Miss rate} \cdot \left(Miss\ \ penalty \right) \right) \cdot Clock\ \ cycle\ \ time$$

- **Five techniques**
  - Read priority over write on miss
  - Subblock placement
  - Early Restart and Critical Word First on miss
  - Non-blocking Caches (Hit under Miss, Miss under Miss)
  - Second Level Cache
- **Can be applied recursively to Multilevel Caches**
  - Danger is that time to DRAM will grow with multiple levels in between
  - First attempts at L2 caches can make things worse, since increased worst case is worse
- **Out-of-order CPU can hide L1 data cache miss (≈3–5 clocks), but stall on L2 miss (≈40–100 clocks)?**