ionos.com

Memcached: the in-memory store in simple terms

8-10 minutes

Memcached is often used to improve the performance of dynamic applications with database connections. This in-memory datastore ensures, for example, that access to the hard drive does not need to retrieve data from the memory. Not only does this relieve the load on back-end systems, but it also significantly reduces latency. Popular websites such as YouTube, Facebook, Twitter, and Wikipedia have been reaping the benefits of this **open-source solution** for a long time now, allowing internet users to enjoy an improved browsing experience. So, how does Memcached work, and how do you get started using it? Here, you will learn everything you need to know about this popular caching system.

Contents

What is Memcached?

Memcached is a high-performance **caching system** that wasdeveloped by Danga Interactive nearly twenty years ago for the internet portal LiveJournal. The cache server was developed with the intent of getting around time-consuming database accesses when using demanding web applications. The solution was to use an in-memory datastore that could provide website users with cached elements quickly. Memcached is very user-friendly and easy to install, set up and use. It is also under the BSD license, so it may be freely used, modified, and copied.

What does Memcached's architecture look like?

The general structure of Memcached's architecture is relatively simple. It is similar to a distributed database system and consists of the application, a client library, and a pool of Memcached instances. Any number of these instances can be installed on a server's main memory. We recommend activating one instance on each server that has memory to spare. These instances work together to acquire the free space that is available for the cache. The client library is the interface between the respective application and Memcached. It takes the data to be stored and places it on an existing server. Through its **multi-thread architecture**, Memcached can also use several process kernels simultaneously.

Note

In addition to Memcached, Redis is also a popular in-memory datastore. While Memcached stands out for its simplicity, Redis brings a wide range of applications and functions to the table. If you want to figure out which software best meets your needs, you can read our article directly comparing Memcached vs. Redis.

How does Memcached work?

Conventional databases store data on a hard drive or preferably a solid-state drive (SSD). With Memcached, on the other hand, data is stored in memory so that it can be retrieved in **microseconds**, eliminating the delay caused by seek times when retrieving data. While it is possible to store data for a prolonged period, most data are automatically deleted after a certain amount of time. This is because Memcached is **just a cache** and not a database in the traditional sense. The least requested data is thus deleted as soon as there is no more space for new data. So, how exactly does the storage process work?

Memcached is also described as an **in-memory key-value store**. Using the TCP and IP protocols, a connection is first established with the server. If a user wants to retrieve specific data, Memcached will check whether it is available in the cache. If it is not, the required data will be retrieved from the main memory. The client then provides a key value for the data in question which is generated by the software library. Using a **hashing algorithm**, the client then determines which Memcached server the data, which is in the form of character strings, is to be stored on. Below are the five most important characteristics of Memcached:

- Data is only sent to one server.
- Data is stored as key-value pairs.
- The various servers do not share data with each other.
- The server only stores data in the memory.
- If there is not enough space, the server will delete the oldest data.

What are the advantages and disadvantages of Memcached?

Determining whether Memcached is the best caching solution for your needs depends on the **requirements** and the **complexity of the application in question**. Its temporary memory system can be useful, for example, when it comes to high-traffic web applications and websites accessing huge databases. Naturally, there are also a few disadvantages that come along with its many advantages. The following is a brief overview:

Advantages	Disadvantages
Extremely fast response times thanks to the in-memory key-value store	Only stores data temporarily and loses that data if a Memcached instance fails
Multi-thread architecture enables vertical scaling of computing capacity	Data cannot be viewed which makes debugging difficult
Sophisticated open-source solution with a publicly available datastore	Key length for values is limited to 250 characters (1 MB)
Easy to use and flexible in terms of application development	Lack of security features necessitates additional firewalls
Supports open data formats and most common clients and programming languages	Not redundant (i.e., no failover using mirroring or data backups)

What is Memcached used for?

Memcached’s uses include **caching** and **storing session data**. Caching makes cached data such as files, metadata, and images available at lightning speed. This allows you to scale in a cost-effective manner when your workload increases and improves the performance of your application. Developers also use in-memory stores to manage session data such as user profiles or online session states. However, this requires that data persistence not be essential. Generally, Memcached is a good solution for applications that require a high-performance large-scale cache.

Getting started with Memcached

Memcached's greatest strengths are undoubtedly its speed, scalability, and support of all common APIs and programming languages, including Ruby, Java, JavaScript, Python, Go, PHP, C, C++, C# and Node.js. It is also simple to install on a Windows or Unix operating system. The Memcached daemon is provided by Danga Interactive as a free download. In addition to the relevant developer package, you also need a **libevent library** that provides asynchronous event notifications. After installation, there will be different configuration options available for the client and server depending on the programming language used.

Since Memcached is an open-source project, it has been used and co-developed by a **large community** for many years. This is an advantage in that detailed information on using, managing, and troubleshooting various APIs and operating systems is available to you online.

Related articles

Varnish Cache: a performance boost for dynamic