

mongoDB

Banco de dados NoSQL - Sharding

Prof. Gustavo Leitão

Sharding



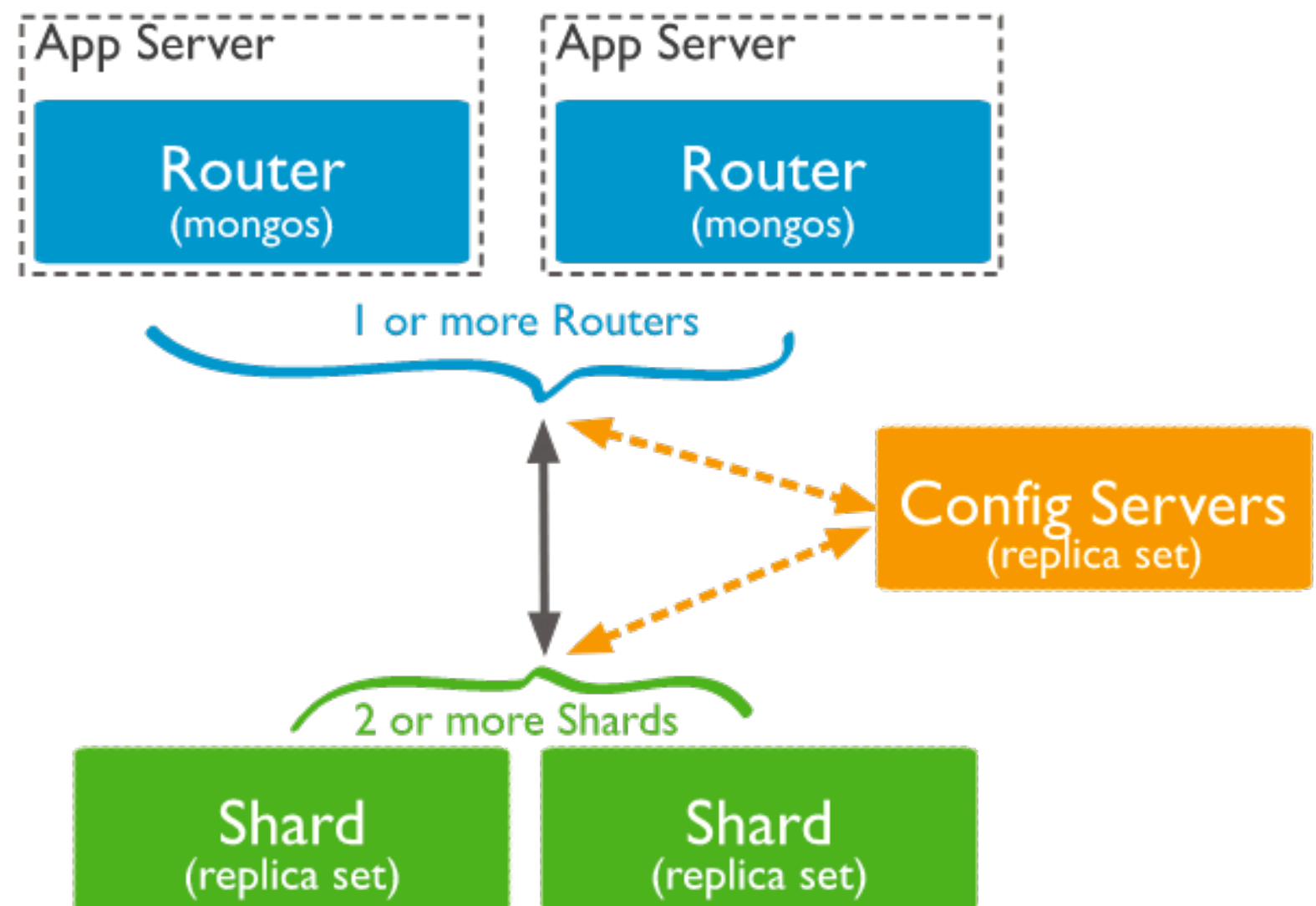
- Sharding é o método que o Mongo utiliza para distribuir os dados nos nós do cluster
- Através do Sharding é possível armazenar um grande conjunto de dados, aumentando a vazão do sistema



- ○ Sharded Cluster consiste nos seguintes componentes:
 - sharded: Cada *shard* contém um subconjunto dos dados. Cada shard pode ser implantado como replica set.
 - mongos: Os mongos são roteadores de consultas promovendo uma interface entre os clientes e os *shards*
 - Config servers: Armazenam configurações de meta dados do cluster. Os config servers devem ser implantados em replica set



- sharded: Cada *shard* contém um subconjunto dos dados.
- mongos: Os mongos são roteadores de consultas
- Config servers: Armazenam configurações de metadados do cluster.

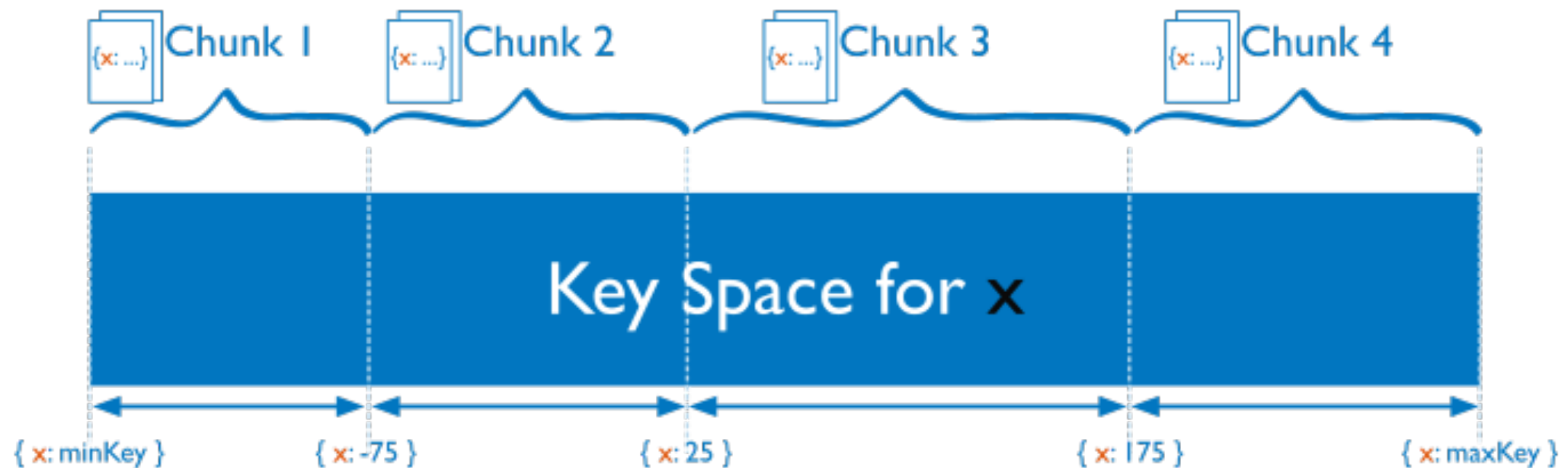




- Shard Keys
 - Utilizada para distribuir os dados entre os nós
 - Consiste em um campo presente obrigatoriamente em todos os documentos de uma "sharded collection"
 - A coleção deve ter um índice pelo campo a ser utilizado como shard key (não pode ser trocado depois)



- Chunks
- Intervalo da shard key que pertence a um nó

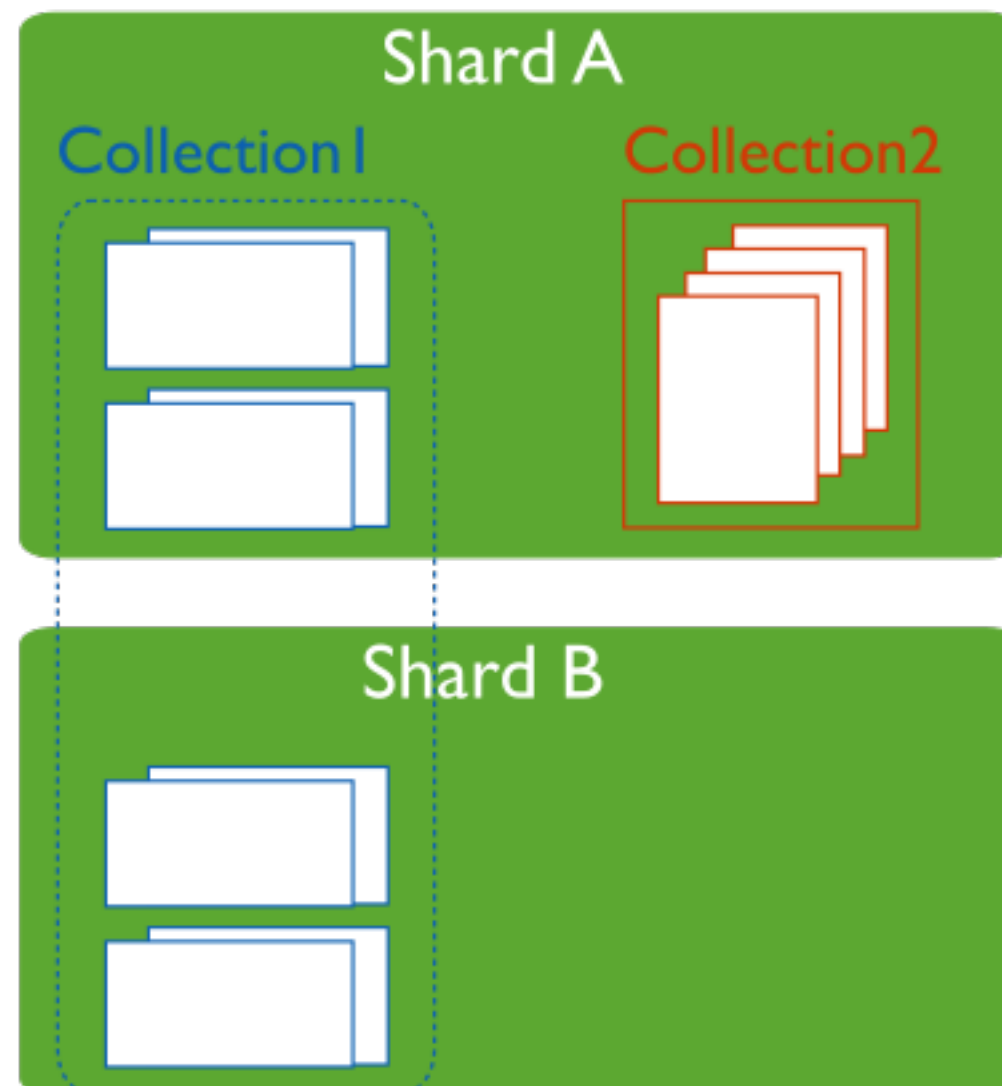




- Vantagens
 - **Read/Write:** MongoDB distribui as leituras e escrita pelos shards permitindo que cada shard processe apenas um subconjunto dos dados
 - **Capacidade:** Como cada shard contém apenas um subconjunto dos dados a capacidade total de armazenamento pode ser ampliada
 - **Alta Disponibilidade:** Um sharded cluster pode continuar operando mesmo caso alguns shards se torne indisponíveis. Apenas operações de leitura/escrita de responsabilidade do shard indisponível não serão realizadas.

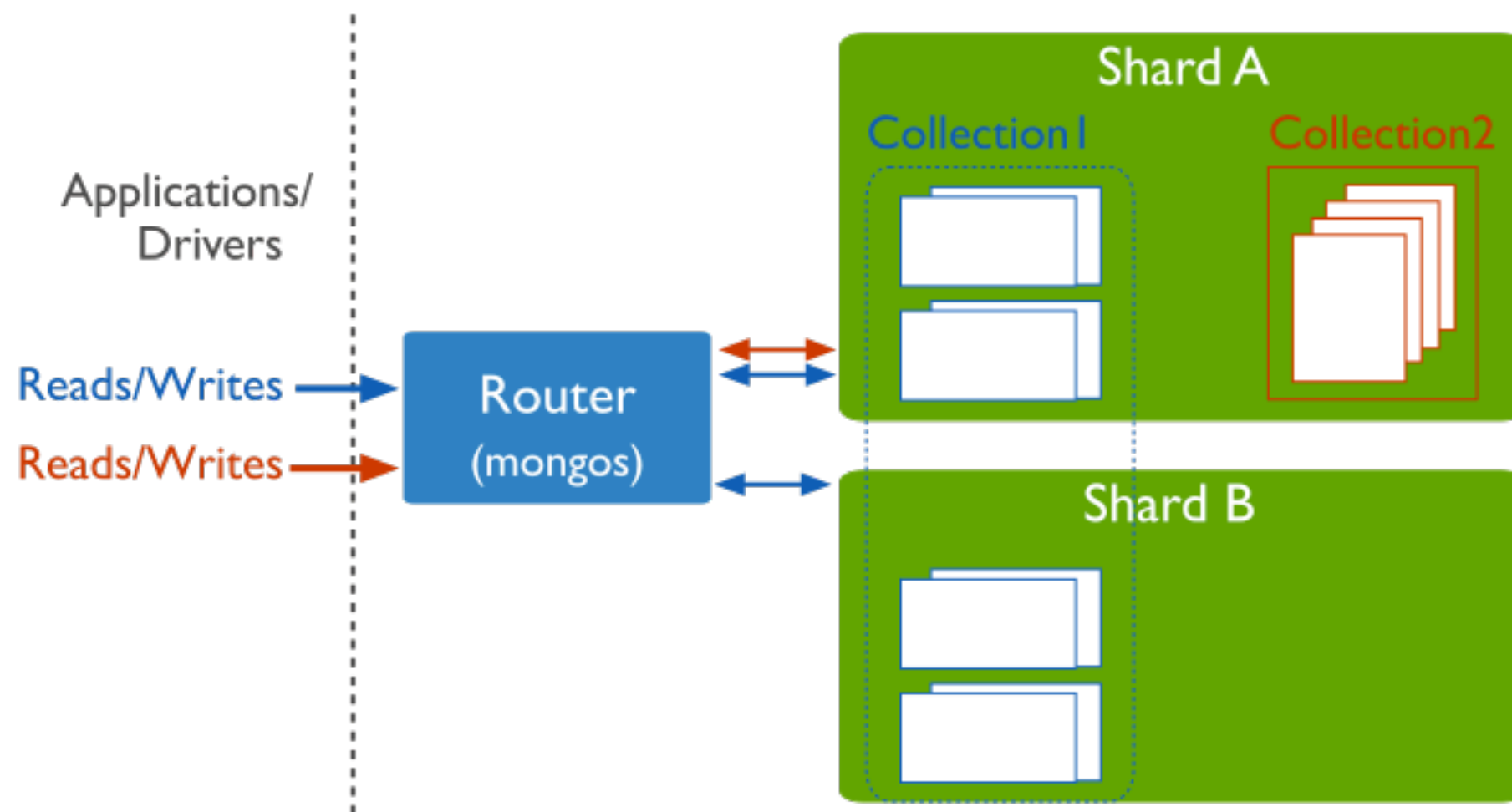


- Ambiente misto: Um banco podem conter coleções sharded e unsharded





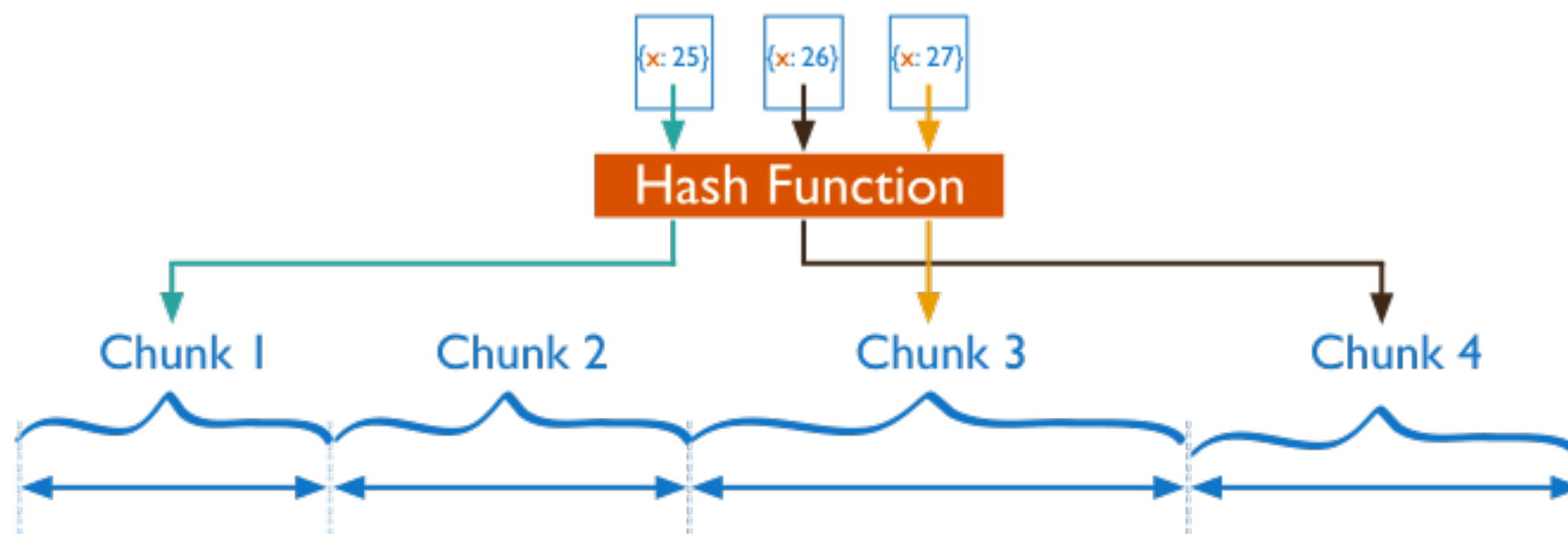
- Tanto coleções sharded como unsharded devem ser acessadas através de roteadores mongos





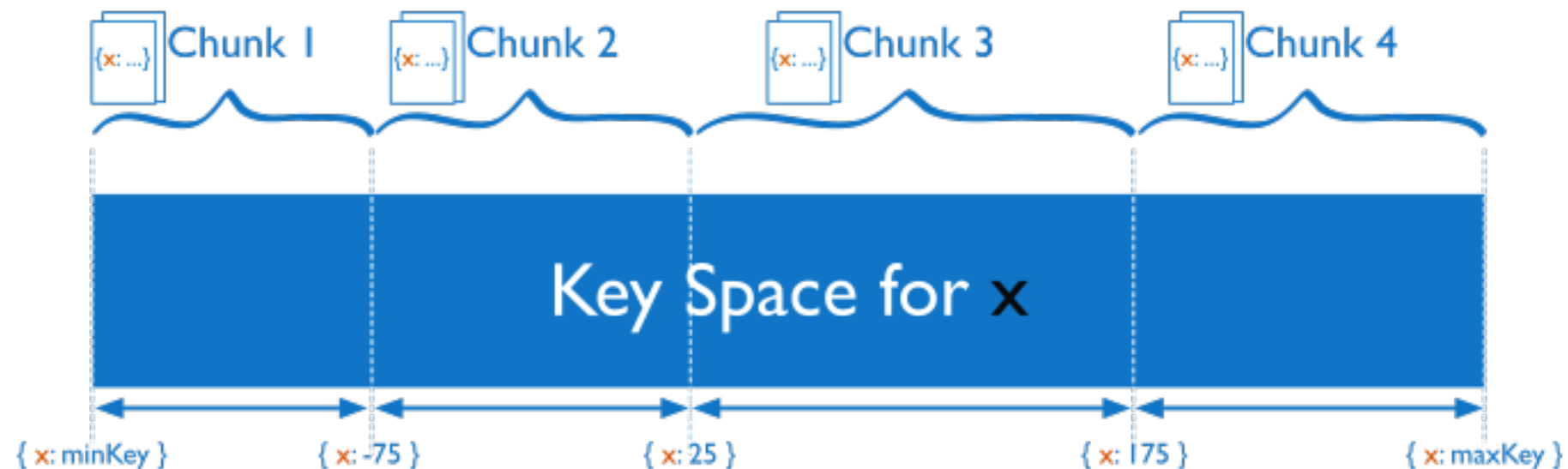
- **Hashed Sharding:**

- Computa função hash da shared key para distribuição dos dados. Cada chunk será responsável por um range do hash
- Proporciona melhor distribuição dos dados, porém consultas sobre um range da shared key podem resultar em acesso a vários nós do cluster





- Range Sharding:
 - Dados são distribuídos de acordo com ranges do shared cluster
 - Dados podem ficar mais concentrados. Porém dados semelhantes tendem a residirem no mesmo shard





Production Configuration

In a production cluster, ensure that data is redundant and that your systems are highly available. Consider the following for a production sharded cluster deployment:

- Deploy Config Servers as a 3 member [replica set](#)
- Deploy each Shard as a 3 member [replica set](#)
- Deploy one or more [mongos](#) routers

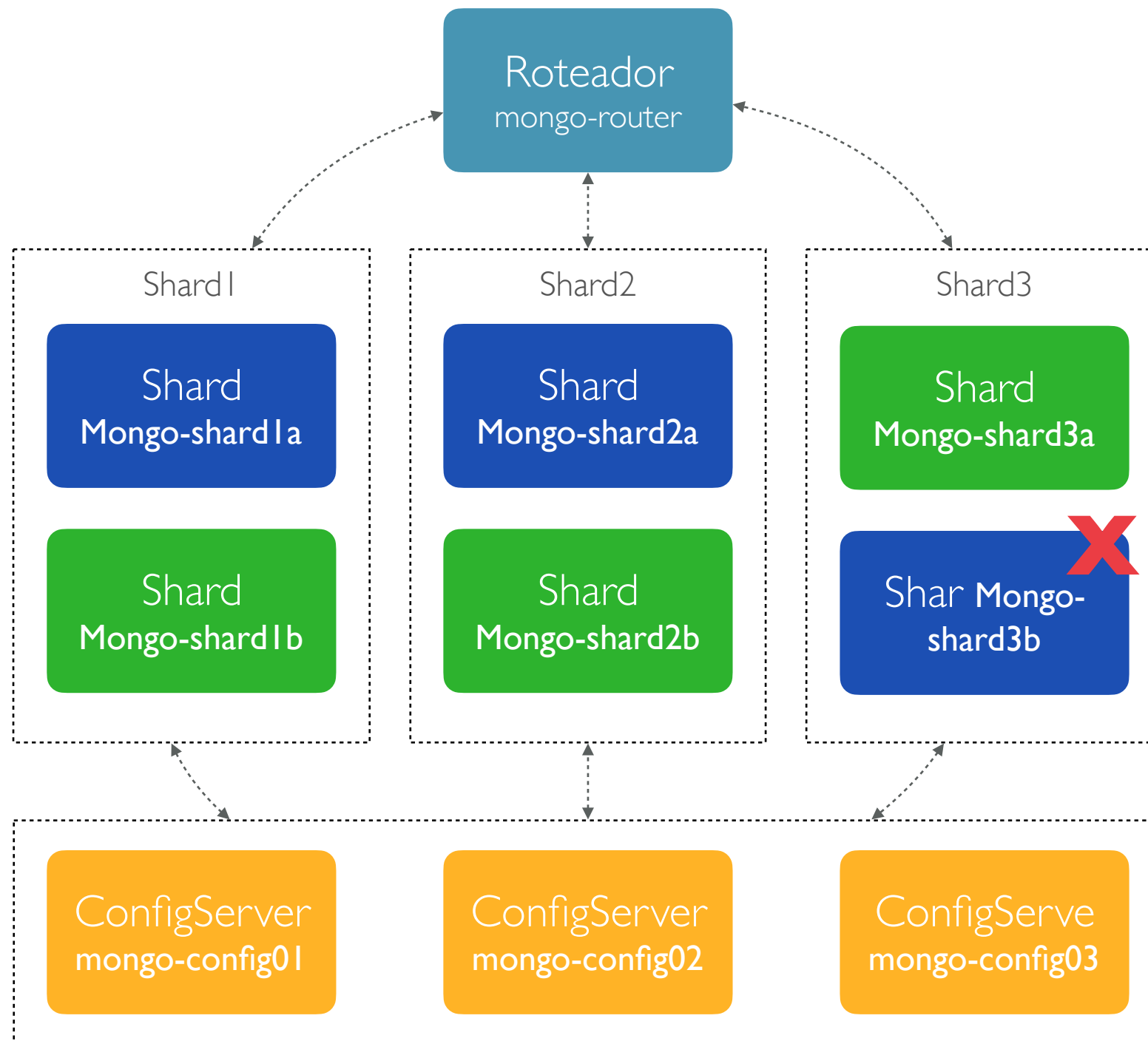


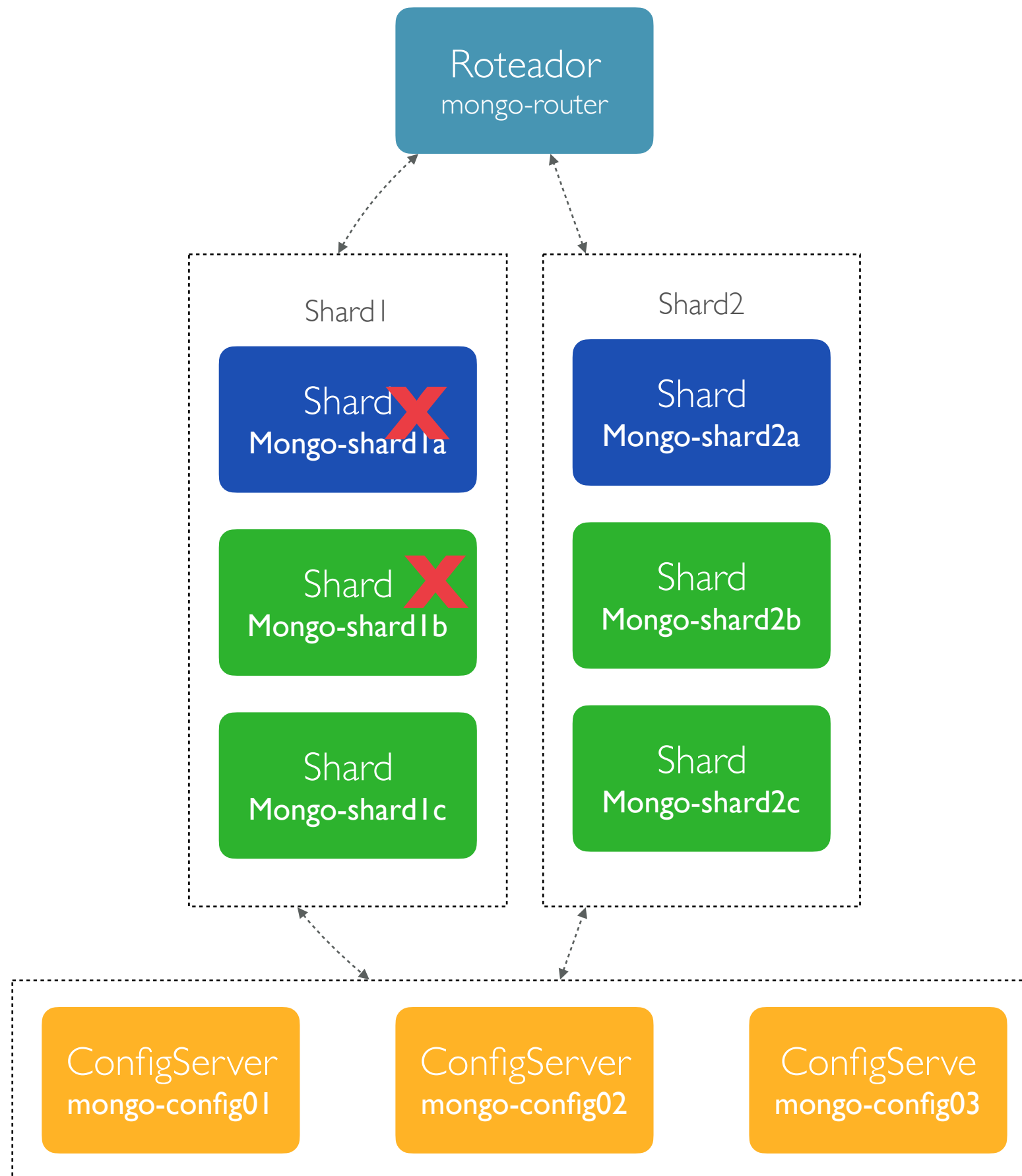
NOTE:

Distributing replica set members across two data centers provides benefit over a single data center. In a two data center distribution,

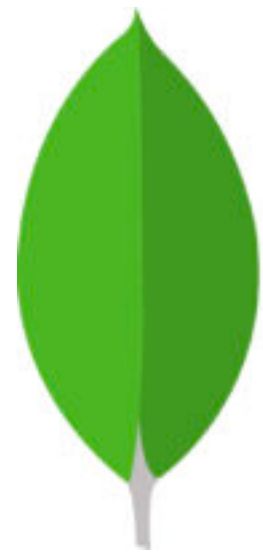
- If one of the data centers goes down, the data is still available for reads unlike a single data center distribution.
- If the data center with a minority of the members goes down, the replica set can still serve write operations as well as read operations.
- However, if the data center with the majority of the members goes down, the replica set becomes read-only.

If possible, distribute members across at least three data centers. For config server replica sets (CSRS), the best practice is to distribute across three (or more depending on the number of members) centers. If the cost of the third data center is prohibitive, one distribution possibility is to evenly distribute the data bearing members across the two data centers and store the remaining member (either a data bearing member or an arbiter to ensure odd number of members) in the cloud if your company policy allows.






```
sh.enableSharding (“mongo-db-shard”)
use mongo-db-shard
  for (var i=0; i < 20000; i++ ) { db.testecollection.insert ( { pos:i, dt: Date() } ) };
db.testecollection.createIndex ( { pos:1 }, { unique : true } )
sh.shardCollection (“mongo-db-shard.testecollection”, {pos:1}, true)
db.testecollection.getShardDistribution()
```



mongoDB

Banco de dados NoSQL - Sharding

Prof. Gustavo Leitão