

Fundamentos de Bancos de Dados: tipos de Banco de Dados NoSQL

Resumo

Referência Bibliográfica

- FEITOSA, M. P. **MongoDB - o banco de dados NoSQL mais utilizado da atualidade: Uma introdução com instalação, comandos e aplicações em Java**. Editor independente; 1ª edição, 2021
- HOWS, D. **Introdução ao MongoDB**. Novatec - 2015
- DATE, C. J. **Introdução a sistemas de banco de dados**. Rio de Janeiro. Ed. Campus, 1991.
- CHEN, Peter. **Modelagem de dados: a abordagem entidade-relacionamento para projeto lógico**. São Paulo: Makron Books, 1990.
- MEDEIROS, L. F., **Banco de dados, princípios e práticas**, 1ª. ed., Ed. Intersaberes, 2013
- PUGA, S., França E., GOYA M., **Banco de dados: Implementação em SQL, PL/SQL e Oracle 11g**, Ed. Pearson, 2013
- ELMASRI R., NAVATHE, S., **Sistemas de Banco de Dados**, 4ªed., Ed. Pearson, 2005

Introdução

Big Data está em ascensão e é matéria-prima dos Bancos de Dados NoSQL. Para entender os NoSQLs, é importante saber que a linguagem SQL sempre foi usada para tratamento de dados em bancos relacionais, ao longo dos anos.

A principal diferença entre os bancos de dados relacionais e NoSQL é que o segundo permite maior velocidade, flexibilidade e escalabilidade ao armazenar e acessar dados não estruturados.

Tipos de Banco de Dados NoSQL

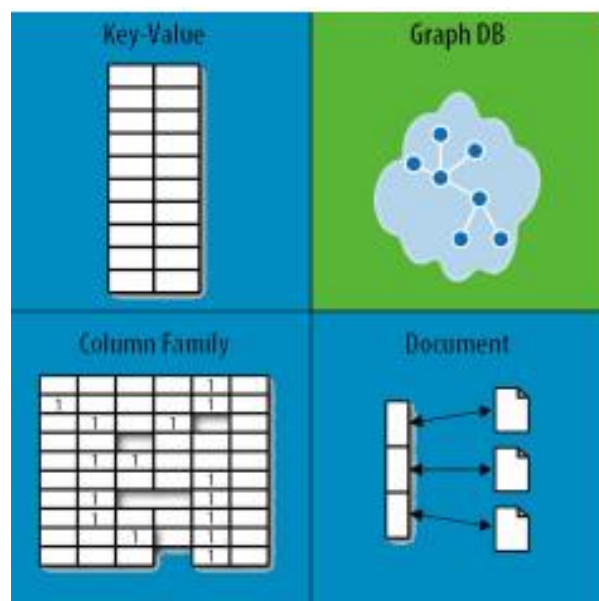


Figura 1: Tipos de Banco de Dados NoSQL

Modelo Colunas

No modelo colunas, o banco de dados faz armazenamento em linhas particulares de tabela. Esse esquema é o perfeito oposto dos bancos relacionais, que armazenam conjuntos de dados em uma única linha. Um exemplo do modelo de colunas é o banco de dados Hbase;

BDs colunares são também conhecidos como BDs de família de colunas, BD de armazenamento de registros extensíveis ou BDs de colunas extensas, todos relacionados à estrutura de colunas esparsas normalmente sem esquema. A Figura 2 mostra um exemplo de estrutura de um BD colunar. Ele pode ser visto como um valor ao qual se associa um nome, sendo este par (nome, valor) chamado coluna. Um conjunto de 28 colunas define uma linha, que é acessada por um identificador (RowKey). Apesar da Figura 2 mostrar linhas com o mesmo número de colunas, linhas diferentes podem ter número de colunas diferentes, sendo adequado à representação de dados heterogêneos.

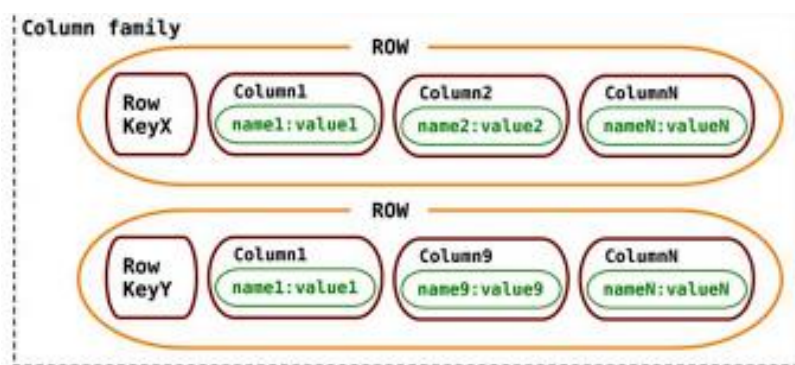


Figura 2: Estrutura de Banco de Dados Colunar

Modelo Chave-valor

O modelo chave-valor é o mais simples dentre os modelos NoSQL. Ele é similar a um sistema de cache em memória no qual todos os dados são indexados apenas por uma chave única e, normalmente, o SGBD não tem controle sobre seu conteúdo (simples ou complexo) para realizar operações, ou seja, o conteúdo é uma "caixa preta" para o SGBD.

Em chave-valor, o banco é formado por conjuntos de chaves, que por sua vez são acompanhados de valores como tabelas *hash*. A estrutura chave-valor também é bem flexível e própria para armazenamento de big data. Por não fazer operações sobre o conteúdo dos dados, as operações sobre esse modelo são simples e rápidas, por exemplo, armazenar (put) e recuperar (get) com base em uma chave. Os BDs chave-valor que mais se destacam são: Redis, Riak, Projeto Voldemort, MemcachedDB, BerkeleyDB, Scalaris e DynamoDB.

Essa família de bancos NoSQL é a que suporta a maior carga de dados, pois o conceito dela é que um determinado valor seja acessado através de uma chave identificadora única.

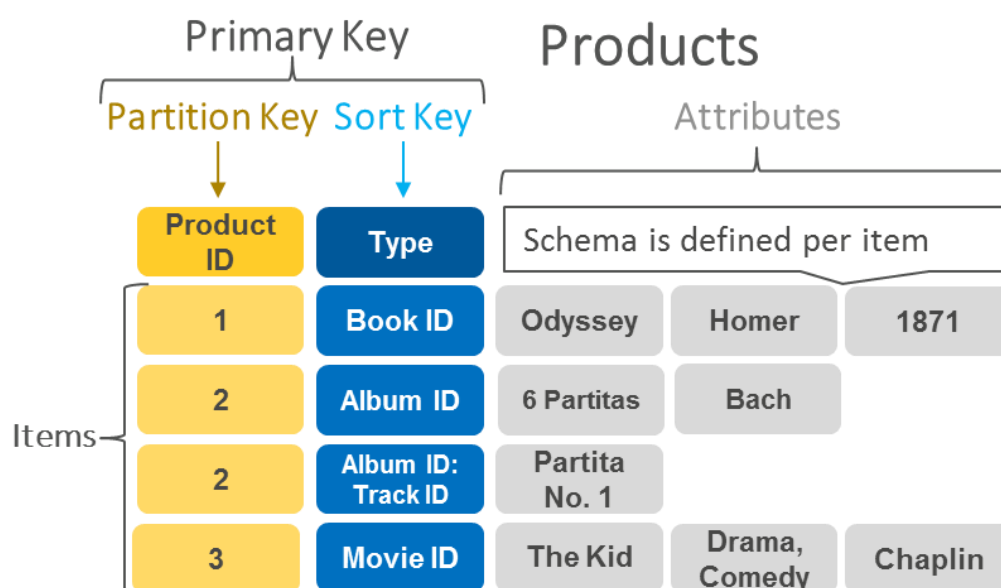


Figura 3: Exemplo de armazenamento chave/valor

Modelo Grafo

A maioria dos BDs NoSQL são pensados para executarem em clusters e, por isso, facilmente incorporam o modelo de dados de agregados, sobre um grande número de registros complexos com conexões implícitas (aninhadas). No caso dos BDs de grafo, o modelo de dados explora os relacionamentos entre os dados de forma explícita, definindo, assim, registros com várias conexões. BDs de grafo são projetados para lidar com dados complexos e altamente conectados, que ultrapassem a capacidade de representação dos BDs relacionais, como é o caso de uma rede social.

O modelo de dados de um BD de grafo é composto por nós e arestas. Nós representam entidades e arestas representam seus relacionamentos, sendo que ambos podem conter atributos. Isto é, aqui os dados são dispostos no formato de arcos conectados por arestas. Podemos definir como um conjunto de linhas conectadas por vértices também.

O modelo de grafos é vantajoso frente à pesquisas complexas, pois a latência e a performance promete ser menor do que no modelo chave-valor, por exemplo.

Os BDs orientados a grafos que mais se destacam são: Neo4J, Infinite Graph e OrientDB.

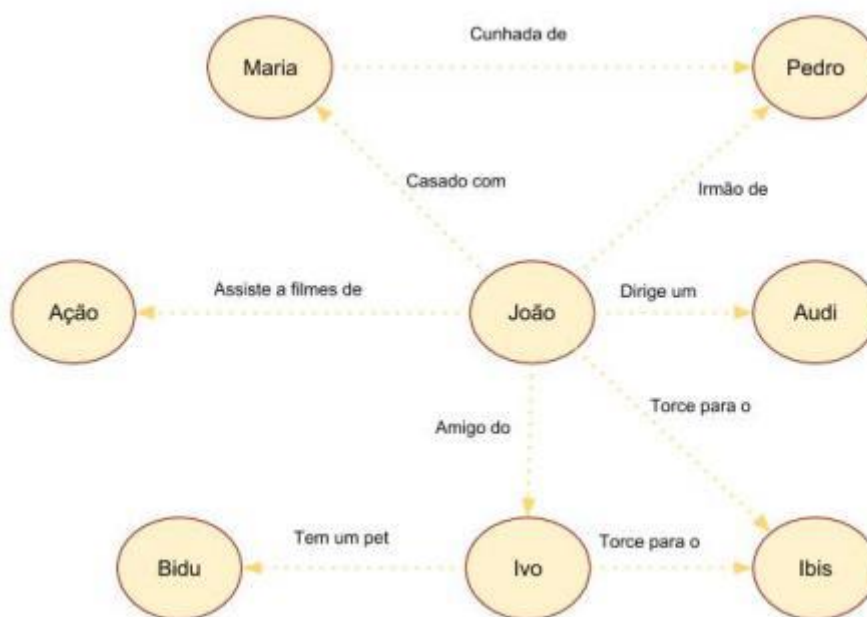


Figura 4: Exemplo de Banco de Dados em Grafo

Modelo Documento

Neste modelo, os dados são “documentos”. É o esquema de armazenamento do MongoDB, por exemplo. Esse modelo é altamente flexível e não carece de colunas pré-montadas, como é o caso do Cassandra. Esse modelo é especialmente eficiente para tratar dados não estruturados, já que uma única coleção pode contar com grupos de dados (documentos) de diversos formatos diferentes.

Esses BDs se assemelham muito aos BDs chave-valor, tendo uma chave para identificar um documento. Os documentos têm um formato padrão, como JSON ou XML (eXtensible Markup Language), e por conta disso, é possível manipular seu conteúdo e realizar consultas. BDs de documentos não são voltados para leituras e escritas concorrentes de alto desempenho, mas sim, para garantir o armazenamento eficiente de um grande volume de dados, bem como uma boa performance para operações de consulta. Como os documentos são hierárquicos, um documento pode conter diversos tipos de dados, como escalares, listas ou até outros documentos, já que geralmente não há esquema fixo. Por não haver um esquema fixo, propriedades podem ser adicionadas ou removidas de um documento sem afetar os demais documentos. Os BDs de documento mais populares são o MongoDB e o CouchDB.

Destaque dos Principais Banco de Dados NoSQL

O MongoDB é um líder de mercado dos bancos de dados NoSQL, pois possui *features* para produção como: replicação, indexação, balanceamento de carga. Para armazenar dados, o MongoDB utiliza alguns documentos muito similares ao formato JSON*. O mais interessante é o que o MongoDB é open source, o que contribui muito para a evolução da sua tecnologia.

O Amazon DynamoDB é um produto excelente da AWS (*Amazon Web Services*). O banco de dados DynamoDB é totalmente *cloud* e viabiliza um desempenho confiável e em escala. Um ponto bem importante: a Amazon confirma que a latência e consistência e fica abaixo de 10 milissegundos. Além disso, tem recursos de segurança, baseados em cache de memória, backup e restauração de dados.

O Cassandra foi desenvolvido no Facebook, atualmente o Cassandra e o HBase são mantidos pela Apache Foundation.

O Cassandra trabalha muito bem com Big Data, pois é otimizado para clusters, especialmente por funcionar sem mestres. O fato de ter mecanismos distribuídos também otimiza bastante a operação com os clusters.

Outro ponto forte do Cassandra é o conceito de orientação por coluna, o que torna a latência bem menor em algumas pesquisas.

O HBase é um banco de dados open source, orientado a colunas e distribuído. Atualmente, Spotify e Facebook são algumas das grandes corporações que utilizam esse modelo de armazenamento. Um dos seus pontos fortes é a pesquisa de dados que oferece uma resposta rápida. Transforma terabytes em milissegundos.

Pode-se concluir que Bancos de Dados NoSQL são extremamente úteis quando assunto é a análise e o tratamento de grande volume de dados.