

# MongoDB Conceitos Básicos

Rafael Gonçalves de Oliveira Viana<sup>1</sup>

<sup>1</sup>Sistemas de Informação – Universidade Federal do Mato Grosso do Sul (UFMS)  
Caixa Postal 79400-000 – Coxim – MS – Brazil

rafael.viana@aluno.ufms.br

**Abstract.** *Nowaday, with the spread of the Internet, a major challenge in the area of computing, the large-scale data manipulation "Big Data", with this new trend, traditinal SGBDs can no longer offer performace storage.*

*Because of this and a host of other issues, a new set of tool platforms Big Data, called NoSQL (not just SQL), was developed, this article will focus on presen-ting one of these MongoDB technologies.*

**Resumo.** *Atualmente com a disseminação da Internet crio-se um grande de-safio na área de computação, a manipulação de dados em larga escala "Big Data", com essa nova tendência os SGBDs tradicionais não são mais capazes, de prover armazenamento com desempenho. Por causa desse problema, e ou-tros demais, um novo conjunto de plataformas de ferramentas voltadas para Big Data, denominado NoSQL (Not only SQL), tem sido desenvolvida, este artigo tem como foco a apresentação de uma dessas tecnologias o MongoDB.*

## 1. Introdução

MongoDB promove diversas soluções inovadoras de armazenamento e processamento de grande volume de dados. Estas soluções foram inicialmente criadas para solucionar problemas gerados por aplicações, com grande volume de dados, tenham uma arquitetura que "escale" com grande facilidade de forma horizontal, além da necessidade de persistência dos dados em aplicações nas nuvens (cloud computing). Para atender essas necessidades, grandes empresas como o Facebook vem utilizando o MongoDB.

## 2. NoSQL

NoSQL é uma classe de sistema de gerenciamento de banco de dados diferentes dos bancos de dados relacionais tradicionais em que os dados não são armazenados usando esquemas de tabela fixa. Principalmente o seu propósito é servir como sistema de banco de dados para aplicações enormes escala web onde eles superam bancos de dados relacionais tradicionais.

Dentro do aspecto do processamento dos dados, o principal paradigma adotado pelos produtos NoSQL é o MapReduce. Em resumo, tal paradigma divide o processamento em duas etapas: (1) Map, que mapeia e distribui os dados em diversos nós de processamento e armazenamento; e (2) Reduce, que agrega e processa os resultados parciais para gerar um resultado final (ou intermediário para outro processo MapReduce).

Além disso, existem diferentes produtos NoSQL disponíveis no mercado, como por exemplo: Column Store (e.g., Hadoop, Casandra, Amazon SimpleDB, Hypertable),

Document Store (e.g., Apache CouchDB, MongoDB), Key-Value/Tuple Store (e.g., Voldemort, Redis, Memcached), Graph Databases (e.g., Neo4J, HyperGraphDB). Assim, serão abordadas implementações existentes para os problemas de distribuição, transações, mineração e a infraestrutura computacional de nuvem.

Em relação a SGBD tradicionais, a distribuição dos dados de forma elástica é inviabilizado pois o modelo de garantia de consistência é fortemente baseado no controle transacional ACID (Atomicity, Consistency, Isolation e Durability). Esse tipo de controle transacional é praticamente inviável quando os dados e o processamento são distribuídos em vários nós. O teorema CAP (Consistency, Availability e Partition tolerance) mostra que somente duas dessas 3 propriedades podem ser garantidas simultaneamente em um ambiente de processamento distribuído de grande porte. A partir desse teorema, os produtos NoSQL utilizam o paradigma BASE (Basically Available, Soft-state Eventually consistency) para o controle de consistência, o que consequentemente traz uma sensível diminuição no custo computacional para a garantia de consistência dos dados em relação a SGBD tradicionais [Marcos Rodrigues Vieira 2012].

### **3. MongoDB**

MongoDB é um sistema de gerenciamento de banco de dados NoSQL lançado em 2009. Ele armazena dados como documentos JSON-like com esquemas dinâmicos (o formato é chamado BSON). MongoDB tem seu foco voltado para quatro coisas: flexibilidade, potência, velocidade e facilidade de uso. Ele suporta servidores replicados e indexação e oferece drivers para várias linguagens de programação.

MongoDB aceita dados com tamanho máximo de 16 MB, a distribuição é horizontalmente escalável, oferece suporte a partilha, foi escrito em C++. MongoDB é um produto gratuito, não tem suporte a compressão de dados, usa funções para as operações de adição de novos registros, atualização e exclusão de registros existentes.

No MongoDB os dados são armazenados usando coleções. As coleções do MongoDB tem nenhuma restrição em relação aos dados armazenados nele. As coleções não tem uma estrutura fixa. Os campos podem ter diferentes tipos de dados.

A linguagem utilizada pelo MongoDB para consultas avançadas, são funções de retorno de chamada. Para o mongo shell, a linguagem utilizada para funções de declaração é JavaScript. Estas funções são usadas para manipular dados retornados pelas consultas básicas. Pode-se percorrer os resultados das consultas e aplicar funções diferentes.

A criação da coleção é feita sobre a primeira inserção. Por causa da flexibilidade do conjunto de MongoDB, a estrutura não tem de ser fixo, sendo assim não há dependências entre as coleções facilitando a exclusão de qualquer tabela e os índices atribuídos a ela [Boicea et al. 2012].

#### **3.1. Inserir**

A inserção de um novo registro é feito usando funções e objetos BSON. As funções usadas para inserir novos dados são 'insert' e 'save'. Ao inserir um novo objeto em uma coleção MongoDB, um novo campo é adicionado automaticamente ao seu novo objeto. Esse campo é chamado de '\_id' e é único, é usado como um índice. Por exemplo, se você quiser adicionar o mesmo usuário na coleção 'user' do banco de dados MongoDB, você tem que escrever no shell mongo algo como isto:

|  |  |
|--|--|
| <pre>db.users.insert ( { 'id_usuario': 1, 'primeiro_nome': 'Rafael', 'ultimo_nome': 'Viana' } );</pre> | <pre>db.users.save ( { 'id_usuario': 1, 'primeiro_nome': 'Rafael', 'ultimo_nome': 'Viana' } );</pre> |
|--|--|

### 3.2. Buscar

No MongoDB, você pode recuperar resultados usando a função "find". Por exemplo, supondo que temos uma coleção de usuários nomeados "usuarios". Se você quiser recuperar todos os itens desta coleção você pode usar algo como isto:

```
db.usuarios.find();
```

Se você quiser apenas para recuperar alguns dos itens na coleção, por exemplo, apenas os usuários que tem o primeiro nome "Rafael" você usa a função de "find" com um objeto extra:

```
db.usuarios.find({primeiro_nome: 'Rafael'});
```

### 3.3. Excluir

Se você quiser remover apenas algumas entradas, como os que têm a 'usuario\_id' maior do que 10, então você deve escrever algo assim:

```
db.users.remove ({ 'id_usuario': { '$ GT': 10}});
```

Se você quiser remover todos itens da coleção pode usar a função "Drop", então deve escrever algo assim:

```
db.usuarios.drop();
```

### 3.4. Atualizar

Em MongoDB pode facilmente atualizar uma linha ou várias linhas. O comando se você deseja atualizar toda a coleção é:

```
db.users.update ({},{ '$ Set': { 'primeiro_nome': 'Luiz'}});
```

O primeiro objeto é usado para selecionar as entradas que você deseja atualizar. Neste caso, a consulta retorna todas as entradas da coleção. Se você quiser atualizar apenas as entradas que o 'user\_id' maior do que 100, a função de atualização parece com isso:

```
db.users.update ({ 'usuario_id': { '$ gt': 100}},
{ '$ Set': { 'primeiro_nome': 'Luiz'}});
```

## 4. Conclusão

O MongoDB, tem o foco em escalabilidade e facilidade, e é um excelente complemento para o modelo relacional, no qual estão sendo aplicado em situações para as quais não foram desenvolvidos. A escolha por um modelo não relacional deve estar diretamente relacionada à compatibilidade da sua modelagem com uma ou outra categoria, adquirindo performace em suas aplicações, que possuem alto fluxo de dados.

## Referências

- Boicea, A., Radulescu, F., and Agapin, L. I. (2012). MongoDB vs oracle–database comparison. In *Emerging Intelligent Data and Web Technologies (EIDWT), 2012 Third International Conference on*, pages 330–335. IEEE.
- Marcos Rodrigues Vieira, Josiel Maimone de Figueiredo, J. M. d. F. A. F. M. V. (2012). Bancos de dados nosql: Conceitos, ferramentas, linguagens e estudos de casos no contexto de big data. SBBD.