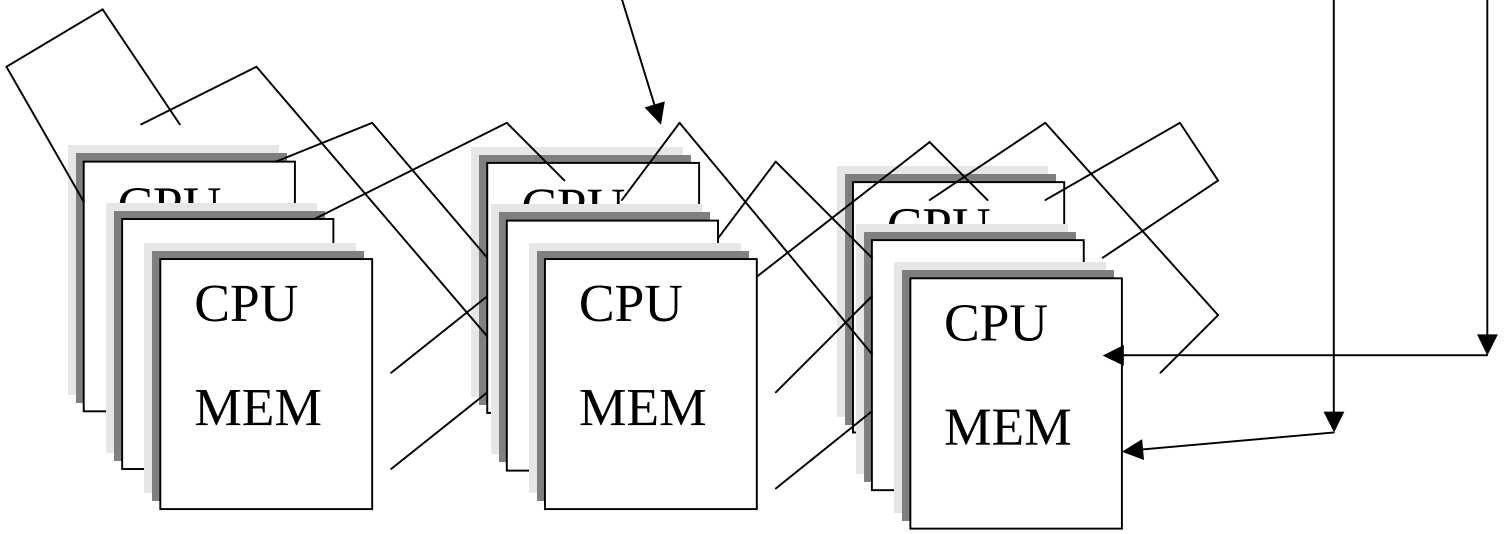
Basic Components of a Parallel (or Serial) Computer

- **Processors**
- **Memory**
- Interconnect Network



Processor Related Terms

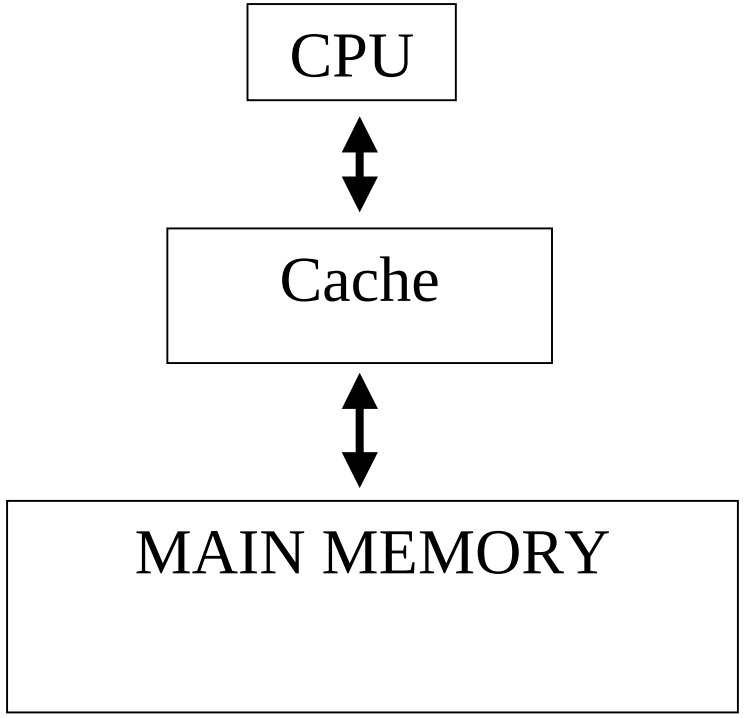
- **RISC: Reduced Instruction Set Computers**
- **PIPELINE : Technique where multiple instructions are overlapped in execution**
- **SUPERSCALAR: Multiple instructions per clock period**

Network Interconnect Related Terms

- **LATENCY : How long does it take to start sending a "message"? Units are generally microseconds or milliseconds.**
- **BANDWIDTH : What data rate can be sustained once the message is started? Units are bytes/sec, Mbytes/sec, Gbytes/sec etc.**
- **TOPLOGY: What is the actual ‘shape’ of the interconnect? Are the nodes connect by a 2D mesh? A ring? Something more elaborate?**

Memory/Cache Related Terms

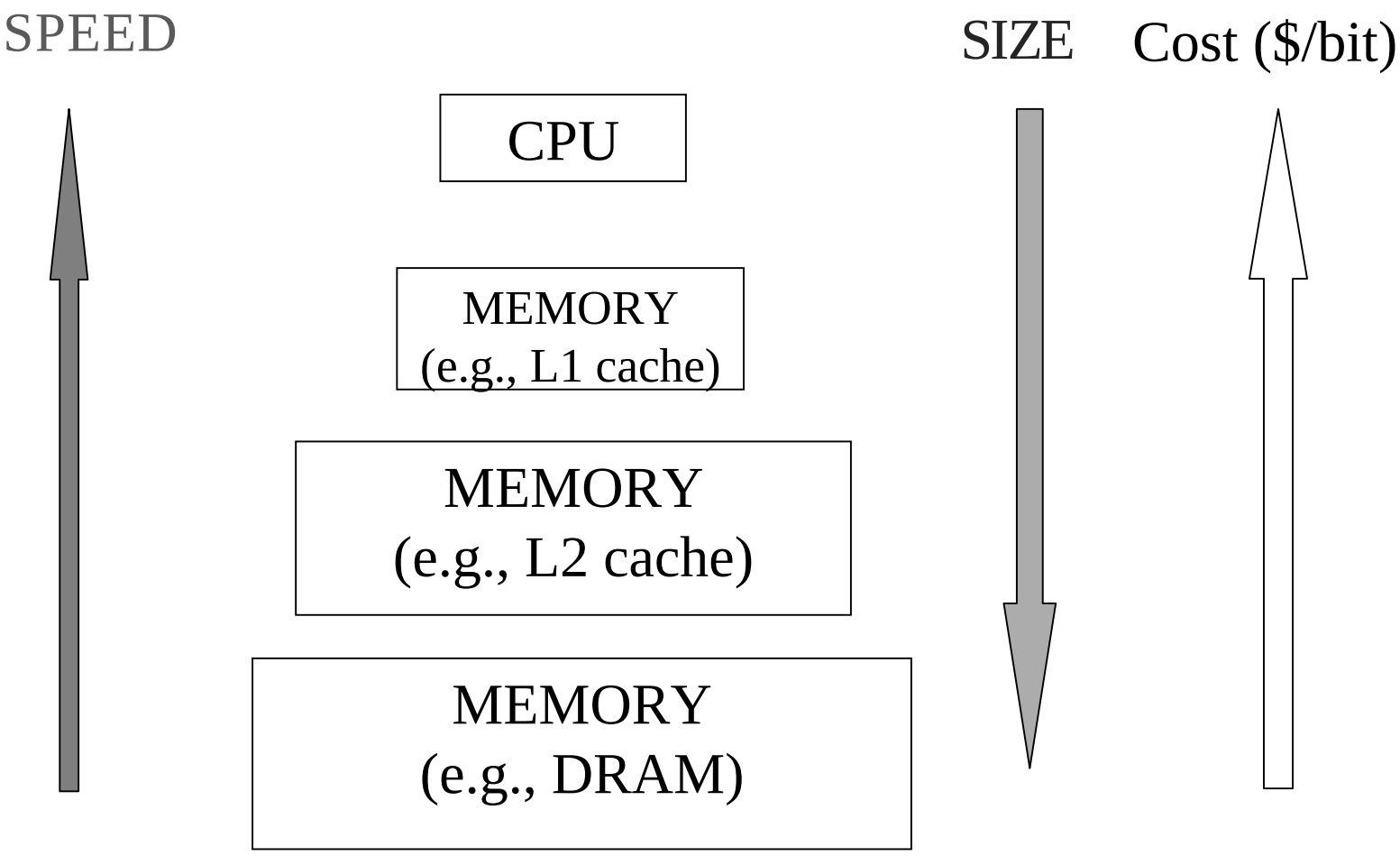
CACHE : Cache is the level of memory hierarchy between the CPU and main memory. Cache is much smaller than main memory and hence there is mapping of data from main memory to cache.



Memory/Cache Related Terms

- **ICACHE : Instruction cache**
- **DCACHE (L1) : Data cache closest to registers**
- **SCACHE (L2) : Secondary data cache**
  - Data from SCACHE has to go through DCACHE to registers
  - SCACHE is larger than DCACHE
  - All processors do not have SCACHE
- **TLB : Translation-lookaside buffer keeps addresses of pages (block of memory) in main memory that have recently been accessed**

Memory/Cache Related Terms (cont.)

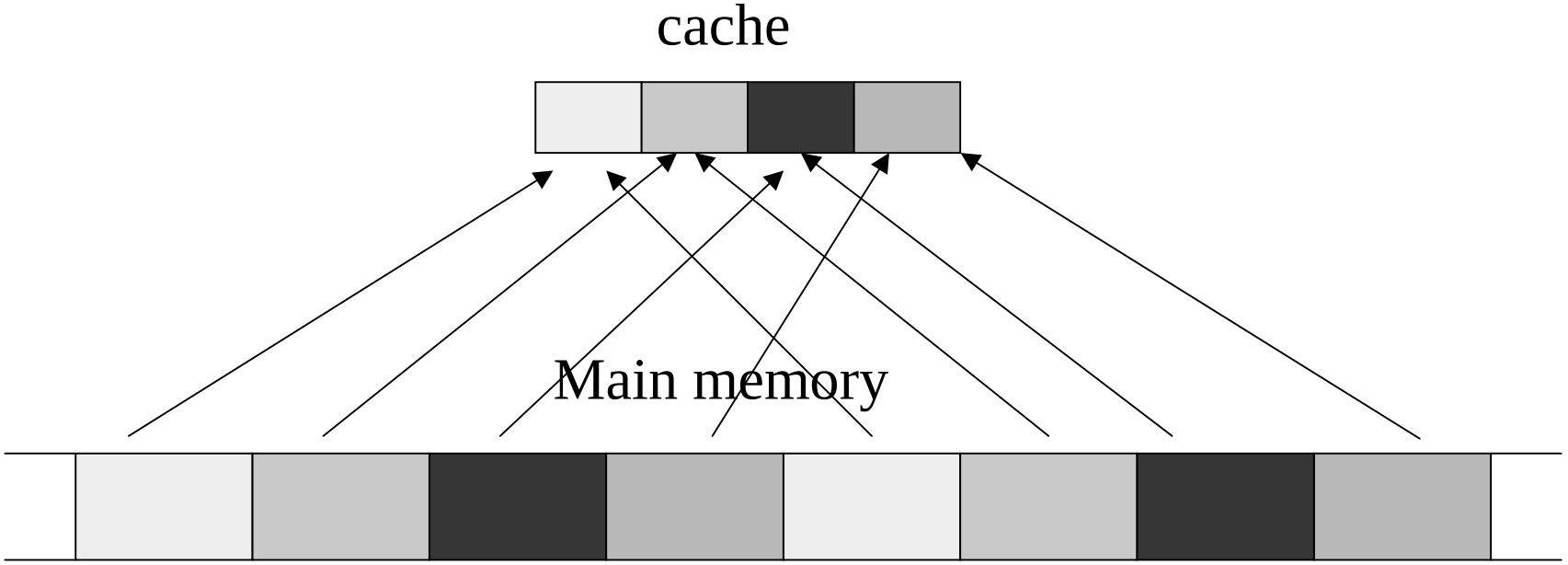


Memory/Cache Related Terms (cont.)

- **The data cache was designed with two key concepts in mind**
  - Spatial Locality
    - When an element is referenced its neighbors will be referenced too
    - Cache lines are fetched together
    - Work on consecutive data elements in the same cache line
  - Temporal Locality
    - When an element is referenced, it might be referenced again soon
    - Arrange code so that date in cache is reused as often as possible

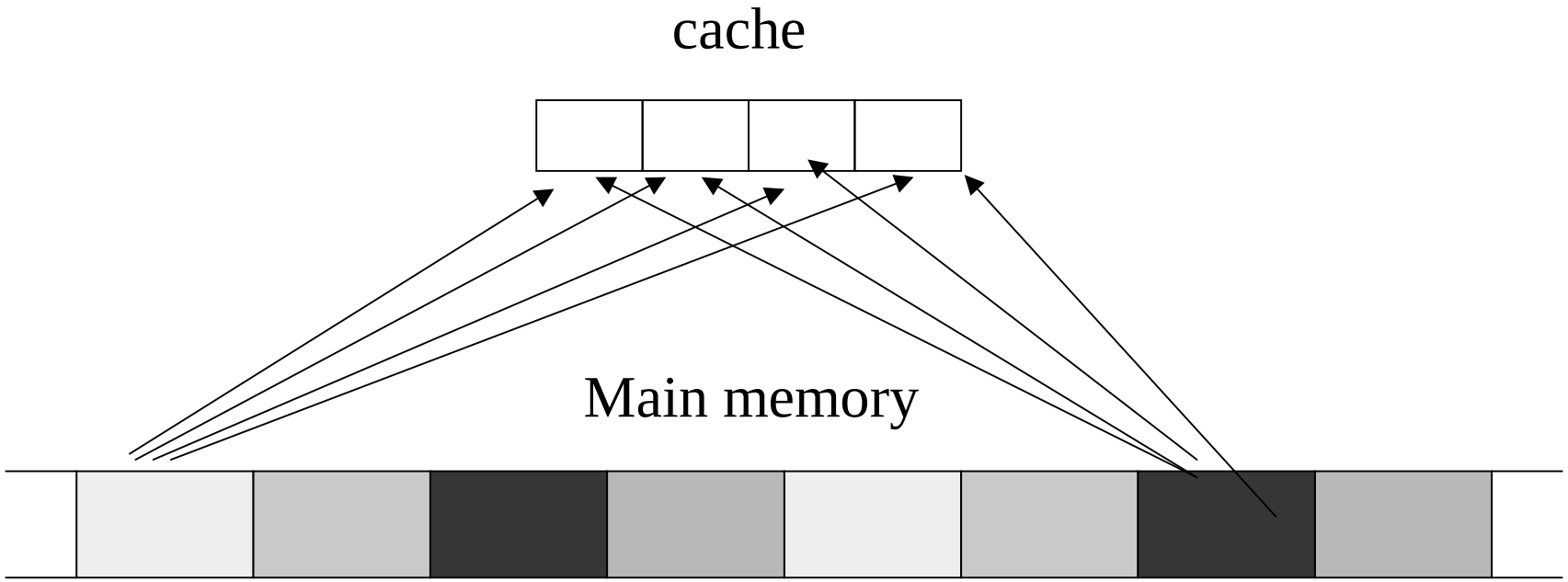
Memory/Cache Related Terms (cont.)

Direct mapped cache: A block from main memory can go in exactly one place in the cache. This is called direct mapped because there is direct mapping from any block address in memory to a single location in the cache.



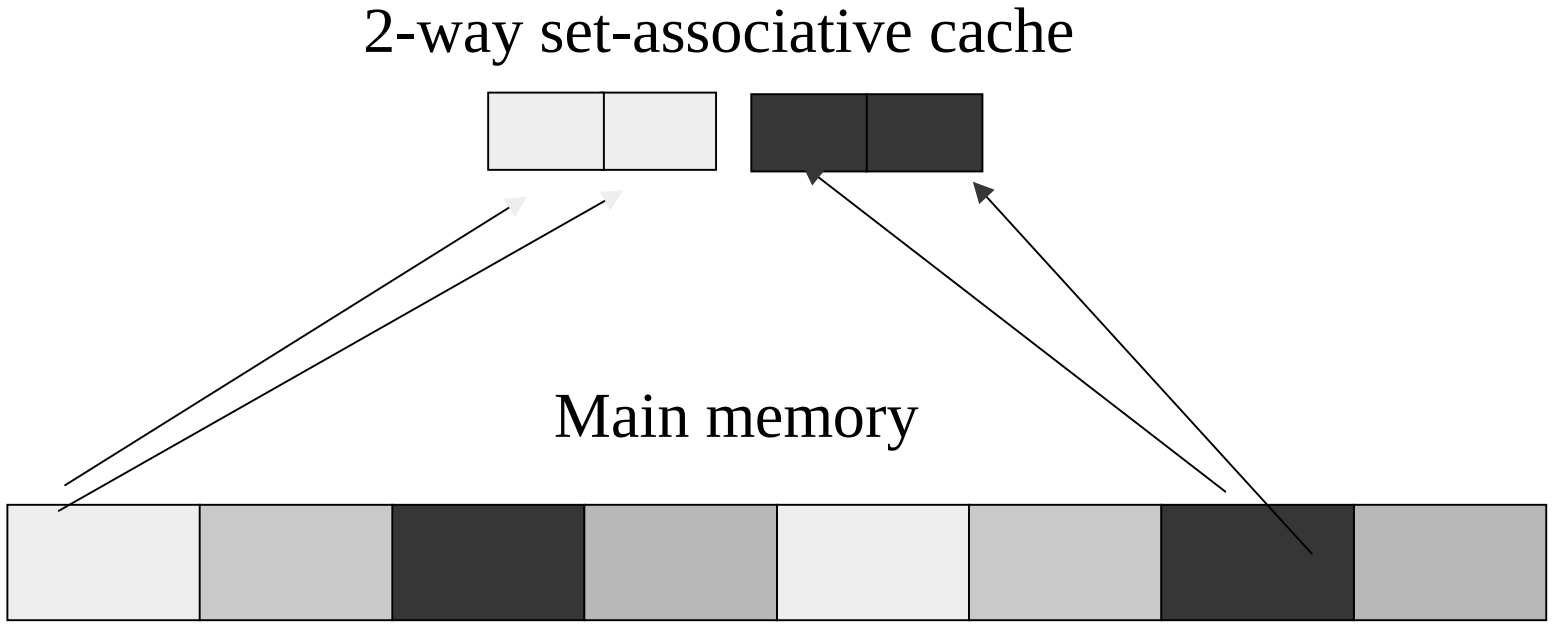
Memory/Cache Related Terms (cont.)

Fully associative cache : A block from main memory can be placed in any location in the cache. This is called fully associative because a block in main memory may be associated with any entry in the cache.



Memory/Cache Related Terms (cont.)

Set associative cache : The middle range of designs between direct mapped cache and fully associative cache is called set-associative cache. In a n-way set-associative cache a block from main memory can go into n (n at least 2) locations in the cache.



Memory/Cache Related Terms (cont.)

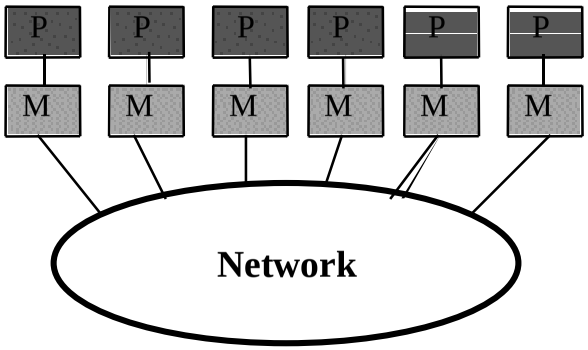
- **Least Recently Used (LRU) : Cache replacement strategy for set associative caches. The cache block that is least recently used is replaced with a new block.**
- **Random Replace : Cache replacement strategy for set associative caches. A cache block is randomly replaced.**

Types of Parallel Computers

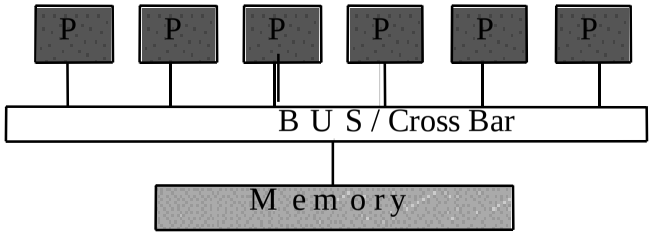
- **Until recently, Flynn's taxonomy was commonly use to classify parallel computers into one of four basic types:**
  - Single instruction, single data (SISD): single scalar processor
  - Single instruction, multiple data (SIMD): Thinking machines CM-2
  - Multiple instruction, single data (MISD): various special purpose machines
  - Multiple instruction, multiple data (MIMD): Nearly all parallel machines

- **However, since the MIMD model "won," a much more useful way to classify modern parallel computers is by their memory model**
  - *shared* memory
  - *distributed* memory
  - (more recently) *hybrid* of the above two (also called multi-tiered, CLUMPS)

Shared and Distributed memory



Distributed memory: each processor has it's own local memory. Must do message passing to exchange data between processors.
(examples: IBM SP, CRAY T3E)



Shared memory: single address space. All processors have access to a pool of shared memory.
(examples: Sun ES10000)

Methods of memory access :
- Bus
- Crossbar