Chapter 09: Caches

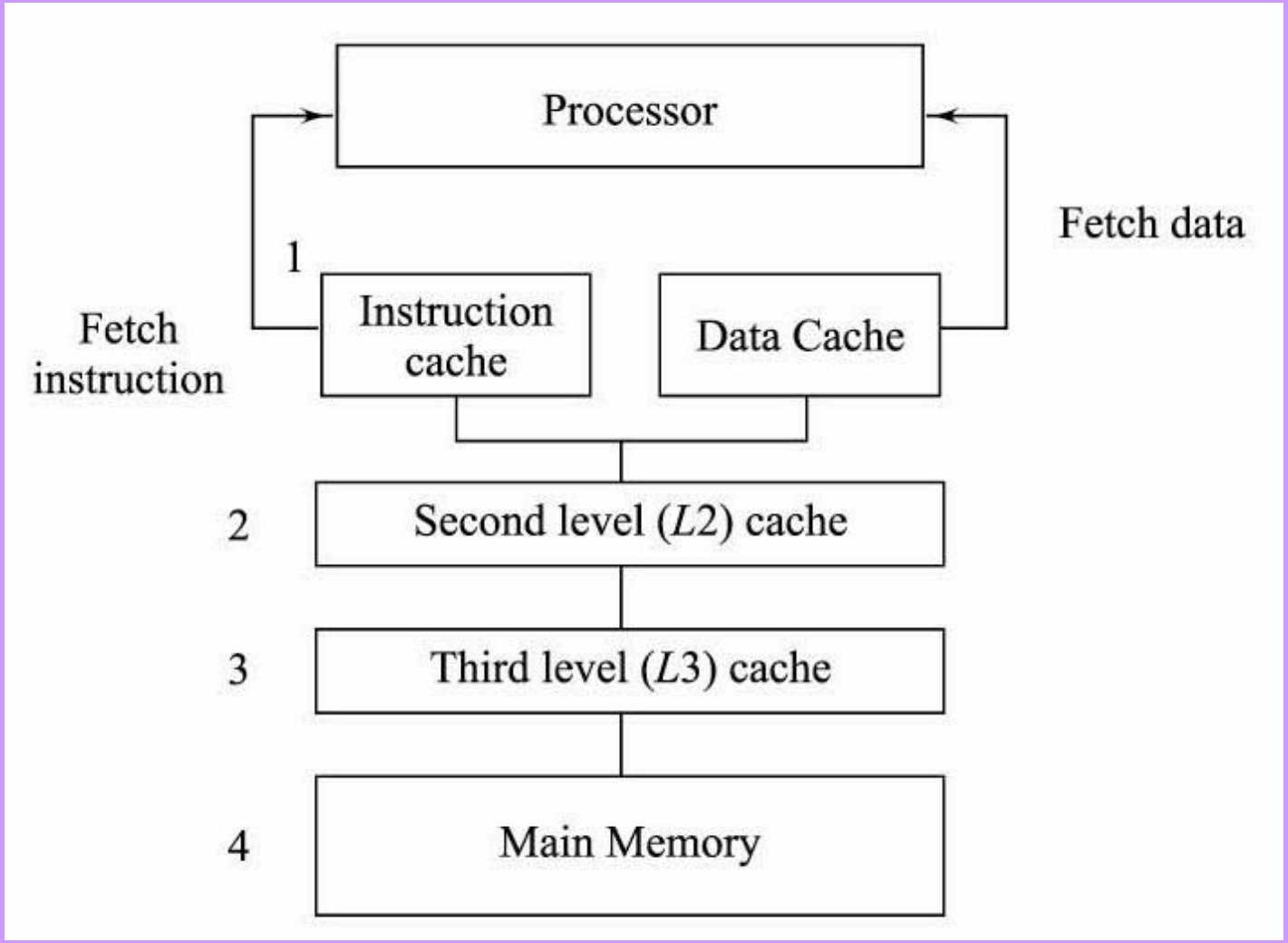
Lesson 06:
Cache Implementation

Objective

- Understand the use of multilevel caches

Implementation of Multilevel caches

Multilevel caches



Multi-level caches advantage

- Gives the processor the additional bandwidth provided by a Harvard architecture at the top level of the memory system, while simplifying the design of the lower levels
- Significantly improves the average memory access time of a system when each level must have a significantly larger capacity than the level above it in the hierarchy

Multi-level caches

- Locality of reference seen by each level decreases as one gets deeper in the hierarchy
- Requests to recently referenced data are handled by the upper levels of the memory system
- Requests that make it to the lower levels tend to be more widely distributed across the address space

Multi-level caches

- Caches with larger capacities tend to be slower
- Speed benefit of separate instruction and data caches are not as significant in lower levels of the memory hierarchy, another argument in favor of using unified caches for these levels

Implementation of Multilevel caches design in computers

Early 1990s' most common hierarchy for computers

- First-level (L1) cache to be relatively small and located on the same chip as the processor
- Lower-level caches were implemented off-chip out of discrete SRAM chips
- Capacities of 4 to 16 kB were not unusual in L1 caches, with L2 caches reaching 64 to 256 kB

late 1990s' hierarchy for personal computers and desktop workstations

- As the number of transistors that can be fabricated on a chip has increased, additional levels of cache have moved onto the processor chip
- Many current systems have both their first-level and second-level caches on the same chip as the processor, or at least in the integrated circuits package
- Third-level caches can be on-chip or off-chip
- Third-level can be many megabytes in size

Summary

We learnt

- Use of multi-level caches

End of Lesson 06 on

Cache Implementation