

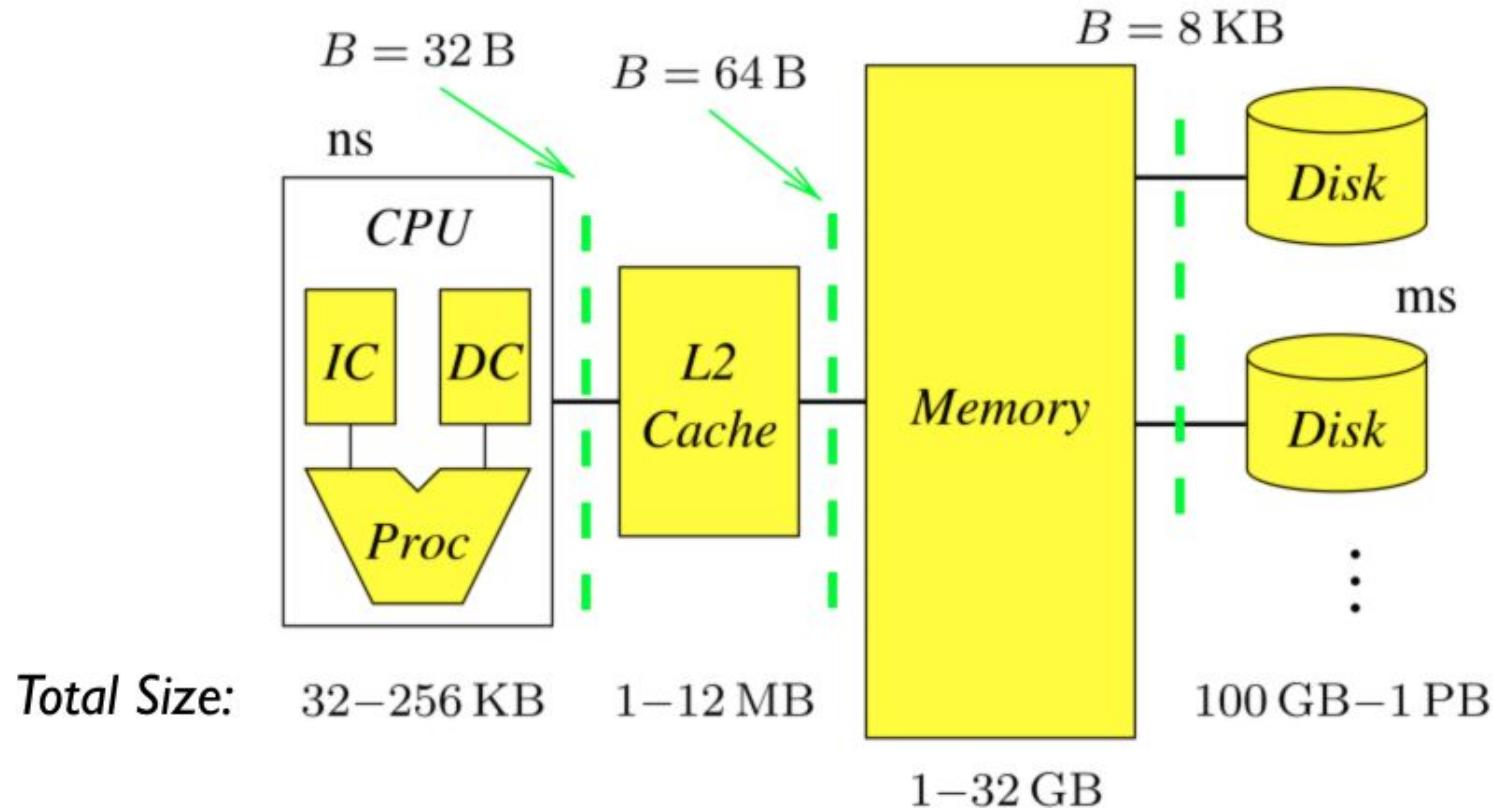
5: The memory Hierarchy

The Memory Hierarchy

- Real systems have several levels storage types:
 - Top of hierarchy: Small and fast storage close to CPU
 - Bottom of Hierarchy: Large and slow storage further from CPU
- Caching is used to transfer data between neighboring levels of the hierarchy.
- To the programmer / compiler does not need to know
 - The hardware provides an **abstraction** : memory looks like a single large array.
- Performance depends on locality of program memory access.

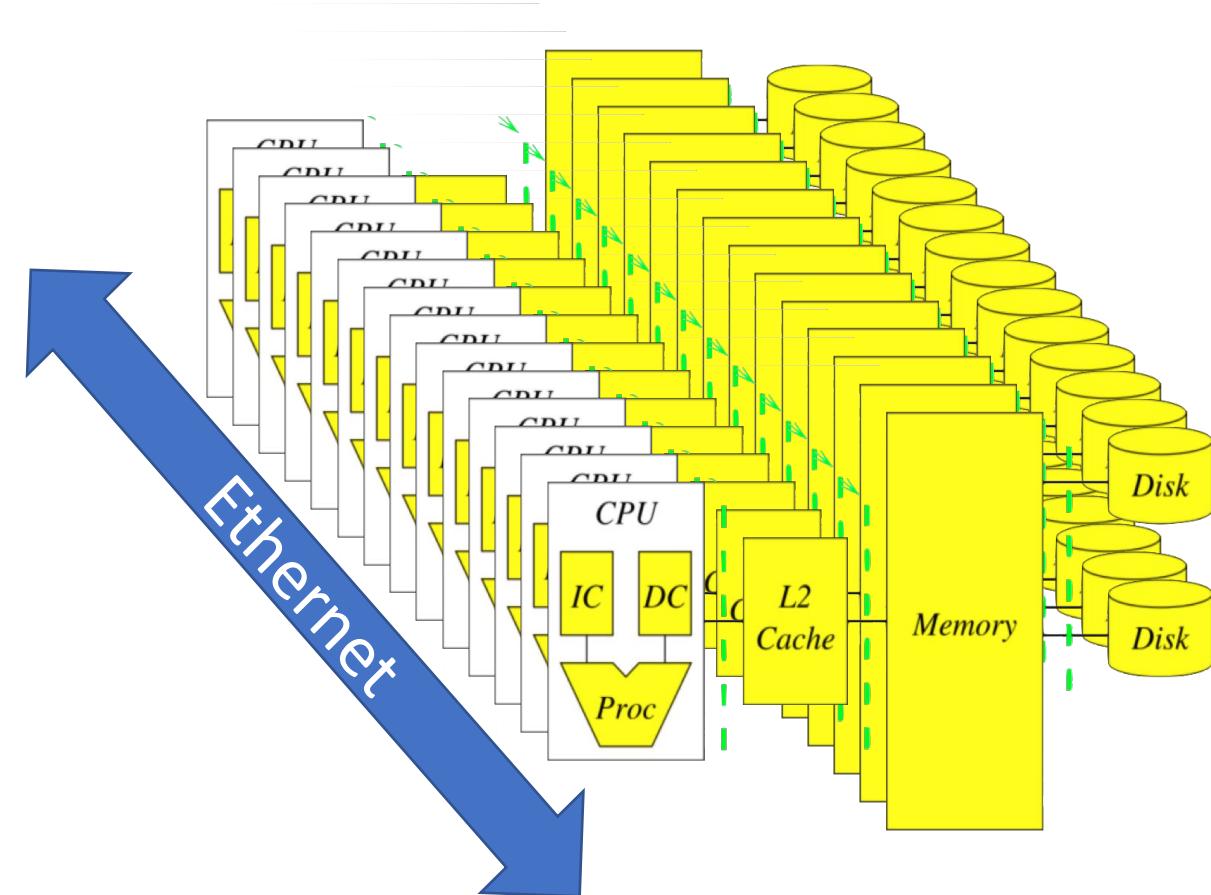
The Memory Hierarchy

B =Block size



Computer clusters extend the memory hierarchy

- A data processing cluster is simply many computers linked through an ethernet connection.
- Storage is shared
- Locality: Data to reside on the computer this will use it.
- “Caching” is replaced by “Shuffling”
- Abstraction is spark RDD.



Sizes and latencies in a typical memory hierarchy.

	CPU (Registers)	L1 Cache	L2 Cache	L3 Cache	Main Memory	Disk Storage	Local Area Network
Size (bytes)	1KB	64KB	256KB	4MB	4-16GB	4-16TB	16TB - 10PB
Latency	300ps	1ns	5ns	20ns	100ns	2-10ms	2-10ms
Block size	64B	64B	64B	64B	32KB	64KB	1.5-64KB

The graph illustrates the exponential increase in size across the memory hierarchy. The y-axis represents size in bytes, and the x-axis represents the memory levels. The curve shows that while the size increases significantly from one level to the next, the latency and block size generally decrease or remain constant.

12 orders of magnitude

6 orders of magnitude

Summary of part 5

- Memory Hierarchy: combining storage banks with different latencies.
- Clusters: multiple computers, connected by ethernet, that share their storage.