



UNIVERSIDADE FEDERAL DE ITAJUBÁ

Banco de Dados II

COM 231

MongoDB
Aula 20

Vanessa Cristina Oliveira de Souza



UNIVERSIDADE FEDERAL DE ITAJUBÁ

PRÁTICA NO MONGODB AGGREGATION FRAMEWORK



Aggregation

- De que adianta conter muitos dados se não for possível extrair informação deles?
- É por esse motivo que agrupamos os dados conforme a necessidade para conseguir o detalhamento necessário, o que nos bancos relacionais normalmente é feito com o comando GROUP BY.
- *Aggregations* são operações que processam dados e retornam os resultados computados.
- MongoDB provê um rico conjunto de operações de agregação que examinam e realizam cálculos sobre os conjuntos de dados.



Aggregation

■ Modalidades

☐ Funções de Agregação Simples

- count, distinct, group

☐ Map Reduce

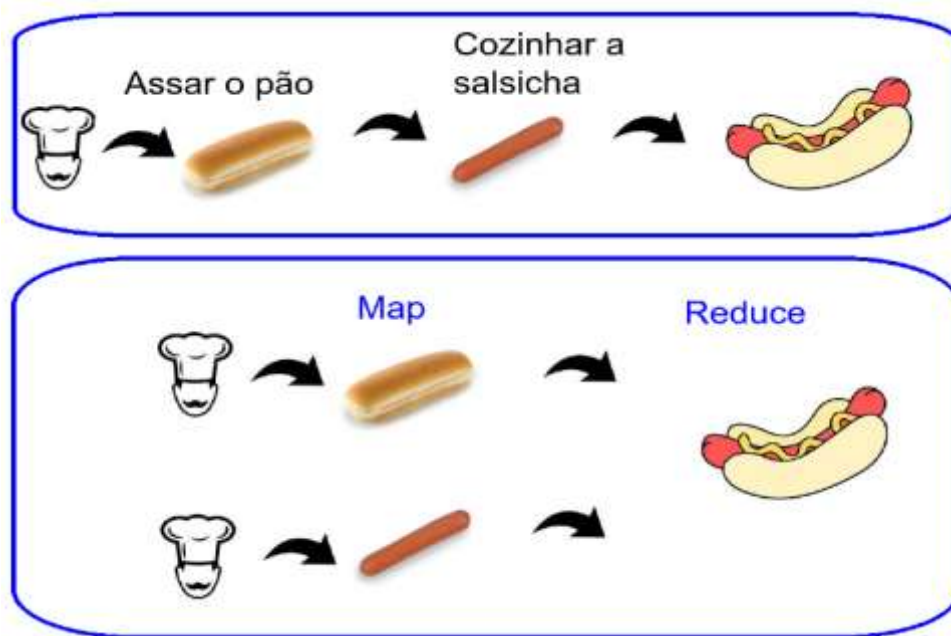
☐ Aggregation Pipelines

- Aggregation Framework
- Versão mais simples e com mais performance do que o tradicional ***map reduce***.



Map Reduce

■ Map-reduce



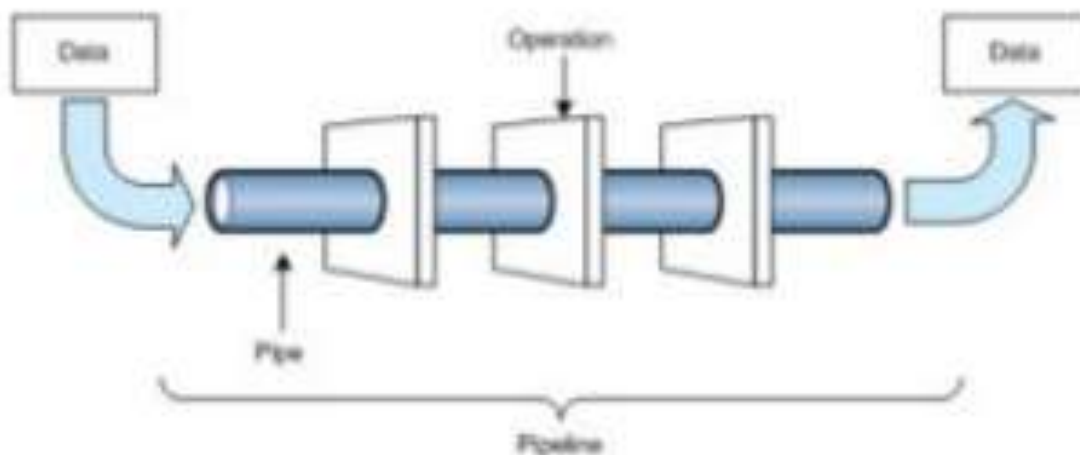
□ **Map** : função JS para extrair (chave, valor) de cada doc

□ **Reduce** : função JS para agregar os valores de cada chave



Aggregation Framework

- No MongoDB não existe apenas um comando semelhante ao Group By, existe na verdade algo bem mais robusto e completo chamado **aggregation framework** (framework de agrupamento).





Aggregation Framework

- O pipeline é dividido em estágios
- Um estágio realiza uma operação com cada documento que recebe, e pode gerar 0, 1 ou mais documentos para o próximo estágio.
- A sintaxe do ***aggregation framework*** é bem diferente do tradicional GROUP BY.
 - `db.collection.aggregate(pipeline, options)`



Aggregation Framework

- O pipeline é dividido em estágios
- Um estágio realiza uma operação com cada documento que recebe, e pode gerar 0, 1 ou mais documentos para o próximo estágio.
- A sintaxe do ***aggregation framework*** é bem diferente do tradicional GROUP BY.
 - `db.collection.aggregate(pipeline, options)`



Aggregation Framework

- Pipeline
 - Sequência de operações de agregação ou estágios.
- Operações
 - Projeção
 - \$project
 - Seleção
 - \$match, \$limit, \$skip
 - Agrupamento
 - \$group
 - Ordenação
 - \$sort
 - Explosão (produto cartesiano)
 - \$unwind



Aggregation Framework

- A operação **match** é equivalente ao *find*
- Filtra os documentos que correspondem à condição especificada, para seguir no pipeline.
 - Select * from bares where RendimentoAnual > 500000;

```
db.bares.aggregate([
  {$match :
    {$gt: "Rendimento_Anual" : 500000}
  }
])
```



Aggregation Framework

- A operação **project** realiza a projeção dos dados
 - Select _id, bairro from bares where RendimentoAnual > 500000;

```
db.bares.aggregate([  
  {$match : {"Rendimento Anual" : {$gt :500000}}},  
  {$project : {_id : 1, Bairro : 1}}  
])
```



Aggregation Framework

- A operação **sort** realiza a ordenação dos dados
 - 1 para crescente
 - -1 para decrescente
 - `Select id, Rendimento Anual from bares where RendimentoAnual > 500000 order by Rendimento Anual;`

```
db.bares.aggregate([
  {$match : {"Rendimento Anual" : {$gt :500000}}},
  {$sort : {"Rendimento Anual" : 1}},
  {$project : {_id : 1, "Rendimento Anual" : 1}}
])
```

A ORDEM NO PIPELINE PODE INTERFERIR NO RESULTADO!!!!



Aggregation Framework

- A operação **limit** limita a quantidade de documentos que vai passar para a próxima etapa do pipeline
 - Select id, Rendimento Anual from bares where RendimentoAnual > 500000 order by Rendimento Anual top1;

```
db.bares.aggregate([
  {$match : {"Rendimento Anual" : {$gt :500000}}},
  {$limit : 1},
  {$sort : {"Rendimento Anual" : 1}},
  {$project : {_id : 1, "Rendimento Anual" : 1}}
])
```

A ORDEM NO PIPELINE PODE INTERFERIR NO RESULTADO!!!!



Aggregation Framework

- A operação **group** agrupa os documentos por alguma expressão especificada
 - Qual a média do Rendimento anual dos bares?
 - `Select mean(Rendimento_Anual) from bares`

```
db.bar.aggregate([
  {$group : {_id:null, media:{$avg:"$Rendimento Anual"}}},
  {$project : {_id : 0, media : 1}}
])
```



Aggregation Framework

■ Options

- Propriedades opcionais para o comando de agregação
- Explain
 - Retorna um documento com informações detalhadas do processamento executado
 -
- allowDiskUse
 - Permite escrever em arquivos temporários
- cursor



Aggregation Framework

- A operação **explain** retorna a forma como o pipeline foi processado
 - `Select id, Rendimento Anual from bares where RendimentoAnual > 500000 order by Rendimento Anual top1;`

```
db.bares.aggregate([
  {$match : {"Rendimento Anual" : {$gt :500000}}},
  {$limit : 1},
  {$sort : {"Rendimento Anual" : 1}},
  {$project : {_id : 1, "Rendimento Anual" : 1}}
],
{explain : 1}
)
```




Aggregation Framework

- Retornar os frequentadores de bares que moram no centro.

```
db.bares.find(  
  {"Frequentadores.Bairro" : "Centro"},  
  {"Frequentadores.nomeFrequentador" : true})
```



```
db.bares.aggregate([  
  {$unwind : "$Frequentadores"},  
  {$match : {"Frequentadores.Bairro" : "Centro"}},  
  {$group : {_id : "$_id", Frequentadores:{$push : "$Frequentadores.nomeFrequentador"}}}  
])
```



Índice no MongoDB

- O mongoDB aceita a criação em qualquer um dos campos do documento.
 - Inclusive nos arrays de objeto

```
db.bar.find({"Frequentadores.nomeFrequentador" : "Karen"}).explain()
```

```
db.bar.createIndex({"Frequentadores.nomeFrequentador" : 1})
```



Referências

- NoSQL – um guia conciso para o Mundo Emergente da Persistência Poliglota.
 - Sadalage & Fowler
 - Editora Novatec
 - 2013

- MongoDB. Construa novas aplicações com novas tecnologias
 - Fernando Boaglio
 - Casa do Código

- MongoDB Manual 3.0
 - <http://docs.mongodb.org/manual/reference/>