Computer Memory

Computers also require memory that can be used by the processor to access and work with data.

Computer memory and its characteristics

Let's take a look at the main characteristics of computer memory:

Capacity: It is the total amount/volume of data that the memory can store. It is preferable for computer memory to have a larger capacity so that more data can be stored and used at any given time.

Access time: It is the time interval between the read/write request and the availability of the data in computer memory. It is essential to have faster access time so that the processor doesn't have to waste time by having to wait for the requested data or to write information into computer memory.

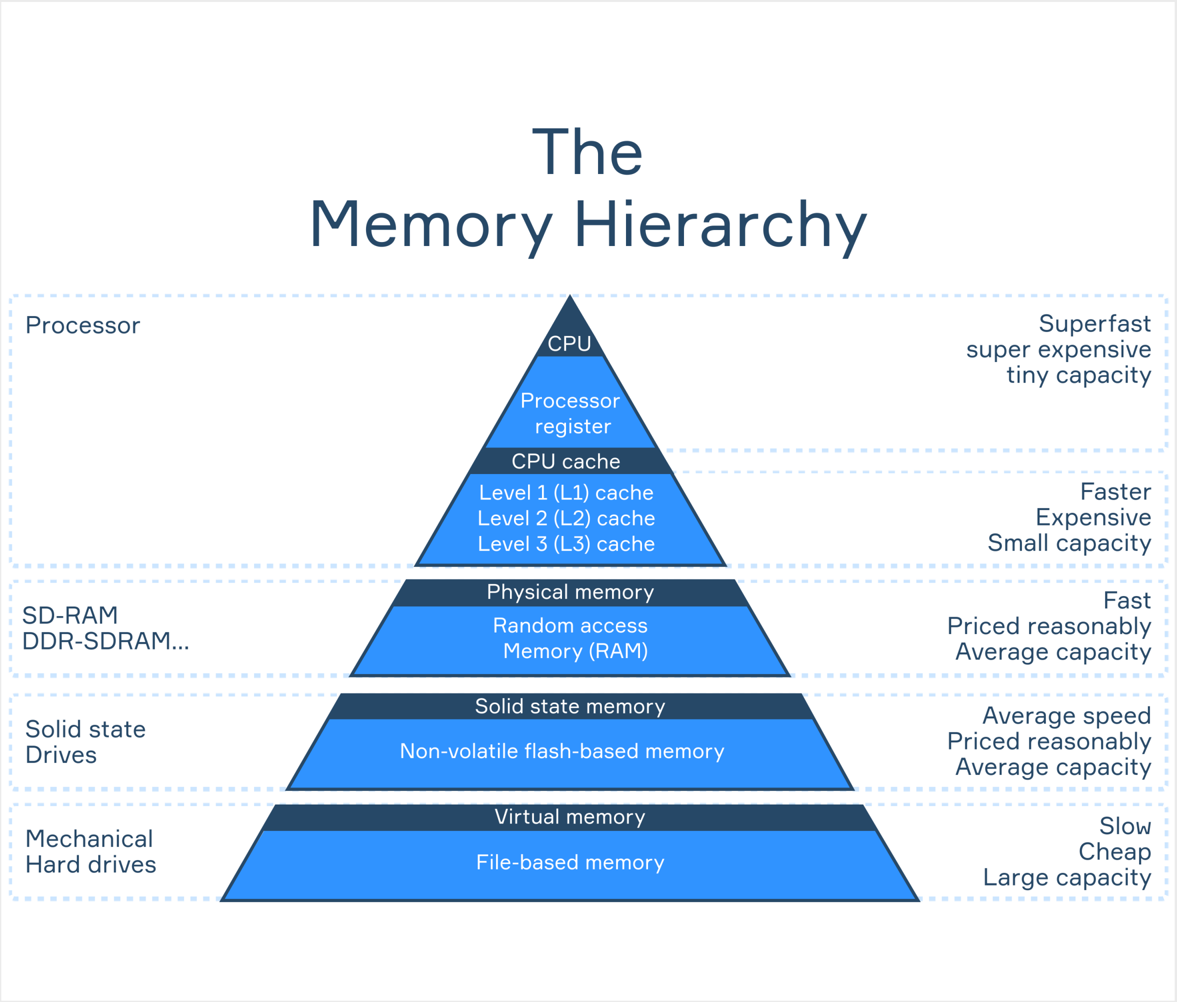
Cost per bit: Different types of computer memory use different technologies and components. Because of this, some are expensive while others are cheap. Cheap memory solutions allow us to use more memory and help increase affordability. Computers have become a basic necessity in our lives so the cost per bit of computer memory is a very important factor for easier accessibility.

Memory hierarchy

However, the storage drive we use in our computer isn't fast enough to keep up with the speed at which a processor processes data. Because of this, the processor has to spend more time waiting to read and write data from and into the memory.

This issue could be solved by using only fast storage solutions but this is not possible due to the cost implications. To solve this problem we use both fast and slow memory in a computer system and efficiently store data across all of them based on the Principle of Locality.

Memory hierarchy is the hierarchical division of different types of computer memory based on their speed or access latency. Fast memory is placed closer to the processor whereas slower memory is placed farther and farther away. This hierarchy shows a relative trend between access time, storage capacity, and cost per byte.



Memory components higher in the hierarchy are placed closer to the processor compared to those placed lower in the hierarchy. This is because placing memory components closer to the processor means the time for data to travel from one place to another is reduced. When considering the fact that modern processors have the ability to compute over a trillion instructions per second; the distance between memory components and the processor is highly relevant.

Principle of Locality

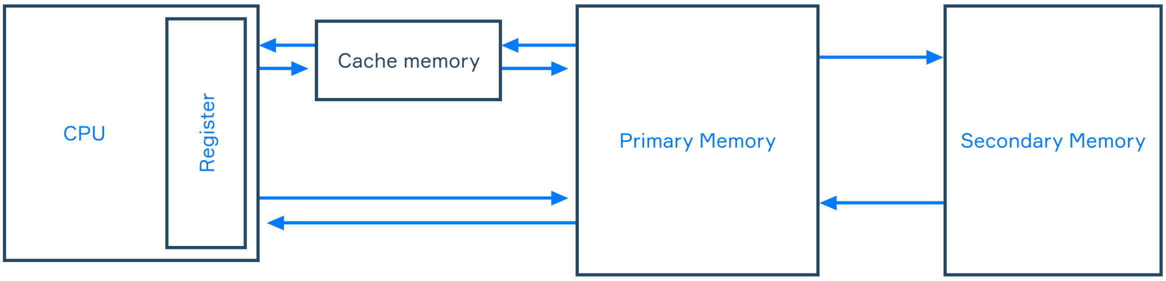
Also referred to as **Locality of reference**, is the tendency of a processor to access the same portion of the address space at a particular instance of time. A common example is the *90/10 rule* which states that a processor approximately spends 90% of its time in 10% of the code. Meaning some code is executed much more often than the other.

Spatial locality: The tendency of referencing memory locations close to the location that was referenced before. An easily understandable example of spatial locality would be arrays. When one item in an array is referenced, it is likely that another one will be referenced in quick succession.

This is a predictable behavior in computer systems that can be exploited to create the illusion of faster memory. Using the principle of locality we can predict what data is more likely to be used and subsequently store them in memory that is faster.

How is computer memory managed?

The memory hierarchy helps us create the illusion of fast memory by using a combination of both fast and slow memory components. Based on the principle of locality, proper memory management can be used as a solution to the cost implications of using only fast memory components.



The CPU works directly with the data inside registers since they are the fastest form of memory. Operations are performed on this data or with the help of it. Registers are unable to store all the necessary information at once. The computer stores all necessary data in its main memory, but it is rather slow and unable to keep up with the demands of the CPU. So, the cache acts as a bridge between fast registers and slow main memory. Cache memory is larger and slower than registers but smaller and faster than main memory and is used to store frequently used data.

When the CPU works on a process, it looks for the data in the memory component that is at the top of the hierarchy. In case the CPU doesn't find the required data, it looks for data in the components that are lower in the hierarchy. When the required data is found, it is duplicated in the components higher in the hierarchy

Our CPU first looks whether the data is present within its registers. If not it looks for them in the cache memory. If not found, the CPU will search for the necessary data in the main memory and lastly in the computer's secondary memory if it is not present in the main memory.

Since there is a lot of back and forth transfer of data between different memory levels, it is very important for memory to be managed properly so as to be efficient. This is done by loading commonly used data and information higher up in the hierarchy, which decreases the average time taken to access data from slower memory.