

SQLxNoSQL

Amanda Ferreira

Ana Elisa Rezende

MongoDB – características gerais

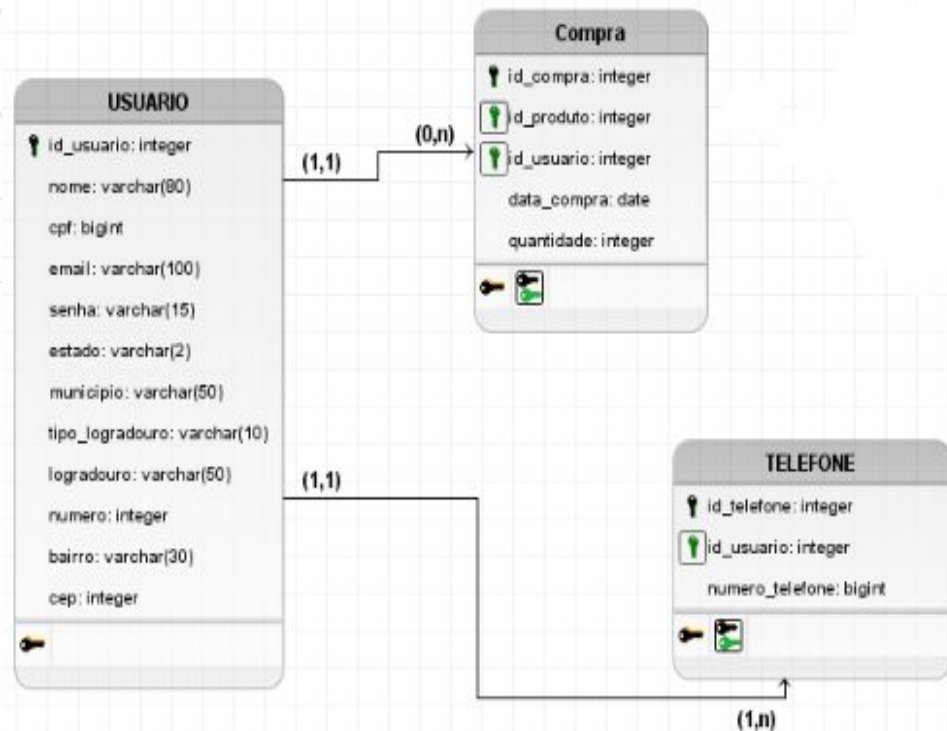
MongoDB é um banco de dados de código aberto, gratuito, de alta performance, sem esquemas e orientado à documentos

Se existe a necessidade de um banco de dados distribuído para aplicativos transacionais e analíticos modernos que trabalham com dados que mudam rapidamente, o MongoDB é o caminho a seguir.

Por ser orientado à documentos JSON, muitas aplicações podem modelar informações de modo muito mais natural, pois os dados podem ser aninhados em hierarquias complexas e ainda serem indexáveis e fáceis de buscar, igual ao que já é feito em JavaScript.



Modelo lógico



```
db.colecao.insert({
  _id:1,
  nome:'Ana Elisa Rezende',
  cpf:123456,
  email:'ana@elisa.com',
  senha:123@456,
  estado:'ES',
  municipio:'Serra',
  tipo_logradouro:'Rua',
  logradouro:'Carapebus',
  numero:105,
  bairro:'Valparaíso',
  cep:29165813,
  telefone:40028922,
  produto:'blusa',
  data_compra:'01-08-2020',
  quantidade:4
})
```

Conexão

```
$ pip install pymongo
```

```
from pymongo import MongoClient
```

```
cliente = MongoClient('localhost', 27017)
```

```
ou
```

```
cliente = MongoClient('mongodb://localhost:27017/')
```



Inclusão

Instruções de esquema do MongoDB

Criado implicitamente na primeira operação `insertOne()` ou `insertMany()`. A chave primária `_id` é adicionada automaticamente se o `_id` campo não for especificado.

```
db.colecao.insert({  
  id_usuario:1,  
  nome:'Ana Elisa Rezende',  
  cpf:123456,  
  email:'ana@elisa.com',  
  ...  
  cep:29165813,  
})
```

Link para os 10 inserts:

<https://drive.google.com/file/d/1qNRdiMfzvGkEt4QxQfe2iVsdM8mdgy8T/view?usp=sharing>

```
1 db.pessoas.insert({  
2   id_usuario:1,  
3   nome:'Ana Elisa Rezende',  
4   cpf:123456,  
5   email:'ana@elisa.com',  
6   senha:123@456,  
7   estado:'ES',  
8   municipio:'Serra',  
9   tipo_logradouro:'Rua',  
10  logradouro:'Carapebus',  
11  bairro:'Valparaíso',  
12  numero:105,  
13  bairro:'Valparaíso',  
14  cep:29165813,  
15  data_compra:'01-08-2020',  
16  quantidade:4,  
17  numero_telefone:40028922  
18 })
```

Seleção

Instruções SQL SELECT

```
SELECIONE *  
DE users
```

Instruções find () do MongoDB

```
db.users.find()
```

Exemplo de sintaxe para pesquisa específica:

```
db.collection.find( { qty: { $gt: 4 } } )
```

Shell Output X

```
1 { "_id" : 1, "nome" : "Ana Elisa Rezende", "cpf" : 123456, "email" : "ana@elisa.com", "senha" : "123@456", "estado" : "ES", "municipio" : "Serra", "tipo_logradouro" :  
2 { "_id" : 2, "nome" : "Pedro Paulo", "cpf" : 12345, "email" : "pedro@gmail.com", "senha" : "secret123", "estado" : "ES", "municipio" : "Vitória", "tipo_logradouro" :  
3 { "_id" : 3, "nome" : "Lucia Gonçalves", "cpf" : 78789, "email" : "lucia@gmail.com", "senha" : "123@456", "estado" : "ES", "municipio" : "Vila Velha", "tipo_logradouro" :  
4 { "_id" : 4, "nome" : "Julia Clarindo", "cpf" : 81254, "email" : "julia@gmail.com", "senha" : "123@456", "estado" : "ES", "municipio" : "Cariacica", "tipo_logradouro" :  
5 { "_id" : 5, "nome" : "Paula Abreu", "cpf" : 96547, "email" : "paula@gmail.com", "senha" : "123@456", "estado" : "ES", "municipio" : "Vila Velha", "tipo_logradouro" :  
6 { "_id" : 6, "nome" : "Joao Junqueira", "cpf" : 79996, "email" : "joao@gmail.com", "senha" : "123@456", "estado" : "ES", "municipio" : "Cariacica", "tipo_logradouro" :  
7 { "_id" : 7, "nome" : "Viviane Reis", "cpf" : 78452, "email" : "viviane@gmail.com", "senha" : "123@456", "estado" : "ES", "municipio" : "Vila Velha", "tipo_logradouro" :  
8 { "_id" : 8, "nome" : "Thyago Barbosa", "cpf" : 54687, "email" : "thyago@gmail.com", "senha" : "123@456", "estado" : "ES", "municipio" : "Serra", "tipo_logradouro" :  
9 { "_id" : 9, "nome" : "Fernando Henrique", "cpf" : 32659, "email" : "fernando@gmail.com", "senha" : "123@456", "estado" : "ES", "municipio" : "Vila Velha", "tipo_logradouro" :  
10 { "_id" : 10, "nome" : "Teresa Hardy", "cpf" : 48173, "email" : "rollin@gmail.com", "senha" : "123@456", "estado" : "ES", "municipio" : "Itarana", "tipo_logradouro" :  
11
```

Exclusão

SQL Delete Statements

```
DELETE FROM people  
WHERE status = "D"
```

MongoDB deleteMany() Statements

```
db.people.deleteMany( { status: "D" } )
```

```
db.users.deleteMany({cpf:48173})
```

Shell Output x Shell Output x Shell Output x

```
1 { "_id" : 1, "nome" : "Ana Elisa Rezende", "cpf" : 123456, "email" : "ana@elisa.com", "senha" : "123@456", "estado" : "ES", "municipio" : "Serra", "tipo_logradouro" : "Rua", "logradouro" : "Rua da Liberdade", "cep" : "13500-000" }
2 { "_id" : 2, "nome" : "Pedro Paulo", "cpf" : 12345, "email" : "pedro@gmail.com", "senha" : "secret123", "estado" : "ES", "municipio" : "Vitória", "tipo_logradouro" : "Rua", "logradouro" : "Rua da Liberdade", "cep" : "13500-000" }
3 { "_id" : 3, "nome" : "Lucia Gonçalves", "cpf" : 78789, "email" : "lucia@gmail.com", "senha" : "123@456", "estado" : "ES", "municipio" : "Vila Velha", "tipo_logradouro" : "Rua", "logradouro" : "Rua da Liberdade", "cep" : "13500-000" }
4 { "_id" : 4, "nome" : "Julia Clarindo", "cpf" : 81254, "email" : "julia@gmail.com", "senha" : "123@456", "estado" : "ES", "municipio" : "Cariacica", "tipo_logradouro" : "Rua", "logradouro" : "Rua da Liberdade", "cep" : "13500-000" }
5 { "_id" : 5, "nome" : "Paula Abreu", "cpf" : 96547, "email" : "paula@gmail.com", "senha" : "123@456", "estado" : "ES", "municipio" : "Vila Velha", "tipo_logradouro" : "Rua", "logradouro" : "Rua da Liberdade", "cep" : "13500-000" }
6 { "_id" : 6, "nome" : "Joao Junqueira", "cpf" : 79996, "email" : "joao@gmail.com", "senha" : "123@456", "estado" : "ES", "municipio" : "Cariacica", "tipo_logradouro" : "Rua", "logradouro" : "Rua da Liberdade", "cep" : "13500-000" }
7 { "_id" : 7, "nome" : "Viviane Reis", "cpf" : 78452, "email" : "viviane@gmail.com", "senha" : "123@456", "estado" : "ES", "municipio" : "Vila Velha", "tipo_logradouro" : "Rua", "logradouro" : "Rua da Liberdade", "cep" : "13500-000" }
8 { "_id" : 8, "nome" : "Thyago Barbosa", "cpf" : 54687, "email" : "thyago@gmail.com", "senha" : "123@456", "estado" : "ES", "municipio" : "Serra", "tipo_logradouro" : "Rua", "logradouro" : "Rua da Liberdade", "cep" : "13500-000" }
9 { "_id" : 9, "nome" : "Fernando Henrique", "cpf" : 32659, "email" : "fernando@gmail.com", "senha" : "123@456", "estado" : "ES", "municipio" : "Vila Velha", "tipo_logradouro" : "Rua", "logradouro" : "Rua da Liberdade", "cep" : "13500-000" }
10 { "acknowledged" : true, "deletedCount" : 0 }
11
```


Atualização

SQL Schema Statements

```
ALTER TABLE people  
ADD join_date DATETIME
```

MongoDB Schema Statements

```
db.people.updateMany(  
  { },  
  { $set: { join_date: new Date() } }  
)
```

```
db.users.update({email:'ana@elisa.com'},{$set:(email:'anaelisa@gmail.com'}})
```

```
Shell Output × Shell Output × Shell Output × Shell Output × Shell Output ×  
1 { "_id" : 1, "nome" : "Ana Elisa Rezende", "cpf" : 123456, "email" : "anaelisa@gmail.com", "senha" : "123@456", "estado" : "ES", "municipio" : "Serra", "tipo_logradouro" : "Rua", "cep" : "13506-900", "data_nascimento" : "1990-01-01", "data_cadastro" : "2023-01-01", "ativo" : true }  
2 { "_id" : 2, "nome" : "Pedro Paulo", "cpf" : 12345, "email" : "pedro@gmail.com", "senha" : "secret123", "estado" : "ES", "municipio" : "Vitória", "tipo_logradouro" : "Rua", "cep" : "13506-900", "data_nascimento" : "1990-01-01", "data_cadastro" : "2023-01-01", "ativo" : true }  
3 { "_id" : 3, "nome" : "Lucia Gonçalves", "cpf" : 78789, "email" : "lucia@gmail.com", "senha" : "123@456", "estado" : "ES", "municipio" : "Vila Velha", "tipo_logradouro" : "Rua", "cep" : "13506-900", "data_nascimento" : "1990-01-01", "data_cadastro" : "2023-01-01", "ativo" : true }  
4 { "_id" : 4, "nome" : "Julia Clarindo", "cpf" : 81254, "email" : "julia@gmail.com", "senha" : "123@456", "estado" : "ES", "municipio" : "Cariacica", "tipo_logradouro" : "Rua", "cep" : "13506-900", "data_nascimento" : "1990-01-01", "data_cadastro" : "2023-01-01", "ativo" : true }  
5 { "_id" : 5, "nome" : "Paula Abreu", "cpf" : 96547, "email" : "paula@gmail.com", "senha" : "123@456", "estado" : "ES", "municipio" : "Vila Velha", "tipo_logradouro" : "Rua", "cep" : "13506-900", "data_nascimento" : "1990-01-01", "data_cadastro" : "2023-01-01", "ativo" : true }  
6 { "_id" : 6, "nome" : "Joao Junqueira", "cpf" : 79996, "email" : "joao@gmail.com", "senha" : "123@456", "estado" : "ES", "municipio" : "Cariacica", "tipo_logradouro" : "Rua", "cep" : "13506-900", "data_nascimento" : "1990-01-01", "data_cadastro" : "2023-01-01", "ativo" : true }  
7 { "_id" : 7, "nome" : "Viviane Reis", "cpf" : 78452, "email" : "viviane@gmail.com", "senha" : "123@456", "estado" : "ES", "municipio" : "Vila Velha", "tipo_logradouro" : "Rua", "cep" : "13506-900", "data_nascimento" : "1990-01-01", "data_cadastro" : "2023-01-01", "ativo" : true }  
8 { "_id" : 8, "nome" : "Thyago Barbosa", "cpf" : 54687, "email" : "thyago@gmail.com", "senha" : "123@456", "estado" : "ES", "municipio" : "Serra", "tipo_logradouro" : "Rua", "cep" : "13506-900", "data_nascimento" : "1990-01-01", "data_cadastro" : "2023-01-01", "ativo" : true }  
9 { "_id" : 9, "nome" : "Fernando Henrique", "cpf" : 32659, "email" : "fernando@gmail.com", "senha" : "123@456", "estado" : "ES", "municipio" : "Vila Velha", "tipo_logradouro" : "Rua", "cep" : "13506-900", "data_nascimento" : "1990-01-01", "data_cadastro" : "2023-01-01", "ativo" : true }  
10 { "acknowledged" : true, "deletedCount" : 0 }  
11 WriteResult({ "nMatched" : 0, "nUpserted" : 0, "nModified" : 0 })  
12
```


Pontos positivos

- Grátis
- Rapidez
- Excelente performance
- Alta disponibilidade
- Flexibilidade
- Suporte a um alto volume de dados
- Forte suporte de queries para consultas

Pontos negativos

- Lock de escrita
- Transações e join
- Consumo de memória ram
- Documentação
-

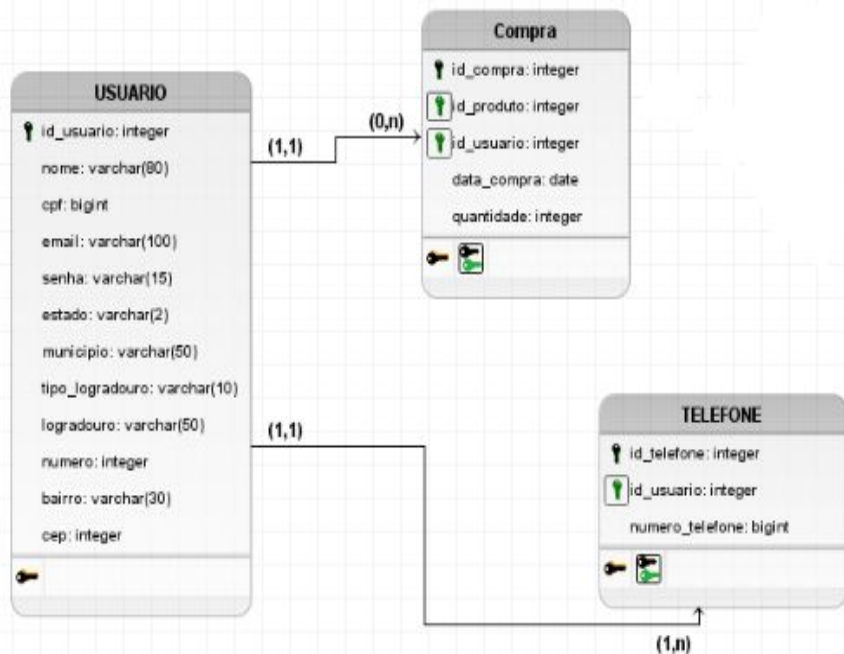
ArangoDB

- tabela ao lado com diferenças da versão gratuita/comercial
- características: Um banco de dados multi-modelo nativo desde o início, com suporte a modelos de chave/valor, documento e gráfico.
- Com ArangoDB, os usuários podem facilmente alterar sua estratégia de acesso a dados, apenas alterando uma consulta.

	Community	Enterprise
Features		
Community Edition Features	✓	✓
SmartJoins		✓
SatelliteCollections		✓
SmartGraphs		✓
Encryption at Rest		✓
Enhanced User Management with LDAP		✓
Encryption Control		✓
Enhanced Data Masking		✓
Auditing		✓
Training		
Free, online education	✓	✓
Private, on-demand training		✓
Support		
SLA	none	24x7
License		
Type	Apache V2	Commercial
Price*		
Annual	free	contact us

* One subscription is valid for a single machine with up to 256 GB of RAM. In a non-cloud environment, this means a single physical machine. In a cloud environment, this means a single virtual machine.

Modelo Lógico Postgres vs representação de Modelo no Database escolhido



```
1  INSERT {
2      id_usuario: 1,
3      nome: "Amanda Ferreira",
4      cpf: 17917917917,
5      email: "amanda.ferreira@gmail.com",
6      senha: "amanda123",
7      estado: "ES",
8      municipio: "Vila Velha",
9      tipo_logradouro: "Rua",
10     logradouro: "Olegário Mariano",
11     numero: 1338,
12     bairro: "Soteco",
13     cep: 29100142,
14     telefone: [
15         {telefone: 27997640777},
16         {telefone: 2730256398}],
17     compra: [
18         {produto: "camisa",
19          data_compra: "2020-10-08",
20          quantidade: 2}]
21     ]
22 } INTO CompraOnline
```

Conexão

```
from pyArango.connection import Connection  
conn = Connection(username='root',  
password='password')
```

arangoURL: 127.0.0.1:8529



Inclusão

```
1 INSERT {  
2     id_usuario: 1,  
3     nome: "Amanda Ferreira",  
4     cpf: 17917917917,  
5     email: "amanda.ferreira@gmail.com",  
6     senha: "amanda123",  
7     estado: "ES",  
8     municipio: "Vila Velha",  
9     tipo_logradouro: "Rua",  
10    logradouro: "Olegário Mariano",  
11    numero: 1338,  
12    bairro: "Soteco",  
13    cep: 29100142,  
14    telefone: [  
15        {telefone: 27997640777},  
16        {telefone: 2730256398}],  
17    compra: [  
18        {produto: "camisa",  
19         data_compra: "2020-10-08",  
20         quantidade: 2}]  
21    ]  
22 } INTO CompraOnline
```

INSERT { <campo> :dado }
INTO nome_da_collection

link para os 10 inserts:

<https://docs.google.com/document/d/1Ak-KHNNuEOMJxbqYngNlwsRdaRctcbW8lldTfYV7Hw4/edit?usp=sharing>

```
1 For user in CompraOnline
2 Return user
```

```
FOR user IN nome_collection
RETURN user
```

Query

20 elements

0.803 ms

JSONTable

_key	_id	_rev	name	cpf	email	senha	estado	municipio	tipo_logradouro	logradouro
5615	CompraOnline/5615	_b08Yj4G--	Pedro Paulo	12345	pedro@gmail.com	secret123	ES	Vitória	Rua	Milton Simões
5616	CompraOnline/5616	_b08Yj4G--A	Lucia Gonçalves	78789	lucia@gmail.com	123@456	ES	Vila Velha	Avenida	Carlos Lindemberg
5617	CompraOnline/5617	_b08Yj4G--C	Julia Clarindo	81254	julia@gmail.com	123@456	ES	Cariacica	Rua	Emilio de Abreu
5618	CompraOnline/5618	_b08Yj4G--E	Paula Abreu	96547	paula@gmail.com	123@456	ES	Vila Velha	Avenida	Carlos Lindemberg
5619	CompraOnline/5619	_b08Yj4G--G	Joao Junqueira	79996	joao@gmail.com	123@456	ES	Cariacica	Rua	Nova
5620	CompraOnline/5620	_b08Yj4G--	Viviane	78452	viviane@gmail.com	123@456	ES	Vila	Avenida	Carlos

Download

CSV

Copy To Editor

Exclusão

```
1 For user in CompraOnline
2 filter user.name IN ["Pedro Paulo", "Viviane Reis"]
3 remove user in CompraOnline
```

FOR user IN nome_collection

FILTER user.<nome_coluna> IN
[<dado_coluna>]

REMOVE user IN nome_collection

Atualização

```
1 For user in CompraOnline
2 filter user.name IN ["Keely Karlee", "Teresa Hardy"]
3 update user with {municipio: "Vila Velha"}
4 in CompraOnline
```

FOR user IN nome_collection

FILTER user.<nome_coluna> IN [<dado_coluna>]

UPDATE user WITH {<nome_campo>: <nome_registro>}

IN nome_collection

5642	CompraOnline/5642	_b08eCru-- A	Teresa Hardy	48173	rollin@gmail.com	123@456	ES	Vila Velha	Rua	Althea Josefa
5643	CompraOnline/5643	_b08eCru-- C	Keely Karlee	24836	bosco@gmail.com	123@456	ES	Vila Velha	Distrito	Bill Lucius

Pontos positivos

- É um banco de dados multi-modelo
- banco de dados de gráficos
- banco de dados de documentos
- armazenamento de valor-chave em um núcleo C++.
- é nativo (usuários podem usar todos os modelos de dados suportados em uma única consulta)
- AQL – ArangoDB Query Language
- pesquisa de texto completo e mecanismo de classificação – ArangoSearch.
- interface de gerenciamento web
-

Pontos negativos

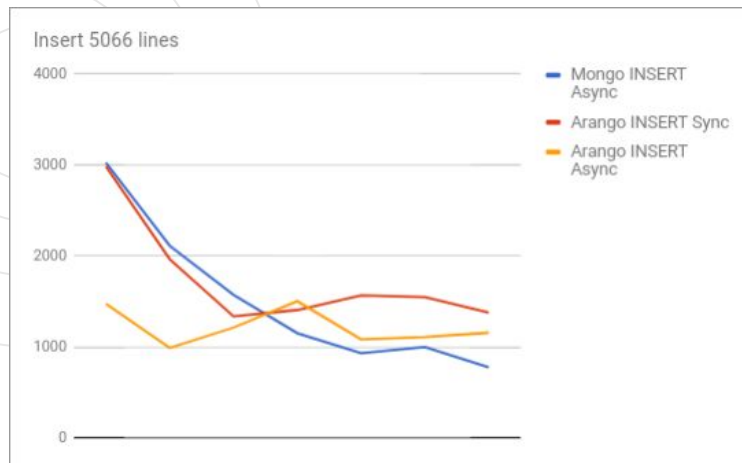
- Suporte a poucas linguagens de programação
- como não é um padrão do mercado, é mais difícil achar exemplos.
- em testes com mais dados o tempo de execução é mais lento (em comparação com o Mongo)

MongoDB x ArangoDB

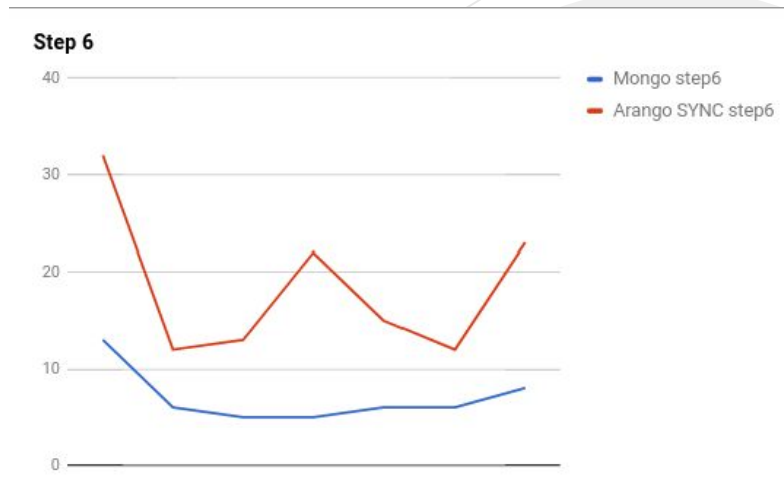
- Modelo único
- Sintaxe JSON
- Grande velocidade
- Sem suporte a junções

- Multi-modelo
- Linguagem própria (AQL)
- Baixa velocidade
- Suporta junções

MongoDB x ArangoDB





Média de tempo de execução na inserção de dados de um arquivo CSV em ms.



Média de tempo de execução para agregar os documentos/bins usando função de agrupamento ("group by") em ms.

ArangoDB x MongoDB

APIs and other access methods	AQL Foxx Framework GraphQL query language HTTP API Java & SpringData JSON style queries VelocityPack/Velocystream	proprietary protocol using JSON
Supported programming languages	C# C++ Clojure Elixir Go Java JavaScript (Node.js) PHP Python R Rust	Actionscript  C C# C++ Clojure  ColdFusion  D  Dart  Delphi  Erlang Go Groovy  Haskell Java JavaScript Lisp  Lua  MatLab  Perl PHP PowerShell  Prolog  Python R  Ruby Rust Scala Smalltalk  Swift

Feature	MongoDB	ArangoDB
General		
Initial Release	2009	2012
License	AGPLv3 / Commercial	Apache 2 / Commercial
Commercial Friendly License	no	yes
Written in	C++	C++
Data-Model	document	multi-model documents, graphs, key-value
Schema free	yes additional schema validation	yes schema validation with Foxx
Data format	JSON / BSON	JSON / VelocityPack
Data Storage	MMAPv1/WIREDTIGER	MMFiles/RocksDB
Persistence to disk	yes	yes
Journaling	yes	yes

Concluindo, o excelente desempenho e a flexibilidade superior de um multi-modelo nativo é uma vantagem chave do ArangoDB. E diferente do MongoDB, ele não tem um alto consumo de memória RAM.

Referências

<https://medium.com/code-rocket-blog/trabalhando-com-python-e-mongodb-1d23ee042658>

<https://medium.com/@neto.acs/crud-em-mongodb-inserindo-dados-eeaa12b57154>

<https://docs.mongodb.com/manual/reference/sql-comparison/>

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

- <https://phpzm.rocks/nosql-no-php-usando-arangodb-parte-1-60c57b83eaf5>
-
- <https://www.arangodb.com/why-arangodb/>
-
- <https://www.arangodb.com/docs>,
<https://www.arangodb.com/why-arangodb/sql-aql-comparison/>
- <https://www.arangodb.com/why-arangodb/multi-model/>

