

Difference Between Buffering and Caching in OS



Most of the people get confused with the terms buffering and caching. Though both holds the data temporarily but, they are different from each other. **Buffering** is basically used to match the transmission speed between sender and receiver.

On the other hands, **Cache** fastens the access speed of the repeatedly used data. They also share some other differences which have been discussed in the comparison chart below.

Content: Buffering Vs Caching

1. [Comparison Chart](#)
2. [Definition](#)
3. [Key Differences](#)
4. [Conclusion](#)

TOP 10 DIFFERENCES

[Difference Between OLTP and OLAP](#)

[Difference Between while and do-while Loop](#)

[Difference Between Guided and Unguided Media](#)

[Difference Between Preemptive and Non-Preemptive Scheduling in OS](#)

[Difference Between LAN, MAN and WAN](#)

[Difference Between if-else and switch](#)

[Difference Between dispose\(\) and finalize\(\) in C#](#)

[Difference Between for and while loop](#)

[Difference Between View and Materialized View](#)

[Difference Between Server-side Scripting and Client-side Scripting](#)

Comparison Chart

BASIS FOR COMPARISON	BUFFERING	CACHING
Basic	Buffering matches the speed between the sender and receiver of the data stream.	Caching fastens the access speed of the repeatedly used data.
Stores	Buffer stores the original copy of data.	Cache stores the copy of original data.
Location	Buffer is an area in primary memory (RAM).	Cache is implemented on the processor it can be implemented on RAM and disk as well.

Definition of Buffering

Buffering is an area in main memory (RAM) that temporarily stores the data when it is being transferred between two devices, or between a device and application. Buffering helps in **matching the speed between the sender and receiver** of the data stream.

If the sender's transmission speed is slower than receiver, then a buffer is created in main memory of the receiver, and it accumulates the bytes received from the sender. When all the bytes of the data has arrived then it provides data for the receiver to operate on.

Buffering also helps **when sender and receiver have different data transfer size**. In computer networking, buffers are used for **fragmentation** and **reassembly** of data. At the sender side, the large data is fragmented into small packets and are

RECENT ADDITION

[Difference Between Blog Post and Article](#)

[Difference Between HashMap and HashSet in Java](#)

[Difference Between VPN and VPS](#)

[Differences Between Open Loop and Closed Loop Control Systems](#)

[Differences Between Private and Public Key](#)

CATEGORIES

[Artificial Intelligence](#)

[DBMS](#)

[Hardware](#)

[Internet](#)

send over the network. At the receiver side, a buffer is created which collect all the data packets and reassemble them to create a large data again.

Buffering also supports **copy semantics for an application I/O**. Copy Semantics can be explained with an example, Suppose an application has a buffer of data to be written to the hard disk. For that, the application calls the write() system call. Now suppose application changes the buffer data before system call returns. In this case, copy semantics provide the version of data, at the time of system call.

Buffers are implemented in three capacities.

- **Zero Capacity:** Here the maximum buffer memory size is Zero. It can not contain any data, so the sender must be blocked until the receiver receives the data.
- **Bounded Capacity:** Here the buffer memory size is finite. At max, the sender can send n block of data. If the buffer memory is full, the sender is blocked till space is available in memory.
- **Unbounded Capacity:** here the buffer memory is potentially infinite. Any number of data blocks can be sent. The sender is never blocked.

Definition of Caching

Cache is a memory implemented in the processor that **stores the copy of original data**. The idea behind caching is that the recently accessed disk blocks must be stored in the cache memory so that when the user again needs to access the same disk blocks, it can be handled locally through cache memory avoiding the network traffic.

Cache size is bounded as it only contains the recently used data. When you modify the cache file, you can view that modification in the original file also. In case the data you require is not in the cache memory, then data is copied from source to the cached memory to make it available to the user when it requests for that data next time.

The cache data can also be kept on disk instead of RAM, as it has one advantage that the **disk cache** are reliable. In case the system crash the cached data is still available on Disk. But data would be lost in volatile memory like RAM. But one advantage of storing the cached data in **RAM** is that it would be accessed **fast**.

Key Differences Between Buffering and Caching in OS

1. The key difference between buffer and cache is that buffer memory is used to cope up with the different speed between sender and receiver

Networking

Operating System

Programming

Software

of the data stream whereas, the cache is a memory which stores the data so that access speed can be fastened for repeatedly used data.

2. Buffer always carry the **original data** to be sent to the receiver.
However, cache carries the **copy of original data**.
3. Buffer is always implemented in the **main memory** (RAM), but, cache can be implemented in **RAM** as well as in **Disk**.

Conclusion

Buffering and Caching both stores the data temporarily but both are used for different purpose. Where buffer matches the speed between two communicating devices and the cache fastens the access to data that is repeated visited.

Related Differences:

1. [Difference Between RAM and ROM Memory](#)
2. [Difference Between Cache and Cookies](#)
3. [Difference Between Copy Constructor and Assignment Operator in C++](#)
4. [Difference Between Virtual and Cache Memory in OS](#)
5. [Difference Between Spooling and Buffering in OS](#)

Leave a Reply

Your email address will not be published. Required fields are marked *

Comment *

Name *

Email *

Website

POST COMMENT

[Copyright © 2023](#) · [Tech Differences](#) · [Contact Us](#) · [About Us](#) · [Privacy](#)