

Conceptos y comandos básicos de la replicación en bases de datos NoSQL

Natalia Stefania Gil Chaparro

Universidad Iberoamericana
Base de datos avanzadas
Tutor Jorge Castañeda
Diciembre 1, 2024

Contenido

1.	Introducción.....	3
2.	Requerimientos no funcionales.....	3
3.	Configuración de la replicación.....	3
4.	Explicación del proyecto	5

1. Introducción

En el presente documento, se establecen los requerimientos de redundancia y disponibilidad 24x7 para el sistema de replicación de MongoDB de la base de datos CopaMundial. Se describen las estrategias necesarias para garantizar que el sistema sea capaz de manejar fallos en los nodos sin afectar la disponibilidad de los datos y rendimiento.

2. Requerimientos no funcionales

Requerimientos No Funcionales

Redundancia

- Implementar replicación en al menos 3 nodos para asegurar que los datos estén disponibles incluso si uno de los nodos falla.
- Configuración de un nodo primario y al menos dos nodos secundarios bajo el modelo Replica Set de MongoDB.

Disponibilidad

- Garantizar un tiempo de actividad del 99.99% mediante la implementación de failover automático.
- Implementar monitoreo y alertas en tiempo real para detectar problemas de conectividad o rendimiento.

3. Configuración de la replicación

Estrategia de Replicación

Se utilizará un Replica Set en MongoDB, que incluye:

- Un nodo primario para recibir escrituras.
- Dos nodos secundarios para mantener copias de los datos y participar en elecciones en caso de fallas.

Comandos para Crear el Entorno de Replicación

Paso 1: Configuración de los Nodos

Inicia cada instancia con el siguiente comando, especificando un puerto y un directorio de datos únicos:

```
Mongod --port27017 --dbpath /data/instancia1 --replSet rs()
```

```
Mongod --port27018 --dbpath /data/instancia2 --replSet rs()
```

```
Mongod --port27019 --dbpath /data/ instancia2 --replSet rs()
```

Luego se inicializa el replica set:

```
rs.initiate().
```

Accedemos a una de las instancias

```
Mongo --port 27017
```

Añadimos el nodo a un arbitro:

```
rs.addArb ("localhost:27017")
```

Configuramos el replica set con:

```
rs.add({
_id:7
```

```
nombre: "Cristian",  
apellido: "González",  
nacionalidad: "Colombiano"  
}  
)
```

Y por último verificamos la configuración:

```
rs.status()
```

4. Explicación del proyecto

El objetivo es replicar la base de datos CopaMundial en tres nodos para garantizar disponibilidad continua y redundancia. Se utilizó el modelo Replica Set de MongoDB, donde los nodos secundarios sirven como respaldo y participan en elecciones en caso de falla del nodo primario.