

## ANÁLISE DE MODELOS DE BANCOS DE DADOS NÃO RELACIONAIS

### ANALYSIS OF NON-RELATIONAL DATABASE MODELS

*Lucas Pereira da Silva*

<https://orcid.org/0000-0003-3476-8651>

*Centro Paula Souza – Fatec Indaiatuba/SP*

*lpsilva510@gmail.com*

*Orientador: Prof<sup>a</sup> Dr<sup>a</sup> Maria das Graças J. M. Tomazela*

<https://orcid.org/0000-0002-5471-2658>

*Centro Paula Souza – Fatec Indaiatuba/SP*

*graca.tomazela@fatec.sp.gov.br*

**RESUMO:** O aumento no volume de dados gerados nos diversos sistemas começou a se tornar um problema para as empresas. Os bancos de dados relacionais já não eram mais o bastante para lidar com tanta informação em tão pouco tempo. Por isso tais empresas buscaram alternativas para conseguirem armazenar todas essas informações e ainda serem capazes de consultá-las em pouquíssimo tempo. A partir disso surgiram os Bancos de Dados Não Relacionais, e com eles, por consequência, a dúvida sobre qual modelo deveria ser utilizado em cada tipo de aplicação. Assim, o objetivo deste trabalho foi testar modelos de banco de dados NoSQL de modo a identificar qual deles tem o melhor desempenho em um determinado cenário. Para alcançar o objetivo proposto, foi realizado um estudo prévio para a escolha de quais gerenciadores de bancos de dados seriam utilizados, dessa forma, algumas API's do Microsoft Azure Cosmos DB foram escolhidas. Para que os testes fossem realizados com dados reais, foram selecionados dados da Síntese de Indicadores Sociais, divulgada pelo IBGE. Esses dados foram manipulados de vinte a trinta vezes em atividades de inserção, consulta e exclusão para que o tempo médio de cada atividade pudesse ser calculado e seus resultados comparados. A partir dessa comparação foi possível concluir que, no cenário avaliado, o modelo de Documentos obteve o melhor desempenho. Espera-se que este estudo possa auxiliar futuros projetistas de banco de dados a escolherem o modelo NoSQL ideal para aplicações de cenários semelhantes a esse, poupando o tempo gasto fazendo essa análise preliminar.

**ABSTRACT:** The increase in the volume of data generated in different systems started to become a problem for companies. The relational databases were no longer enough to handle so much information in a short time. Therefore, these companies are looking for alternatives to be able to store all this information and are still able to consult them in a short time. From that, Non-Relational Databases arose, and with them, consequently, a doubt about which model should be used in each type of application. Thus, the objective of this work was to test NoSQL database models to identify which one has the best performance in a specific scenario. To achieve the proposed objective, a preliminary study was carried out to choose which database managers used, thus, some Microsoft Azure Cosmos DB's APIs were selected. For which tests were performed with real data, data from the Synthesis of Social Indicators, released by IBGE, were selected. These data were manipulated twenty to thirty times on insert, select and delete activities for the average time of each activity that can be calculated, and their results compared. From this comparison, it was possible to conclude that, in the evaluated scenario, the model of recorded documents or the best performance. It is hoped that this study can help future database designers to choose the

ideal NoSQL model for applications like this scenario, recovering the time spent doing this preliminary analysis.

**PALAVRAS-CHAVE:** Comparação de Modelos. Modelo Não Relacional. Sistema Gerenciador de Banco de Dados. Volume de Dados.

**KEYWORDS:** Model Comparison. Non-Relational Model. Data Base Management System. Data Volume.

## 1 INTRODUÇÃO

Por meio de pesquisas de funções de automação de escritório realizadas pela IBM, durante as décadas de 1960 e 1970, surgiram os primeiros fundamentos de bancos de dados relacionais. À época as empresas percebiam que era muito caro manter uma equipe de funcionários para organizar seus arquivos, portanto os investimentos em pesquisas valiam muito a pena. Nesse período diversas pesquisas foram realizadas, criando grandes descobertas, algumas utilizadas até os dias de hoje. Uma delas era um sistema no qual o usuário seria capaz de acessar informações guardadas em tabelas, por meio de comandos em inglês.

Segundo Silberschatz et al. (2006, p. 4), “Um sistema de banco de dados é uma coleção de dados inter-relacionados e um conjunto de programas que permitem aos usuários acessar e modificar esses dados”.

Um Sistema Gerenciador de Banco de Dados (SGBD), é um software que permite a criação, alteração, inserção e eliminação de dados em um banco de dados. Seus principais requisitos são: a segurança; a integridade; o controle de concorrência; e a recuperação e tolerância de falhas. Um sistema gerenciador de banco de dados funcional necessita conseguir recuperar dados de maneira eficiente. Silberschatz et al. (2006), ainda diz que, bancos de dados tem a importante função de conceder aos usuários um olhar abstrato dos dados, em outras palavras, o SGBD esconde certas informações de como tais dados estão guardados.

O atual padrão de banco de dados em aplicações comerciais é o Modelo Relacional, em que os dados são armazenados em conjuntos de tabelas bidimensionais simples. Porém, o volume de dados gerado nas mais diversas aplicações existentes nos dias de hoje é imenso e, quanto mais o tempo passa, maior é a quantidade de dados que será armazenada. Assim, o modelo relacional não está mais se enquadrando a todas as situações como afirma Diana e Gerosa (2010, p.1) “A quantidade de dados gerados, armazenados e processados atingiu escalas inéditas com a Web 2.0. Soluções além de

banco de dados relacionais estão sendo implementadas e utilizadas por organizações da Web 2.0 para lidar com dados desse volume.”.

Um bom exemplo de sistema que possui um gigantesco volume de dados sendo atualizado a todo momento são as redes sociais, que a cada segundo tem algum usuário compartilhando textos, fotos, vídeos, notícias e o que mais for possível compartilhar.

Com essa crescente demanda por volume de dados e desempenho é que chega um novo modelo de SGBD, conhecido como NoSQL, Not only SQL, que segundo Souza et al. (2014, p.1), “Este novo modelo foi proposto com o objetivo de atender ao gerenciamento de grandes volumes de dados semiestruturados ou não estruturados, que necessitam de alta disponibilidade e escalabilidade.”.

Diante desse contexto, Diana e Gerosa (2010, p.2) afirmam “Como essas tecnologias são muito recentes, existem poucas recomendações indicando em que contexto usar determinado paradigma.”. Portanto, a realização de um estudo que possa levantar as principais características e funções de alguns SGBD’s pode ser de grande ajuda para projetistas de banco durante a escolha de um gerenciador não relacional.

Tendo como base o cenário exposto, esse trabalho foi norteado pela pergunta “Qual o modelo de bancos de dados NoSQL mais adequado para o cenário dos testes executados?”. Desta forma teve como objetivo analisar modelos de banco de dados não relacionais para, a partir dos resultados obtidos, identificar os pontos positivos e negativos de cada modelo para o cenário.

## 2 METODOLOGIA

A metodologia utilizada durante o desenvolvimento deste estudo foi a Pesquisa Experimental que, segundo Gil (2002), consiste na determinação de um objeto de estudo, seleção das variáveis capazes de influenciá-lo e definição de meios de controle e observação dos efeitos que esta variável produz neste objeto.

Para que tal método fosse utilizado foi necessário definir a hipótese do trabalho, que é a de que caso exista um estudo detalhado que auxilie futuros projetistas de banco de dados a escolherem o melhor modelo NoSQL para seus projetos, poderá haver uma redução no tempo de desenvolvimento de tais projetos.

Desse modo foi executada uma comparação entre os modelos de bancos de dados NoSQL, focando em características das arquiteturas e assim permitindo a avaliação do desempenho de cada modelo durante a inserção, consulta e exclusão de uma grande quantidade de dados simultaneamente.

Para o desenvolvimento deste trabalho foi utilizado o Azure Cosmos DB da Microsoft com as APIS MongoDB (Colunar), Tabelas (Chave-Valor) e SQL (Documentos).

O modelo de dados utilizado para a realização dos testes comparativos deste trabalho foi desenhado com base na tabela divulgada pelo IBGE em 2016, com dados sobre o ano de 2015, que expõe informações sobre a média de horas semanais trabalhadas no trabalho principal, média de horas gastas em afazeres domésticos e a jornada total das pessoas com 16 anos ou mais de idade. Após a definição e criação do modelo de dados foram criadas as bases de dados para a realização dos testes.

O trabalho consistiu na realização de testes de comparação entre os modelos de bancos de dados não relacionais. Para isso foram efetuadas inserções, consultas e exclusões com o objetivo de medir o tempo de resposta destes modelos. Após a execução de tais testes foi realizada uma análise dos valores obtidos, verificando o tempo médio para que os dados fossem.

### **3 DESENVOLVIMENTO**

Durante a realização desta análise foram realizadas atividades de inserção, consulta e exclusão de dados, sendo que cada atividade foi executada repetidas vezes. Foram realizados 20 testes de inserção; 30 testes de consulta, sendo 10 consultando todos os dados contidos no banco, 10 filtrando os dados e 10 realizando a união dos dados originados por mais de uma entidade; e 20 testes de exclusão, sendo 10 utilizando filtros na exclusão dos dados e 10 excluindo todos os dados da base.

Durante a execução da atividade de inserção foi utilizada uma estrutura de repetição, comando for, esse laço de repetição percorria um dicionário que continha um número de identificação para cada pessoa e o sexo de cada uma, assim todos os bancos de todos os modelos teriam exatamente o mesmo número de homens e mulheres, de tal modo que os testes de consulta e deleção sempre teriam os mesmos dados, dessa forma os testes ficaram padronizados. Para que esta análise obtivesse resultados mais justos e coerentes com os tempos de resposta de cada um dos modelos utilizados, o tempo de execução da estrutura de repetição foi desconsiderado.

### **4 RESULTADOS OBTIDOS**

Após a realização das atividades de teste de desempenho, pôde-se observar que cada modelo apresentou tempos diferentes de execução. O tempo médio de cada modelo encontra-se na Tabela 1.

**Tabela 1:** Tempo Médio das Execuções

Atividade	Condição	Documento	Colunar	Chave-Valor
<b>Inserção</b>	100000 dados do tipo 'pessoa'	05:20:50	05:24:03	05:29:57
<b>Consulta</b>	dados do tipo 'pessoa' (SELECT *)	01:09:22	01:08:10	01:12:26
<b>Consulta</b>	pessoas do sexo 'masculino' (49.974 pessoas)	00:37:22	00:35:32	00:36:45
<b>Consulta</b>	junção de coleções (9.713 pessoas)	00:15:59	00:16:04	00:16:11
<b>Exclusão</b>	todos os dados do tipo 'pessoa'	05:14:28	05:15:42	05:12:59
<b>Exclusão</b>	pessoas do sexo 'masculino' (49.974 pessoas)	03:27:08	03:27:35	03:25:54

**Fonte:** Elaborado pelo autor

Como observa-se, por meio da Tabela 1, a API que representa o modelo de documento foi a que obteve o melhor tempo médio de resposta para o teste de inserção, enquanto a API que estava representando o modelo chave-valor a que obteve o pior tempo médio.

Na atividade de consulta por todos os dados do tipo pessoa, o modelo que apresentou o melhor tempo médio foi colunar e, novamente, o modelo chave-valor foi o que mais demorou para responder.

Ao analisar os resultados das consultas utilizando um filtro, foi possível perceber que todos os modelos tiveram um tempo de resposta parecido, com pouco mais de um minuto de diferença entre um e outro – observe que, apesar do tempo parecer curto, um minuto pode ser considerado muito tempo para a execução de uma busca em um banco de dados, além disso, a base de teste possui 100 mil dados na coleção, coleções com mais dados podem ter seu tempo aumentado em decorrência de seus tamanhos. Nesse experimento, o modelo que se destacou nessa operação foi o Colunar.

Observando os resultados referentes às consultas que envolvem transações de diferentes coleções de dados, pode-se notar que, o modelo documentos se saiu melhor, obtendo o menor tempo de resposta, e o modelo chave-valor obteve o pior tempo médio.

Analisando na Tabela 1 os resultados obtidos com os testes de exclusão, em que todos os 100 mil dados do tipo ‘pessoa’ deveriam ser excluídos, pode-se notar que o modelo chave-valor obteve um resultado superior aos demais enquanto que o modelo colunar terminou os testes com o pior tempo médio.

Analisando os resultados do último dos cinco testes executados, a exclusão de todos os dados da coleção ‘pessoa’ em que o campo ‘sexo’ fosse igual à ‘M’, pode-se observar que o modelo chave-valor, novamente, sobressaiu-se aos demais, tendo o melhor resultado entre os três. Ressalta-se que dessa vez o modelo colunar obteve o pior dos resultados.

O Quadro 1 traz em seus resultados uma comparação consolidando qual o modelo obteve o melhor e o pior resultado em cada um dos testes executados.

**Quadro 1:** Comparativo entre os Melhores e Piores Resultados

Atividade	Condição	Melhor Resultado	Pior Resultado
<b>Inserção</b>	100000 dados do tipo ‘pessoa’	Documento	Chave-Valor
<b>Consulta</b>	dados do tipo ‘pessoa’ (SELECT *)	Colunar	Chave-Valor
<b>Consulta</b>	pessoas do sexo ‘masculino’ (49.974 pessoas)	Colunar	Documento
<b>Consulta</b>	junção de coleções (9.713 pessoas)	Documento	Chave-Valor
<b>Exclusão</b>	todos os dados do tipo ‘pessoa’	Chave-Valor	Colunar
<b>Exclusão</b>	pessoas do sexo ‘masculino’ (49.974 pessoas)	Chave-Valor	Colunar

**Fonte:** Elaborado pelo autor

Observa-se a partir da Quadro 1 que os três modelos apareceram a mesma quantidade de vezes na coluna “Melhor Resultado”, sendo que, para que se afirme qual o

modelo melhor se encaixaria no cenário de testes proposto pela análise seria necessário também analisar a coluna “Piores Resultados”.

A partir dos resultados da coluna “Piores Resultados” pode-se observar que o modelo Documentos melhor se adapta para o cenário proposto, pois ele aparece nesta coluna apenas uma vez, enquanto os demais aparecem no mínimo duas vezes.

## 5 CONSIDERAÇÕES FINAIS

Para início deste trabalho foi levado em consideração a questão norteadora que constituiu o problema da pesquisa “Qual o modelo de bancos de dados NoSQL mais adequado para o cenário dos testes executados?”

Para encontrar uma resposta a essa pergunta, foi preciso documentar quais seriam os critérios de avaliação dos modelos, concluindo que o indicador utilizado para a comparação seria o tempo médio de resposta, ou seja, após todos os testes serem realizados e terem seus tempos de execução documentados deveria ser realizado o cálculo da média aritmética simples.

Durante a execução dos testes observou-se que era preciso padronizar os dados que seriam utilizados durante a análise, pois só assim todos os resultados das consultas e exclusões seriam os mesmos e, portanto, a comparação do tempo de execução seria justa. Assim, em meio à execução dos testes foi necessário criar um padrão e realizar novamente a execução de alguns testes já realizados.

Com os dados já padronizados, os testes tiveram um novo início, e desta vez todos os bancos utilizados possuíam os mesmos dados na mesma quantidade. Assim realizaram-se os testes de inserção, consulta e exclusão e seus resultados foram documentados para posterior análise.

Após a realização desses passos, pôde-se observar que o modelo de documentos obteve melhores resultados em dois testes (inserção e consulta utilizando junções de coleções) e o pior resultado em um dos testes (consulta por pessoas do sexo ‘masculino’); já o modelo colunar conseguiu melhores resultados, também, em dois testes (consulta por todos os dados do tipo ‘pessoa’ e consulta por pessoas do sexo ‘masculino’) e os piores resultados em dois dos testes (exclusão de todos os dados do tipo ‘pessoa’ e exclusão de pessoas do sexo ‘masculino’); e por fim, o modelo chave-valor obteve os melhores resultados em dois testes (exclusão de todos os dados do tipo ‘pessoa’ e exclusão de pessoas do sexo ‘masculino’) e os piores em três testes (inserção, consulta por todos os dados do tipo ‘pessoa’ e consulta utilizando junções de coleções).

Assim, diante desses resultados percebe-se que, para cenários que se assemelham ao deste trabalho, o modelo de banco de dados não relacional mais indicado seria o modelo de Documentos e que, para cenários que diferenciam-se deste, faz-se necessário a realização de um novo estudo, pois como todos os modelos obtiveram resultados próximos, a realização de testes em um novo cenário poderá levar a resultados diferentes.

Dado o cenário específico em que os testes se realizaram, o objetivo do trabalho foi atingido, pois pode-se chegar à conclusão de qual o melhor modelo de banco não relacional melhor se encaixaria ao ambiente avaliado.

## REFERÊNCIAS

DATE, C. J. **Introdução a Sistemas de Banco de Dados**. 8.ed. Rio de Janeiro: Elsevier, 2003.

DIANA, M. de; GEROSA, M. A.. NoSQL na Web 2.0: Um Estudo Comparativo de Bancos Não-Relacionais para Armazenamento de Dados na Web 2.0. In: WORKSHOP DE TESES E DISSERTAÇÕES EM BANCO DE DADOS, 9., 2010, São Paulo. **Workshops ...** São Paulo: Departamento de Ciência da Computação – Universidade de São Paulo (USP), 2010. p.1-8.

GIL, A. C.; **Como Elaborar Projetos de Pesquisa**. 4. ed. São Paulo: Atlas, 2002.

INSTITUTO BRASILEIRO DE GEOGRAFIA E ESTATÍSTICA. **Síntese de Indicadores Sociais – SIS**. Disponível em: <<https://www.ibge.gov.br/estatisticas-novoportal/sociais/populacao/9221-sintese-de-indicadores-sociais.html?&t=downloads>> Acesso em: 13 out. 2017

MARTINS, M. V. **Comparação Entre Arquitetura Relacional e NoSQL para o Desenvolvimento do Sistema de Controle de Informações do Aeroporto de Viracopos**. 2015. 57p. Monografia (Graduação de Tecnologia em Análise e Desenvolvimento de Sistemas) – Faculdade de Tecnologia de Indaiatuba, FATEC, Indaiatuba, 2015.

MELO, V. N. **Análise de Desempenho Entre os Modelos de Banco de Dados Relacional e Não Relacional – NoSQL**. 2013. 66p. Monografia (Graduação de Tecnologia em Análise e Desenvolvimento de Sistemas) – Faculdade de Tecnologia de Indaiatuba, FATEC, Indaiatuba 2013.

PANIZ, D. **NoSQL: Como armazenar os dados de uma aplicação moderna**. São Paulo: Casa do Código, 2016.

SILBERSCHATZ, A.; KORTH, H. F.; SUDARSHAN, S. **Sistema de Banco de Dados**. 5.ed. Rio de Janeiro: Elsevier, 2006.

SOUZA, A. M. de *et al.* Critérios para Seleção de SGBD NoSQL: o Ponto de Vista de Especialistas com Base na Literatura. In: SIMPÓSIO BRASILEIRO DE SISTEMAS DE INFORMAÇÃO, 10., 2014, Londrina. **Anais...** São Paulo: Escola de Artes, Ciências e Humanidades – Universidade de São Paulo (USP), 2014. p.149-160.