

O Papel dos Bancos de Dados SQL e NoSQL na Engenharia de Dados

Visão geral

Bancos de dados relacionais e não relacionais são dois tipos de sistemas de gerenciamento de bancos de dados que nos ajudarão no gerenciamento de dados complexos.

Um banco de dados relacional armazena dados estruturados, enquanto o NoSQL os dados são semiestruturados ou não estruturados. No banco de dados relacional (SQL), ele consiste em tabelas (campos e registros), enquanto os dados do banco de dados não relacional são armazenados na forma de gráficos, documentos como XML e JSON.

Exemplos de Bancos de Dados Relacionais: MySQL, Oracle.

Exemplos de Bancos de Dados Não Relacionais: MongoDB, Apache Cassandra.

Bancos de dados relacionais

Conforme já definido, um banco de dados relacional consiste em tabelas e uma tabela consiste em linhas e colunas. Cada tabela tem uma coluna especial que contém apenas valores distintos e exclusivos que são chamados de chave primária (primary key). Esta chave primária é usada para definir o relacionamento entre as tabelas, e se a chave primária for usada em outra tabela então essa coluna é chamada de chave estrangeira (foreign key).

Para construir um banco de dados é necessário fazer um design do nosso banco de dados ou melhorar esse design de banco de dados existente, chamado de processo de normalização do banco de dados (é o processo de estruturar o banco de dados para remover a redundância de dados e melhorar a integridade dos dados).

Vantagens dos Bancos de Dados Relacionais

Os bancos de dados relacionais têm 4 propriedades: Atomicidade, Consistência, Isolamento e Durabilidade.

Atomicidade: Refere-se à integridade da transação do banco de dados. Cada transação é considerada como uma unidade, não existindo meio-termo.

Consistência: Apenas os dados que seguem os requisitos determinados podem ser armazenados no banco de dados. Caso contrário, o comando terá um erro impedindo a gravação do dado que esteja fora da conformidade das regras estabelecidas.

Isolamento: Capacidade de processar simultaneamente várias transações de forma que uma não afete a outra.

Durabilidade: Uma vez concluída a execução da transação, as atualizações e modificações no banco de dados sejam armazenadas e gravadas em disco em memória não volátil. Os efeitos da transação, portanto, não serão perdidos.

Banco de dados não relacional

Bancos de dados não relacionais também são chamados de bancos de dados No-SQL, que não requerem nenhuma tabela, campos ou registros.

Os bancos de dados NoSQL lidam com dados semiestruturados ou não estruturados. Em vez de conter tabelas, consiste em arquivos dentro de várias pastas. Eles podem possuir qualquer tipo de dado, seja JSON, XML, etc. Assim, criar e gerenciar dados em NoSQL é fácil e rápido.

Existem vários tipos de bancos de dados não relacionais, como:

Bancos de dados de documentos.

Bancos de dados de gráficos.

Bancos de dados de colunas largas.

Bancos de dados de valores-chave.

Vantagens dos bancos de dados não relacionais

Flexibilidade: Por armazenar dados não estruturados e estruturados, esses tipos de banco de dados são mais flexíveis que os bancos de dados relacionais que só podem armazenar dados de maneira estruturada.

Escalabilidade: esses bancos de dados podem ser dimensionados para acomodar qualquer tipo de dados, mantendo um baixo custo.

Alta performance: Os bancos de dados No-SQL são definidos para um bom desempenho e contêm dados não estruturados, enquanto nos bancos de dados relacionais os dados são armazenados em tabelas, portanto, o acesso aos dados é um pouco mais lento.

Tipos NoSQL:

- Grafo
- Colunas
- Chave-valor
- Documento

Conhecendo os tipos de banco de dados NoSQL

Grafo

Os bancos de dados de **grafos** foram criados especificamente para possibilitar o armazenamento de relacionamentos e a navegação por eles. Os **grafos** contêm nós, arestas e propriedades, todos usados para representar e armazenar **dados** de uma forma que os **bancos de dados** relacionais não estão equipados a fazer.

Comum em detecção de fraudes, mecanismos de recomendação, redes sociais, sistemas de arquivos, games, entre outros

Coluna

- Keyspace: agrupamento de famílias de colunas → database
- Column Family/table: agrupamento de colunas → table
- Row Key: chave que representa uma linha de coluna → Primary Key
- Column: representa um valor contendo: Name, Value Timestamp

Chave-valor

Armazena um conjunto de dados, seja ele simples ou complexo, NoSQL Key-Value (chave-valor): **consiste em uma modelagem que indexa os dados a uma chave**. Com base similar a um dicionário python

- Bom desempenho em aplicações na nuvem.
- Menor capacidade de busca.

Uso: cache, sessão de usuário, carrinhos de compra

Iremos utilizar o Redis um banco de dados, cache, mensageria e fila.

- Alto desempenho;
- Estrutura de dados na memória;
- Versatilidade de uso;
- Replicação e persistência.

Documento

No modelo de banco de dados orientado a documentos é projetado para armazenar, recuperar e gerenciar informações orientadas a documentos, também conhecidas como dados semi-estruturados.

Introdução ao MongoDB

MongoDB:

- Código aberto
- Orientado a documentos
- Alta performance
- Schema-free
- Utiliza json para armazenamento dos dados

- Suporte a índices
- Auto-sharding (escalonamento horizontal)
- Map-reduce
- GridFS
- Rica linguagem de consulta