



Banco de dados NoSQL - Georreferenciamento

Prof. Gustavo Leitão

Utilizando coordenadas

- O MongoDB é otimizado para lidar com coordenadas geográficas
- Para aproveitar seus ferramentas, é importante que as coordenadas estejam no formato GeoJSON

```
<field>: { type: <GeoJSON type> , coordinates: <coordinates> }
```

copy

For example, to specify a [GeoJSON Point](#):

```
location: {  
  type: "Point",  
  coordinates: [-73.856077, 40.848447]  
}
```

copy

Utilizando coordenadas

- Há diversos tipos possíveis:
 - Point
 - LineString
 - Polygon
 - Multipoint
 - MultilineString
 - MultiPolygon
 - GeometryCollection

Utilizando coordenadas

- Para utilizar as ferramentas de consulta do mongo, é necessário criar um índice especial. São eles:
 - 2dsphere index
 - 2d index (legacy)

2d sphere

- Um índice 2dsphere suporta consultas que calculam geometrias em uma **esfera semelhante à terra**.
- O índice 2dsphere suporta todas as consultas geoespaciais do MongoDB: consultas de inclusão, interseção e proximidade.

Create a 2dsphere Index

To create a **2dsphere** index, use the `db.collection.createIndex()` method and specify the string literal `"2dsphere"` as the index type:

```
db.collection.createIndex( { <location field> : "2dsphere" } )
```

copy

Exemplo

copy

```
db.places.insert(  
  {  
    loc : { type: "Point", coordinates: [ -73.97, 40.77 ] },  
    name: "Central Park",  
    category : "Parks"  
  }  
)
```

```
db.places.insert(  
  {  
    loc : { type: "Point", coordinates: [ -73.88, 40.78 ] },  
    name: "La Guardia Airport",  
    category : "Airport"  
  }  
)
```

Exemplo

Create a **2dsphere** Index

The following operation creates a **2dsphere** index on the location field **loc**:

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

[copy](#)

Exemplo

- É possível criar um índice composto onde um dos campos seja uma coordenada.

```
db.places.createIndex( { loc : "2dsphere" , category : -1, name: 1 } )
```

copy

Unlike the [2d](#) index, a compound **2dsphere** index does not require the location field to be the first field indexed.
For example:

```
db.places.createIndex( { category : 1 , loc : "2dsphere" } )
```

copy

Exemplo

- **\$geoWithin** - Busca por pontos que estejam dentro do polígono passado.

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

[copy](#)

Exemplo

- **\$geoWithin** - O exemplo a seguir consulta as coordenadas e retorna todos os documentos em um raio de 10 milhas da coordenada [-88, 30]. O exemplo converte a distância, 10 milhas, em radianos, dividindo pelo raio equatorial aproximado da terra, 3963,2:

```
db.places.find( { loc :  
    { $geoWithin :  
        { $centerSphere :  
            [ [ -88 , 30 ] , 10 / 3963.2 ]  
        }  
    }  
} )
```

[copy](#)

Exemplo

- **\$near** - Retorna os pontos mais próximos de um ponto definido (ordenados do mais próximo para o mais distante).

```
db.places.find(  
  {  
    loc:  
      { $near :  
        {  
          $geometry: { type: "Point", coordinates: [ -73.9667, 40.78 ] },  
          $minDistance: 1000,  
          $maxDistance: 5000  
        }  
      }  
    }  
  )
```

Distâncias definida em metros

Exemplo

- **\$nearSphere** - Retorna os pontos mais próximos de um ponto definido (ordenados do mais próximo para o mais distante). Calcula a distância utilizando geometria esférica.

```
db.places.find(  
  {  
    loc:  
      { $near :  
        {  
          $geometry: { type: "Point", coordinates: [ -73.9667, 40.78 ] },  
          $minDistance: 1000,  
          $maxDistance: 5000  
        }  
      }  
    }  
  )
```

Distâncias definida em metros

Exercício

- Crie um programa e siga os seguintes passos:
 - Converta a coordenada legadas da base de dados dos restaurantes para padrão GeoJSON
 - Crie um índice 2dsphere para o novo campo
 - Liste os restaurantes distantes até 1 quilômetro do Port Authority Bus Terminal (NY)



Banco de dados NoSQL - Georreferenciamento

Prof. Gustavo Leitão