

Introducción

Las bases de datos no relacionales, también conocidas como NoSQL, son un tipo de sistema de gestión de bases de datos diseñado para almacenar datos de manera flexible y escalable, sin seguir el modelo tradicional de tablas y relaciones de las bases de datos relacionales. Estas bases de datos permiten almacenar datos en formatos más variados, como documentos JSON, pares clave-valor o grafos, lo que las hace ideales para manejar datos no estructurados o semi estructurados generados por aplicaciones modernas como redes sociales, dispositivos IoT y aplicaciones web.

Bases de datos no relacionales

Las bases de datos no relacionales, también conocidas como NoSQL ("Not Only SQL"), son un tipo de sistema de gestión de bases de datos (SGBD) diseñado para almacenar y administrar datos de forma flexible y escalable, sin seguir el modelo de tablas y relaciones de las bases de datos relacionales tradicionales.

Las bases de datos relacionales almacenan datos en tablas estructuradas con filas y columnas, donde cada columna tiene un tipo de dato específico. Las relaciones entre las tablas se definen mediante claves foráneas.

Las bases de datos no relacionales, por el contrario, permiten almacenar datos en formatos más flexibles, como documentos JSON, pares clave-valor o grafos. Esto las hace más adecuadas para manejar datos no estructurados o semi estructurados, como los generados por las redes sociales, los dispositivos IoT o las aplicaciones web.

Características principales de las bases de datos no relacionales:

Flexibilidad: Permiten almacenar datos en una variedad de formatos, sin necesidad de definir un esquema rígido de antemano.

Escalabilidad: Pueden manejar grandes volúmenes de datos de manera eficiente, lo que las hace ideales para aplicaciones que requieren mucha capacidad de almacenamiento.

Alto rendimiento: Ofrecen un rendimiento rápido para consultas y escrituras, incluso con grandes conjuntos de datos.

Disponibilidad: Están diseñadas para ser altamente disponibles, lo que significa que pueden funcionar sin interrupciones incluso si falla un servidor.

Ventajas y desventajas de las bases de datos no relacionales

Ventajas:

- **Flexibilidad:** Las bases de datos no relacionales permiten almacenar datos en una variedad de formatos, como documentos JSON, pares clave-valor o grafos. Esto las hace más adecuadas para manejar datos no estructurados o semi estructurados, como los generados por las redes sociales, los dispositivos IoT o las aplicaciones web.

- **Escalabilidad:** Pueden manejar grandes volúmenes de datos de manera eficiente, lo que las hace ideales para aplicaciones que requieren mucha capacidad de almacenamiento.
- **Alto rendimiento:** Ofrecen un rendimiento rápido para consultas y escrituras, incluso con grandes conjuntos de datos.
- **Disponibilidad:** Están diseñadas para ser altamente disponibles, lo que significa que pueden funcionar sin interrupciones incluso si falla un servidor.
- **Simplicidad:** Suelen ser más simples de diseñar y administrar que las bases de datos relacionales, lo que puede reducir los costos de desarrollo y mantenimiento.

Desventajas:

- **Falta de estandarización:** No existe un lenguaje de consulta estándar para las bases de datos no relacionales, lo que puede dificultar la escritura de consultas complejas y la migración de datos entre diferentes sistemas.
- **Consistencia:** Las bases de datos no relacionales no siempre ofrecen el mismo nivel de consistencia que las bases de datos relacionales. Esto puede ser un problema para aplicaciones que requieren datos altamente consistentes.
- **Seguridad:** Puede ser más difícil implementar medidas de seguridad en las bases de datos no relacionales que en las bases de datos relacionales.
- **Menos madurez:** Las bases de datos no relacionales son una tecnología relativamente nueva y es posible que no tengan el mismo nivel de madurez y soporte que las bases de datos relacionales.

MONGODB

MongoDB es una base de datos NoSQL popular, lo que significa que a diferencia de las bases de datos relacionales tradicionales que usan tablas y filas, MongoDB almacena datos en **documentos JSON flexibles**.

Almacena datos en documentos: En lugar de tablas con filas y columnas estructuradas, MongoDB utiliza documentos JSON para almacenar datos. Estos documentos pueden tener distintos formatos e incluir diversos tipos de datos, como números, texto, arrays e incluso documentos anidados. Esta flexibilidad la hace adecuada para información compleja o que cambia con el tiempo.

Escalable: MongoDB puede manejar grandes cantidades de datos de manera eficiente. Además, permite escalar horizontalmente agregando más servidores a su cluster. Esto significa que a medida que tu aplicación y la data crezcan, puedes añadir servidores para gestionar la carga.

Alto rendimiento: MongoDB es conocida por su velocidad de lectura y escritura, incluso para conjuntos de datos voluminosos. Esta característica es crucial para aplicaciones que requieren acceso a la información en tiempo real.

Licencia parcialmente open-source: El servidor principal de MongoDB está disponible bajo una licencia que te permite ver y modificar el código fuente (open-

source). Sin embargo, algunas funciones más novedosas requieren una licencia comercial.

RavenDB

RavenDB es una potente base de datos NoSQL que almacena información en documentos JSON flexibles. Ofrece varias características clave que la convierten en una opción atractiva para aplicaciones específicas:

Énfasis en consultas complejas: RavenDB destaca por su capacidad para manejar consultas de datos intrincadas. Esto la hace adecuada para escenarios donde necesitas realizar análisis profundos o recuperar información basada en relaciones complejas dentro de tus datos.

Transacciones ACID: RavenDB garantiza las propiedades ACID (Atomicidad, Consistencia, Aislamiento, Durabilidad) para las transacciones. Esto asegura la integridad y consistencia de los datos, especialmente crucial para aplicaciones que requieren actualizaciones de datos confiables.

Modelo de datos flexible: Al igual que MongoDB, RavenDB utiliza documentos JSON para almacenar datos. Esta estructura flexible te permite almacenar varios tipos de datos dentro de un mismo documento y adaptarse a las necesidades cambiantes de la información.

Escalabilidad: RavenDB ofrece escalabilidad horizontal, lo que te permite agregar más servidores a tu cluster a medida que aumenta el volumen de tus datos. Esto garantiza un rendimiento fluido incluso con conjuntos de datos grandes.

Alta disponibilidad: RavenDB prioriza la alta disponibilidad, lo que significa un tiempo de inactividad mínimo y un servicio continuo incluso si un servidor tiene problemas.

CouchDB

CouchDB es un sistema de gestión de bases de datos (DBMS) NoSQL, de código abierto y distribuido desarrollado por Apache Software Foundation. Se enfoca en proporcionar almacenamiento, recuperación y administración de datos altamente flexible y escalable.

Modelo de Datos: Utiliza un modelo de datos basado en documentos JSON (JavaScript Object Notation). Cada registro en CouchDB es un documento JSON que contiene información estructurada en pares de clave-valor.

Escalabilidad y Replicación: CouchDB está diseñado para ser altamente escalable y tolerante a fallos. Permite la replicación bidireccional de datos entre múltiples nodos, lo que mejora la disponibilidad y la resistencia a fallos del sistema.

API RESTful: Ofrece una API HTTP RESTful para realizar operaciones CRUD (Crear, Leer, Actualizar, Eliminar) en la base de datos. Esto facilita la integración con aplicaciones web y otros sistemas que utilizan protocolos HTTP estándar.

Consultas y Índices: CouchDB proporciona un motor de consultas basado en MapReduce que permite realizar consultas complejas y análisis de datos en grandes conjuntos de datos. También admite la creación de índices para mejorar el rendimiento de las consultas.

Documentación y Comunidad: CouchDB cuenta con una documentación completa y una comunidad activa de desarrolladores que contribuyen al proyecto y brindan soporte a los usuarios.

Firestore

Firestore es una plataforma de desarrollo de aplicaciones móviles y web desarrollada por Google. Ofrece una amplia gama de herramientas y servicios que facilitan la creación, el desarrollo y la administración de aplicaciones de alta calidad de manera rápida y eficiente.

Base de datos en tiempo real: Firestore ofrece una base de datos en tiempo real basada en JSON que permite a las aplicaciones mantener datos sincronizados entre usuarios y dispositivos en tiempo real. Es especialmente útil para aplicaciones colaborativas, de chat en tiempo real y aplicaciones que requieren actualizaciones en tiempo real.

Autenticación de usuarios: Proporciona servicios de autenticación de usuarios con integración con proveedores populares como Google, Facebook, Twitter y otros proveedores de identidad. Esto facilita la implementación de sistemas de inicio de sesión seguros y eficientes.

Almacenamiento en la nube: Firestore ofrece un servicio de almacenamiento en la nube que permite a las aplicaciones almacenar y servir archivos estáticos y multimedia de forma segura y escalable. Es útil para alojar imágenes, videos, archivos PDF y otros tipos de archivos en la nube.

Hosting web: Firebase Hosting permite a los desarrolladores alojar y servir aplicaciones web estáticas y dinámicas de forma rápida y segura. Proporciona integración con HTTPS, CDN (Content Delivery Network) y herramientas de desarrollo como Firebase CLI.

Funciones en la nube: Firebase Functions permite a los desarrolladores escribir y desplegar funciones en la nube que se ejecutan automáticamente en respuesta a eventos específicos. Esto facilita la implementación de lógica empresarial compleja y automatización de tareas en la nube.

Notificaciones en tiempo real: Firebase Cloud Messaging (FCM) proporciona un servicio de mensajería en la nube que permite enviar notificaciones push y mensajes a dispositivos móviles y web de manera eficiente y escalable.

Característica	Bases de datos relacionales	Bases de datos no relacionales (NoSQL)
Modelo de datos	Tabular	Esquemas flexibles (documentos, gráficos, clave-valor, etc.)
Estructura	Altamente estructurada	Estructura flexible
Relaciones entre datos	Se establecen mediante claves foráneas	Se pueden establecer de diversas formas, según el modelo de datos
Consistencia	Fuerte consistencia de datos ACID	Menor consistencia, pero mayor escalabilidad
Escalabilidad	Buena escalabilidad vertical	Alta escalabilidad horizontal
Consultas	Lenguaje SQL estructurado	Lenguajes de consulta específicos para cada modelo
Ejemplos	MySQL, PostgreSQL, Oracle	MongoDB, Cassandra, CouchDB

Formatos de datos utilizados por MongoDB, RavenDB, CouchDB y Firebase

MongoDB:

- **JSON (JavaScript Object Notation):** Es el formato de datos predeterminado en MongoDB. Es un formato de texto ligero y fácil de leer que se utiliza para almacenar datos estructurados en pares clave-valor.
- **BSON (Binary JSON):** Es una representación binaria de JSON que es más eficiente en cuanto a espacio y rendimiento.
- **Otros formatos:** MongoDB también admite otros formatos de datos como YAML, CSV y Protocol Buffers.

RavenDB:

- **JSON:** RavenDB utiliza JSON como formato de datos predeterminado al igual que MongoDB.
- **BSON:** RavenDB también admite BSON.
- **RavenQL:** RavenDB tiene su propio lenguaje de consulta llamado RavenQL que puede utilizarse para crear consultas complejas.

CouchDB:

- **JSON:** CouchDB utiliza JSON como formato de datos predeterminado.
- **CommonJS:** CouchDB también admite CommonJS, un formato de datos basado en JavaScript que se utiliza para almacenar módulos de código.
- **Attachments:** CouchDB permite almacenar archivos adjuntos binarios junto con los documentos JSON.

Firestore:

- **JSON:** Firestore utiliza JSON como formato de datos predeterminado.
- **Firestore Realtime Database:** Firestore Realtime Database es una base de datos NoSQL que almacena datos JSON en tiempo real.
- **Cloud Firestore:** Cloud Firestore es una base de datos NoSQL escalable que almacena datos JSON estructurados.

Conclusión

las bases de datos no relacionales ofrecen flexibilidad, escalabilidad y alto rendimiento para manejar grandes volúmenes de datos en entornos dinámicos y exigentes. Aunque presentan ventajas como la flexibilidad en el formato de datos y la capacidad de escalabilidad, también tienen desafíos como la falta de estandarización en los lenguajes de consulta y la necesidad de implementar medidas de seguridad específicas. Sin embargo, su creciente popularidad y la variedad de opciones disponibles, como MongoDB, RavenDB, CouchDB y servicios como Firestore, demuestran su relevancia y utilidad en el panorama actual de desarrollo de aplicaciones.

E-grafía

<https://www.acens.com/comunicacion/wp-content/images/2014/02/bbdd-nosql-wp-acens.pdf>

<https://www.certia.net/tipos-de-bases-de-datos-no-relacionales/>

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

<https://www.mongodb.com/docs/>