

Universidad Mariano Gálvez de Guatemala

Ingeniería en sistemas

Sede Jalapa

Marco Tulio Valde Albizurez

Introducción a los Sistemas

Investigación

Hugo Enrique Vásquez Ortiz

0907-24-19574

Introducción

Las bases de datos no relacionales están diseñadas para ejecutarse sobre clústeres, es decir que pueden tener muchas maquinas pequeñas (bajo costo) ejecutando instancias de estas bases de datos.

Bases de datos no relacionales.

Una base de datos no relacional se diferencia de las bases de datos relacionales en cuanto a su forma de funcionar y su estructura. Una base de datos no relacional es un conjunto de datos que están almacenados en una forma no tabular. Por lo tanto, pueden establecerse en lugares tales como un documento, una tabla o un vídeo, mientras que las relacionales comparten el mismo formato.

Cualquiera de estos lugares, a su vez, **contienen distintos tipos de formatos o de información.**

Es por esta razón que se considera que una base de datos no relacional es mucho más flexible, pues permite que se organicen y comuniquen diferentes tipos de información en diferentes formatos.

Ventajas y Desventajas

Ventajas:

- Trabajar con bases de datos no relacionales ofrece muchas ventajas, entre las cuales destacan las siguientes:
- Su flexibilidad para elaborar esquemas de información, algo que permite almacenar y gestionar datos que son semiestructurados o no estructurados.
- Son fácilmente escalables horizontalmente y, por tanto, son utilizadas para trabajar con volúmenes grandes de información.
- Su rendimiento es alto debido a que se han preparado para trabajar con patrones y modelos de datos específicos.
- Gracias a sus API exclusivas y la proporción de modelos creados para gestionar los distintos tipos de datos se han convertido en un sistema de almacenamiento muy funcional.

Desventajas:

Aunque utilizar bases de datos no relacionales para almacenar información ofrece múltiples beneficios, también hay que tener en cuenta una serie de desventajas:

- No cumplen con las propiedades ACID como sí lo hacen las relacionales.
- Son incompatibles con las consultas realizadas en lenguaje SQL.
- Todavía no existe un sistema estandarizado.

- Al ser de código abierto, no poseen el soporte que proporciona una empresa comercial, siendo este muchas veces limitado.
- Dado que no comparten un estándar, se pueden presentar problemas de compatibilidad

MongoDB, RavenDB, CouchDB, Firebase

MongoDB:

es un sistema de base de datos NoSQL, orientado a documentos y de código abierto.

En lugar de guardar los datos en tablas, tal y como se hace en las bases de datos relacionales, MongoDB guarda estructuras de datos BSON (una especificación similar a JSON) con un esquema dinámico, haciendo que la integración de los datos en ciertas aplicaciones sea más fácil y rápida.

MongoDB es una base de datos adecuada para su uso en producción y con múltiples funcionalidades. Esta base de datos se utiliza mucho en la industria,¹ contando con implementaciones en empresas como MTV Network, Craigslist, Foursquare.

RavenDB:

es una base de datos documental de código abierto con soporte transaccional nativo escrita íntegramente en C# y desarrollada por Hibernating Rhinos Ltd.⁴⁵⁶¹⁷ Es una base de datos multiplataforma compatible con Windows, Linux y MacOS. RavenDB almacena datos como documentos JSON y soporta clústeres distribuidos con replicación maestro-maestro

CouchDB:

es un gestor de bases de datos de código abierto, cuyo foco está puesto en la facilidad de su uso y en ser "una base de datos que asume la web de manera completa".¹ Se trata de una base de datos NoSQL que emplea JSON para almacenar los datos, JavaScript como lenguaje de consulta por medio de MapReduce y HTTP como API.¹ Una de sus características más peculiares es la facilidad con la que permite hacer replicaciones. CouchDB fue liberada por primera vez en 2005, transformándose en un proyecto Apache en 2008.

A diferencia de una base de datos relacional, CouchDB no almacena los datos y sus relaciones en tablas. En cambio, cada base de datos es una colección de documentos independientes. Cada documento mantiene sus propios datos y su esquema autocontenido. Una aplicación puede acceder a múltiples bases de datos, por ejemplo, una residente en el teléfono móvil del usuario y otra residente en un servidor. Los metadatos del documento contienen información acerca de la versión del mismo, permitiendo refundir cualesquiera

diferencias que puedan haberse producido mientras las bases de datos estaban desconectadas.

CouchDB implementa una forma de control de concurrencia multiversión (Multiversion Concurrency Control) a fin de evitar la necesidad de bloquear el archivo de base de datos durante las escrituras. La resolución de conflictos se delega a la aplicación. Para resolver un conflicto generalmente se requiere primero refundir los datos en uno de los documentos para luego eliminar el otro.²

Otras características son una semántica ACID con consistencia eventual, MapReduce, replicación incremental y tolerancia a los fallos. Las labores de administración se facilitan por medio de una aplicación web incorporada, llamada Futon.

Firestore:

es una plataforma para el desarrollo de aplicaciones web y aplicaciones móviles lanzada en 2011 y adquirida por Google en 2014.

Es una plataforma ubicada en la nube, integrada con Google Cloud Platform, que usa un conjunto de herramientas para la creación y sincronización de proyectos que serán dotados de alta calidad, haciendo posible el crecimiento del número de usuarios y dando resultado también a la obtención de una mayor monetización.

Cuadro comparativo Bases de datos relacionales vs Bases de datos no relacionales.

| Base de datos relacionales | Base de datos no relacionales |
|-------------------------------------------------------------------------|-----------------------------------------------------------------------------|
| Cuando el volumen de mis datos no crece o lo hace poco. | Cuando el volumen de mis datos crece muy rápidamente en momentos puntuales. |
| Cuando las necesidades de proceso se pueden asumir en un solo servidor. | Cuando las necesidades del proceso no se pueden proveer. |

Conclusión

Las **bases de datos NoSQL** vinieron para quedarse, aparecieron como pequeños proyectos aislados para resolver problemas específicos, que eran muy complejos hacerlos con las bases de datos relacionales.

Bibliografías

<https://codigoonclick.com/nosql-databases-introduccion/>

<https://www.ui1.es/blog-ui1/bases-de-datos-no-relacionales>

<https://es.wikipedia.org/wiki/MongoDB>

<https://es.wikipedia.org/wiki/RavenDB>

<https://es.wikipedia.org/wiki/CouchDB>

<https://es.wikipedia.org/wiki/Firebase>