

TECNOLOGIAS DE PROGRAMAÇÃO DE SISTEMAS DE
INFORMAÇÃO

MongoDB

SISTEMAS GESTORES DE BASES DE DADOS II | Prof. Magno
Andrade | Prof. Jorge Louro

UNIDADE CURRICULAR : SISTEMAS GESTORES DE BASES DE DADOS II



- **Conteúdo da apresentação**
 - *Sharding*
 - Componentes
 - *Shard key*
 - Vantagens
 - Técnicas

SHARDING

UNIDADE CURRICULAR : SISTEMAS GESTORES DE BASES DE DADOS II



- **Introdução**

- É um método para distribuir dados por várias máquinas.
- Utilizado para suportar conjuntos de dados muito grandes e operações de alto rendimento.
- Bases de dados com conjuntos de dados muito grandes ou aplicações com pedidos constantes desafiam a capacidade de um só servidor.

UNIDADE CURRICULAR : SISTEMAS GESTORES DE BASES DE DADOS II



- **Introdução**

- Por exemplo, constantes execuções de *queries* podem esgotar a capacidade do processador do servidor.
- Conjuntos de dados maiores do que a *RAM*, trazem problemas de desempenho ao discos rígidos.

UNIDADE CURRICULAR : SISTEMAS GESTORES DE BASES DE DADOS II



- **Crescimento de um sistema**
 - Duas formas de acomodar o crescimento:
 - Escalonamento vertical
 - Envolve aumentar a capacidade de um só servidor, usar um processador mais potente, adicionar mais memória *RAM* ou aumentar a quantidade de espaço de armazenamento.
 - Escalonamento horizontal
 - Envolve a divisão do conjunto de dados do sistema e o carregamento em múltiplos servidores, adicionando servidores adicionais para aumentar a capacidade conforme o necessário.

UNIDADE CURRICULAR : SISTEMAS GESTORES DE BASES DE DADOS II



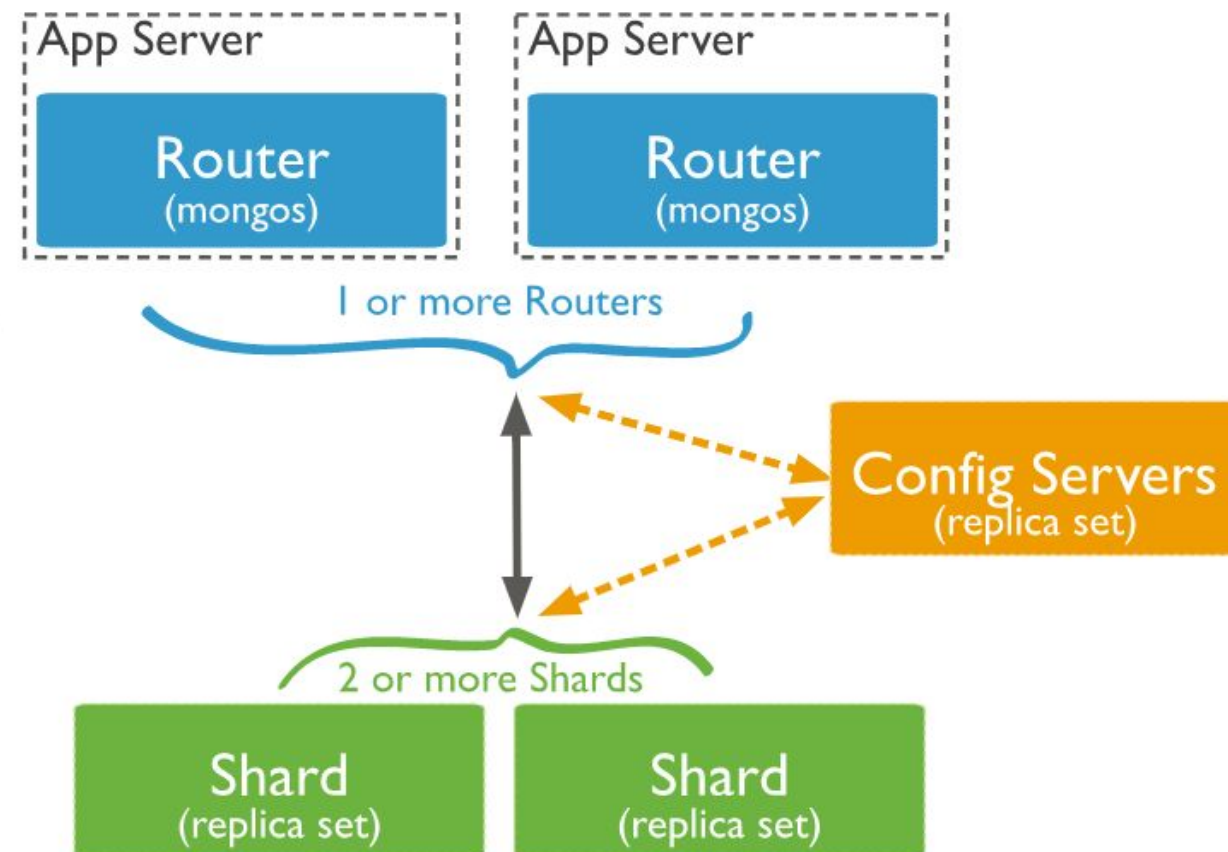
- **Sharding - Componentes**

- Um **cluster sharded** tem os seguintes componentes:
 - **shard**
 - Cada *shard* contém um subconjunto dos dados fragmentados. Cada *shard* pode ser “lançado” como um *replica set*.
 - **mongos**
 - Actua como um *router* de *queries*, fornecendo uma interface entre as aplicações de cliente e o *cluster sharded*.
 - **config servers**
 - Guardam metadados e configurações para o *cluster*. Tem de ser “lançados” como um *replica set*.

UNIDADE CURRICULAR : SISTEMAS GESTORES DE BASES DE DADOS II

- **Sharding - Componentes**

- O MongoDB fragmenta os dados ao nível da colecção, distribuindo os dados da colecção por vários *shards* do *cluster*.



UNIDADE CURRICULAR : SISTEMAS GESTORES DE BASES DE DADOS II



- ***Sharding*** - Componente *shard*

- Um *shard* é uma instância *mongod* ou *replica set*.
- Um *shard* contém um subconjunto dos dados fragmentados num *cluster sharded*.
- Juntos, os *shards* do *cluster* mantêm **todo** o conjunto de dados do *cluster*.

UNIDADE CURRICULAR : SISTEMAS GESTORES DE BASES DE DADOS II



- ***Sharding* - Componente *shard***

- Os *shards* devem ser “lançados” como *replica set*, para garantir redundância e alta disponibilidade.
- Utilizadores, clientes ou aplicações só devem ligar-se directamente a um *shard*, para realizar operações de administração ou de manutenção.
- Efectuar *queries* num único *shard*, retorna apenas subconjuntos de dados.

UNIDADE CURRICULAR : SISTEMAS GESTORES DE BASES DE DADOS II



- ***Sharding - Componente shard***

- *As queries* devem ser efectuadas utilizando o *router (mongos)*.
- O *MongoDB* não garante que dois *chunks* (pedaços) contínuos estão no mesmo *shard*.

UNIDADE CURRICULAR : SISTEMAS GESTORES DE BASES DE DADOS II



- **Sharding - Componente *config servers***
 - Os *config servers* guardam os metadados, de um *cluster sharded*.
 - Os metadados reflectem o estado e organização de todos os dados e componentes, dentro de um *cluster sharded*.
 - Estes metadados incluem a lista de *chunks* (pedaços) em cada *shard* e o intervalo que define cada *chunk* (pedaço).

UNIDADE CURRICULAR : SISTEMAS GESTORES DE BASES DE DADOS II



- **Sharding - Componente *config servers***
 - As instâncias de ***mongos*** guardam estes dados em *cache* e utilizam esta informação para encaminhar as operações de leitura e escrita, para os *shards* correctos.
 - Os ***mongos*** actualizam a *cache* quando existem alterações nos metadados do *cluster*, como por exemplo, divisão de *chunks* ou adição de um *shard*.
 - Os *shards* também lêem metadados de *chunks* através dos *config servers*.

UNIDADE CURRICULAR : SISTEMAS GESTORES DE BASES DE DADOS II



mongoDB®

- **Sharding - Componente *config servers***
 - Alguns comandos utilizados neste componente:
 - Como criar uma instância *config server*:
 - Utilizando o seguinte comando:
 - `mongod --configsvr --replSet configReplSet --bind_ip localhost,<hostname(s)|ip address(es)>`

UNIDADE CURRICULAR : SISTEMAS GESTORES DE BASES DE DADOS II



- **Sharding** - Componente *mongos*

- As instâncias ***mongos*** encaminham as operações de *queries* e escrita para os *shards* do *cluster sharded*.
- É a única interface que deve ser utilizada pelas aplicações para aceder aos dados de um *cluster sharded*.
- **Nota:** as aplicações nunca são ligadas ou comunicam directamente com os *shards*.

UNIDADE CURRICULAR : SISTEMAS GESTORES DE BASES DE DADOS II

- **Sharding - Componente *mongos***

- Os ***mongos*** guardam em *cache* os metadados dos *config servers* e localizam que dados estão em cada *shard*.
- ***Mongos*** utilizam estes metadados para encaminhar as operações pedidas pelas aplicações e clientes, para as instâncias *mongod*.
- A instância ***mongos*** junta os dados de cada um dos ***shards*** escolhidos e retorna os dados.



UNIDADE CURRICULAR : SISTEMAS GESTORES DE BASES DE DADOS II

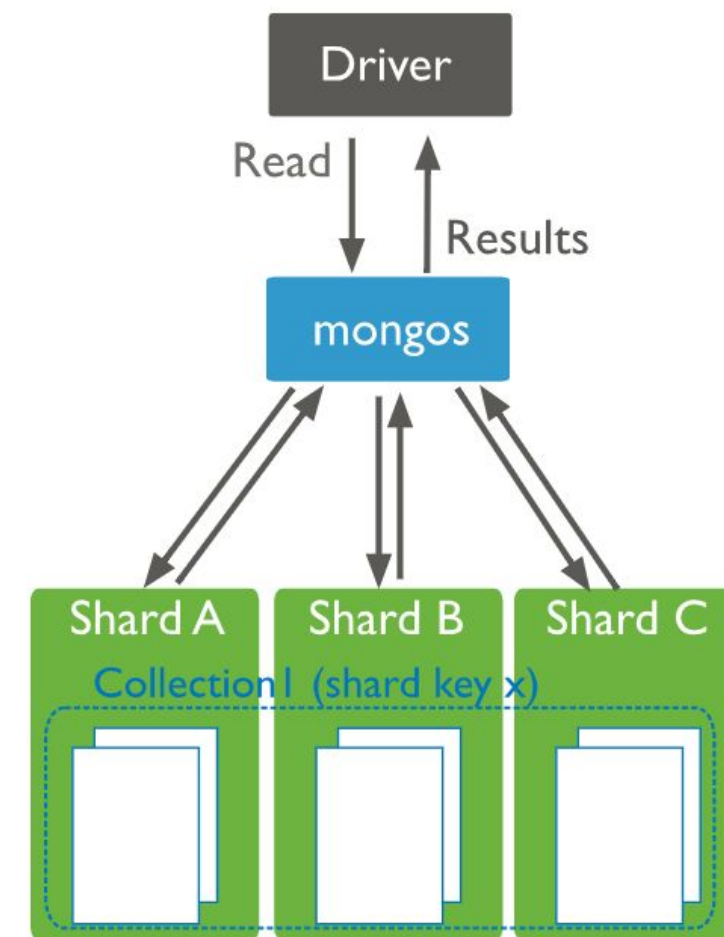
- **Sharding - Componente *mongos***

- Operações de transmissão

- As instâncias ***mongos*** consultam por transmissão todos os *shards* da colecção, a não ser que consiga determinar em que *shard* ou subconjunto de *shards* guardam os dados pretendidos.
- Após receber todas as respostas de todos os *shards*, faz a junção dos dados e retorna-os.



mongoDB®



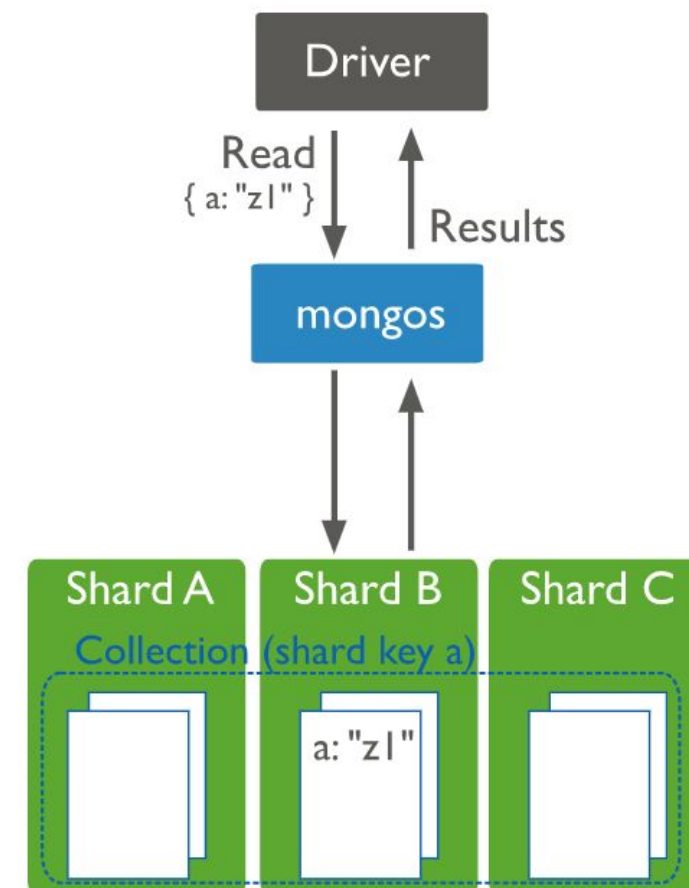
UNIDADE CURRICULAR : SISTEMAS GESTORES DE BASES DE DADOS II

- **Sharding** - Componente *mongos*

- Operações direccionadas
 - A instância **mongos** pode encaminhar as *queries*, que incluem a *shard key*.
 - É usada o valor da *shard key* para localizar o *chunk* (pedaço), em que o intervalo do *chunk* inclui o valor da *shard key* e direcciona a *query* para o *shard* que contém o *chunk*.



mongoDB®



UNIDADE CURRICULAR : SISTEMAS GESTORES DE BASES DE DADOS II



mongoDB®

- **Sharding - Componente *mongos***

- Alguns comandos utilizados neste componente:
 - Adicionar um *shard replica set* a um *cluster sharded*:
 - `sh.addShard("<replica_set>/<hostname><:port>")`
 - `sh.addShard("repl0/mongodb3.example.net:27327")`
 - Activar o *sharding* numa determinada base de dados:
 - `sh.enableSharding("database")`
 - `sh.enableSharding("test")`
 - **Nota:** este comando tem de ser executado antes de utilizar o comando `shardCollection()`

UNIDADE CURRICULAR : SISTEMAS GESTORES DE BASES DE DADOS II



mongoDB®

- **Sharding** - Componente *mongos*
 - Alguns comandos utilizados neste componente:
 - Efectuar o *sharding* de uma colecção:
 - `sh.shardCollection(namespace, key, unique, options)`
 - Nota: este comando só pode ser utilizado **uma única vez** numa colecção.

```
{  
  shardCollection: "<database>.<collection>",  
  key: <shardkey>,  
  unique: <boolean>,  
  numInitialChunks: <integer>,  
  collation: { locale: "simple" },  
  writeConcern: <document>  
}
```

UNIDADE CURRICULAR : SISTEMAS GESTORES DE BASES DE DADOS II



mongoDB®

- **Sharding - Componente *mongos***

- Alguns comandos utilizados neste componente:
 - Efectuar o *sharding* de uma colecção (cont.):
 - `sh.shardCollection(namespace, key, unique, options)`
 - *namespace* → é o nome da base de dados e a colecção no seguinte formato: "database.collection"
 - *key* → é a *shard key*, com o valor 1, -1 ou "hashed"
 - *unique* → assegura que o índice da *shard key* é exclusivo.
 - Exemplos do comando anterior:
 - `sh.shardCollection("records.people", { zipcode: 1 })`
 - `sh.shardCollection("records.people", { zipcode: -1 }, true)`

UNIDADE CURRICULAR : SISTEMAS GESTORES DE BASES DE DADOS II



mongoDB®

- **Sharding - Componente *mongos***

- Alguns comandos utilizados neste componente:
 - Remover um *shard* do cluster:
 - `db.adminCommand({ removeShard : "bristol01" })`
 - só é possível remover um *shard* de cada vez, se existirem remoções em progresso é apresentado um erro.
 - O processo de balanceamento, migra todos os *chunks* desse *shard*, para outros *shards* do *cluster*.
 - Após o *shard* estar vazio, o *MongoDB* remove o *shard* do *cluster*.

UNIDADE CURRICULAR : SISTEMAS GESTORES DE BASES DE DADOS II



mongoDB®

- **Sharding - Componente *mongos***

- Alguns comandos utilizados neste componente:
- Determinar se a instância é do tipo ***mongos***:
 - Para detectar se a instância conectada é deste tipo, é usado o comando:
 - `db.runCommand({ isMaster: 1 })`
 - Se for deste tipo, retorna um documento com o campo “***msg***” e o valor “***isdbgrid***”.

```
{  
  "ismaster" : true,  
  "msg" : "isdbgrid",  
  "maxBsonObjectSize" : 16777216,  
  "ok" : 1,  
  ...  
}
```


UNIDADE CURRICULAR : SISTEMAS GESTORES DE BASES DE DADOS II



mongoDB®

- **Sharding - Componente *mongos***
 - Alguns comandos utilizados neste componente:
 - Ver informações sobre o *cluster sharded*:
 - Utilizando o comando ***sh.status()*** ou ***db.printShardingStatus()***
 - Estes comandos apresentam informações sobre o *cluster*. Por exemplo, estas informações incluem a distribuição de *chunks* pelos vários *shards*.

UNIDADE CURRICULAR : SISTEMAS GESTORES DE BASES DE DADOS II



mongoDB®

- **Sharding - Componente *mongos***

- Alguns comandos utilizados neste componente:
- Como criar uma instância ***mongos***:
 - Utilizando o seguinte comando (é um exemplo):
 - `mongos --configdb configReplSet/mongoddb07.example.net:27019,mongoddb08.example.net:27019,mongoddb09.example.net:27019 --bind_ip localhost,<hostname(s)|ip address(es)>`

UNIDADE CURRICULAR : SISTEMAS GESTORES DE BASES DE DADOS II



mongoDB®

- **Sharding - Componente *mongos***
 - Alguns comandos utilizados neste componente:
 - Listar todos os *shards*:
 - Utilizando o seguinte comando:
 - `db.adminCommand({ listShards : 1 })`
 - Listar as bases de dados com o *sharding* activo:
 - Utilizando os seguintes comandos:
 - `use config`
 - `db.databases.find({ "partitioned": true })`

UNIDADE CURRICULAR : SISTEMAS GESTORES DE BASES DE DADOS II



- ***Sharding - Shard Key***

- Para distribuir os documentos de uma colecção, o *MongoDB* fragmenta a colecção utilizando a *shard key*.
- A *shard key* determina a distribuição dos documentos da colecção pelos *shards* do *cluster*.
- A *shard key* consiste num campo imutável ou campos que existem em todos os documentos dessa colecção.

UNIDADE CURRICULAR : SISTEMAS GESTORES DE BASES DE DADOS II



- ***Sharding - Shard Key***

- Estes campos podem ser:
 - um campo indexado.
 - campos compostos indexados.
- A *shard key* é escolhida no momento em que a colecção é fragmentada.
- Após a fragmentação, **já não é possível (v4.0)** alterar a *shard key*.
- Uma colecção fragmentada (***sharded***) só pode ter uma *shard key*.

UNIDADE CURRICULAR : SISTEMAS GESTORES DE BASES DE DADOS II

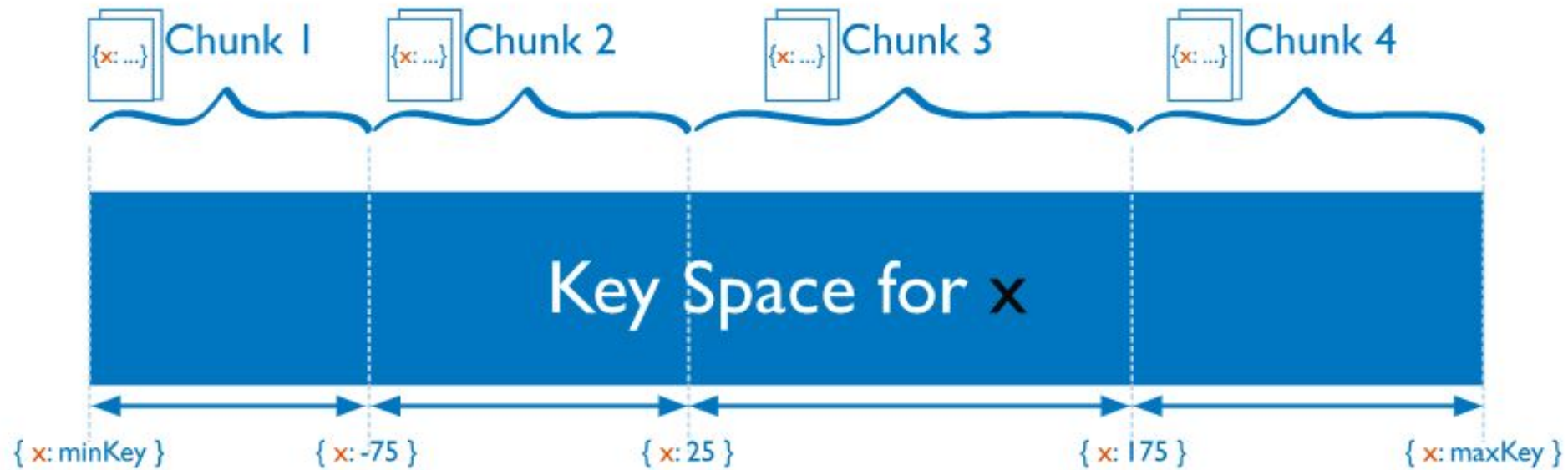


- ***Sharding - Shard Key***

- O *MongoDB*, particiona os dados da colecção utilizando intervalos dos valores da *shard key*.
- Cada intervalo é definido como um intervalo não sobreposto de valores da *shard key* e está associado a um fragmento (*chunk*).
- O *MongoDB* tenta distribuir os fragmentos/pedaços (*chunks*) de forma uniforme pelos *shards* do *cluster*.

UNIDADE CURRICULAR : SISTEMAS GESTORES DE BASES DE DADOS II

- **Sharding - Shard Key**



UNIDADE CURRICULAR : SISTEMAS GESTORES DE BASES DE DADOS II



- **Sharding - Shard Key**
 - Escolha da *shard key*
 - Afecta:
 - Desempenho
 - Eficiência
 - Escalabilidade
 - Um *cluster* com o melhor *hardware* e infraestrutura pode ficar comprometido pela escolha da *shard key*.
 - A escolha desta e a indexação pode também afectar a estratégia de fragmentação (*sharding*).

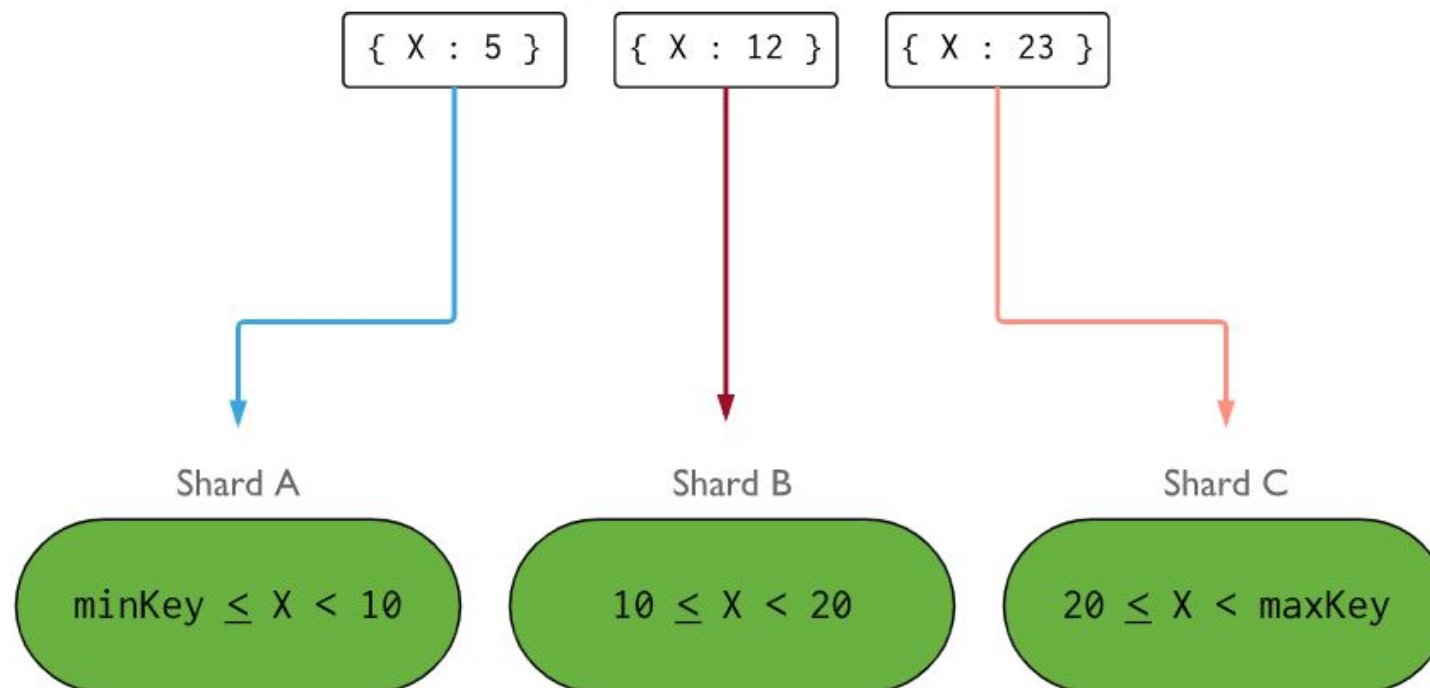
UNIDADE CURRICULAR : SISTEMAS GESTORES DE BASES DE DADOS II



- **Sharding - Shard Key**

- Escolha da *shard key*

- A *shard key* perfeita, permite que o *MongoDB* distribua documentos uniformemente em todo o *cluster*.

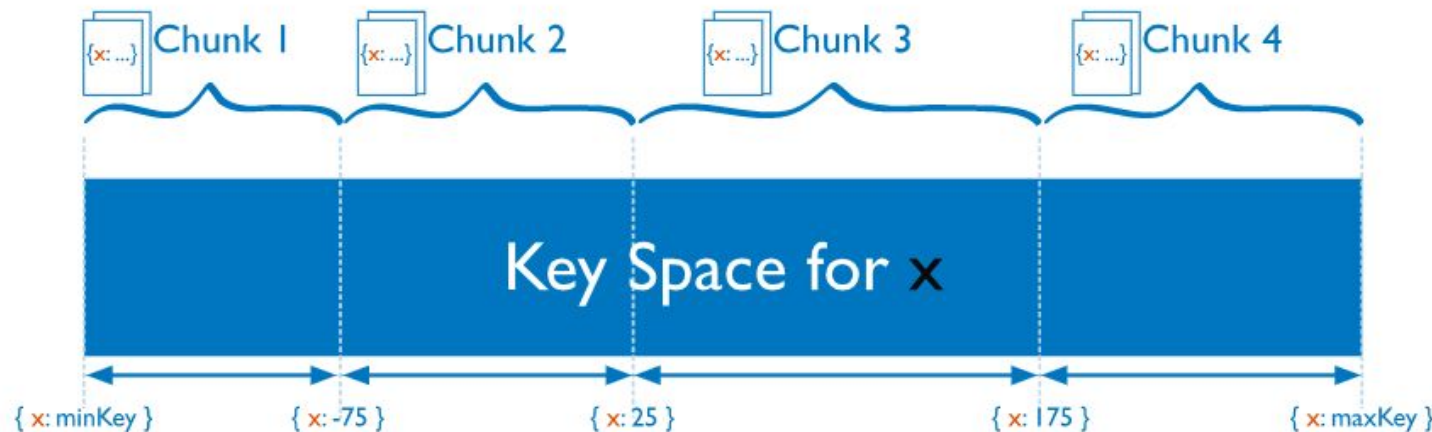


UNIDADE CURRICULAR : SISTEMAS GESTORES DE BASES DE DADOS II



- **Sharding - Chunks**

- O *MongoDB* fragmenta os dados em pedaços (*chunks*).
- Cada *chunk* tem um intervalo inclusivo inferior e exclusivo superior baseado na *shard key*.



UNIDADE CURRICULAR : SISTEMAS GESTORES DE BASES DE DADOS II



- **Sharding - Chunks**

- Os *mongos* encaminham as escritas para os *chunks* apropriadas com base no valor da *shard key*.
- O *MongoDB* divide os *chunks* quando estes ultrapassam o limite definido no *chunk size*. Inserções e actualizações podem despoletar uma divisão de um *chunk*.
- O intervalo mais pequeno que um *chunk* pode ter, é um único valor exclusivo de *shard key*. Um *chunk* que contém apenas documentos com um único valor de *shard key* não pode ser dividido.

UNIDADE CURRICULAR : SISTEMAS GESTORES DE BASES DE DADOS II



- ***Sharding - Chunks***

- **Tamanho do *chunk***

- O tamanho padrão é 64 megabytes.
- É possível aumentar ou diminuir este tamanho.
- *Chunks* mais pequenos permitem uma distribuição mais uniforme, em detrimento de migrações mais frequentes.
- *Chunks* maiores permitem menos migrações, em detrimento de uma distribuição mais uniforme.

UNIDADE CURRICULAR : SISTEMAS GESTORES DE BASES DE DADOS II



- **Sharding - Chunks**
 - **Migração de *chunks***
 - As migrações ocorrem para distribuir uniformemente os chunks pelo shards do cluster.
 - Duas formas:
 - manual → é realizado com comandos próprios para o efeito.
 - automática → o processo de balanceamento automaticamente migra os *chunks*, quando existe desequilíbrio na distribuição dos *chunks* pelos *shards*.

UNIDADE CURRICULAR : SISTEMAS GESTORES DE BASES DE DADOS II



- **Vantagens do *Sharding***

- **Leituras/Escritas**

- As operações de leitura e escrita são distribuídas uniformemente pelos vários *shards* do *cluster*.
- Permite cada *shard* processar uma subparte da operação.
- É possível fazer o escalonamento destas operações de leitura/escrita com a adição de mais *shards*.

UNIDADE CURRICULAR : SISTEMAS GESTORES DE BASES DE DADOS II



- **Vantagens do *Sharding***
 - **Capacidade de armazenamento**
 - O *sharding* permite distribuir os dados por vários *shards* do *cluster*.
 - Permite cada *shard* conter um subconjunto do total de dados existente no *cluster*.
 - Com o crescimento de dados, a adição de novos *shards* permite aumentar a capacidade de armazenamento do *cluster*.

UNIDADE CURRICULAR : SISTEMAS GESTORES DE BASES DE DADOS II



- **Vantagens do *Sharding***

- **Alta disponibilidade**

- Um *cluster sharded* pode continuar a realizar operações de leitura/escrita, mesmo que um ou mais *shards* estejam indisponíveis.
- Os *config servers* podem ser lançados como *replica sets*. O *cluster* pode continuar a processar as leituras e escritas desde que a maioria do *replica set* esteja disponível.
- Em ambientes de produção, os *shards* individuais devem ser “lançados” como *replica sets*, fornecendo redundância e disponibilidade.

UNIDADE CURRICULAR : SISTEMAS GESTORES DE BASES DE DADOS II



mongoDB®

- **Considerações antes de efectuar o *Sharding***

- Uma infraestrutura de *cluster sharded* requer um cuidado especial, no planeamento, execução e manutenção.
- Ter atenção a escolha da *shard key*, para assegurar o desempenho e eficiência.
- Não é possível tornar uma colecção *sharded*, numa não *sharded*.

UNIDADE CURRICULAR : SISTEMAS GESTORES DE BASES DE DADOS II



mongoDB®

- **Considerações antes de efectuar o *Sharding***
 - Não é possível escolher outra *shard key*, após a fragmentação.
 - Ter atenção às *queries* realizadas, que não utilizam a *shard key* ou o prefixo de um índice composto da *shard key*, dado que são realizadas consultas em todos os *shards* e estas podem demorar muito tempo.

UNIDADE CURRICULAR : SISTEMAS GESTORES DE BASES DE DADOS II



- **Interacções com um *cluster sharded***

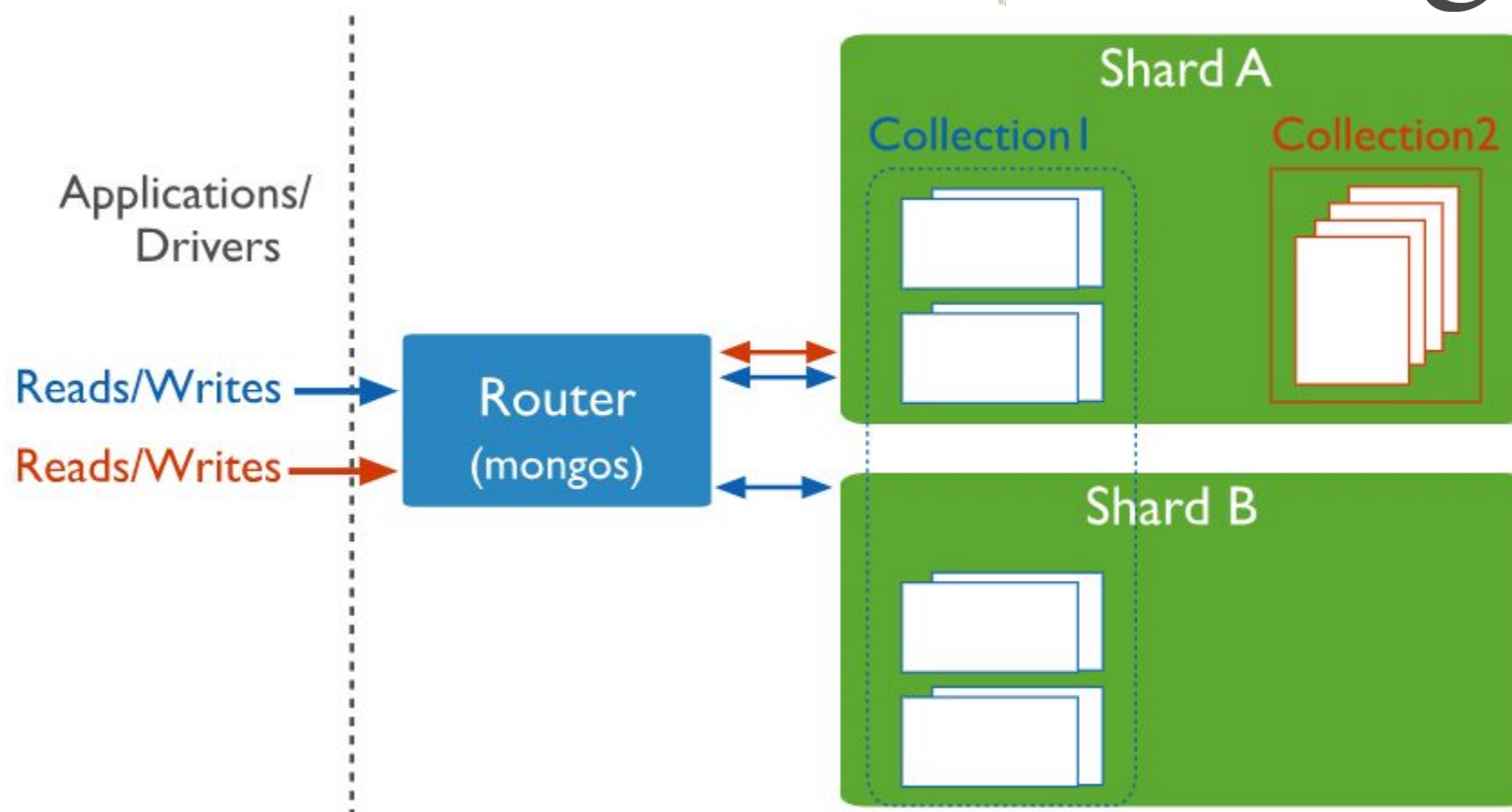
- A conexão deve ser feita ao *router mongos*, para interagir com qualquer colecção do *cluster sharded*.
- Isto inclui colecções *sharded* e não *sharded*.
- Os clientes nunca devem ligar-se a um único *shard*, para efectuar operações de leitura e escrita.

UNIDADE CURRICULAR : SISTEMAS GESTORES DE BASES DE DADOS II

- Interacções com um *cluster sharded*



mongoDB®



UNIDADE CURRICULAR : SISTEMAS GESTORES DE BASES DE DADOS II



- **Estratégias de *Sharding***

- O *MongoDB* suporta duas técnicas de *sharding*, para distribuir dados num *cluster sharded*.
- Estas duas técnicas são:
 - *Hashed Sharding*
 - *Ranged Sharding*

UNIDADE CURRICULAR : SISTEMAS GESTORES DE BASES DE DADOS II



- **Estratégias de *Sharding***

- *Hashed sharding*

- Esta técnica utiliza o índice *hashed*, para dividir os dados pelo *cluster sharded*.
- Os índices *hashed* computam o valor *hash* de um campo único, como o valor de índice.
- Este valor é utilizado como *shard key*.

UNIDADE CURRICULAR : SISTEMAS GESTORES DE BASES DE DADOS II



- **Estratégias de *Sharding***

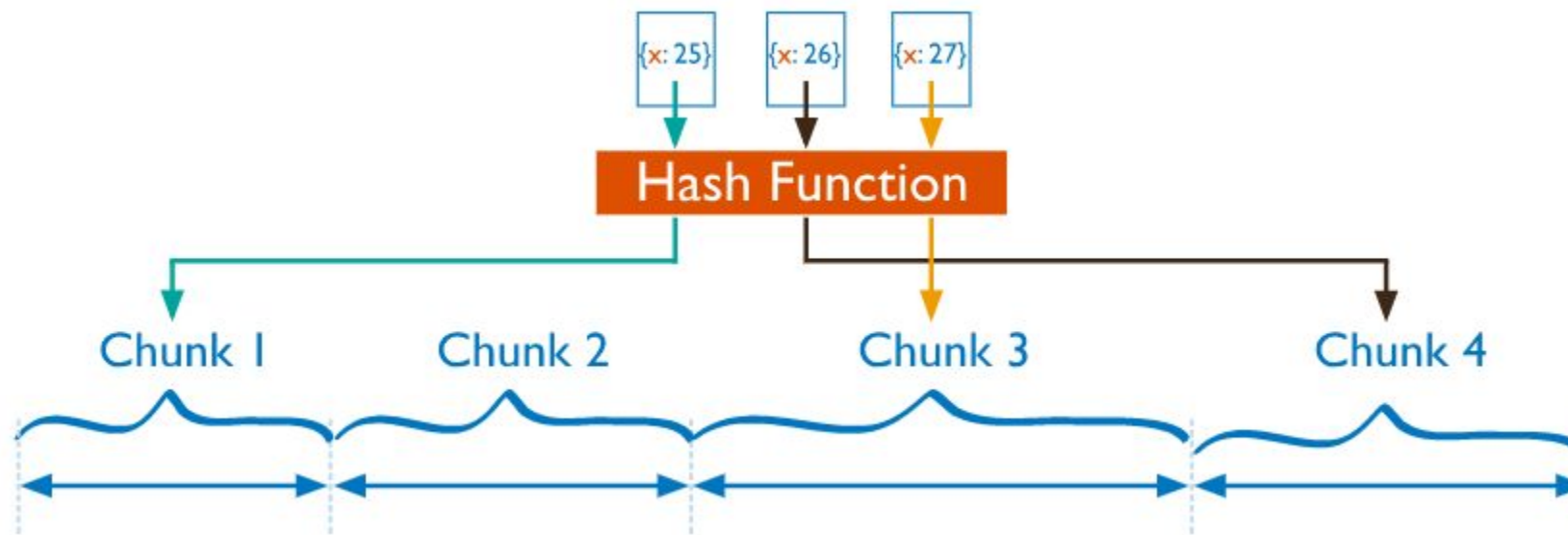
- *Hashed sharding*

- Os *chunks* são criados com esses valores *hash*, computados anteriormente.
- A cada *chunk* é associado um intervalo, baseado no valor da *shard key hashed*.

UNIDADE CURRICULAR : SISTEMAS GESTORES DE BASES DE DADOS II



- Estratégias de *Sharding*
 - *Hashed sharding*



UNIDADE CURRICULAR : SISTEMAS GESTORES DE BASES DE DADOS II



- **Estratégias de *Sharding***

- *Hashed sharding*

- Vantagens:

- fornece uma distribuição de dados mais equilibrada no *cluster sharded*.
 - os ***mongos*** podem direcionar *queries* com correspondências de igualdade para um único *shard*.

- Desvantagens:

- documentos com valores de *shard key* próximos são improváveis de estarem no mesmo *chunk* ou *shard*.
 - os ***mongos*** realizam operações de transmissão para *queries* que possuam intervalos, logo são operações ineficientes.

UNIDADE CURRICULAR : SISTEMAS GESTORES DE BASES DE DADOS II



- **Estratégias de *Sharding***

- *Hashed sharding*

- Utilizando o seguinte comando:

- `sh.shardCollection("database.collection", { <field> : "hashed" })`

- Seleccionado o campo que têm um índice *hashed*, que será usado como *shard key*.

UNIDADE CURRICULAR : SISTEMAS GESTORES DE BASES DE DADOS II



- **Estratégias de *Sharding***

- *Hashed sharding*

- Se a colecção não é vazia, é preciso criar o índice *hashed* para a *shard key* manualmente, com o comando `db.collection.createIndex()`, antes de utilizar o comando `shardCollection()`.
- Se a colecção é vazia, o *MongoDB* cria o índice *hashed* automaticamente ao executar o comando `sh.shardCollection()`.

UNIDADE CURRICULAR : SISTEMAS GESTORES DE BASES DE DADOS II



- **Estratégias de *Sharding***

- *Hashed sharding* - Exemplo

- Obter o valor *hashed* para uma determinada *shard key* de um documento.
- Usando o seguinte comando:
 - `convertShardKeyToHashed(<Object>)`
- Este comando utiliza a função de *hashing*, que é também utilizada na indexação *hashed*.

UNIDADE CURRICULAR : SISTEMAS GESTORES DE BASES DE DADOS II



- **Estratégias de *Sharding***

- *Hashed sharding* - Exemplo

- Considerando a seguinte colecção *sharded*, que utiliza o campo ***_id*** como *shard key*:

- use test
 - `db.orders.createIndex({ _id: "hashed" })`
 - `sh.shardCollection("test.orders", { _id : "hashed" })`

UNIDADE CURRICULAR : SISTEMAS GESTORES DE BASES DE DADOS II



- **Estratégias de *Sharding***

- *Hashed sharding* - Exemplo

- Se existir um documento deste género na colecção:

- {
 _id: ObjectId("5b2be413c06d924ab26ff9ca"),
 "item" : "Chocolates",
 "qty" : 25
}

UNIDADE CURRICULAR : SISTEMAS GESTORES DE BASES DE DADOS II



- **Estratégias de *Sharding***

- *Hashed sharding* - Exemplo

- Para determinar o valor *hash* para o campo ***_id*** desse documento, realizamos o seguinte comando:

- `convertShardKeyToHashed(ObjectId("5b2be413c06d924ab26ff9ca"))`

UNIDADE CURRICULAR : SISTEMAS GESTORES DE BASES DE DADOS II



- **Estratégias de *Sharding***

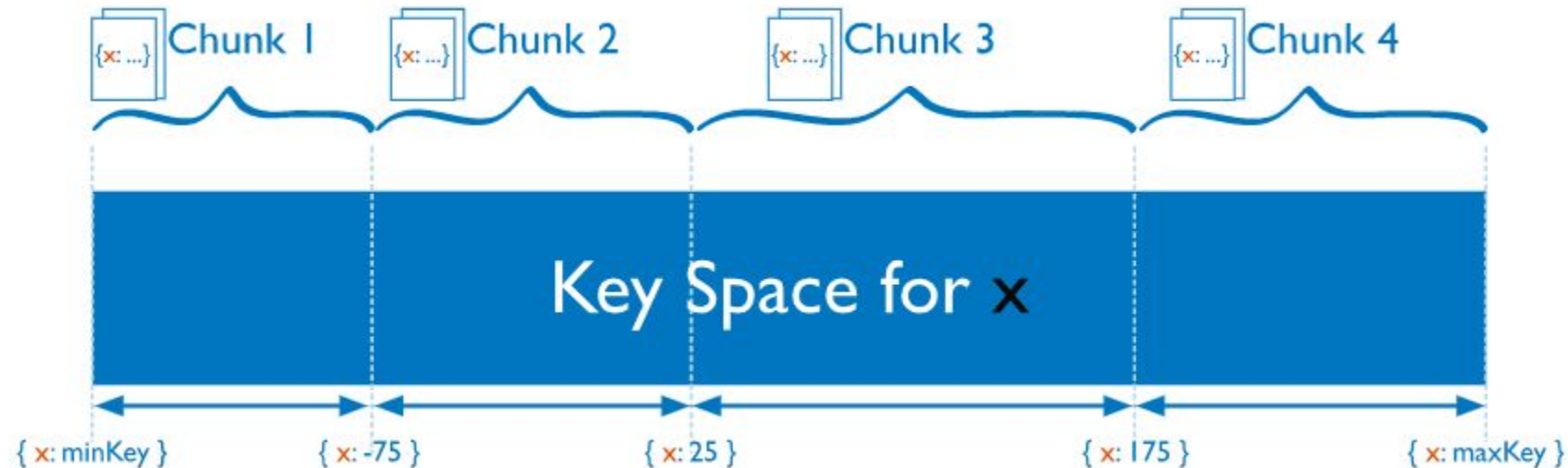
- *Ranged sharding*

- Envolve a divisão dos dados em intervalos contínuos, determinados pelos valores da *shard key*.
 - *Ranged sharding* é a metodologia *sharding* padrão.

UNIDADE CURRICULAR : SISTEMAS GESTORES DE BASES DE DADOS II



- Estratégias de *Sharding*
 - *Ranged sharding*



UNIDADE CURRICULAR : SISTEMAS GESTORES DE BASES DE DADOS II



- **Estratégias de *Sharding***

- *Ranged sharding*

- Vantagens:

- documentos com valores de *shard key* próximos são prováveis de estar no mesmo *chunk* ou *shard*.
- isto permite *queries* eficientes, em que as leituras são direccionadas a documentos dentro de um intervalo contínuo.

- Desvantagens:

- o desempenho da leitura e escrita pode diminuir com uma má escolha da *shard key*.

UNIDADE CURRICULAR : SISTEMAS GESTORES DE BASES DE DADOS II



- **Estratégias de *Sharding***

- *Ranged sharding*

- Utilizando o seguinte comando:

- `sh.shardCollection("database.collection", { <shard key> })`

- Seleccionado o campo que têm um índice único ou composto para usar como *shard key*.

UNIDADE CURRICULAR : SISTEMAS GESTORES DE BASES DE DADOS II



- **Estratégias de *Sharding***

- *Ranged sharding*

- Se a colecção não é vazia, é preciso criar o índice para a *shard key* manualmente, com o comando `db.collection.createIndex()`, antes de utilizar o comando `shardCollection()`.
- Se a colecção é vazia, o *MongoDB* cria o índice automaticamente ao executar o comando `sh.shardCollection()`.

UNIDADE CURRICULAR : SISTEMAS GESTORES DE BASES DE DADOS II

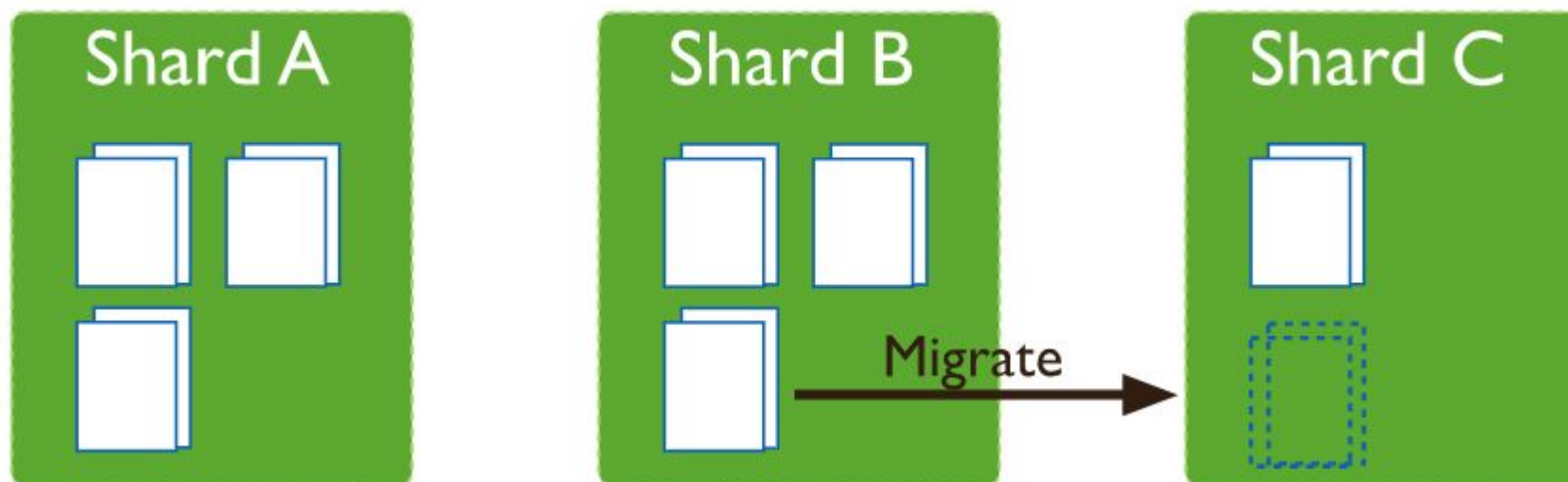


- **Balanceamento**

- É um processo que é executado no *background* e que monitoriza o número de fragmentos/pedaços (*chunks*) em cada *shard*.
- Este processo tenta automaticamente migrar os *chunks* entre os *shards*, obtendo um número igual de *chunks* por *shard*.
- Este procedimento é totalmente transparente em relação a camada de utilizador e aplicação, no entanto, podem ocorrer problemas no desempenho enquanto este procedimento está a ocorrer.

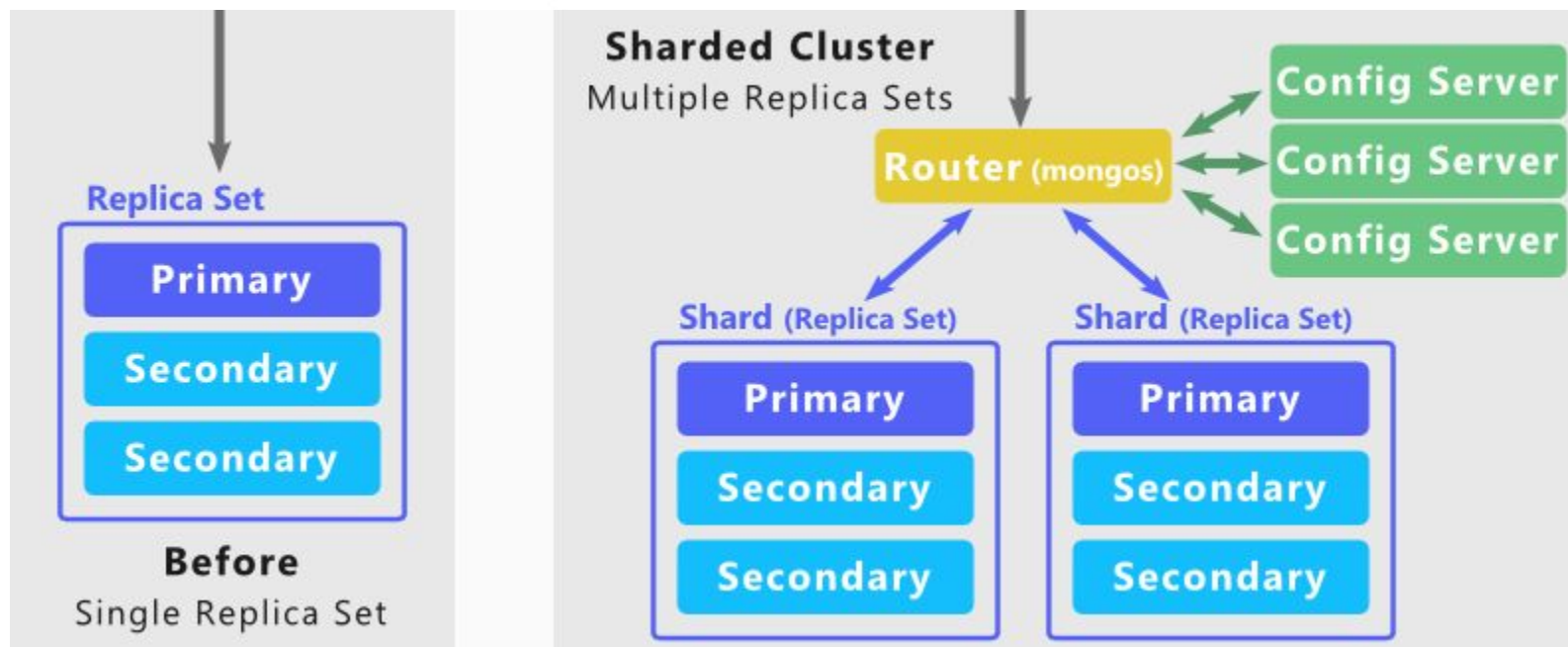
UNIDADE CURRICULAR : SISTEMAS GESTORES DE BASES DE DADOS II

- **Balanceamento**



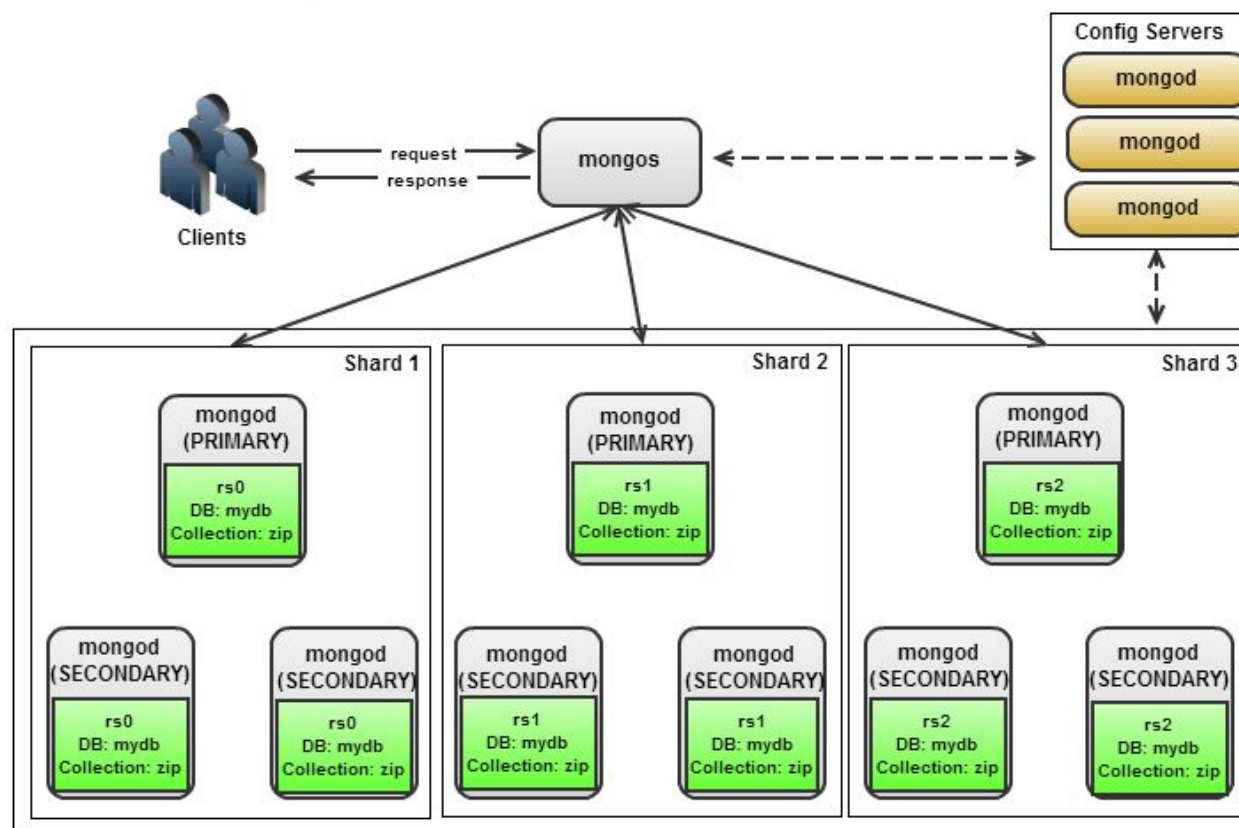
UNIDADE CURRICULAR : SISTEMAS GESTORES DE BASES DE DADOS II

- Exemplos de *cluster sharded*



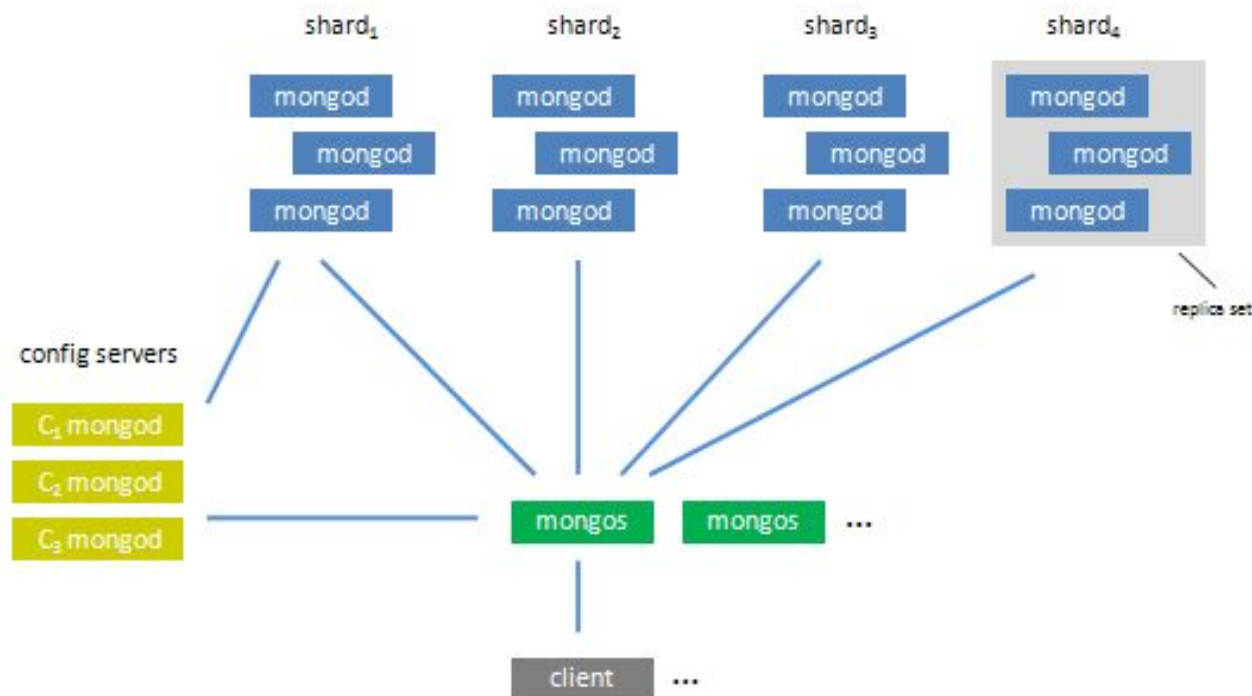
UNIDADE CURRICULAR : SISTEMAS GESTORES DE BASES DE DADOS II

- Exemplos de *cluster sharded*



UNIDADE CURRICULAR : SISTEMAS GESTORES DE BASES DE DADOS II

- Exemplos de *cluster sharded*



UNIDADE CURRICULAR : SISTEMAS GESTORES DE BASES DE DADOS II



- Referências

- <https://docs.mongodb.com/manual/reference/sharding/>
- <https://docs.mongodb.com/manual/sharding/>
- <https://docs.mongodb.com/manual/core/sharded-cluster-components/>
- <https://docs.mongodb.com/manual/core/sharding-shard-key/>
- <https://severalnines.com/blog/turning-mongodb-replica-set-sharded-cluster>

UNIDADE CURRICULAR : SISTEMAS GESTORES DE BASES DE DADOS II



- Referências (cont.)

- <https://severalnines.com/blog/turning-mongodb-replica-set-sharded-cluster>
- <https://blog.scottlogic.com/2014/08/08/sharded-clusters-mongodb-considerations.html>



CTeSP

CURSOS TÉCNICOS
SUPERIORES PROFISSIONAIS