

FACULDADE DE TECNOLOGIA DA PARAÍBA
ANÁLISE E DESENVOLVIMENTO DE SISTEMAS
BANCO DE DADOS AVANÇADO

THALLES RAMON PINHEIRO DE SOUSA
ANDRÉ LUIZ DE SANTANA SILVA ROCHA

NoSQL

CABEDELO-PB
NOVEMBRO/2010

THALLES RAMON PINHEIRO DE SOUSA
ANDRÉ LUIZ DE SANTANA SILVA ROCHA

NoSQL

Trabalho apresentado à disciplina de Banco de dados avançado sob a orientação da Prof. Renata Viegas para a obtenção de parte da nota do 2º estágio do período 2010.2.

CABEDELO-PB
NOVEMBRO/2010

NoSQL:

Princípios e Características

Thalles Ramon P. de Sousa, André Luiz de S. Silva Rocha

Departamento de Sistemas - Faculdade de Tecnologia da Paraíba (FATECPB)

João Pessoa – PB - Brasil

thallesrps0@gmail.com, andreluiz@live.com

Resumo. *Este artigo visa abordar as principais características sobre o NoSQL, banco de dados não relacional que atualmente está em grande utilização e desenvolvimento. O NoSQL consiste em um banco de dados que utiliza a escalabilidade como seu ponto forte, aumentando assim sua capacidade de armazenamento de dados e velocidade. Grandes empresas mundiais já utilizam esta tecnologia, tais como Google, Twitter, Facebook.*

1. Informações Gerais

Com o mundo cada vez mais informatizado e o crescente fluxo de informações, os antigos Modelos Relacionais estão dando espaço a mais nova tecnologia desenvolvida para grandes empresas que necessitam de alta capacidade de armazenamento, o NoSQL.

Antes de falarmos no NoSQL, precisamos saber o porque do desenvolvimento desta tecnologia. A idéia principal dos projetistas do NoSQL é promover uma alternativa de alto armazenamento com velocidade e disponibilidade elevada. Algumas das grandes empresas de tecnologia atualmente, investem no desenvolvimento dos seus próprios SGBDs baseado na idéia NoSQL.

O Google, por exemplo, investe desde 2004, no BigTable, um banco de dados proprietário, desenvolvido para suprir as necessidades de armazenamento da empresa, totalmente baseado na filosofia de alto desempenho, escalabilidade e disponibilidade. Existem vários SGBDs atualmente, no qual vamos abordar-los mais a frente, contanto um exemplo importante e famoso é o Cassandra, projetado para ser um sistema de alta disponibilidade e escalabilidade, foi desenvolvido pelo site de relacionamento Facebook, para lidar com seu grande fluxo de informações. Em 2010, o Cassandra se consolidou de vez, sendo utilizado como alternativa pelo Twitter, ante o MySQL utilizado anteriormente^[1].

2. Por que “Not Only SQL”?

O Termo NoSQL, foi utilizado pela primeira vez em 1998, por Carlo Strozzi, como nome de seu SGBD, baseado no Modelo Relacional, sem interface SQL. ^[2]

Not Only SQL é um termo genérico para uma classe definida de banco de dados que rompe uma longa história de banco de dados baseados nos Modelos Relacionais. ^[3]

Este tipo de banco de dados começou a ser desenvolvido para suprir as necessidades de alto armazenamento, já discutidas neste artigo anteriormente. O NoSQL, está sendo tratado como o futuro do grande armazenamento de informações, geradas todos os dias. A importância é tanta, que as maiores empresas atualmente em tecnologia, já recorrem a este recurso para o tratamento de suas informações, desenvolvendo a cada dia, novas soluções para auxiliar e incrementar o NoSQL.

A idéia retirar as estruturas impostas aos SGBDs baseados no Modelo Relacional, garante que o NoSQL, em ocasiões, visto como retrocesso por alguns especialistas, armazene, busque, acelere e organize os dados de maneira objetiva, direta e livre sem nenhum relacionamento.

3. BD Relacional x BD NoSQL

Os bancos de dados relacionais, desde a década de 1970, revolucionaram o mundo da tecnologia, com sua prática forma de armazenamento de dados. Por outro lado, o nosso mundo é ágil, prático e necessita de inovações. A seguir, vamos discutir pontos positivos e negativos das duas tecnologias, com suas características marcantes, e onde é mais correto aplicar cada um dos casos.

A diferença mais importante a ser discutida, é a do escalonamento, é nesse ponto onde o NoSQL contém suas principais vantagens, ante os SGBDs relacionais, pois o NoSQL foi criado nesse intuito, de melhorar a escalabilidade dos servidores.

Os atuais bancos de dados relacionais são muito restritos em relação à escalabilidade, pois utilizam escalonamento vertical nos servidores, quanto mais dados, mais espaço no servidor e memória necessita. A diferença do NoSQL, é que utiliza escalonamento horizontal e tem grande facilidade de distribuição do dados, ou seja, com um aumento

de dados, aumenta-se o numero de servidores, que podem ser ou no de alta performance, barateando e otimizando o armazenamento. ^[2]

Outra vantagem da utilização do escalonamento horizontal, é que com a divisão dos dados em vários servidores, o volume dos dados por servidor é minimizado, pois, conjuntos de dados menores são mais fáceis de serem processados, armazenados ou gerenciados. ^[1]

Os bancos de dados relacionais oferecem como seus pontos fortes, segurança das informações, controle de ocorrência, recuperação de falhas, otimização de consultas, alguns processos, tais como de validação, verificação e garantias. Estes bancos garantem plenamente a integridade das informações, pois utilizam a questão das entidades e relacionamentos, baseados no Modelo Relacional.

Como outro ponto positivo, os bancos de dados relacionais têm ao seu favor a questão de já estarem no mercado em grande utilização há anos, creditando em seus usuários mais confiança e estabilidade.

A seguir veremos uma tabela comparativa sobre as questões mais discutidas neste artigo, a escalabilidade, consistência e disponibilidade.

	BD Relacional	BD NoSQL
Escalabilidade	Devido à estruturação do modelo, é possível, mas complexo.	É sem dúvida a principal vantagem do NoSQL. Possui mais flexibilidade na inclusão de dados porá não possuir estrutura.
Consistência	Ponto forte do Modelo Relacional. A regra de consistência presente garante um maior rigor à consistência das informações.	Não garante a consistência da informação, caso nenhuma informação seja atualizada, retornará a todos os pedintes o mesmo valor.
Disponibilidade	Este modelo não suporta eficientemente grande demanda, dado a dificuldade e distribuição de dados.	Outro ponto forte do BD NoSQL. Possui um alto grau de distribuição de dados, e garante um maior

		numero de solicitações.
--	--	-------------------------

Tabela 1: Pontos Fundamentais: BD Relacionais x BD NoSQL. ^[1]

Ao favor dos dois tipos de bancos de dados, está a possibilidade de serem multiusuários, garantindo que os usuários possam utilizar simultaneamente da base de dados.

Um fator importante, para o processo de escolha em utilização do NoSQL ou de um SGBD relacional, é a questão da disponibilidade do servidor. Se por exemplo, for uma aplicação web, que necessita de disponibilidade imediata, o NoSQL leva vantagem, pois, com a divisão de vários servidores, podendo os usuários acessarem os vários servidores ou no caso de queda de um deles, o sistema não será totalmente prejudicado. Caso não seja necessária uma disponibilidade tão imediata, o indicado será algum SGBD relacional.

As principais características do NoSQL que serão discutidas posteriormente se caracterizam por: escalabilidade horizontal, alta disponibilidade, menor tempo de busca/resposta e paralelismo de atualização de dados.

A seguir, conforme ilustrado na Tabela 1, uma breve comparação entre os banco de dados relacionais e os NoSQL.

Banco de Dados Relacionais	Banco de Dados NoSQL
O que é: Baseia-se no conceito de entidade e relacionamento, no qual, todos os dados são guardados em tabelas. Os dados são separados de forma única tentando diminuir ao máximo a redundância. Pois a informação é criada pelo conjunto dos dados, onde são as relações entre as tabelas que fazem esse serviço.	O que é: Uma solução alternativa para os bancos de dados relacionais. Possuem uma alta escalabilidade, disponibilidade, desempenho e menor tempo de resposta a consultas.
Características: tabelas, esquema definido, hierarquia, redundância mínima, entidade e relacionamento, formas normais, transações ACID (Atomicidade, Consistência, Isolamento, Durabilidade).	Características: registros, schema-free, tolerância a falha, escalabilidade, clusterização, mapreduce, sharding.
Necessidades: Sistemas locais,	Necessidades: Sistemas em

sistemas financeiros, sistemas corporativos, segurança da informação, consistência dos dados.	nuvem, análises sociais, alta escalabilidade, performance na consulta/escrita, replicação.
SGBDs: DB2, Firebird, InterBase, Microsoft SQL Server, MySQL, Oracle, PostgreSQL.	SGBDs: Cassandra, BigTable, MongoDB, CouchDB, Dynamo.
Algumas Empresas: SAP, OpenERP, Previdência Social, Caixa, Itaú, Salesforce, Vale.	Algumas Empresas: Twitter, Facebook, Digg, Amazon, LinkedIn, Google, Yahoo, The New York Times, Bit.ly, eBay.

Tabela 2: Comparação entre Banco de Dados Relacionais e Banco de Dados NoSQL. ^[2]

4. Características NoSQL

4.1 Escalabilidade Horizontal (scale out) significa adicionar mais nós ao sistema, tais como adicionar um novo servidor e um sistema de software que permita a distribuição do trabalho entre múltiplas máquinas. ^[4]

4.2 Replicação – Escalar por duplicação de informações

Consiste na copia das informações em mais de um banco para aumentar nossa capacidade de recuperar estas informações. Podemos dividir em duas “arquiteturas” principais:

- **Master-Slave:** Cada escrita em banco resulta em X escritas onde X é o número de slaves. Neste caso temos um banco “Master” que propaga cada write para os bancos “slaves”. Isto aumenta a nossa velocidade de leitura, mas não melhora a capacidade de escrita.
- **Multi-Master:** Aumentamos o número de Masters em nosso sistema e assim aumentamos nossa capacidade de escrita.

4.3 Schema-free: Um dos fatores que contribuem para um banco de dados NoSQL escalar é por causa da ausência de um schema (schema free). Todos os novos bancos tem em comum que eles são key-value stores, ou seja salvam, como o nome sugere, um conjunto de entradas formadas por uma chave associada a um valor e o valor poderia ser de qualquer tipo, um binário ou string que está sendo salvo de forma desnormalizada (schema-free).

4.4 Clusterização: Basicamente compreende um banco de dados armazenado e gerenciado por mais de um servidor, provê uma alta disponibilidade e um alto desempenho do sistema. Assim, a organização se beneficia diminuindo o tempo de inoperabilidade do banco de dados. Esse

processo vem como uma solução para reduzir gastos com estrutura de hardware.^[8]

4.5 Mapreduce: É um algoritmo, patenteado pela Google para gerenciamento em larga escala.^[7] Existem duas fases:

- Map: O nó principal recebe os dados, divide e partes menores e as envia aos outros nós para serem processados. Ao final do processamento estes nós devolvem o resultado ao nó principal.
- Reduce: O nó principal combina as respostas obtidas pelos outros nós gerando o resultado final do processamento.

4.6 Sharding: Consiste em dividir os dados horizontalmente, ou seja, quebrar as tabelas, diminuindo o seu número de linhas e separando-as em ambientes diferentes. Neste ponto todos os dados de uma partição não devem conter referências aos dados de outras partições, sendo que os dados em comum deverão ser replicados entre as bases.^[7]

5. Classificação dos Bancos de dados NoSQL

5.1 Sistemas baseados em armazenamento chave-valor

Baseia-se numa coleção de chaves únicas e de valores, os quais são associados com as chaves.

Exemplo: Dynamo

5.2 Sistemas orientados a documentos

Os documentos são as unidades básicas de armazenamento e estes não utilizam qualquer tipo de estruturação pré-definida, como é o caso das tabelas do modelo relacional.

Exemplo:

CouchDB -> Um documento no CouchDB é um objeto que possui um identificador único e que consiste de certos campos. Esse documento segue o formato JSON (Java Script Object Notation), padrão que visa a fácil legibilidade e independência da linguagem. Desenvolvido em 2008, oferece API estilo REST para documentos centrados em operações CRUD (create, retrieve, update, delete). Ele pode fazer retorno baseado em valores-chave e operar com Hadoop Map-Reduce para pesquisas não-triviais. Também é possível gerar visões usando funções Java-Script. Criar uma visão pode ser exaustivo, mas os retornos subseqüentes com a visão podem ser mais

rápidos. Suporta replicação multimaster e distribuição de dados por múltiplas instâncias; usa a ferramenta SpiderMonkey JavaScript, tornando-o apropriado para aplicativos Web, como Erlang, HTTP, JavaScript, PHP, Python e Ruby.

5.3 Sistemas orientados a colunas

Mudou-se de orientação a registro para orientação a colunas (tributos).

Exemplo: Cassandra, BigTable.

5.4 Sistemas baseados em grafos

Os dados são armazenados em nós de um grafo cujas arestas representam o tipo de associação entre esses nós.

Exemplo:

Neo4J -> Depois de muitos anos de desenvolvimento, a NeoTechnology lançou a versão 1.0 do Neo4j, um banco de dados em forma de grafo feito em Java. A InfoQ falou com o COO da NeoTechnology Peter Neubauer para entender melhor o novo lançamento do N4j e o que ele traz de novo para os desenvolvedores. ^[6]

O Neo4j kernel JAR, que tem cerca de 440k, está disponível sobre as licenças AGPLv3 e sobre uma licença comercial, com a última deve ser utilizada apenas para aplicação com código fechado. As informações salvas no Neo4j são representadas usando três pilares:

- **Nó:** Possui o mesmo conceito de uma instância de um objeto, e possui um ID único.
- **Relacionamento (arestas)** - Fornece uma ligação entre dois nós, ambos os nós possuem uma direção e um tipo de relacionamento.
- **Propriedade (atributo)** - São pares de String key/Objeto valor que podem existir tanto em um nó quanto em um relacionamento. ^[6]

Comparado com os bancos de dados relacionais, os bancos de dados em forma de grafos tendem a serem melhores quando temos um grande volume de complexidade, estruturas de dados com baixo acoplamento que mudam e sofrem frequentes consultas - em um modelo relacional queries complexas podem gerar um grande número de joins, o que pode causar problemas de performance. Neubauer explica esse problema com mais detalhes, dizendo:

O Neo4j vem solucionar o problema de queda de performance entre queries que envolvem muitos joins em um RDBMS. Através da representação de dados utilizando grafos, o Neo4j consegue navegar entre os nós e os relacionamentos com uma velocidade constante, independente da quantidade de dados que constituem o grafo. “Isso nos trás alguns outros efeitos como algoritmos de grafos muito rápidos, sistema de recomendação e um estilo OLAP de análise o que atualmente não é possível com configuração normais de um RDBMS”. [6]

6. SGBDs NoSQL

Com a necessidade do NoSQL, grandes empresas começaram a desenvolver seus próprios SGBDs baseados na filosofia NoSQL. A seguir, vamos listar alguns exemplos importantes, e definir algumas características.

Apache Cassandra

Desenvolvido inicialmente pelo site de relacionamento Facebook, é um projeto de sistema de banco de dados distribuído, altamente escalável, que foi desenvolvido na plataforma Java, reuni a arquitetura do Dynamo, da Amazon e o modelo de dados do BigTable, do Google. O Cassandra exerce com excelência a função de repositório de dados. Teve seu código-fonte aberto à comunidade em 2008. Agora é mantido por desenvolvedores da fundação Apache e colaboradores de outras empresas.

BigTable

Foi desenvolvido pelo Google, para distribuir dados por centenas de servidores e escalar por conjuntos de dados de até 1 petabyte. Uma grande variedade de aplicativos da empresa usa o Bigtable, incluindo índices Web, Google Earth, Maps, Blogger, Youtube e Gmail. O BigTable é proprietário, mas o modelo de dados existe em implementações de código aberto. Ele pode ser usado como input ou output para o MapReduce, que ativa o processo de distribuição de arquivos ou banco de dados usando funções de mapeamento e redução.

Dynamo

Desenvolvido pela Amazon em 2007, foi criado para oferecer armazenamento de valores-chaves de dados de alta disponibilidade, permitindo atualizações para sobreviver às falhas de servidor e rede.

Apache CouchDB

É um banco de dados orientado a documentos e de código fonte aberto escrito na linguagem Erlang. Foi desenvolvido e mantido pela Fundação Apache e busca replicação e escalabilidade horizontal.

MongoDB

É um banco de dados não relacional que combina as melhores funcionalidades de orientação a documentos, hashes e RDBMSs. É um banco de dados orientado a documentos, escalável, livre de esquema, de alto desempenho e código aberto escrito em C++.

7. Principais Empresas Utilizadoras

Grandes empresas hoje utilizam o NoSQL em grande parte dos seus sistemas, tais como as gigantes da tecnologia, Google, Amazon, Yahoo, eBay.

Outras grandes empresas que contam com grande fluxo de informações utilizam esta tecnologia, Facebook, Twitter, The New York Times são alguns exemplos. Abaixo, vamos descrever as principais características destas empresas.

Google Inc.: “É uma multinacional estadunidense de computação em nuvem pública, buscador na Internet e uma corporação de tecnologias de publicidade. O Google hospeda e desenvolve uma série de serviços e produtos baseados na Internet e gera lucro principalmente através da publicidade pelo seu programa AdWords. A missão declarada da empresa desde o início foi "organizar a informação mundial e torná-la universalmente acessível e útil". Em 2006, a empresa mudou-se para sua atual sede, em Mountain View, Califórnia.” [5]

Amazon: “É uma empresa de comércio eletrônico dos Estados Unidos da América com sede em Seattle, estado de Washington. Foi uma das primeiras companhias com alguma relevância a vender produtos na Internet. Amazon inclui, igualmente, a Alexa Internet, a9.com, e a Internet Movie Database (IMDb).” [5]

Yahoo! Inc.: “É uma empresa norte-americana de serviços de Internet com a missão de ser "o serviço de Internet global mais essencial para consumidores e negócios". Opera um portal de Internet, um diretório web, e outros serviços, incluindo o popular Yahoo! Mail. Foi fundado por David

Filo e Jerry Yang, formandos da Universidade de Stanford em janeiro de 1994 e incorporado no dia 2 de março de 1995. A sede da empresa é em Sunnyvale, Califórnia. Enquanto a popularidade do Yahoo! crescia, novos serviços iam surgindo, tornando o Yahoo! uma parada obrigatória para todas novas tendências da Internet. Estes incluem: o Yahoo! Messenger, um mensageiro instantâneo, o Yahoo! Groups, serviço muito popular de criação de mailing lists por assunto de interesse, bate-papo e jogos online, vários portais de notícias e informação, compras online e leilões.”^[5]

eBay: “É uma empresa de comércio eletrônico fundada nos Estados Unidos, em Setembro de 1995, por Pierre Omydiar. Atualmente é o maior site do mundo para a venda e compra de bens, é o mais popular shopping da internet, e possivelmente foi a pioneira neste tipo de trabalho. O eBay possuía mais de 181 milhões de membros registrados em todo do mundo ao fim de 2005. O eBay tem por finalidade fornecer uma plataforma global de negociações, onde qualquer pessoa pode negociar qualquer coisa. Em 2005, o eBay gerou mais de 21 bilhões de dólares em mercadorias transacionadas.”^[5]

Facebook: “É uma rede social lançada em 4 de fevereiro de 2004. Foi fundado por Mark Zuckerberg, um ex-estudante de Harvard. Inicialmente, a adesão ao Facebook era restrita apenas aos estudantes da Universidade Harvard. Ela foi expandida ao Instituto de Tecnologia de Massachusetts (MIT), à Universidade de Boston, ao Boston College e a todas as escolas Ivy League dentro de dois meses. Desde 11 de setembro de 2006, apenas usuários com 13 anos de idade ou mais podem ingressar.”^[5]

Twitter: “É uma rede social e servidor para microblogging, criado em 2006 por Jack Dorsey, que permite aos usuários enviar e receber atualizações pessoais de outros contatos em textos de até 140 caracteres, conhecidos como "tweets", por meio do website do serviço, por SMS e por softwares específicos de gerenciamento. As atualizações são exibidas no perfil de um usuário em tempo real e também enviada a outros usuários seguidores que tenham assinado para recebê-las.”^[5]

Referências

- [1] Ricardo W. Brito, “Banco de Dados NoSQL x SGBDs Relacionais: Análise Comparativa”
- [2] Jean Carlo Nascimento aka Suissa, “Introdução ao NoSQL” <http://www.nosqlbr.com.br/introducao-ao-nosql.html>
- [3] Wikipédia, “NoSQL” <http://pt.wikipedia.org/wiki/NoSQL>
- [4] Edmar Ferreira, “Escolhendo entre escalabilidade horizontal e escalabilidade vertical”. <http://escalabilidade.com/2010/09/21/escolhendo-entre-escalabilidade-horizontal-e-escalabilidade-vertical/>
- [5] Wikipédia, <http://pt.wikipedia.org/wiki>
- [6] Michel Hunger, “Neo4j – Base de Dados NoSQL baseada em Java”
- [7] Edmar Ferreira, “Introdução ao NoSQL parte II” <http://escalabilidade.com/2010/04/06/introducao-ao-nosql-parte-ii/>
- [8] InfoWester “Cluster: Principais Conceitos”, <http://www.infowester.com/cluster.php>