

Memcached: Instalación y uso para mejorar la carga de tu web



Cuando un portal web tiene éxito y el número de visitas aumenta considerablemente, tarde o temprano nos podemos encontrar que esto baje el rendimiento del servidor, incluso llegándose a colapsar, sin poder dar respuesta a todas las peticiones realizadas por los usuarios. Para dar solución a este problema, es recomendable, además de optar por una estructura de máquinas más potente como puede ser el uso del **Cloud**, **utilizar herramientas de cacheo que permitan reducir los recursos utilizados** por las distintas peticiones de información realizadas por los visitantes.

Mediante el uso de estas herramientas de cacheo logramos reducir ese número de consultas y obtener el valor almacenado en la cache, en aquellos sistemas donde se realice la misma consulta a la base de datos varias veces. Por ejemplo, imagínense una tienda de consumibles electrónicos en el que en un breve periodo de tiempo 500 usuarios realizan la misma consulta sobre un determinado producto. ¿Tiene sentido realizar 500 consultas idénticas a la base de datos? Parece que no, ya que podemos llegar a colapsar el **servidor**.

Los sistemas de cache lo que hacen es **almacenar el resultado de la primera consulta realizada en memoria**, de esta forma cuando alguien solicite esos datos, en vez de ejecutar la consulta en la base de datos acudirán a la memoria, que siempre será más rápido que acceder a la información en la base de datos. Si está ahí la muestra, y si no lo estuviera la trae por medio de la consulta a la tabla correspondiente, mostrándole la información al usuario y almacenándolo en memoria para futuros usos.

Son varias las herramientas de cache que podemos utilizar para este fin, pero hoy nos centraremos en la explicación de cómo funciona Memcached.

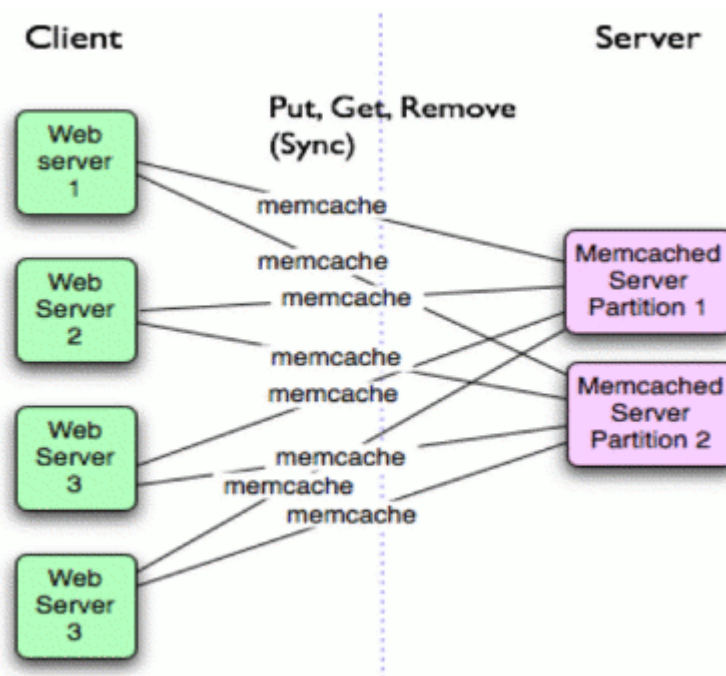
Qué es Memcached

Memcached es una herramienta desarrollada por la empresa Danga Interactive, que mantiene el proyecto bajo licencia BSD y que fue pensado para **incrementar la velocidad de aplicaciones web dinámicas**, reduciendo la carga de la base de datos.

El funcionamiento de este sistema es muy sencillo. Memcached hace uso de una **tabla hash** donde va almacenando los valores asociados a una clave única para cada elemento. Por medio del uso de instrucciones sencillas, se puede almacenar y recuperar información almacenada en la memoria de forma muy rápida, haciendo uso de la clave.

Cabe decir que hasta ahora hemos estado hablando de que Memcached permite almacenar los resultados devueltos de una consulta a la base de datos, pero además de estos datos, se puede almacenar cualquier información que se nos ocurra, por ejemplo resultados de cálculo, información de sesión de los usuarios...

Funcionamiento de Memcached



Como hemos comentado anteriormente, el uso de Memcached reduce el número de consultas en la base de datos. Cuando uno hace uso de este sistema para la creación de **páginas web**, lo primero que hay que hacer es consultar al servidor de Memcached si tiene almacenada la información solicitada. Si éste tiene la información la devuelve y si no debemos hacer la petición a la base de datos e introducir los datos en Memcached, para que puedan ser accedidos directamente la próxima vez que los necesitemos.

Cada dato introducido en Memcached tiene un tiempo de expiración; pasado ese tiempo, el servicio Memcached lo elimina, dejando ese espacio para el almacenamiento de otro objeto.

Para realizar estas acciones, Memcached hace uso de una serie de simples instrucciones que pasamos a detallarlas a continuación.

- **set**: es la instrucción encargada de actualizar el objeto si ya existe o bien lo agrega en caso de que no exista.
- **add**: se encarga de agregar el objeto sólo si no existe otro con la misma clave.
- **replace**: por medio de esta instrucción se actualiza el objeto sólo si existe.
- **get**: recupera el objeto que coincida con la clave indicada.
- **delete**: borra el objeto que le indicamos por medio de una clave.

Ventajas de usar Memcached

Son varias las ventajas que puede ofrecernos el uso de Memcached en el desarrollo de una web. Entre las principales ventajas que nos ofrece, podemos destacar:

- Reducir los niveles de carga de los servidores. Ésta es la primera gran ventaja del uso de este sistema, ya que gracias a él se reducen el número de consultas a la base de datos.
- Memcached permite ajustar el espacio de memoria dedicada, permitiendo indicar la cantidad de memoria destinada para el almacenamiento de información.
- Fácil comunicación entre cliente y servidor, basada en sencillas instrucciones, tal y como hemos visto en el punto anterior.
- Posibilidad de controlar el tiempo de vida de la información almacenada, indicándolo a la hora de almacenar el objeto.

Instalación de Memcached

El proceso de instalación de Memcached es sencillo. Únicamente hay que seguir unos pocos pasos y en poco tiempo lo tendremos instalado en nuestra máquina. Nosotros vamos a explicar el proceso de instalación para un servidor que funciona con sistema operativo Ubuntu, aunque la instalación bajo otra distribución **Linux** es similar a la que explicaremos.

Lo primero que haremos será actualizar el repositorio “**apt-get**” para que los paquetes que instalemos sean la última versión disponible. Para ello ejecutamos desde consola Linux la siguiente instrucción:

apt-get update

Una vez que tengamos actualizado el repositorio, empezaremos la instalación de Memcached.

apt-get install php5-memcache

El paso siguiente será instalar “**php-pear**” si no lo tuviéramos instalado ya.

apt-get install php-pear

Por último, utilizamos PECL (PHP Extension Community Library) para finalizar la instalación.

pecl install memcache

Completados los pasos anteriores, debemos editar el archivo **memcache.ini** para activar el paquete. Este archivo lo podemos encontrar en la ruta **/etc/php5/conf.d**. Para su edición podemos utilizar cualquier editor de texto.

vi /etc/php5/conf.d/memcache.ini

En el contenido de ese archivo, buscamos la línea “**extensión=memcache.so**” y le quitamos el “;” que tendrá delante, quedando algo similar a lo que os mostramos a continuación:

; uncomment the next line to enable the module

extension=memcache.so

Para finalizar el proceso, reiniciamos el servidor Apache para que los cambios surjan efecto.

/etc/init.d/apache restart

Podemos comprobar que la instalación se ha realizado correctamente mediante la función “**phpinfo()**” de PHP. Deberíamos ver algo como lo siguiente:

memcache

memcache support	enabled
Active persistent connections	0
Revision	\$Revision: 1.86 \$

Directive	Local Value	Master Value
memcache.allow_failover	1	1
memcache.chunk_size	8192	8192
memcache.default_port	11211	11211
memcache.hash_function	crc32	crc32
memcache.hash_strategy	standard	standard
memcache.max_failover_attempts	20	20

Configuración de Memcached

Una vez que hemos completado la instalación ya podemos hacer uso de él, pero si queremos, podemos modificar los parámetros de su configuración y cambiar los valores que trae por defecto.

Para ello es necesario que editemos el fichero “**memcached**” que nos encontraremos en la ruta “**/etc/sysconfig**”. Dentro de este archivo, nos encontraremos una serie de directivas que podemos configurar según necesitemos.

- **PORT**: indica el puerto por el que nos deberemos conectar al servidor Memcached. Por defecto el puerto es el 11211, pero podemos poner cualquier otro que esté libre.
- **MAXCONN**: Por medio de este parámetro, indicamos el número máximo de conexiones simultáneas que se le puede hacer al servidor Memcached. Por defecto el valor es de 1024, aunque se puede aumentar hasta las 4096 conexiones simultáneas.
- **CACHESIZE**: Aquí se le indica el tamaño de la RAM que se quiere utilizar para el almacenamiento de los datos. Por defecto el valor que tiene es de 64 megas, pero puede ser ampliado hasta 1024 megas, siempre y cuando el servidor disponga de esa memoria RAM.
- **OPTIONS**: Aquí podemos asignar una IP del servidor para que los servidores basados en apache/php/nginx puedan conectarse.

Ejemplo de uso

Para nuestro ejemplo de uso supondremos que tenemos una tabla en nuestra base de datos donde almacenamos a los clientes de nuestra **tienda online**. En esta tabla tendremos todos sus datos personales (nombre, apellido, dirección, DNI...)

Lo primero de todo será crearnos un objeto de la clase Memcached. Para ello será necesario indicarle la dirección IP o nombre del servidor donde está instalada y el número del puerto donde nos podemos conectar. En nuestro caso, supondremos que está instalado en el mismo servidor con el puerto por defecto, por lo que el nombre del servidor utilizaremos “localhost” y el puerto “11211”.

```
$memcache = new Memcache();
```

```
$memcache->connect('localhost',11211);
```

Ahora supongamos que queremos recuperar los datos del cliente cuyo DNI es el “00000000T”. Al ser el DNI único para cada persona, es la clave que hemos utilizado para guardar los datos en nuestro sistema de cache.

Lo primero que debemos hacer es solicitar esta información al servidor Memcached.

```
$cliente = $memcache->get("00000000T");
```

Si el cliente estaba en memoria, los datos del cliente los tendremos en la variable **\$cliente**. Si por el contrario no estuviera, en **\$cliente** tendríamos un valor “false”, y por tanto tendríamos que acudir a la base de datos para traernos la información.

```

if ( $cliente === false) {
    .....
}

```

Como hemos dicho, comprobamos si **\$cliente** tiene el valor **false**, ya que de ser así, tendremos que acudir a la base de datos.

A continuación lo que tendremos que hacer es realizar la conexión a nuestra base de datos para traernos la información.

```

mysql_connect("localhost", "usuario", "password") or die(mysql_error());

mysql_select_db("base_datos") or die(mysql_error());

```

Ejecutamos la sentencia que nos devolverá la información del usuario almacenada en nuestra base de datos.

```

$sql = "SELECT * FROM clientes WHERE dni = '00000000T'";

$cliente = mysql_fetch_array($sql);

```

Una vez que tengamos el cliente recuperado, lo almacenamos en el servidor Memcached. En nuestro ejemplo le indicaremos que permanezca almacenado durante 1 hora, lo que equivale a 3.600 segundos que es la forma de indicárselo al servidor.

```

$memcache->set("00000000T",$cliente,0,3600);

```

El código de ejemplo, todo junto quedaría de la siguiente forma.

```

$memcache = new Memcache();

$memcache->connect('localhost',11211);

$cliente = $memcache->get("00000000T");

if ( $cliente === false) {

    mysql_connect("localhost", "usuario", "password") or die(mysql_error());

    mysql_select_db("base_datos") or die(mysql_error());

    $sql = "SELECT * FROM clientes WHERE dni = '00000000T'";

    $cliente = mysql_fetch_array($sql);

    $memcache->set("00000000T",$cliente,0,3600);

}

```

Memcached es una herramienta muy útil que hará que nuestras aplicaciones ofrezcan tiempos de respuestas mejores.