

# Spring Data com MongoDB

Fabiano Guizellini Modos

twitter: @fmodos

stackoverflow.com: fmodos

github.com/fmodos



# Agenda

- NOSQL vs SQL
- MongoDB
- MongoDB Java API
- Spring Data MongoDB



SQL: Estrutura e tipo de dados são fixos

NOSQL: Estrutura dinâmica e varios modelos de dados.



# NoSQL - Modelo de Dados

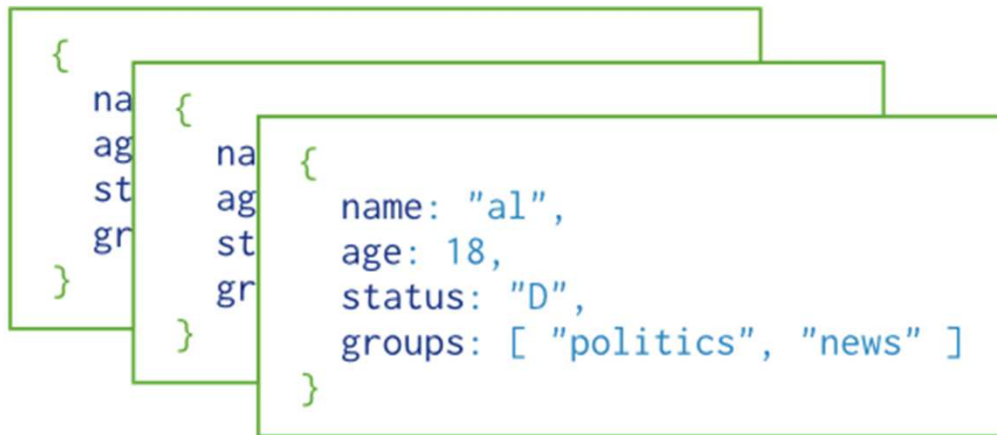
- Document – orientado a JSON (MongoDB)
- Graph – orientado a grafos (Neo4j)
- Key-Value – chave-valor (Redis)
- Wide-columns – orientado a colunas (Cassandra)

# MongoDB

JSON ->

```
{  
  name: "sue",  
  age: 26,  
  status: "A",  
  groups: [ "news", "sports" ]  
}
```

← field: value  
← field: value  
← field: value  
← field: value



Collection

# MongoDB

Collection  
↓  
db.users.insert(  
Document  
↓  
{  
 name: "sue",  
 age: 26,  
 status: "A",  
 groups: [ "news", "sports" ]  
}

Document

```
{  
  name: "sue",  
  age: 26,  
  status: "A",  
  groups: [ "news", "sports" ]  
}
```

insert

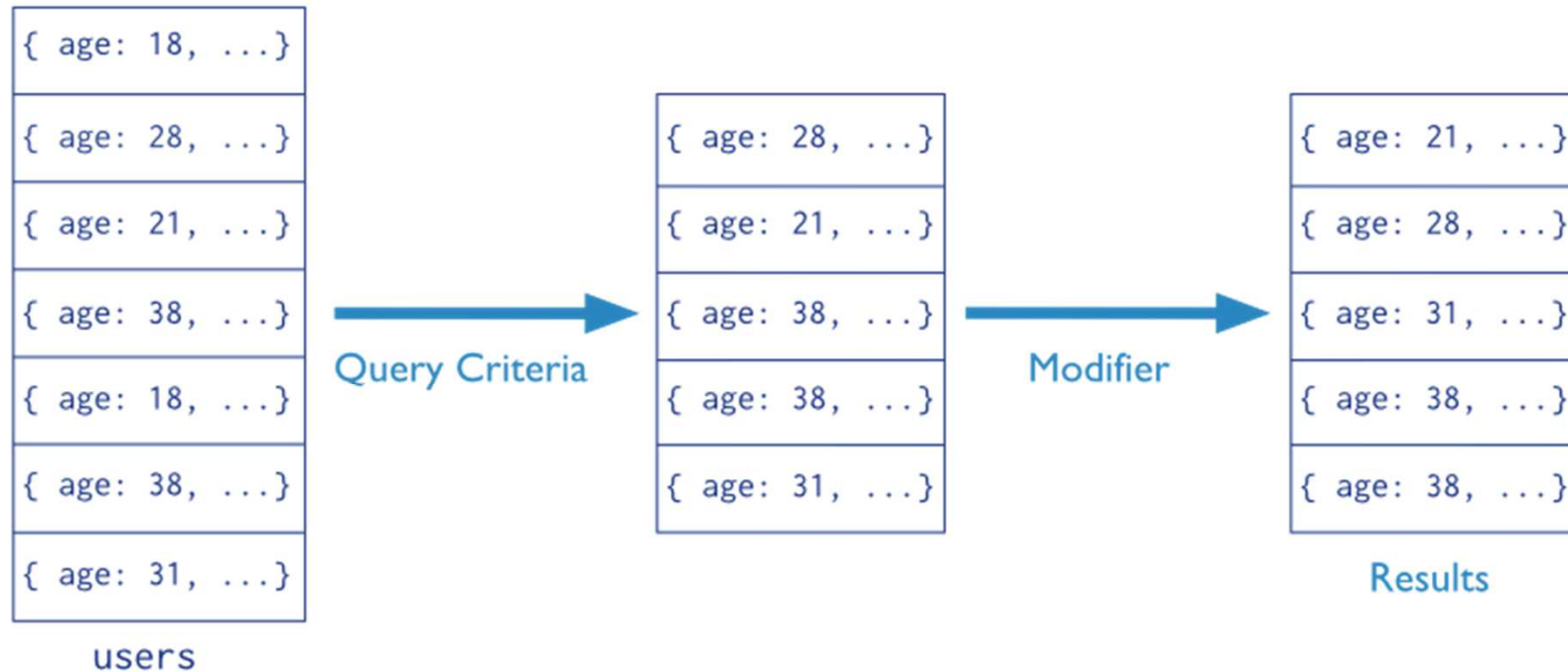
Collection

{ name: "al", age: 18, ... }
{ name: "lee", age: 28, ... }
{ name: "jan", age: 21, ... }
{ name: "kai", age: 38, ... }
{ name: "sam", age: 18, ... }
{ name: "mel", age: 38, ... }
{ name: "ryan", age: 31, ... }
{ name: "sue", age: 26, ... }

users

# MongoDB

Collection                      Query Criteria                      Modifier  
`db.users.find( { age: { $gt: 18 } } ).sort( {age: 1 } )`





# Java MongoDB API

- MongoClient
- DB
- DBCollection
- DBObject
- DBCursor

```
BasicDBObject dbobject = new BasicDBObject("nome",  
"Joel").append("sobrenome", "Santana")
```

```
....
```

```
dbobject.get("nome")
```



# Spring Data

“This is an umbrella project which contains many subprojects that are specific to a given database”

- MongoDB
- Neo4j
- Cassandra
- JDBC Extensions
- Redis
- Etc...

# Spring Data MongoDB

“The Spring Data MongoDB project provides integration with the MongoDB document database.”

- Spring configuration support using Java based @Configuration classes or an XML namespace
- MongoTemplate helper class
- Java based Query, Criteria, and Update DSLs
- GeoSpatial integration
- Etc...

# Configuração

Dependência Maven -6

```
<dependency6
  <groupId6org.springframework.data</groupId6
  <artifactId6spring-data-mongodb</artifactId6
  <version61.6.0.RELEASE</version6
</dependency6
```

Instanciando um MongoClient:

```
MongoTemplate template = new MongoClient(new SimpleMongoDbFactory(new
MongoClient(new ServerAddress("localhost")), "database"))
```

SpringContext.xml:

```
<mongo:mongo host="localhost" port="27017"/>
<bean id="mongoTemplate"
class="org.springframework.data.mongodb.core.MongoTemplate">
  <constructor-arg ref="mongo"/>
  <constructor-arg name="databaseName" value="database"/>
</bean>
```

# MongoTemplate operations

- save - salva o objeto
- insert - insere o objeto, caso não exista
- insertAll - insere uma lista de objetos
- updateFirst - atualiza o primeiro registro de uma query
- updateMulti - atualiza todos registros de uma query

# Update methods

- addToSet - adiciona no array
- inc - soma o numero
- pop - remove o primeiro item do array
- pull - remove o valor do array
- pullAll - remove os valores do array
- push - adiciona no array
- rename - renomeia o campo
- set - seta um valor
- unset - limpar o valor

# MongoTemplate operations

- `upsert` - atualiza ou insere o registro da query
- `findAndModify` – encontra e altera o registro da query
- `Remove` – remove o objeto



# MongoTemplate operations

- findAll – retorna todos objetos de uma classe
- findOne – retorna um objeto da query
- findById – retorna o objeto desse id
- find – retorna todos objetos da query
- findAndRemove – remove e retorna o primeiro objeto da query

# Criteria methods

- and – operador ‘e’
- exists –se campo existe
- gt, gte – maior que, maior igual que
- lt, lte – menor que, menor igual que
- is – operador ‘=’
- in – valor esta na lista
- not – operador de negação
- regex – expressão regular
- size – tamanho do array

# GeoSpatial Query

- withinCenter - na area de um **Circle**
- withinCenterSphere - na are de uma **Circle**
- withinBox – na area de uma **Box**
- near – perto de um **Point**
- nearSphere - perto de um **Circle**
- maxDistance – com a distância maxima de

# Exemplos - GeoSpatial

```
private double[] location;
```

## 1. Buscar por uma area

```
Box box = new Box(new Point(-73.99756, 40.73083), new Point(-73.988135,  
40.741404));  
List<Player6 players =  
    template.find(new Query(Criteria.where("location").withinBox(box)), Player.class);
```

## 2. Busca por uma distancia maxima

```
Point location = new Point(-73.99171, 40.738868);  
NearQuery query = NearQuery.near(location).maxDistance(new Distance(10,  
Metrics.MILES));  
  
GeoResults<GuriaSolteira6 = operations.geoNear(query, GuriaSolteira.class);
```

# Full Text Query

- Busca por texto em mais de uma propriedade de um documento
- `@TextIndexed(weight=1)`
- `Query q = TextQuery.searching(new TextCriteria().matching("java"));`

# MapReduce

- Processamento paralelo de grande massa de dados

Código Java para iniciar processo

```
mongoOperations.mapReduce("ideias", "classpath:map.js", "classpath:reduce.js",  
options().outputCollection("ranking"), Object.class);
```

map.js

```
var map = function() {  
    emit(1, {  
        curtidas: this.Curtidas.length,  
        visualizacoes: this.Visualizacoes,  
        novas: this.Situacao == 0 ? 1 : 0  
    });  
}
```

reduce.js

```
var reduce = function(key, values) {  
    var ret = {};  
    for (var idx = 0; idx < values.length; idx++) {  
        ret.curtidas += values[idx].curtidas;  
        ret.visualizacoes +=  
values[idx].visualizacoes;  
        ret += values[idx].novas;  
    }  
    return ret; }
```

# Referências

- <http://www.mongodb.org/>
- <http://docs.mongodb.org/ecosystem/drivers/java/>
- <http://projects.spring.io/spring-data/>
- <http://docs.spring.io/spring-data/mongodb/docs/current/reference/html/>
- <http://kkovacs.eu/cassandra-vs-mongodb-vs-couchdb-vs-redis>
- [http://info.mongodb.com/rs/mongodb/images/10gen\\_Top\\_5\\_NoSQL\\_Considerations.pdf?\\_ga=1.135548657.110404807.1413325853](http://info.mongodb.com/rs/mongodb/images/10gen_Top_5_NoSQL_Considerations.pdf?_ga=1.135548657.110404807.1413325853)

**OBRIGADO**  
**THAT'S ALL FOLKS!!!**