

# **NOSQL - Armazenamento e Recuperação de dados não-relacional**

**Bruno Santos de Lima<sup>1</sup>, Leandro Ungari Cayres<sup>1</sup>**

<sup>1</sup>Faculdade de Ciências e Tecnologia  
Universidade Estadual Paulista  
"Júlio de Mesquita Filho"

Caixa Postal 19060-900 – Presidente Prudente – SP – Brasil

brunoslima4@gmail.com, leandroungari@gmail.com

**Abstract.** Traduzir o resumo...

**Resumo.** Fazer... (última coisa)

## **1. Introdução**

Irrefutavelmente, é possível notar o crescimento da tecnologia com isso o número de dispositivos eletrônicos, sejam eles supercomputadores, computadores desktop ou aplicações moveis, vem aumento em grandes proporções tornando-se presentes nas mais variadas situações do dia a dia, cada qual buscando manipulando diversos dados referentes ao contexto em que eles são aplicados, seja no âmbito empresarial, hospitalar, juridico, academico e diversos outros setores.

Com as diversas aplicações e diversos dispositivos atuando diariamente na vida da sociedade no que diz respeito a manipulação de dados, surge a necessidade de realizar inicialmente o armazenamento desses dados afim de poder manipula-los de forma eficiente e segura para abstrair informações relevantes sobre determinado contexto onde esses dados estão inseridos e com isso facilitar, ajudando a sociedade a utilizar essas informações para realizar algumas atividades diarias, o responsavel por realizar esse armazenamento e consequentemente a manipulação desses dados é o chamado Banco de dados.

Quando o assunto Banco de dados é abordado logo é pensado como os dados não armazenados e manipulados? para isso é seguido um modelo, de certa forma uma padronização, para responder esta pergunta, o modelo mais conhecido e utilizado em larga escala por Sistemas Gerenciadores de Banco de Dados (SGBD's) é denominado modelo relacional [Codd 1970] criado por Edgar Frank Codd em 1970, o modelo relacional substituiu o modelo anterior chamado de modelo hierarquico tornando-se um padrão referencia para este quesito, sendo utilizado pela grande maioria dos SGBD's, onde podemos citar por exemplo os mais conhecidos atualmente: MySQL, Oracle, SQL Server e PostgreSQL.[Brito 2010]

Contudo atualmente o modelo relacional produz a flexibilidade que o cenario atual exige e vem sendo substituido em algumas areas pelos chamados NoSQL, este artigo tem como foco abordar o modelo NoSQL apresentando suas caracteristicas, formas de armazenamento e recuperação dos dados e alguns contrastes ao modelo relacional.

Este artigo esta organizado como segue: na seção 2.....

## 2. NOSQL

Com o decorrer do tempo foi pensado em modelagens alternativas ao tão utilizado modelo relacional, tais pensamentos justificam-se devido a algumas limitações presente no modelo relacional, limitações essas relacionadas a falta de flexibilidade fornecida pelo modelo relacional.

O NoSQL foi idealizado em 1998 e tem como significado de seu termo "não somente SQL", pois a ideia é que ambos os modelos, tanto o relacional quanto o NoSQL, possam coexistir, cada um em seu espaço de atuação, ou seja, existem aplicações em que o modelo relacional é mais interessante de se utilizar, porém também existem aplicações em que o NoSQL é mais viável, por exemplo em aplicações distribuídas, onde o armazenamento dos dados é distribuído, sendo esses dados de grande escala, como por exemplo redes sociais tais como Facebook e Twitter e podem acumular Terabits de dados todos os dias, não existindo um esquema fixo, assim para este tipo de aplicação o NoSQL é mais vantajoso.[Gueidi et al. 2016]

Os Banco de dados NoSQL fornecem mecanismos de armazenamento e recuperação de dados que são modelados como não estruturado, diferente da forma disposta por tabelas utilizada pelos Banco de dados Relacionais [Zhao et al. 2014], sendo assim o NoSQL não é relacional, essa seria a principal característica do NoSQL, porém seu conceito é muito amplo podendo existir banco de dados NoSQL com algumas características diferentes de outros banco de dados NoSQL, isso deve-se a flexibilidade que ele prove, entretanto existem um grupo de características comuns entre os banco de dados NoSQL, tais características são descritas abaixo:

- Distribuído: O banco de dados NoSQL são em geral distribuídos, assim várias máquinas são responsáveis por armazenar e prover esses dados quando os mesmos forem requisitados.
- Escalabilidade horizontal:
- Grande volume de dados: Um dos propósitos do NoSQL é ter a capacidade de armazenar uma grande quantidade de dados de modo mais rápido possível.
- SQL não é suportado: De uma modo geral nos Banco de dados NoSQL não é utilizado a linguagem SQL, assim cada sistema tem seu próprio modo de realizar consultas, existem estudos e também já existe uma alternativa proposta de unificar as consultas em Banco de dados NoSQL.
- Base:

## 3. Categorização dos Banco de dados NOSQL

O NOSQL pode ser classificado em cinco categorias [Nayak et al. 2013] sendo elas: Banco de dados de armazenamento chave-valor, Banco de dados orientado por colunas, Banco de dados orientado a documentos, Banco de Dados orientado a grafos e Banco de Dados orientado a objetos.

Em Banco de dados de armazenamento chave-valor os dados tem duas informações associadas a ele, uma sendo a chave e a outra sendo o valor, onde esse valor é literalmente o dado armazenado, como isso seu mecanismo de funcionamento atua de forma similar a um hash, trazendo como um aspecto positivo a rapidez em sua manipulação, não só isso mas também alta concorrência e armazenamento em massa, porem o grande ponto negativo deve-se a falta de uma esquematização o que deixa a visualização da organização desses dados dificultada.[Nayak et al. 2013]

Em Banco de dados orientado por coluna consiste em uma organização onde o dados não são armazenados em linhas mas sim em colunas, ou seja, ocorre uma mudança de orientação agora sendo por atributos, colunas, e não mais por registros, tuplas. Dois exemplos dessa categoria de Banco de dados orientado por colunas é o Cassandra e o BigTable.[Brito 2010][Jatana et al. 2012]

Em Banco de dados orientado a documentos os dados são armazenados na forma de documentos que tem uma chave unica associada a ele, essa chave é uma string que representa o caminho, ou URI, onde o documento está armazenado, no geral é fonecido uma API ou uma linguagem de consulta para que esses documentos possam ser recuperados rapidamente. [Jatana et al. 2012] Uma vantagem relacionado a essa categoria está no aproveitamento de espaço de armazenamento, visto que um documento tem um tamanho necessario para armazenar os dados em que nele estão sem que ocorra desperdicio de espaço, já no modelo relacional os dados são armazenados em tabelas sendo que alguns campos dessa tabela podem ficar vazios desperdiçando espaço, um exemplo de SGBD que utiliza este modelo o MongoDB. Esse modelo deve ser evitado em aplicações que envolvem muitos relacionamentos.[Nayak et al. 2013]

Em Banco de dados orientado a grafos armazenam os dados na forma de um grafo, sendo que cada nó é um objeto e o relacionamento entre esses objetos é expresso pelas arestas que ligam dois nós. Esse tipo de modelo é indicado para redes sociais, sistemas bioinformatica entre outros, pois os nós podem representar por exemplo pessoas ou negocios semelhantes aos objetos utilizados em determinadas linguagens de programação, entretanto apresenta um desvantagem pois é difícil realizar um agrupamento desses dados.[Nayak et al. 2013][Jatana et al. 2012]

Em Banco de dados orientado a objetos os dados são armazenados na forma de objetos, objetos analogos aos do paradigma de programação orientado a objetos oferecendo recursos como encapsulamento, polimorfismo e herança, cada objeto tem um identificador unico que representa o objeto unicamente.[Nayak et al. 2013]

#### **4. Sistemas Gerenciadores de Banco de Dados Não relacionais**

##### **5. Banco de dados: relacional vs não relacional**

##### **6. NoSQL: Vantagens e desvantagens**

Como toda tecnologia o NoSQL também apresenta um conjunto vantagens e desvantagens em sua utilização, é importante acrescentar que essas vantagens e desvantagens em sua utilização pode sofrer variações de acordo com a necessidade de uma aplicação ou organização, ou seja, dependendo do proposito da aplicação as desvantagens são minimizadas devido a grande parcela de beneficios. A seguir é discutido algumas vantagens e desvantagens em sua utilização.

## 6.1. Vantagens

Os Bancos de dados NoSQL tem como sua grande vantagem o processamento mais rápido dos dados que estão armazenados se comparado com os Bancos de dados Relacionais. O fator que torna o Banco de dados relacional mais lento em processamento é caracterizado pelas chamadas restrições ACID, cujo a sigla representa Atomicidade, consistência, isolamento e durabilidade.

Atomicidade significa que toda atualização é executada por completa ou não, já a consistência é o fato de toda transação estar proibida de quebrar as regras do banco de dados, isolamento deve-se ao fato de que cada aplicação faça transações de modo independente a outras aplicações que estão atuando em paralelo. Essas restrições são importantes no quesito precisão, porém se utilizadas causam uma perda de desempenho no processamento, o NoSQL não utiliza o suporte para as restrições ACID aumentando o seu desempenho de processamento.[Leavitt 2010]

A não utilização das restrições ACID não torna o NoSQL um Banco de dados inconsistente ou inseguro, apenas diminui essas capacidades, porém não as elimina por completo devido a utilização da BASE como uma alternativa ao conjunto ACID.[Pritchett 2008]

Um outro aspecto para o processamento mais rápido de dados nos Bancos de Dados NoSQL é a simplificação do modelo, ou seja, de um modo geral seus modelos são mais simples facilitando seu processamento. Outra vantagem interessante do NoSQL está na flexibilidade que o modelo prove, essa flexibilidade permite que as organizações e desenvolvedores utilizem seus aplicativos da forma que melhor atenda suas necessidades.[Leavitt 2010]

## 6.2. desvantagens

O NoSQL apresenta ainda alguns desafios, sendo um deles a maior complexidade devido a não utilização de SQL, todas as consultas devem ser realizadas manualmente através da programação, esse fator pode não ser um problema em consultas simples, porém podem potencializar um problema para as tarefas mais difíceis. A falta de ferramentas para gerenciamento ou mesmo para utilização como suporte parte do cliente ainda é um problema, onde essa falta de ferramentas pode estar relacionado ainda ao desconhecimento da tecnologia que pode ser um outro problema.[Leavitt 2010]

Devido a não utilização de restrições ACID quesitos como confiabilidade e a consistência tornam-se um problema em Banco de dados NoSQL sendo uma desvantagem, principalmente para certos tipos de aplicações como bancárias este modelo não é recomendado.[Leavitt 2010]

## 7. Armazenamento e recuperação em banco de dados não relacional

## 8. Conclusão

### References

Brito, R. W. (2010). Bancos de dados nosql x sgbd's relacionais: análise comparativa. *Faculdade Farias Brito e Universidade de Fortaleza*.

- Codd, E. F. (1970). A relational model of data for large shared data banks. *Commun. ACM*, 13(6):377–387.
- Gueidi, A., Gharsellaoui, H., and Ahmed, S. B. (2016). A nosql-based approach for real-time managing of embedded data bases. In *Computer Applications & Research (WSCAR), 2016 World Symposium on*, pages 110–115. IEEE.
- Jatana, N., Puri, S., Ahuja, M., Kathuria, I., and Gosain, D. (2012). A survey and comparison of relational and non-relational database. *International Journal of Engineering Research Technology IJERT*.
- Leavitt, N. (2010). Will nosql databases live up to their promise? *Computer*, 43(2). IEEE.
- Nayak, A., Poriya, A., and Poojary, D. (2013). Type of nosql databases and its comparison with relational databases. *International Journal of Applied Information Systems*, 5(4):16–19. Published by Foundation of Computer Science, New York, USA.
- Pritchett, D. (2008). Base: An acid alternative. *Queue*, 6(3):48–55.
- Zhao, G., Lin, Q., Li, L., and Li, Z. (2014). Schema conversion model of sql database to nosql. In *P2P, Parallel, Grid, Cloud and Internet Computing (3PGCIC), 2014 Ninth International Conference on*, pages 355–362. IEEE.