

# 01 Introducción

Sharding se refiere al proceso de dividir datos entre máquinas; el término particionamiento también se usa a veces para describir este concepto. Al colocar un subconjunto de datos en cada máquina, es posible almacenar más datos y manejar más carga sin requerir máquinas más grandes o más potentes, solo una mayor cantidad de máquinas menos potentes.

El fragmentado también se puede utilizar para otros fines, como colocar datos a los que se accede con más frecuencia en un hardware más eficiente o dividir un conjunto de datos basado en la geografía para ubicar un subconjunto de documentos en una colección (por ejemplo, para usuarios basados en una localidad particular) cerca de servidores de aplicación desde los que se accede más comúnmente.

La fragmentación manual se puede hacer con casi cualquier software de base de datos. Con este enfoque, una aplicación mantiene conexiones a varios servidores de bases de datos diferentes, cada uno de los cuales es completamente independiente. La aplicación gestiona el almacenamiento de diferentes datos en diferentes servidores y consultas en el servidor apropiado para recuperar los datos. Esta configuración puede funcionar bien, pero se vuelve difícil de mantener cuando se agregan o eliminan nodos del clúster o ante cambios en las distribuciones de datos o patrones de carga.

MongoDB admite el particionamiento automático, que intenta abstraer la arquitectura de la aplicación y simplificar la administración de dicho sistema. MongoDB permite que su aplicación ignore el hecho de que no está hablando con un servidor MongoDB independiente, hasta cierto punto. En el lado de las operaciones, MongoDB automatiza el equilibrio de datos entre fragmentos y facilita la adición y eliminación de capacidad.

Sharding es la forma más compleja de configurar MongoDB, tanto desde un punto de vista de desarrollo como operativo. Hay muchos componentes para configurar y monitorear, y los datos se mueven automáticamente por el clúster. Debe sentirse cómodo con los servidores independientes y los conjuntos de réplicas antes de intentar implementar o utilizar un clúster fragmentado.

Además, al igual que con los conjuntos de réplicas, el medio recomendado para configurar e implementar clústeres fragmentados es a través de MongoDB Ops Manager o MongoDB Atlas. Se recomienda Ops Manager si necesita mantener el control de su infraestructura informática. Se recomienda MongoDB Atlas si puede dejar la administración de la infraestructura a MongoDB (tiene la opción de ejecutarse en Amazon AWS, Microsoft Azure o Google Compute Cloud).

## Componentes

El Sharding de MongoDB le permiten crear un grupo de muchas máquinas (fragmentos) y dividir una colección entre ellos, colocando un subconjunto de datos en cada fragmento. Esto permite que su aplicación crezca más allá de los límites de recursos de un servidor independiente o un conjunto de réplicas.

NOTA Muchas personas están confundidas acerca de la diferencia entre replicación y fragmentación. Recuerde que la replicación crea una copia exacta de sus datos en varios servidores, por lo que cada servidor es una imagen reflejada de todos los demás servidores. Por el contrario, cada fragmento contiene un subconjunto diferente de datos.

Uno de los objetivos de los fragmentos es hacer que un grupo de 2, 3, 10 o incluso cientos de fragmentos parezca una sola máquina para su aplicación. Para ocultar estos detalles de la aplicación, ejecutamos uno o más procesos de enrutamiento llamados mongos frente a los fragmentos. Un mongos mantiene una "tabla de contenido" que le dice qué fragmento contiene qué datos.

Las aplicaciones pueden conectarse a este enrutador y emitir solicitudes normalmente, como se muestra en la Figura 14-1. El enrutador, sabiendo qué datos están en qué fragmento, puede redireccionar las solicitudes a los fragmentos apropiados. Si hay respuestas a una solicitud, el enrutador las recoge y, si es necesario, las fusiona y las envía de vuelta a la aplicación.

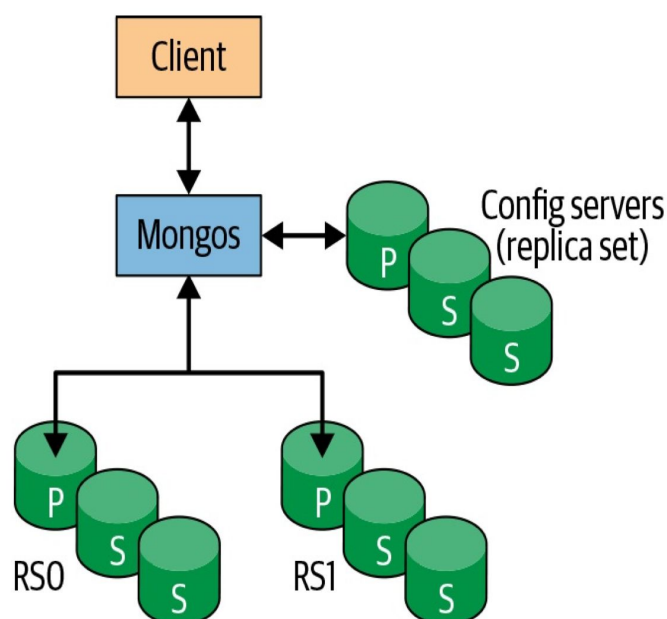


Figure 14-1. Sharded client connection

Hasta donde la aplicación sabe, está conectado a un mongod independiente.