UNIDADE 2 Tipos de Dados e Meios de Armazenamento (Complemento)

Disciplina: Tópicos Especiais III (DEC7553)

Prof. Alexandre L. Gonçalves

E-mail: a.l.goncalves@ufsc.br



- Termo genérico que define bancos de dados não-relacionais (Not Onty SQL);
- O termo NoSQL foi inicialmente cunhado em 1998;
- Segundo Carlo Strozzi (criador do Strozzi NoSQL) o movimento NoSQL "é completamente distinto do modelo relacional e, portanto, deveria ser mais apropriadamente chamado de "NoREL" ou algo que produzisse o mesmo efeito".



- Com a popularização da internet, novos dados surgiram e o tratamento desses dados se tornou mais complexo e caro;
- Tecnologia iniciada por companhias voltadas à internet (Google[®], Facebook[®], Amazon[®], Linkedin[®]) com o intuito de superar as limitações de banco de dados relacionais para aplicações web modernas;



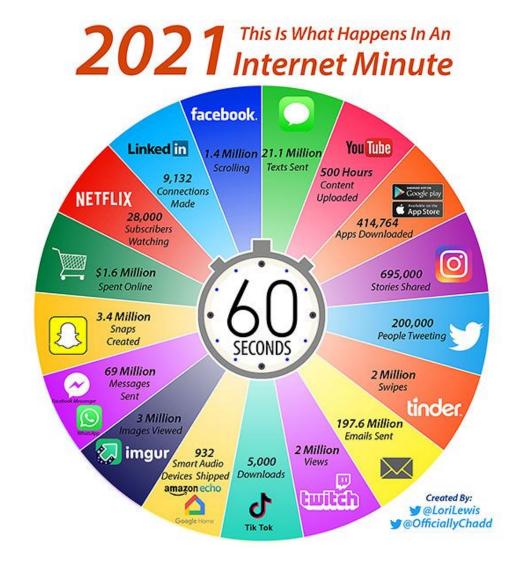
- Em 2006, o artigo "BigTable: A Distributed Storage System for Structured Data", publicado pelo Google[®] faz ressurgir o conceito de NoSQL;
- Outros eventos ocorrem nos anos seguintes e catapultaram esta tecnologia;
- Desde então, esta classe de banco de dados passou a ser chamada de NoSQL e vem se tornando cada vez mais popular;



- Tecnologia motivada por quatro tendências tecnológicas:
 - Big Users
 - Big Data
 - Internet of Things
 - Cloud Computing



Um minuto na Internet



Fonte:

e

https://www.allaccess.c om/merge/archive/3297 2/infographic-what-happ ens-in-an-internet-minut

6



Características

- Utilização de processamento paralelo:
 - Divide determinada tarefa em várias outras menores distribuindo-as entre os vários processadores disponíveis;
 - Pode utilizar muitos processadores baratos tornando-se uma solução economicamente interessante;
 - Permite escalar o sistema horizontalmente evitando limitar a organização à poucos fornecedores de hardware mais poderoso.



Características

- Distribuição em escala global:
 - A distribuição, combinada com *hardware* de baixo custo, demanda que sistemas desta natureza sejam robustos para tolerar falhas de infraestrutura;
 - A partir dessas necessidades várias alternativas de bancos de dados não relacionais surgiram, entre elas:
 - Esquema Chave/Valor
 - Orientado a Documentos
 - Orientado a Colunas
 - Baseado em Grafos



Ranking de Popularidade

- DB-Engines
 - https://db-engines.com/en/ranking_trend



MongoDB[®]

- Banco de Dados orientado a documentos multiplataforma;
- Utiliza documentos no estilo JSON com esquema;
- O projeto iniciou em 2007 pela 10gen[®] e a primeira versão foi liberada em 2009. Em 2013 a 10gen[®] mudou de nome para MongoDB Inc[®].



MongoDB® – Principais Características

- Consultas Ad hoc
- Indexação de conteúdo
- Replicação
- Balanceamento de carga
- Pode ser utilizado como um sistema de arquivo
- Funções avançadas de agregação de tarefas, tais como Map-reduce
- Suporte à transações no modelo ACID



- MongoDB[®] Formas de Acesso
 - Mongo Shell
 - Interface Gráfica

Linguagem de Programação



- Acesso à aplicação mongo.exe (disponível no diretório bin);
- Para apresentar as bases de dados disponíveis:
 - show databases
- Para utilizar ou criar a base de dados:
 - use <database>
- Para apresentar a base de dados em uso:
 - db
- Para eliminar a base de dados em uso:
 - db.dropDatabase()
- Para criar uma collection:
 - db.createCollection("nome da collection")
- Para apresentar as coleções de uma base de dados:
 - show collections
- Para eliminar uma coleção da base de dados:
 - db.nome da coleção.drop()

https://docs.mongodb.com/manual/mongo/



Inserção de Documentos

```
db.clientes.insertOne({ _id: 1, nome: "cliente 1", idade: 29 , estado_civil:"C"})
db.clientes.insertOne({ _id: 2, nome: "cliente 2", idade: 40, email:
   "cliente_2@email.com", estado_civil:"C" })

db.clientes.insertMany([
        { _id: 3, nome: "cliente 3", idade: 21, estado_civil: "S" },
        { _id: 4, nome: "cliente 4", idade: 49, email: "cliente_4@email.com",
        estado_civil:"C" }
])
```

Obs: por padrão o campo _id é gerado pelo MongoDB



Consulta de Documentos

```
db.clientes.find()
db.clientes.find().pretty()
db.clientes.find( {idade: {$lte: 30} })
db.clientes.find({ idade: { $in: [ 21, 50] } })
db.clientes.find({ estado civil: "C", idade: { $gt: 25} })
db.clientes.find({ $or: [ {estado_civil: "S" }, {idade: { $gt: 30} } ] } ).sort({idade: -1})
db.clientes.find({ $or: [ {estado civil: "S" }, {idade: { $gt: 30} } ] }, { id: 0, nome: 1,
idade: 1} ).sort({idade: -1})
db.clientes.find( {
   estado civil: "C",
   $or: [ { idade: { $gt: 40 } }, { nome: /^cliente 1/ } ]
})
```



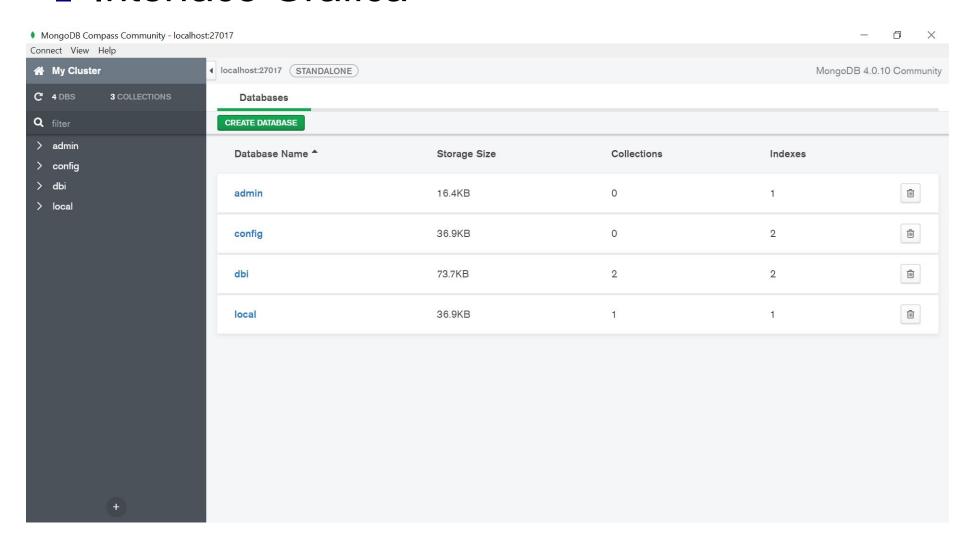
Atualização de Documentos



Eliminação de Documentos

```
db.clientes.deleteOne( { _id: 3 } )
db.clientes.remove( { _id: 3 } )
db.clientes.deleteOne( { estado_civil: "C" } )
db.clientes.deleteMany( { estado_civil: "C" } )
db.clientes.drop()
```

Interface Gráfica





Linguagem de Programação

- Várias linguagens possuem drivers de acesso ao MongoDB, entre elas:
 - C, C++, C#, Go, Java, Node.js, Perl, PHP, Python, Ruby, Scala



- Acesso à aplicação mongo.exe (disponível no diretório bin);
- Para apresentar as bases de dados disponíveis:
 - show databases
- Para utilizar ou criar a base de dados:
 - use ds_database
- Para apresentar a base de dados em uso:
 - db
- Para criar uma collection:
 - db.createCollection("textual_search")
- Para apresentar as coleções de uma base de dados:
 - show collections



Inserção de dados

```
db.stores.insertOne({ id: 1, title: "Segmented drain systems", abstract: "A segmented
   drain system comprises a drain body operable"})
db.textual search.insert( [
   { id: 2, title: "Aircrew ensembles", abstract: "A method of countering the effects of g
   forces" },
   { id: 3, title: "Protective hand covering", abstract: "A protective hand covering has a
   membrane" },
   { _id: 4, title: "Perspiration concealing brassiere", abstract: "A perspiration concealing
   brassiere" },
   { id: 5, title: "Low profile medical kit", abstract: "A first aid systems for an ultra
   compact" },
   { id: 6, title: "Accessory for shoe laces hat brims and the like", abstract: "A decorative
   and or promotional accessory" } ]
```



Criação do índice

db.textual_search.createIndex({ title: "text", abstract: "text" })



Exemplos de buscas

```
db.textual_search.find( { $text: { $search: "Protective Medical"} } )
db.textual search.find( { $text: { $search: "Protective -Medical"} } )
db.textual_search.find( { $text: { $search: "\"Medical Kit \""} } )
db.textual_search.find( { $text: { $search: "+profile +kit"} } )
db.textual search.find(
  { $text: { $search: "Protective Medical" } },
  { score: { $meta: "textScore" } }
).sort( { score: { $meta: "textScore" } } )
```



Documentação

https://docs.mongodb.com/manual/text-search/

https://api.mongodb.com/python/1.8/tutorial.html



Exercício Complementar

- Considerando os conceitos vistos nesta unidade realize uma implementação em Pyhon que:
 - Se conecte a um banco MongoDB e crie um database e uma collection.
 - Realize a carga do arquivo 2015_USPTOf.json que representa um fragmento da base de patentes americanas para um DataFrame.
 - Crie um índice para os campos "title" e "abstract".
 - A partir dos dados de patentes que constam no DataFrame, realize a inserção no Banco de Dados MongoDB.
 - Realize algumas consultas, entre elas, pelo menos uma consulta utilizando o conceito de score.