

Introducción

Son muchas las aplicaciones web que utilizan algún tipo de bases de datos para funcionar. Hasta ahora estábamos acostumbrados a utilizar bases de datos SQL como son MySQL, Oracle o MS SQL, pero desde hace ya algún tiempo han aparecido otras que reciben el nombre de NoSQL (Not only SQL – No sólo SQL) y que han llegado con la intención de hacer frente a las bases relacionales utilizadas por la mayoría de los usuarios.

Qué son las bases de datos NoSQL



Se puede decir que la aparición del término NoSQL aparece con la llegada de la web 2.0 ya que hasta ese momento sólo subían contenido a la red aquellas empresas que tenían un portal, pero con la llegada de aplicaciones como [Facebook](#), Twitter o Youtube, cualquier usuario podía subir contenido, provocando así un crecimiento exponencial de los datos.

Es en este momento cuando empiezan a aparecer los primeros problemas de la gestión de toda esa información almacenada en bases de datos relacionales. En un principio, para solucionar estos problemas de accesibilidad, las empresas optaron por utilizar un mayor número de máquinas pero pronto se dieron cuenta de que esto no solucionaba el problema, además de ser una solución muy cara. La otra solución era la creación de sistemas pensados para un uso específico que con el paso del tiempo han dado lugar a soluciones robustas, apareciendo así el movimiento NoSQL.

Por lo tanto hablar de bases de datos NoSQL es hablar de estructuras que nos permiten almacenar información en aquellas situaciones en las que las bases de datos relacionales generan ciertos problemas debido principalmente a problemas de escalabilidad y rendimiento de las bases de datos relacionales donde se dan cita miles de usuarios concurrentes y con millones de consultas diarias.

Además de lo comentado anteriormente, las bases de datos NoSQL son sistemas de almacenamiento de información que no cumplen con el esquema entidad-relación. Tampoco utilizan una estructura de datos en forma de tabla donde se van almacenando los datos sino que para el almacenamiento hacen uso de otros formatos como clave-valor, mapeo de columnas o grafos (ver epígrafe 'Tipos de bases de datos NoSQL').

Ventajas de los sistemas NoSQL

Esta forma de almacenar la información ofrece ciertas ventajas sobre los modelos relacionales. Entre las ventajas más significativas podemos destacar:

- **Se ejecutan en máquinas con pocos recursos:** Estos sistemas, a diferencia de los sistemas basados en SQL, no requieren de apenas computación, por lo que se pueden montar en máquinas de un coste más reducido.
- **Escalabilidad horizontal:** Para mejorar el rendimiento de estos sistemas simplemente se consigue añadiendo más nodos, con la única operación de indicar al sistema cuáles son los nodos que están disponibles.
- **Pueden manejar gran cantidad de datos:** Esto es debido a que utiliza una estructura distribuida, en muchos casos mediante **tablas Hash**.
- **No genera cuellos de botella:** El principal problema de los sistemas SQL es que necesitan transcribir cada sentencia para poder ser ejecutada, y cada sentencia compleja requiere además de un nivel de ejecución aún más complejo, lo que constituye un punto de entrada en común, que ante muchas peticiones puede ralentizar el sistema.

Principales diferencias con las bases de datos SQL

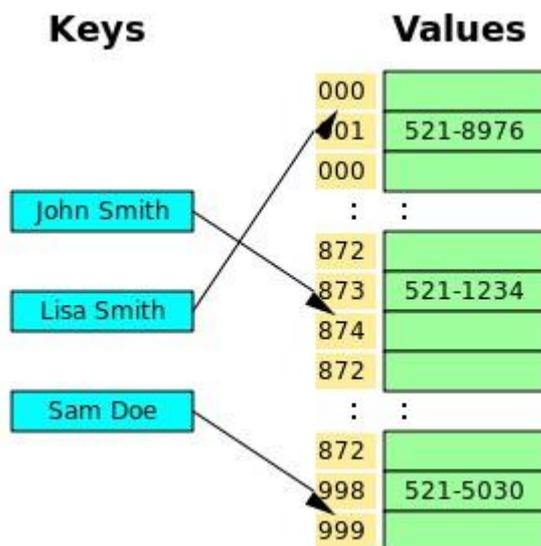
Algunas de las diferencias más destacables que nos podemos encontrar entre los sistemas NoSQL y los sistemas SQL están:

- **No utilizan SQL como lenguaje de consultas.** La mayoría de las bases de datos NoSQL evitan utilizar este tipo de lenguaje o lo utilizan como un lenguaje de apoyo. Por poner algunos ejemplos, Cassandra utiliza el lenguaje CQL, MongoDB utiliza JSON o BigTable hace uso de GQL.
- **No utilizan estructuras fijas como tablas para el almacenamiento de los datos.** Permiten hacer uso de otros tipos de modelos de almacenamiento de información como sistemas de clave-valor, objetos o grafos.
- **No suelen permitir operaciones JOIN.** Al disponer de un volumen de datos tan extremadamente grande suele resultar deseable evitar los JOIN. Esto se debe a que, cuando la operación no es la búsqueda de una clave, la sobrecarga puede llegar a ser muy costosa. Las soluciones más directas consisten en desnormalizar los datos, o bien realizar el JOIN mediante software, en la capa de aplicación.
- **Arquitectura distribuida.** Las bases de datos relacionales suelen estar centralizadas en una única máquina o bien en una estructura máster-esclavo, sin embargo en los casos NoSQL la información puede estar compartida en varias máquinas mediante mecanismos de tablas Hash distribuidas.

Tipos de bases de datos NoSQL

Dependiendo de la forma en la que almacenen la información, nos podemos encontrar varios tipos distintos de bases de datos NoSQL. Veamos los tipos más utilizados.

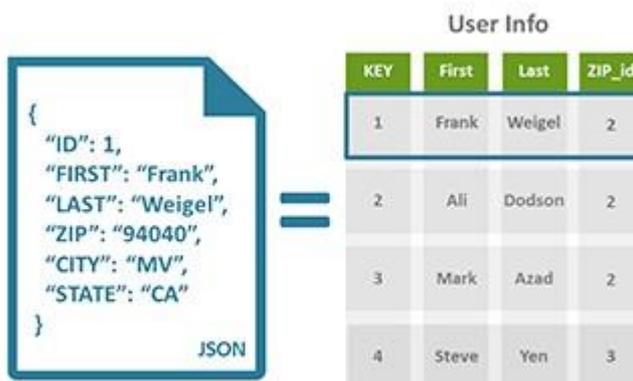
1.- Bases de datos clave – valor



Son el modelo de base de datos NoSQL más popular, además de ser la más sencilla en cuanto a funcionalidad. En este tipo de sistema, cada elemento está identificado por una llave única, lo que permite la recuperación de la información de forma muy rápida, información que habitualmente está almacenada como un objeto binario (BLOB). Se caracterizan por ser muy eficientes tanto para las lecturas como para las escrituras.

Algunos ejemplos de este tipo son Cassandra, BigTable o HBase.

2.- Bases de datos documentales

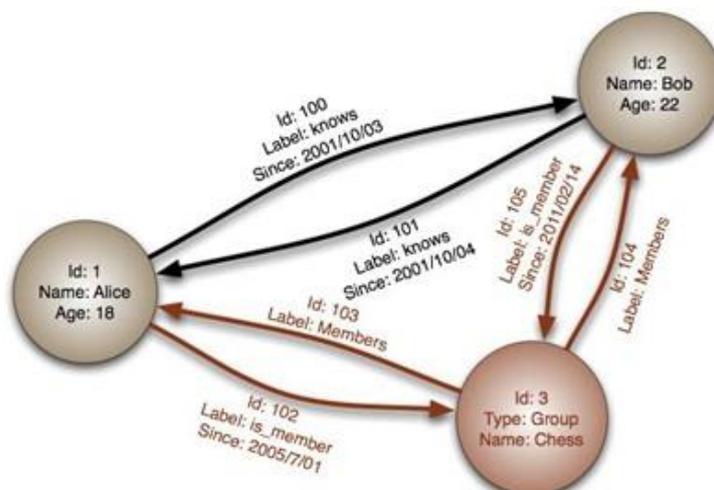


Este tipo almacena la información como un documento, generalmente utilizando para ello una estructura simple como JSON o XML y donde se utiliza una clave única para cada registro. Este tipo de implementación permite, además de realizar búsquedas por clave–valor, realizar consultas más avanzadas sobre el contenido del documento.

Son las bases de datos NoSQL más versátiles. Se pueden utilizar en gran cantidad de proyectos, incluyendo muchos que tradicionalmente funcionarían sobre bases de datos relacionales.

Algunos ejemplos de este tipo son MongoDB o CouchDB.

3.- Bases de datos en grafo



En este tipo de bases de datos, la información se representa como nodos de un grafo y sus relaciones con las aristas del mismo, de manera que se puede hacer uso de la teoría de grafos para recorrerla. Para sacar el máximo rendimiento a este tipo de bases de datos, su estructura debe estar totalmente normalizada, de forma que cada tabla tenga una sola columna y cada relación dos.

Este tipo de bases de datos ofrece una navegación más eficiente entre relaciones que en un modelo relacional.

Algunos ejemplos de este tipo son Neo4j, InfoGrid o Virtuoso.

4.- Bases de datos orientadas a objetos

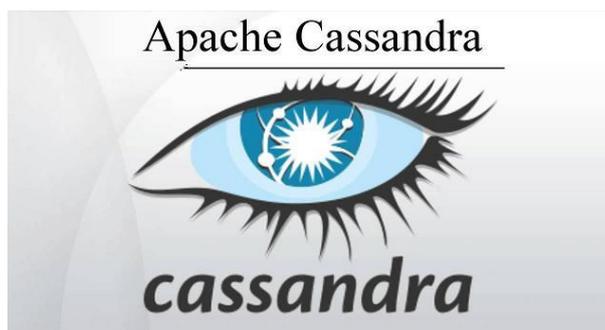
En este tipo, la información se representa mediante objetos, de la misma forma que son representados en los lenguajes de programación orientada a objetos (POO) como ocurre en JAVA, C# o Visual Basic .NET.

Algunos ejemplos de este tipo de bases de datos son Zope, Gemstone o Db4o.

Ejemplos de bases de datos NoSQL

Veamos a continuación algunos tipos de bases NoSQL más utilizadas actualmente.

1.- Cassandra



Se trata de una base de datos creada por Apache del tipo clave–valor.

Dispone de un lenguaje propio para realizar consultas CQL (Cassandra Query Language). Cassandra es una aplicación Java por lo que puede correr en cualquier plataforma que cuente con la JVM.

2.- Redis



Se trata de una base de datos creada por Salvatore Sanfilippo y Pieter Noordhuis y está apoyado por VMWare. Se trata de una base de datos del tipo clave–valor. Se puede imaginar como un array gigante en memoria para almacenar datos, datos que pueden ser cadenas, hashes, conjuntos de datos o listas.

Tiene la ventaja de que sus operaciones son atómicas y persistentes. Por ponerle una pega, Redis no permite realizar consultas, sólo se puede insertar y obtener datos, además de las operaciones comunes sobre conjuntos (diferencia, unión e inserción).

Creado en ANSI C, por lo tanto es compatible y funciona sin problemas en sistemas Unix, Linux y sus derivados, Solaris, OS/X sin embargo no existe soporte oficial para plataformas Windows.

3.- MongoDB



Se trata de una base de datos creada por 10gen del tipo orientada a documentos, de esquema libre, es decir, que cada entrada puede tener un esquema de datos diferente que nada tenga que ver con el resto de registros almacenados. Es bastante rápido a la hora de ejecutar sus operaciones ya que está escrito en lenguaje C++.

Para el almacenamiento de la información, utiliza un sistema propio de documento conocido con el nombre BSON, que es una evolución del conocido JSON pero con la peculiaridad de que puede almacenar datos binarios.

En poco tiempo, MongoDB se ha convertido en una de las bases de datos NoSQL favoritas por los desarrolladores.

4.- CouchDB



Se trata de un sistema creado por Apache y escrito en lenguaje Erlang que funciona en la mayoría de sistemas POSIX, incluyendo GNU/LINUX y OSX, pero no así en sistemas Windows.

Como características más importantes cabe destacar el uso de Restfull HTTP API como interfaz y JavaScript como principal lenguaje de interacción. Para el almacenamiento de los datos se utiliza archivos JSON. Permite la creación de vistas, que son el mecanismo que permite la combinación de documentos para retornar valores de varios documentos, es decir, CouchDB permite la realización de las operaciones JOIN típicas de SQL.

SQL o NoSQL that's the question

Algunas de las razones que nos pueden llevar a decantarnos por el uso de las bases de datos NoSQL en lugar de las clásicas SQL son:

- Cuando el volumen de los datos crece muy rápidamente en momentos puntuales, pudiendo llegar a superar el **Terabyte** de información.
- Cuando la escalabilidad de la solución relacional no es viable tanto a nivel de costes como a nivel técnico.
- Cuando tenemos elevados picos de uso del sistema por parte de los usuarios en múltiples ocasiones.
- Cuando el esquema de la base de datos no es homogéneo, es decir, cuando en cada inserción de datos la información que se almacena puede tener campos distintos.

Grandes compañías que utilizan este tipo de bases de datos

Son muchas las grandes empresas que hacen uso de este tipo de bases de datos no relacionales, como:

- **Cassandra:** Facebook, [Twitter](#)...
- **HBase:** Yahoo, Adobe...
- **Redis:** Flickr, Instagram, Github...
- **Neo4j:** Infojobs...
- **MongoDB:** FourSquare, SourceForge, CERN...