



Actividad 3 - Requerimientos no funcionales



Reybert Andrés Peñuela Sepúlveda

Corporación universitaria Iberoamericana

Facultad de Ingeniería de software

Bases de datos avanzadas

Profesor William Ruiz

Cali, Colombia

16 de abril del 2023

Escenario donde se requeriría un particionamiento (sharding) en MongoDB

Imaginemos que somos el administrador de una plataforma de comercio electrónico de gran envergadura que recibe millones de visitantes cada día y vende una amplia variedad de productos en todo el mundo. La plataforma almacena la información de los productos, usuarios, pedidos y transacciones en una base de datos MongoDB. Con el paso del tiempo, el tráfico a la plataforma sigue creciendo, lo que se traduce en un aumento del volumen de datos y solicitudes a la base de datos.

En este escenario, nos enfrentaremos a los siguientes desafíos:

Escalabilidad de la base de datos: La base de datos MongoDB experimenta un crecimiento constante en términos de volumen de datos y solicitudes, lo que dificulta el manejo de las operaciones de lectura y escritura. El rendimiento de la base de datos empieza a degradarse, afectando la experiencia del usuario en la plataforma.

Distribución de datos: La base de datos almacena información de usuarios y transacciones de diferentes regiones del mundo. Algunas regiones pueden tener más tráfico y actividad que otras, lo que puede provocar una carga desequilibrada en los servidores de la base de datos.

Alta disponibilidad: Para garantizar un servicio ininterrumpido a los usuarios, es fundamental mantener una alta disponibilidad de la base de datos, incluso en caso de fallos del servidor o problemas de red.

Para abordar estos desafíos y garantizar un rendimiento óptimo de la base de datos, es necesario implementar el particionamiento o sharding en MongoDB.

El sharding permitirá dividir los datos de la base de datos en fragmentos y distribuirlos en múltiples servidores, conocidos como shards. Esto mejorará la escalabilidad de la base de datos, ya que cada shard será responsable de una parte del conjunto total de datos, lo que permitirá distribuir la carga y las solicitudes entre los servidores.

Además, el sharding también ayudará a equilibrar la distribución de datos, ya que se puede configurar para dividir los datos según criterios específicos, como la ubicación geográfica. Esto garantizará una distribución uniforme de la carga y un mejor rendimiento en todas las regiones.

Finalmente, la implementación del sharding también contribuirá a la alta disponibilidad de la base de datos. En caso de fallos del servidor o problemas de red, los servidores adicionales (shards) estarán disponibles para manejar las solicitudes, lo que permitirá a la plataforma seguir funcionando sin interrupciones.

Requerimientos no funcionales para el escenario de particionamiento (sharding) en MongoDB

Rendimiento:

- a) **Tiempo de respuesta:** La base de datos debe mantener tiempos de respuesta rápidos para las consultas y escrituras, incluso cuando el volumen de datos y las solicitudes aumenten.
- b) **Tasa de transacción:** La base de datos debe ser capaz de manejar un alto número de transacciones simultáneas sin degradar significativamente el rendimiento.

Escalabilidad:

- a) **Escalabilidad horizontal:** La solución de sharding debe permitir la escalabilidad horizontal, lo que significa que se puedan agregar más servidores (shards) al sistema para manejar el crecimiento del volumen de datos y las solicitudes.
- b) **Balanceo de carga:** La distribución de los datos entre los shards debe garantizar un balanceo de carga adecuado para evitar cuellos de botella y maximizar el rendimiento.

Disponibilidad:

- a) **Alta disponibilidad:** La base de datos debe estar disponible en todo momento, incluso en caso de fallos del servidor o problemas de red.
- b) **Recuperación ante desastres:** La solución de sharding debe incluir mecanismos de recuperación ante desastres para garantizar la continuidad del negocio y la protección de los datos en caso de fallos catastróficos.

Seguridad:

- a) **Autenticación y autorización:** La base de datos debe proporcionar mecanismos de autenticación y autorización para garantizar que solo los usuarios autorizados puedan acceder y modificar los datos.
- b) **Cifrado:** La solución de sharding debe garantizar la protección de los datos en tránsito y en reposo mediante el cifrado.

Mantenibilidad:

- a) **Monitorización:** La solución de sharding debe ofrecer herramientas de monitorización para evaluar el rendimiento, la salud y el estado de los shards y la base de datos en su conjunto.
- b) **Gestión de configuración:** La solución debe permitir una gestión de configuración eficiente y sencilla, facilitando la adición, eliminación y modificación de shards según sea necesario.

Conclusión

En conclusión, los requerimientos no funcionales abordados en este documento son cruciales para garantizar el éxito de la implementación de sharding en MongoDB en el escenario de la plataforma de comercio electrónico. El rendimiento, la escalabilidad, la disponibilidad, la seguridad y la mantenibilidad son factores clave que deben ser cuidadosamente considerados y optimizados para satisfacer las necesidades del negocio y garantizar una experiencia de usuario óptima.

El rendimiento y la escalabilidad permitirán a la plataforma manejar de manera eficiente un alto volumen de datos y transacciones simultáneas, lo que es esencial para una plataforma de comercio electrónico en crecimiento. La alta disponibilidad y la recuperación ante desastres asegurarán la continuidad del negocio y protegerán los datos en caso de fallos del sistema o problemas de red. La seguridad de los datos es crucial para proteger la información confidencial de los usuarios y cumplir con las regulaciones aplicables. La mantenibilidad, incluida la monitorización y la gestión de configuración, facilitará la administración del sistema y permitirá adaptarse rápidamente a las necesidades cambiantes del negocio.

Al abordar estos requerimientos no funcionales, la solución de sharding en MongoDB garantizará un rendimiento óptimo, escalabilidad y confiabilidad, lo que permitirá a la plataforma de comercio electrónico prosperar y expandirse en un mercado altamente competitivo.

ENLACES

- Repositorio GitHub: <https://github.com/ReybertAPS/MongoDB-Sharding>
- Video: <https://drive.google.com/drive/folders/1Kui2Fhzq9MCV0jLqLIOWdmk4zbhd-Zjp?usp=sharing>

Nota: El video lo podrá encontrar tanto en el repositorio de GitHub como en el enlace de drive. Esto lo hago porque a veces drive da problemas para descargar o visualizar videos.