

```
{  
  "title": "Geolocalização com MongoDB",  
  "speakers": [  
    "Venilton Falvo Jr",  
    "Guilherme Bueno"  
  ],  
  "place": {  
    "lat": -21.978419,  
    "long": -47.8833706,  
    "name": "IFSP São Carlos"  
  }  
}
```

Sumário

- NoSQL
- MongoDB
- Geo-localização

NoSQL

1. Não relacional
2. Distribuído
3. Escala horizontalmente
4. API simples para se trabalhar e que suporte bem uma base com muitos dados.

NoSQL (Schemaless)

```
[{  
  "name": "Guilherme Bueno",  
  "age": 20,  
  "emails": [  
    "bueno.guis@gmail.com",  
    "guilherme.bueno@live.com"  
  ]  
},  
{  
  "name": "Venilton Falvo Jr.",  
  "emails": [  
    "falvojr@gmail.com"  
  ]  
}]
```

NoSQL (núcleos)

1. wide column store
2. document store
3. key value store
4. graph database
5. outros tipos



MongoDB

MongoDB

É um banco de dados não relacional baseado em documentos cujos princípios são alta performance e alta disponibilidade.

MongoDB (BSON)

```
{
  "_id" : ObjectId("5627d2b85db9a6a1512df125"),
  "loc" : {
  },
  "name" : "Central Park",
  "category" : "Parks"
}
{
  "_id" : ObjectId("5627d2c55db9a6a1512df126"),
  "loc" : {
  },
  "name" : "La Guardia Airport",
  "category" : "Airport"
}
```


MongoDB (Embedded Docs)

```
{
  "_id" : ObjectId("5627d2b85db9a6a1512df126"),
  "name" : "Guilherme",
  "tweets": [
    {
      "message": "Olá denovo!",
      "date": 1445462927679
    },
    {
      "message": "Primeiro tweet!",
      "date": 1445462907427
    }
  ]
}
```

MongoDB (escalar)

Relacional



Mais hardware para
escalar verticalmente

Não Relacional



Mais núcleos processando,
distribuição do banco

MongoDB (Leitura)

```
db.users.find(  
  { age: { $gt: 18 } },  
  { name: 1, address: 0 }  
) .limit(5)
```

collection
query criteria
projection
cursor modifier

```
SELECT _id, name  
FROM users  
WHERE age > 18  
LIMIT 5;
```

projection
table
select criteria
cursor modifier

MongoDB (Leitura)

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

MongoDB (Leitura)

```
{
  "_id" : ObjectId("562808b6c1a12507b96a5a6b"),
  "name" : "admin",
  "age" : 20,
  "status" : "B"
}
```

sers.find(collection
age: { \$gt: 18 } }, query criteria

MongoDB (Escrita)

```
db.users.insert(                                     collection
{
  name: "admin",
  passwd: "cGFzc3dk"                                documents
  status: "B"
})
```

```
INSERT INTO users                                     table
      (name, passwd, status)                         columns
VALUES ("admin", "cGFzc3dk", "B");                  values
```

MongoDB (Escrita)

```
db.users.update(  
  { age: { $gt: 18 } },  
  { $set: { status: "A" } },  
  { multi: true }  
)
```

b.users.remove() (criteria da mesma forma que com o find)

Geo-localização

- Index geo espacial
- Queries geo espaciais

Geo-localização (Index)

```
db.places.insert({  
  loc : {  
    type: "Point",  
    coordinates: [ -21.978419, -47.8833706 ]  
  },  
  name: "IFSP",  
  category : "Education"  
})
```

```
db.places.createIndex( { loc: "2dsphere" } )
```

Geo-localização (Queries)

```
db.<collection>.find(  
  { <location field> :  
    { $geoWithin :  
      { $centerSphere : [ [ <x>, <y> ] , <radius> ]  
    } } } )
```

Geo-localização (Queries)

```
db.places.find(  
  { loc: {  
    $geoIntersects: {  
      $geometry: {  
        type: "Polygon" ,  
        coordinates: [  
          [ [ 0, 0 ], [ 3, 6 ], [ 6, 1 ], [ 0, 0 ] ]  
        ]  
      }  
    }  
  } } )
```

Geo-localização (Queries)

```
db.<collection>.find( { <location field> :  
  { $near :  
    { $geometry :  
      { type : "Point" ,  
        coordinates : [ <longitude> , <latitude> ] } ,  
      $maxDistance : <distance in meters> ,  
      $minDistance: <distance in meters>  
    } } } )
```

Coding...

Referencias

<http://nosql-database.org/>

<https://docs.mongodb.org/>

<https://university.mongodb.com/>

<https://docs.mongodb.org/manual/applications/geospatial-indexes/>

Obrigado!

<https://github.com/falvojr/ext-mongodb-geo>