

Sistemas de Caché

Para mejorar la velocidad de carga de una web



Introducción

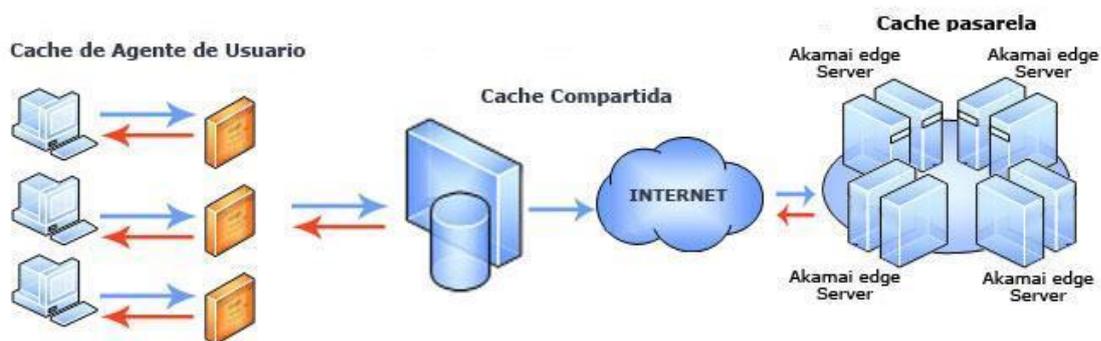
Llega el momento en la vida de un sitio web que debido al número de visitas, o por la complejidad de las consultas que se realizan a su base de datos, es necesario de contar con algún sistema de caché que permita mostrar la información solicitada de una forma más rápida.

De manera general se podría definir la caché como el proceso en el que un conjunto de datos son duplicados con el objetivo de reducir el tiempo de acceso a la información original, optimizando de esta forma el rendimiento del sistema.

Si extrapolamos el concepto de caché al terreno web, podemos hablar que se trata de un proceso en que se realiza el almacenamiento de algún documento web (página HTML, imágenes, librería JavaScript, consulta a base de datos...), con el objetivo de reducir el ancho de banda que es consumido por los visitantes y la carga de los servidores que atienden las peticiones.

Si a la hora del desarrollo de un proyecto intuimos que éste tendrá un tráfico elevado, es fundamental que establezcamos una política de cacheo que nos evite tener problemas a la hora de atender las peticiones que realizarán los visitantes del portal.

Durante el proceso de solicitud de información, existen tres tipos de caché que pueden o no actuar en ese proceso:



- **Caché de agente de usuario:** Es una caché que se da en todos los navegadores web y que funciona para cada usuario.
- **Caché compartida:** Se trata de una caché que es compartida por todos los usuarios que accedan. Este tipo de caché es utilizada por los proveedores de Internet para ahorrar ancho de banda.
- **Caché pasarela:** Es un sistema de cacheo de la información a nivel de servidor que se realiza de forma transparente al usuario.

Mecanismos para la gestión de la caché

En la actualidad nos podemos encontrar con tres mecanismos distintos para la gestión de la caché del contenido web.

1.- Mecanismo por Validación

En este tipo de caché el servidor comprueba si la respuesta que tiene almacenada el navegador no ha sufrido variación desde la última vez que se visitó. Si es así, se mostrará la información que mantiene el navegador. En este mecanismo nos podemos encontrar dos tipos distintos:

a) Last-Modified

La primera vez que un servidor devuelve una página a un navegador, éste le adjunta también la fecha de modificación del fichero. La próxima vez que el navegador solicite esa página, el servidor le enviará un mensaje de no modificación en caso de que la página no se haya modificado desde su último acceso, por lo que el navegador mostrará al usuario la página que tiene almacenada en su caché.

b) Entity Tag (ETag)

ETag es un identificador único que identifica a cada fichero cada vez que se crea o se modifica. En este caso, en vez de comprobar la fecha de modificación, lo que se comprueba es que coincida el valor almacenado con el del fichero del servidor.

2.- Mecanismos por Frescura

Este tipo de mecanismos hace que la información solicitada al servidor lleven asociadas una fecha de caducidad, pudiendo ser utilizada sin necesidad de tener que consultar al servidor si ha sido modificado. En este caso también nos podemos encontrar dos variantes de este sistema:

a) Expires

En esta versión se le asigna una fecha de caducidad al fichero, por lo que el navegador no necesitará solicitar al servidor de nuevo la información mientras no se haya superado la fecha de expiración indicada.

b) Max-age

El funcionamiento de este sistema es similar al anterior pero en este caso se le asigna una fecha de caducidad relativa, es decir, se le indica el tiempo que debe transcurrir para que esa información ya no sea válida. La información del tiempo tiene que ser en segundos.

3.- Mecanismo por invalidación

Es un mecanismo que se deriva de otra petición que se pasa por la caché. Por ejemplo, si una dirección cacheada es solicitada más tarde por medio de una petición POST, PUT o DELETE, la respuesta que estuviera cacheada quedaría invalidada. Esto es lo que suele ocurrir cuando intentamos navegar hacia atrás desde el navegador una vez enviados los datos de un formulario.

-
- **Mayor seguridad.** Los centros de datos de acens cuentan con los últimos mecanismos en seguridad, para proporcionar la máxima protección de los datos de nuestros clientes.
 - **Rapidez al disponer de tu servidor virtual.** Como hemos comentado anteriormente, sólo son necesarios tres clics para poder tener en marcha nuestra máquina. Si en un futuro necesitáramos más recursos, lo podríamos ampliar en cualquier momento y en unos pocos minutos.
 - **Control de gastos.** Algo a tener muy en cuenta viendo la actual situación económica. Con Instant Servers sólo pagarás por los recursos que consumas en cada momento.
 - **Mayor eficiencia.** El cliente se desentiende completamente del mantenimiento de los servidores, pudiendo dedicar ese tiempo a potenciar su negocio.
 - **Mayor rendimiento.** Se ha comprobado que este tipo de servidores virtuales ofrece entre un 20% y un 30% de mayor rendimiento, gracias al uso de la tecnología Joyent, y todo ello a pesar de usar menos máquinas.
 - **Alta disponibilidad.** Sabemos que para nuestros clientes disponer en todo momento de su negocio online es fundamental, por eso Instant Servers garantiza una disponibilidad del 99,9% trimestral por máquina.
 - **Redimensionamiento en 'caliente'.** Una de las principales ventajas de Instant Servers es que no se necesita reiniciar las máquinas virtuales, lo cual es, ya que posibilita un mejor control de gastos, adecuando éstos a la demanda en cada momento. Esto es posible gracias al uso del sistema operativo SmartOS, basado en Illumos y las tecnologías de virtualizaciones de zonas tradicional de Solaris.

Caché en la programación

Dependiendo del tipo de datos que queremos cachear, podemos encontrar varias técnicas que podemos utilizar centrándonos en la programación que utilizemos en el desarrollo.

- **Caché SQL:** Nos permite almacenar en memoria el resultado de consultas recurrentes a la base de datos, y de esta forma reducir el consumo de recursos por la ejecución de las mismas consultas.
- **Caché de objetos:** En este tipo de caché se pueden almacenar arrays, textos, resultados de funciones, respuestas a consultas de bases de datos... indicándole el tiempo de expiración. Ejemplos de este tipo de cacheo es APC o Memcached.
- **Caché de templates:** Es un sistema que permite cachear los resultados HTML al invocar diferentes URL. Por poner un ejemplo, el sistema de plantillas Smarty para PHP integra este tipo de cacheo para devolver resultados más eficientes.

Haciendo uso de una buena planificación de caché podremos conseguir resultados espectaculares, que harán que la visita de los internautas a nuestro sitio sea mucho más agradable, al mostrar el contenido de una forma mucho más rápida.