Práctica 15. Bases de datos no relacionales (NoSQL)

Apuntes de BD para DAW, DAM y ASIR

José Juan Sánchez Hernández

Curso 2023/2024

Índice

Índice de figuras

Índice de cuadros

1 Bases de datos NoSQL

Las bases de datos **no relacionales** o **NoSQL** nos permiten almacenar información en situaciones donde las bases de datos relaciones pueden tener problemas de escalabilidad y rendimiento. Estas bases de datos están diseñadas para modelos de datos específicos y tienen esquemas flexibles.

Ejemplo:

Por ejemplo, suponga que queremos modelar el esquema de una base de datos sencilla para almacenar libros.

Si utilizásemos una **base de datos relacional** almacenaríamos la información en diferentes tablas, que estarían relacionadas entre sí mediante restricciones de claves primarias y ajenas. El modelo relacional está diseñado para permitir que exista integridad referencial entre tablas y para reducir la redundancia de la información mediante la normalización. En este ejemplo, tendríamos las siguientes tablas:

- **libros**(isbn, título, año_edición)
- **autores**(id_autor, nombre_autor)
- **autor_isbn**(id_autor, isbn)

En una **base de datos NoSQL**, el registro de un libro se podría almacenar en un único documento JSON. Para cada libro, tendríamos los siguientes elementos: isbn, título, año_edición, id_autor y nombre_autor .

```
{
    "isbn": 9788448190330,
    "título": "Fundamentos de Bases de Datos",
    "año_edición": 2015,
    "id_autor": 11,
    "nombre_autor": "Abraham Silverschatz"
}
```

1.1 Características de las bases de datos NoSQL

- No cumplen con el esquema entidad–relación.
- No utilizan estructuras fijas de almacenamiento como las tablas.
- Estas bases de datos tienen esquemas de datos flexibles, lo que hace que sean ideales para aplicaciones que trabajan con datos semiestructurados y no estructurados.
- Normalmente no soportan operaciones de tipo JOIN.
- No garantizan que se puedan realizar transacciones, por lo tanto no cumplen las propiedades ACID (Atomicidad, Consistencia, Aislamiento y Durabilidad).

1.2 Tipos de bases de datos NoSQL

1.2.1 Bases de datos de tipo clave-valor

Las bases de datos clave-valor son altamente divisibles y permiten un escalado horizontal a escalas que otros tipos de bases de datos no pueden alcanzar.

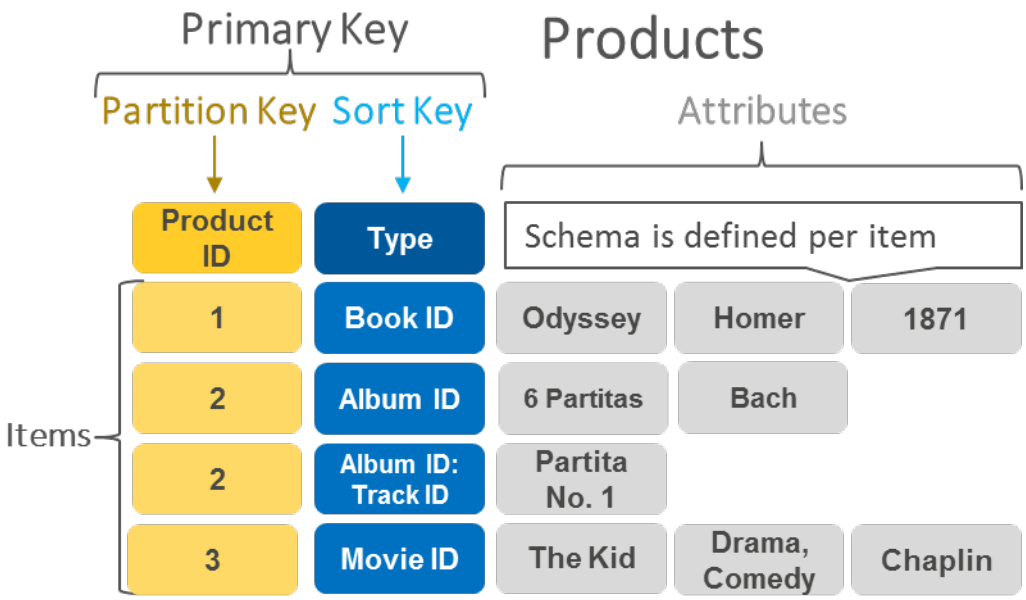

Imagen: Ejemplo de un base de datos clave-valor. Amazon Web Services.

Casos de uso

- Almacén de sesiones.
- Carrito de la compra.
- Juegos.
- IoT.

Ejemplos de bases de datos de tipo clave-valor

- Redis.
- Amazon DynamoDB.
- Riak.

Tutoriales

- https://riptutorial.com/redis
- https://riptutorial.com/Download/redis.pdf
- Quickstart: How to Use Redis on Java. 2019. DZone.
- A Docker/docker-compose setup with Redis and Node/Express. 2018.

Práctica con Redis

- Servidor:

Iniciamos un contenedor Docker con `redis` en el puerto 6379.

```
docker run --rm --name redis-server -p 6379:6379 redis
```

- Cliente:

Iniciamos un contenedor Docker con `redis-cli` que conecta con el servidor `redis` que hemos creado en el paso anterior.

```
docker run -it --rm --link redis-server:redis-server redis redis-cli -h redis-
    server -p 6379
```

Para almacenar valores utilizamos `SET`.

```
redis-server:6379> SET miclave "¡Hola mundo!"
OK
```

Para recuperar valores utilizamos `GET`.

```
redis-server:6379> GET miclave
"\xc2\xa1Hola mundo!"
```

1.2.2 Bases de datos de tipo documentos JSON

En algunas aplicaciones, los datos se representan como un objeto o un documento de tipo `JSON` porque es un modelo de datos intuitivo para los desarrolladores. Las bases de datos de `JSON` tienen una naturaleza flexible y facilitan a los desarrolladores el almacenamiento y la consulta de datos en una base de datos mediante el uso del mismo formato de modelo de documento que emplean en el código de aplicación.

Ejemplo

Ejemplo de un documento `JSON`que describe una película.

```
{
    "year" : 2019,
    "title" : "Avengers: Endgame",
    "info" : {
        "directors" : [ "Anthony Russo", "Joe Russo"],
        "release_date" : "2019-04-25T00:00:00Z",
        "rating" : 8.8,
        "genres" : ["Action", "Adventure", "Sci-Fi"],
        "image_url" : "https://your.server.com/images/avengers.jpg",
        "plot" : "After the devastating events of Vengadores: Infinity War
            (2018), the universe is in ruins. With the help of remaining allies,
             the Avengers assemble once more in order to undo Thanos' actions
            and restore order to the universe.",
        "actors" : ["Robert Downey Jr.", " Chris Evans", "Mark Ruffalo"]
    }
}
```

Casos de uso

- Administración de contenidos, como blogs y plataformas de video.
- Catálogos, como los productos de una aplicación de e-commerce.

Ejemplos de bases de datos comerciales de tipo documentos `JSON`

- MongoDB.
- Couchbase.
- Amazon DocumentDB.

Tutoriales

- MongoDB Hello World—How to connect, insert and find data. 2018. Medium.

Práctica con MongoDB

- Servidor:

Iniciamos un contenedor Docker con `mongo`.

```
docker run -d --rm --name mongo-server mongo
```

- Cliente:

Iniciamos un contenedor Docker con un cliente de `mongo` que conecta con el servidor `mongo` que hemos creado en el paso anterior.

```
docker run -it --rm --link mongo-server mongo mongo --host mongo-server
```

Una vez que hemos conectado con el servidor mongo podemos ejecutar los siguientes comandos. En primer lugar, listamos todas las bases de datos que existen en el servidor.

```
show databases
```

Para seleccionar una base de datos existente o para crear una nueva base de datos utilizamos el comando `use`. Por ejemplo, el siguiente comando creará una base de datos llamada `tienda`.

```
use tienda
```

Para listar las colecciones de documentos que existen en la base de datos utilizamos el comando `show collections`.

```
show collections
```

Como acabamos de crear la base de datos, el comando anterior no devolverá ningún resultado porque todavía no existe ninguna colección.

Vamos a crear una colección llamada `libros`.

```
db.createCollection("libros")
```

Ahora insertamos el primer documento JSON en la colección `libros`.

```
db.libros.insertOne({titulo: "Bases de datos NoSQL", autor: "Pepe"})
```

También es posible insertar una lista de documentos JSON en una única operación.

```
db.libros.insertMany([{titulo: "Libro A", autor: "María"},{titulo: "Libro B",
    autor: "Juan"}])
```

Para obtener la lista de documentos de la base de datos utilizamos el comando `find`.

```
db.libros.find()
```

Podemos formatear la salida con el comando `pretty`, para mejorar la legibilidad de los documentos JSON.

```
db.libros.find().pretty()
```

Si quisiéramos realizar una búsqueda sobre algún autor en concreto podemos ejecutar el siguiente comando.

```
db.libros.find({autor: "Pepe"})
```

1.2.3 Bases de datos de tipo grafo

Las bases de datos no relacionales basadas en grafos utilizan **nodos** para almacenar las entidades de datos y **aristas** para almacenar las relaciones entre las entidades. Una arista siempre tiene un nodo de inicio, un nodo final, un tipo y una dirección, y puede describir relaciones principales y secundarias, las acciones, la propiedad y cosas similares. No hay límite para la cantidad y el tipo de relaciones que puede tener un nodo.

Ejemplo

Este ejemplo muestra cómo sería el grafo de una red social. Las personas serían los nodos y us relaciones son las aristas. De este modo es posible conocer quiénes son los amigos de los amigos de una persona específica.

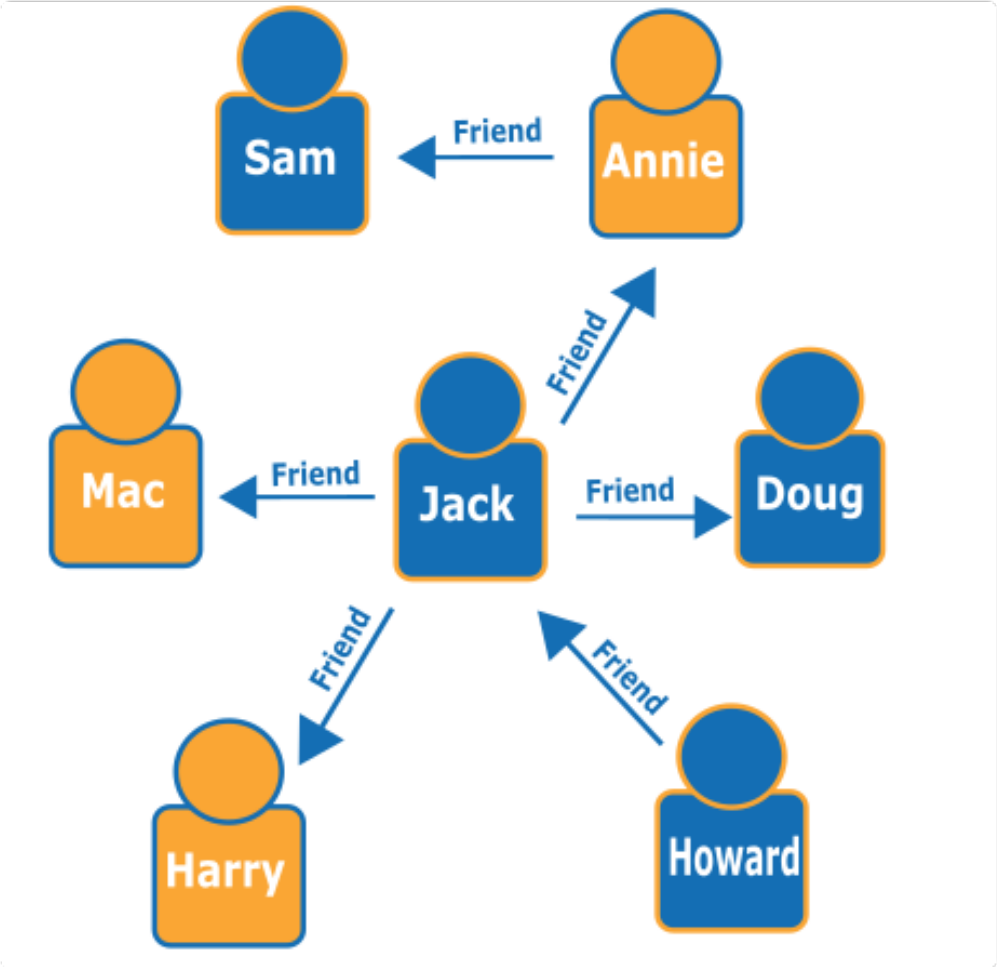

Imagen: Ejemplo de un base de datos basada en grafos. Amazon Web Services.

Casos de uso

- Detección de fraudes.
- Motores de recomendaciones.

Ejemplos de bases de datos de tipo grafo

- Neo4j.
- InfinteGraph.

1.2.4 Bases de datos orientadas a columnas

Las bases de datos orientadas a columnas están optimizadas para obtener columnas de datos de una forma muy rápida y eficiente. Han sido diseñadas para gestionar grandes volúmenes de datos en clústeres distribuidos, lo

que las hace ideales para el procesamiento de Big Data.

Ejemplos de bases de datos orientadas a columnas

- Apache Cassandra.
- HBase.

Referencias

- Sistema gestor de base de datos orientado a columnas.
- Base de datos columnar.

1.3 Referencias

El contenido de esta web ha sido extraído de las siguientes referencias:

- NoSQL.
- Bases de datos NoSQL. Qué son y tipos que nos podemos encontrar.
- ¿Qué son las bases de datos NoSQL?. Amazon Web Services.

2 Licencia