

Gestores de BD NoSQL

Aponte Roldán, Sigfredo (xxxxxxxxxx)), Gonzales Cave, Angel Gabriel
(2017057861)), Pacora Silva, Jorge Carlos (xxxxxxxxxx)), Quispe Mamani,
José Luis (xxxxxxxxxx))

Tacna, Perú

Abstract

EDITAR

1. Resumen

EDITAR

2. Introducción

EDITAR

- A
- B
- C

3. Marco Teórico

3.1. A

3.1.1. A.1

EDITAR [1]

3.1.2. **A.2**

3.2. **Base de datos NoSQL orientadas a Documentos**

Una base de datos orientada a documentos está diseñada para gestionar información orientada a documentos o datos semiestructurados. Tienen un grado de complejidad y flexibilidad superior a las bases de datos clave – valor. A diferencia de las conocidas bases de datos relacionales con su definición de “tabla”, los sistemas documentales están diseñados entorno a la definición abstracta de un “documento”. Un documento es la unidad principal de almacenamiento de este tipo de base de datos, y toda la información que aquí se almacena, se hace en formato de documento.

Este tipo de Base de Datos NoSQL 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.

3.2.1. **Ventajas**

- Almacenar y recuperar todos los datos relacionados como una sola unidad puede entregar ventajas enormes en el rendimiento y la escalabilidad.
- Los gestores no tienen que hacer operaciones complejas como las uniones para encontrar los datos que normalmente están relacionados, ya que todo se encuentra en un mismo lugar.

3.3. **MongoDB**

MongoDB es una base de datos NoSQL Open Source orientada a documentos y ha sido diseñada con la idea de que fuera fácil tanto desarrollar para ella como de ser administrada. Un documento es una estructura de datos compuesta de pares de campos y valores. Además, estos objetos son muy similares a lo que serían objetos en notación JSON.

Utilizar documentos tiene ciertas ventajas:

- Los documentos son tipos de datos nativos en muchos lenguajes de programación.
- Los documentos embebidos y los arrays minimizan la necesidad de realizar joins muy pesados.

- Tener un esquema dinámico es la base del polimorfismo.

3.3.1. *Características generales*

- **Indexado**

MongoDB soporta índices secundarios, permitiendo una buena aceleración en muchas queries, pudiendo indexar cualquier campo de la base de datos.

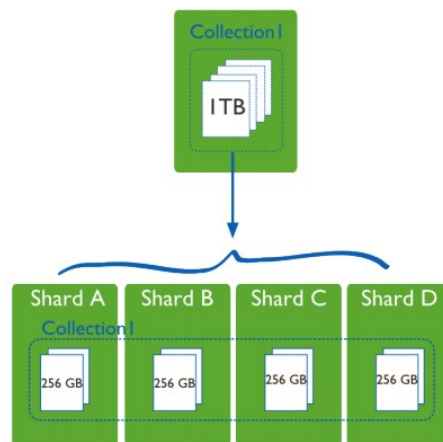
- **Agregación**

MongoDB viene con muchas funciones predefinidas que facilitan en gran medida el tratamiento de las colecciones almacenadas.

- **Replicación y Balanceo de Carga**

MongoDB utiliza un sistema de replicación empleando la arquitectura maestro – esclavo con el resto de máquinas. Además, hace uso de un tipo de particionado horizontal llamado sharding. Estas dos herramientas de escalado hacen de mongo una herramienta poderosa cuando se trata de crecer mucho en poco tiempo.

Mientras que la replicación busca realizar copias de los datos, repartirlos por nuevos hosts, permitiendo ofrecer alta disponibilidad, balanceo de carga y copias automáticas de seguridad, el sharding busca particionar esas tablas que, previsiblemente, se van haciendo más grandes y cada vez son más costosas de replicar.



3.3.2. A.3

EDITAR

- x

- y

- z

3.4. B

3.4.1. B.1

EDITAR

3.4.2. B.2

EDITAR

3.4.3. B.3

EDITAR

4. Análisis (Aplicación)

EDITAR

5. Conclusiones

- Conclusion 1 :
A

- Conclusion 2 :
B

- Conclusion 3 :
C

Referencias

- [1] EncuRed (ne). Sql. Recuperado de <https://www.ecured.cu/SQL>. Accedido 15-08-2019.