Chapter 10

Multicore Computers

Multi-Core Computer

- A multi-core processor is a processing system composed of two or more independent cores (or CPUs). The cores are typically integrated onto a single integrated circuit die (known as a chip multiprocessor or CMP).

- A many-core processor is one in which the number of cores is large enough that traditional multi-processor techniques are no longer efficient
  - Somewhere in the range of several tens of cores - and likely requires a network on chip.

Multi-Core Computer

- dual-core processor contains two independent microprocessors.
- A dual core set-up is somewhat comparable to having multiple, separate processors installed in the same computer.
  - But because the two processors are actually plugged into the same socket, the connection between them is faster.
- Ideally, a dual core processor is nearly twice as powerful as a single core processor.
  - In practice, performance gains are about 50%:
    - A dual core processor is likely to be about one-and-a-half times as powerful as a single core processor.

Multi-Core Computer

- A multi-core processor implements multiprocessing in a single physical package.
  - Cores may or may not share caches
  - May implement message passing or shared memory inter-core communication methods.
- All cores are identical in symmetric multi-core systems.
  - EX: Intel Core 2 Duo
- They are not identical in asymmetric multi-core systems.
  - EX: IBM Cell Processor

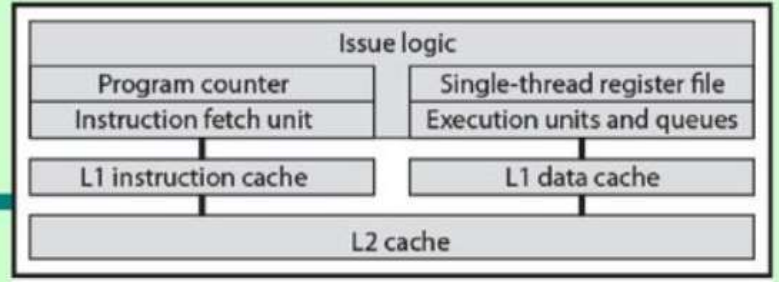
CMP benefits

- with a shared on-chip cache memory, communication events can be reduced to just a handful of processor cycles.
- therefore with low latencies, communication delays have a much smaller impact on overall performance.
- threads can also be much smaller and still be effective.
- automatic parallelization more feasible.

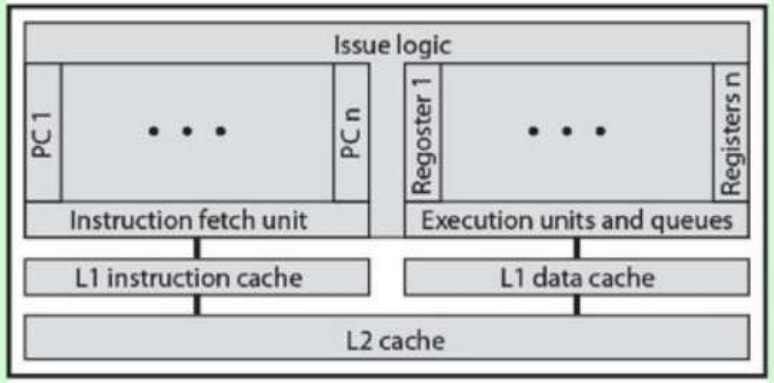
Hardware Performance Issues

- Microprocessors have seen an exponential increase in performance
  - —Improved organization
  - —Increased clock frequency
- Increase in Parallelism
  - —Pipelining
  - —Superscalar
  - —Simultaneous multithreading (SMT)
- Diminishing returns
  - —More complexity requires more logic
  - —Increasing chip area for coordinating and signal transfer logic
    - – Harder to design, make and debug

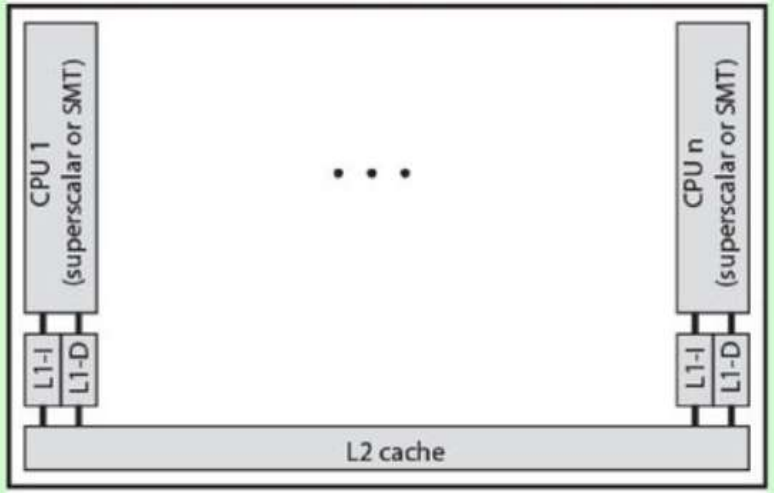
Alternative Chip Organizations



(a) Superscalar

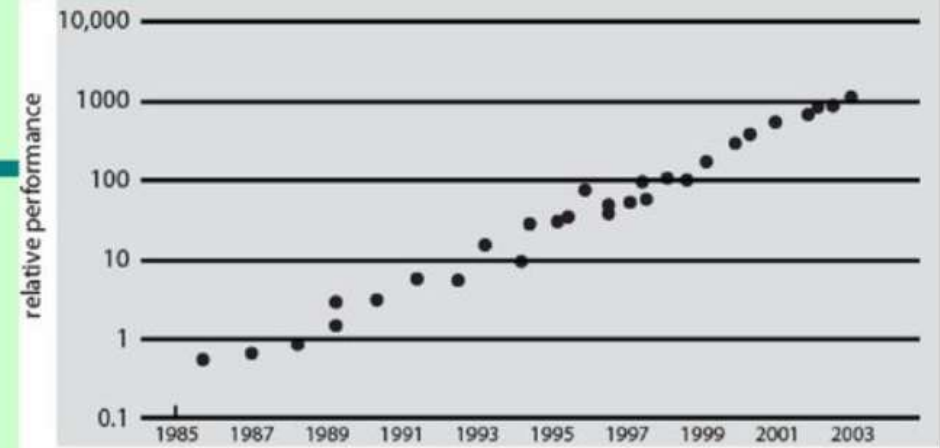


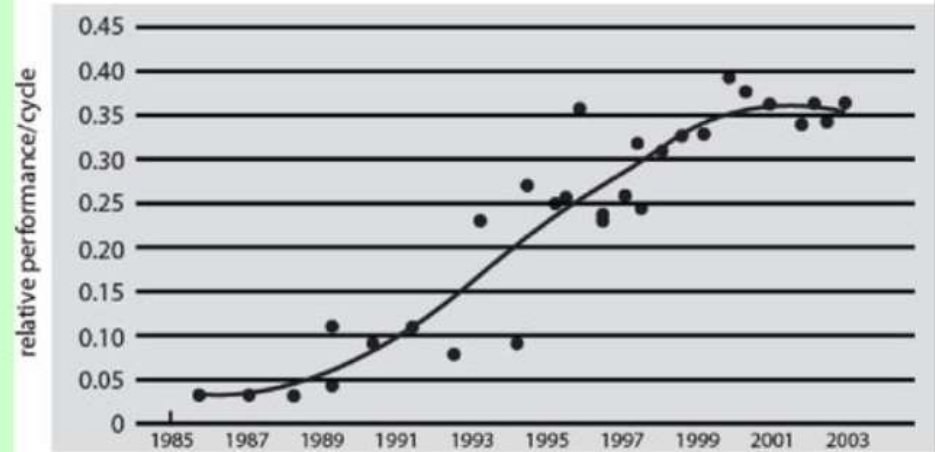
(b) Simultaneous multithreading

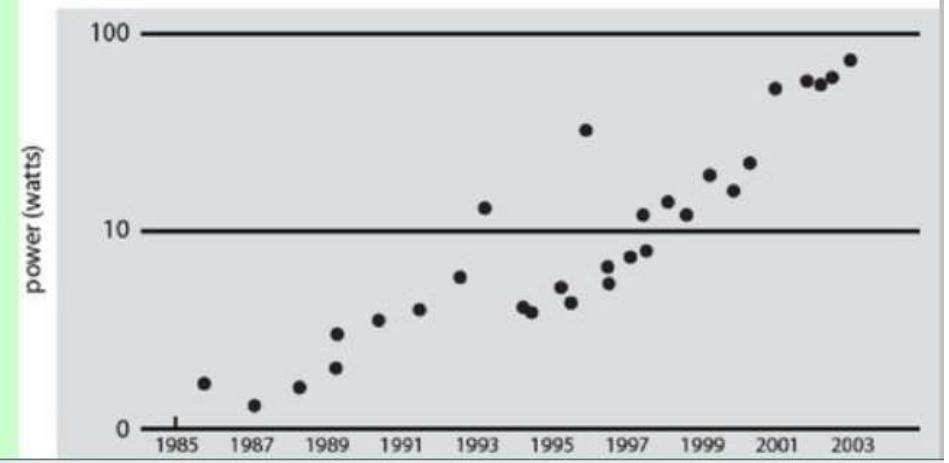


(c) Multicore

Intel Hardware Trends



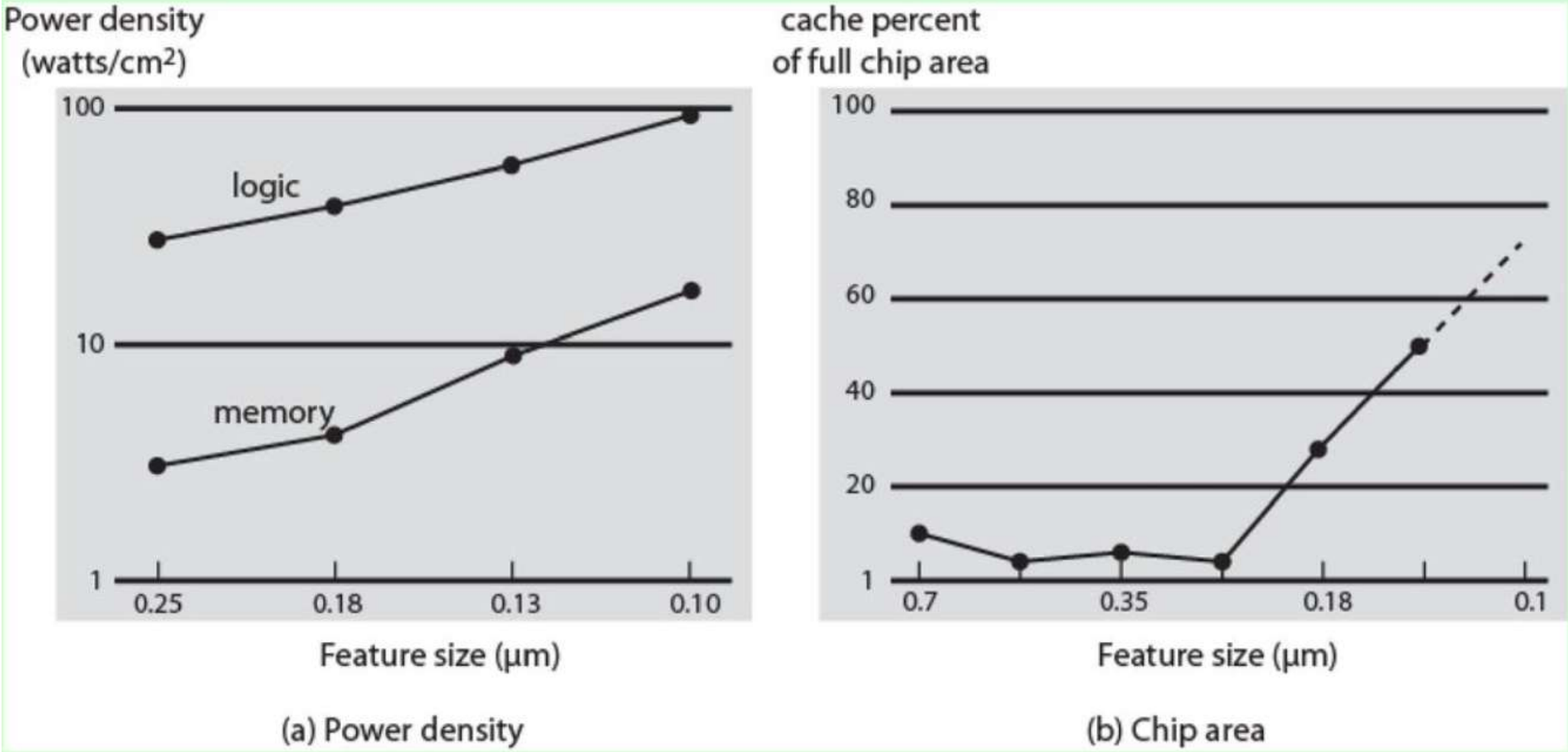




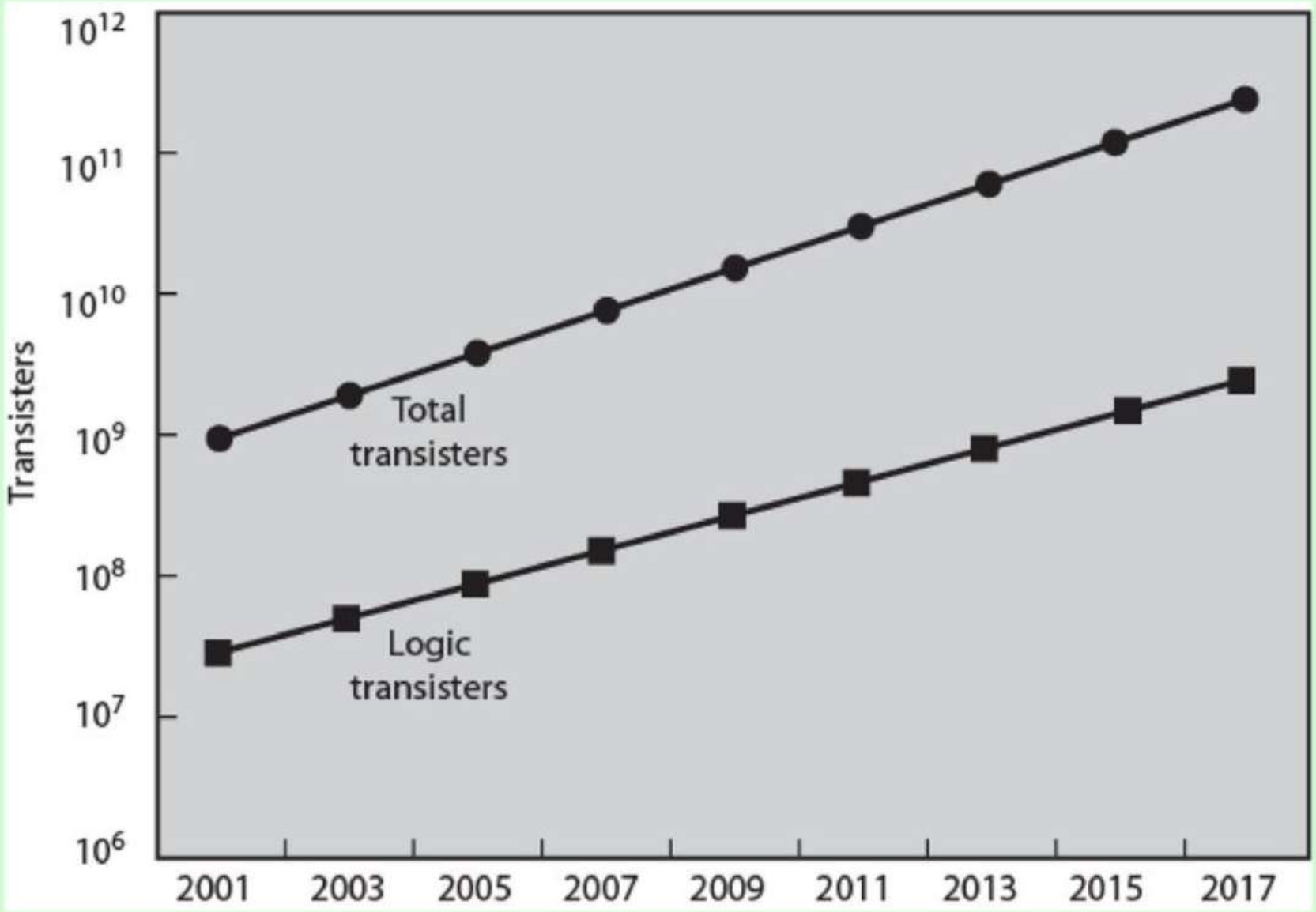
Increased Complexity

- Power requirements grow exponentially with chip density and clock frequency
  - Can use more chip area for cache
    - Smaller
    - Order of magnitude lower power requirements
- By 2015
  - 100 billion transistors on 300mm^2 die
    - Cache of 100MB
    - 1 billion transistors for logic
- Pollack's rule:
  - Performance is roughly proportional to square root of increase in complexity
    - Double complexity gives 40% more performance
- Multicore has potential for near-linear improvement
- Unlikely that one core can use all cache effectively

Power and Memory Considerations



Chip Utilization of Transistors



Software Performance Issues

- Performance benefits dependent on effective exploitation of parallel resources
- Even small amounts of serial code impact performance
  - —10% inherently serial on 8 processor system gives only 4.7 times performance
- Communication, distribution of work and cache coherence overheads
- Some applications effectively exploit multicore processors

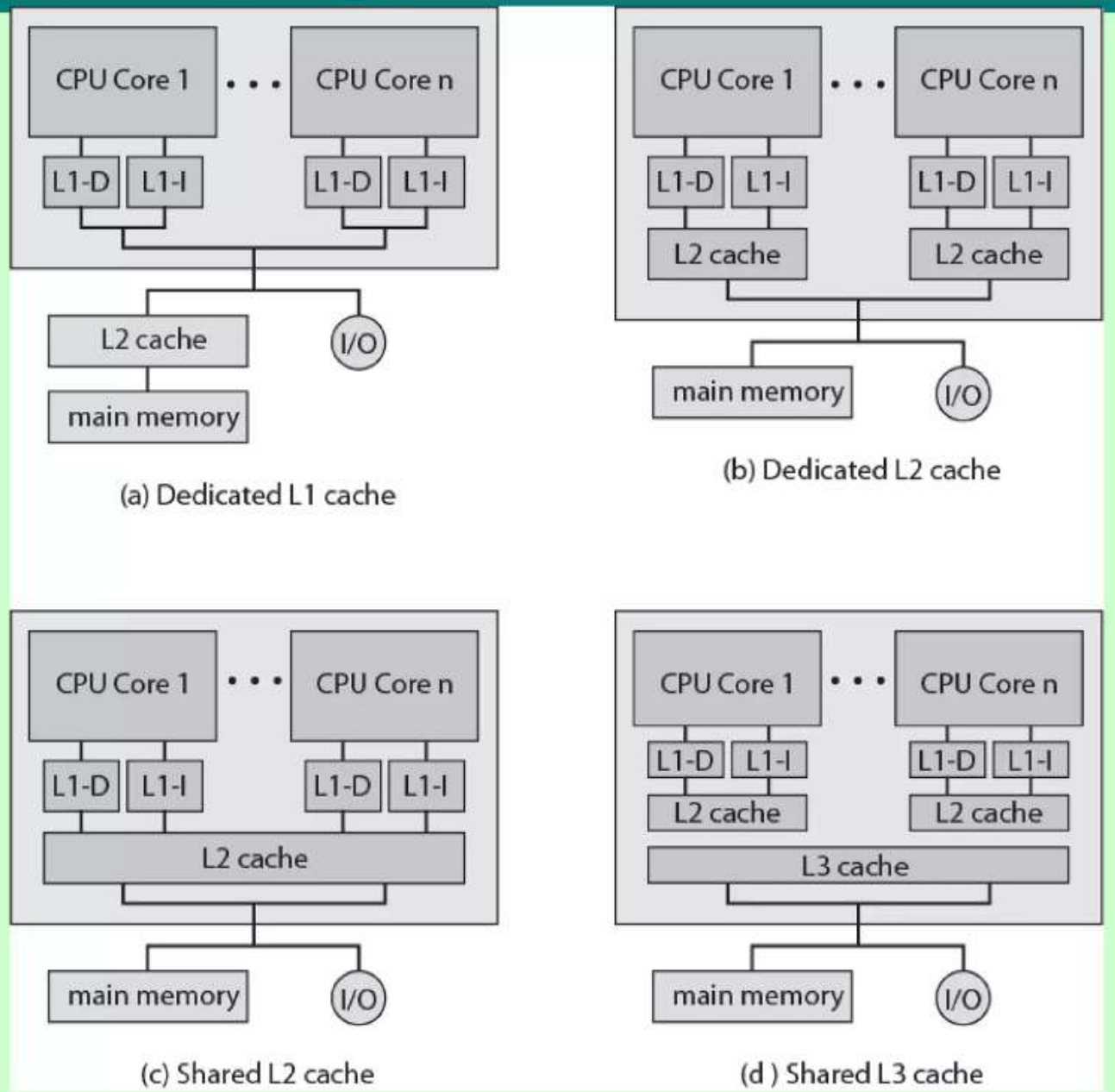
Effective Applications for Multicore Processors

- Database
- Servers handling independent transactions
- Multi-threaded native applications
  - Lotus Domino, Siebel CRM
- Multi-process applications
  - Oracle, SAP, PeopleSoft
- Java applications
  - Java VM is multi-thread with scheduling and memory management
  - Sun's Java Application Server, BEA's Weblogic, IBM Websphere, Tomcat
- Multi-instance applications
  - One application running multiple times

Multicore Organization

- Number of core processors on chip
- Number of levels of cache on chip
- Amount of shared cache
- Next slide examples of each organization:
- (a) ARM11 MPCore
- (b) AMD Opteron
- (c) Intel Core Duo
- (d) Intel Core i7

Multicore Organization Alternatives



Advantages of shared L2 Cache

- Constructive interference reduces overall miss rate
- Data shared by multiple cores not replicated at cache level
- With proper frame replacement algorithms mean amount of shared cache dedicated to each core is dynamic
  - Threads with less locality can have more cache
- Easy inter-process communication through shared memory
- Cache coherency confined to L1
- Dedicated L2 cache gives each core more rapid access
  - Good for threads with strong locality
- Shared L3 cache may also improve performance