

Afinal, o que é SQL?

SQL é a sigla para “*Structured Query Language*”, que significa, traduzindo para o português, “Linguagem de Consulta Estruturada”. Trata-se de uma linguagem de consulta a banco de dados relacionais.

Com o SQL você pode executar vários comandos para criar, alterar, gerenciar, consultar, etc no seu banco de dados.

Costumamos dizer que **bancos SQL** seguem uma modelagem relacional, pois estes se baseiam no fato de que todos seus dados sejam guardados em tabelas.

E NoSQL? O que é?

NoSQL (*Not Only SQL*) é o termo utilizado para banco de dados não relacionais de alto desempenho onde, geralmente, o SQL não é utilizado como linguagem de consulta.

O **NoSQL** foi criado para ter uma performance melhor e uma escalabilidade mais horizontal para suprir necessidades onde os bancos relacionais não são eficazes.

No geral, temos 5 tipos de bancos de dados **NoSQL**:

Documento

Os dados são armazenados como documentos. Os documentos podem ser descritos como dados no formato de chave-valor, como por exemplo, o padrão **JSON**.

Um exemplo de banco de dados neste formato é o **MongoDB**.

Colunas

Os dados são armazenados em linhas particulares de tabela no disco, podendo suportar várias linhas e colunas, além de permitir sub-colunas.

Um banco de dados dessa família, por exemplo, é o **Cassandra**.

Grafos

Os dados são armazenados na forma de grafos (vértices e arestas).

O **Neo4j** é um banco que utiliza grafos.

Chave-valor

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

Um exemplo é o banco de dados **Riak**.

SQL vs NoSQL

O conceito de modelo relacional (SQL) se baseia no fato de que todos os dados sejam guardados em tabelas.

Ao modelo não-relacional (NoSQL) se aplica o conceito de *schema*: uma chave de valor é que é utilizada para recuperar valores, conjunto de colunas ou documentos.

Quais são as diferenças?

O SQL tem certa dificuldade em conciliar a demanda por escalabilidade. Quanto à escalabilidade do NoSQL, devemos levar em consideração a modelagem do sistema. Do que adianta termos um sistema super simples e querermos utilizar o NoSQL apenas pela escalabilidade? Todo o ganho de performance seria perdido quando rodássemos a aplicação.

Um ponto forte do SQL é quanto à consistência das informações.

Já o NoSQL garante o último valor atualizado, isso se nenhuma atualização for realizada até o momento da consulta.

Quanto à **segurança**, ambos estão suscetíveis a ataques.

NewSQL: o melhor dos dois mundos?

Os bancos de dados NewSQL buscam promover a mesma melhoria de desempenho e escalabilidade dos sistemas NoSQL, não abrindo mão dos benefícios dos bancos de dados tradicionais da linguagem SQL, nem das propriedades ACID.

Mike Stonebreaker, fundador do **VoltDB** (um dos bancos de dados desse novo modelo), destacou a vantagem dos bancos de dados NewSQL por proporcionarem consultas em tempo real, além de maior capacidade de processamento.

Segundo *Mike*, há um custo grande em não usar SQL, sendo exigido trabalho excessivo dos desenvolvedores para compensar sua ausência.

Diferente dos *SGBD* tradicionais, que eram considerados soluções para qualquer tipo de aplicação, os *NewSQL* utilizam uma estratégia diferente, onde cada novo sistema desenvolvido visa atender a uma necessidade específica do mercado.

E eles fazem isso de forma separada, terminando com o antigo conceito de ter um único sistema que sirva para qualquer tipo de aplicação, fazendo com que os bancos de dados sejam especialistas para um propósito.

Isso impede que um número absurdo de funções e comportamentos desnecessários para uma determinada aplicação sejam gerados.

STONEBRAKER e CATTEL (2011) definem cinco características de um **SGBD NewSQL**:

- Linguagem SQL como meio de interação entre o *SGBD* e a aplicação;
- Suporte para transações *ACID*;
- Controle de concorrência não bloqueante, para que as leituras e escritas não causem conflitos entre si;
- Arquitetura que forneça um maior desempenho por nó de processamento;
- Arquitetura escalável, com memória distribuída e com capacidade de funcionar em um aglomerado com um grande número de nós.

Vantagens e desvantagens: SQL, NoSQL e NewSQL

Na tabela a seguir temos um comparativo das características dos bancos de dados relacionais (*RDBMS*), NoSQL, e NewSQL, com suas capacidades e pontos fortes.

Característica	RDBMS	NoSQL	NewSQL
Cumprimento ACID (dados, integridade de transações)	Sim	Não	Sim
OLAP / OLTP	Sim	Não	Sim
A análise dos dados (agregados, transformar, etc.)	Sim	Não	Sim
Rigidez do esquema (mapeamento rigoroso da modelo)	Sim	Não	Talvez
Flexibilidade formato de dados	Não	Sim	Talvez
A computação distribuída	Sim	Sim	Sim
Escala para cima (vertical) / Dimensionar (horizontal)	Sim	Sim	Sim
Desempenho com crescimento de dados	Rápido	Rápido	Muito Rápido
Sobrecarga de desempenho	Enorme	Moderado	Mínimo
Popularidade / Suporte comunidade	Enorme	Crescente	Crescendo lentamente

A partir dessa tabela podemos observar que o NewSQL é um aprimoramento do RDBMS, utilizando o melhor dos dois mundos.

Ele manteve totalmente as características dos bancos de dados padrões, implementando as inovações do NoSQL.

Os itens em que foram marcados com “Talvez” dependem do Banco NewSQL que estamos utilizando.

Como vimos, cada banco tem um propósito específico. Existem alguns bancos NoSQL especialistas com o poder do NewSQL, como é o caso do **MariaDB** (Desenvolvido pelo criador do MySQL).

Os melhores bancos de dados

Vou listar aqui algum dos bancos de dados NoSQL e SQL que eu considero que valem a pena serem mencionados.

NoSQL

Aerospike

Banco de dados NoSQL que oferece uma vantagem de velocidade de memória, atraindo empresas de anúncios de alta escala e aquelas que precisam de tempos de resposta em milissegundo.

Aerospike está apostando em novas categorias, incluindo jogos, *e-commerce* e segurança, onde a baixa latência é tudo.

Apache Cassandra

Os pontos fortes são a modelagem de dados *NoSQL* e escalabilidade linear flexível em *hardware commodity* por conta do uso de cluster.

Amazon DynamoDB

Foi desenvolvido pela Amazon para incrementar o seu próprio negócio *e-commerce* em rápido crescimento, tendo seus serviços altamente escaláveis.

Inspirou o **Cassandra**, **Riak**, e outros projetos NoSQL no processo.

MongoDB

É o banco de dados mais popular NoSQL, com mais de sete milhões de downloads e centenas de milhares de implantações.

Sua popularidade se deve à facilidade de desenvolvimento e manejo flexível dos dados. Muito utilizado em aplicações de redes sociais web e móvel.

HBase

É o banco de dados que roda em cima do **HDFS** (*Hadoop Distributed File System* – sistema de arquivos distribuído projetado para rodar em *hardware commodity*), por isso dá aos usuários a capacidade única de trabalhar diretamente com os dados armazenados no Hadoop.

As características incluem grande escalabilidade.

NewSQL

MemSQL

Como o próprio nome sugere, é operado em memória.

Ele é um sistema de banco de dados de alta escala por sua combinação de desempenho e compatibilidade com o SQL transacional e ACID na memória, adicionando uma interface relacional em uma camada de dados *in-memory*.

VoltDB

Projetado por vários pesquisadores de sistema de banco de dados bem conhecidos, esse banco oferece a velocidade e a alta escalabilidade dos bancos de dados NoSQL, mas com garantias ACID, e sua latência em milissegundo e integração com **Hadoop**.

SQLFire

Servidor de banco de dados *NewSQL* da **VMware**, desenvolvido para escalar em plataformas nas nuvens e tomar as vantagens de infraestrutura virtualizadas.

MariaDB

Foi desenvolvido pelo criador do *MySQL* e é totalmente compatível com o *MySQL*. Também pode interagir com os bancos de dados NoSQL, como *Cassandra* e *LevelDB*.

NoSQL, NewSQL e sua relação com Big Data

As **aplicações Big Data** originaram-se com a chegada da Web 2.0, evoluindo com grande velocidade, e tem como proposta dar ao usuário informações úteis de forma rápida ou permitindo que a empresa tome boas decisões no nível de negócios.

Com a utilização dessas aplicações por gigantes da internet, como **Google, Facebook e Youtube**, o conceito de *BigData* se tornou ainda mais popular, sendo considerada pela equipe da **Gartner** umas das tecnologias que terão maior importância na modernização da gestão da informação, com fortes impactos na infraestrutura da informação e governança das empresas.

Segundo a **Gartner**, o *Big Data* suporta soluções de processamento para uma variedade de dados novos e já existentes, trazendo benefícios reais para os negócios. Ainda assim, o processamento do volume e variedade de dados continuam sendo soluções técnicas, a menos que sejam parte das metas de negócio.

O aspecto essencial do Big Data não é a habilidade técnica de processamento de dados, mas os benefícios que podem ser alcançados utilizando suas análises.

Os dados hoje são gerados de várias maneiras e formam uma grande massa para análise, originando o problema com os bancos de dados relacionais, devido a sua falta de capacidade de processar tanta informação com alta disponibilidade, otimização e escalabilidade.

Com isso surge a necessidade de sistemas de bancos de dados mais eficientes, motivando a utilização dos bancos de dados NoSQL e NewSQL.

A velocidade da disponibilização das informações é essencial para a tomada de decisões, já que uma informação tardia não tem utilidade, por isso os cinco “Vs” do Big Data (Volume, Velocidade, Variedade, Veracidade e Valor).

A necessidade de sistemas de bancos de dados mais eficientes para essas tarefas motivou a utilização do NoSQL e do NewSQL.

Os bancos NoSQL tiveram um crescente aumento com a difusão do Big Data algumas empresas chegaram a desenvolver bancos internos para atender inicialmente suas próprias demandas.

Esses bancos acabaram se tornando um produto, como o caso do **Big Table** e **Dynamo**, desenvolvidos para solucionar problemas com larga escala que sofriam no momento.

Outra coisa interessante é que os bancos NoSQL e os NewSQL são concebidos e evoluídos como soluções para os problemas enfrentados pelo Big Data.

Como as empresas gigantes são as que sofrem com problemas de larga escala, o que impulsiona de fato as melhorias nos bancos de dados é o uso de soluções analíticas como BI, Data Warehouse, ferramentas de ETL e de cubos.