

Uma Análise Comparativa entre Sistemas Gerenciadores de Bancos de Dados NoSQL no contexto de Internet das Coisas

Allexandre Sampaio Santos Soares, Pablo Freire Matos

Instituto Federal de Educação, Ciência e Tecnologia da Bahia (IFBA)
Av. Amazonas, 3150, Zabelê – 45.075-265 – Vitória da Conquista – BA – Brasil

{allexandre.soares, pablofmatos}@ifba.edu.br

Abstract. *The search for improvement on applications performance is constant, especially in environments where data is analyzed in real time, as in Internet of Things (IoT). This paper aims to analyze the performance of non-relational Database Management Systems (DBMS), inside the IoT context. To do so, three DBMS were analyzed with a benchmarking tool, using a real IoT dataset. The tests evaluated response time, throughput, error rate and the use of CPU, memory and storage. The results showed each DBMS has characteristics that make it be the recommendation to a specific scenario, depending on the application's objective.*

Resumo. *É constante a busca pela melhoria no desempenho de aplicações, principalmente em um ambiente onde dados são analisados em tempo real, como na Internet das Coisas (IoT). Este artigo objetiva analisar o desempenho de sistemas gerenciadores de bancos de dados (SGBDs) não relacionais, no contexto da IoT. Para tanto, três SGBDs foram avaliados por meio de uma ferramenta de benchmarking, utilizando uma base de dados real de IoT. Os testes avaliaram o tempo de resposta, a vazão, a taxa de erros e o consumo de CPU, memória e armazenamento. Os resultados demonstraram que cada SGBD tem características que o faz ser recomendado a um cenário específico, a depender do objetivo da aplicação em questão.*

1. Introdução

Os Bancos de Dados (BDs) têm sido amplamente adotados como solução no armazenamento de dados. Mais especificamente, o modelo de BD relacional se tornou um grande sucesso, sendo implementado por vários desenvolvedores e utilizado na maioria dos projetos de BD ou sistemas computacionais [Sadalage e Fowler 2013]. No entanto, a evolução da computação, o aumento do tráfego de dados e a necessidade de visualização dos dados em tempo real, dentre outros motivos, fizeram com que as aplicações modernas buscassem alternativas de armazenamento de dados que priorizem escalabilidade e velocidade [Sadalage e Fowler 2013].

Como alternativas para evitar esse problema, pode-se verificar o uso de BDs de modelos hierárquico, de rede, orientado a objetos, objeto-relacional, dentre outros. Dentre essas abordagens, destaca-se o modelo NoSQL, do inglês *Not Only SQL*. Os BDs NoSQL têm como características a arquitetura flexível, a alta velocidade de acesso e gravação de dados e escalabilidade, que os tornam muito compatíveis com aplicações de alto desempenho e que priorizam a velocidade de acesso em detrimento à confiabilidade dos dados [Strauch 2011].

Dentre as diversas aplicações que podem se beneficiar da utilização de BDs NoSQL, estão aquelas ligadas à Internet das Coisas (IoT – *Internet of Things*). Geralmente, essas aplicações, por funcionarem em tempo real, necessitam de grande velocidade de manipulação de dados, uma vez que nesse ambiente há uma grande quantidade de dados de múltiplas fontes sendo trafegados [Porto, Ziviani e Ogasawara 2015].

A análise de Sistemas Gerenciadores de Bancos de Dados (SGBDs) é uma prática muito difundida, utilizada para se medir a aplicabilidade desses a um caso específico ou para se medir o desempenho entre diversos SGBDs para um determinado propósito. Apesar de existirem alguns trabalhos que enfoquem a utilização de SGBDs NoSQL no contexto de IoT, ainda são poucos os que tratam da análise do desempenho desses.

Pinto *et al.* (2013) realizaram um comparativo entre os SGBDs NoSQL MongoDB e DB4O, buscando compreender o comportamento desses quando submetidos a grandes cargas de dados. Uma aplicação foi desenvolvida especificamente para a realização do experimento, sendo possível especificar a quantidade de registros a serem inseridos no BD. Os resultados do trabalho sugerem o uso de um ou outro SGBD de acordo com o foco da aplicação em questão, seja desempenho ou economia de recursos. Carniel *et al.* (2012), por sua vez, compararam alguns BDs NoSQL orientados a colunas e documentos com o BD relacional PostgreSQL no contexto de processamento de consultas sobre Data Warehouse. Os resultados alcançados demonstraram superioridade do SGBD orientado a colunas, em comparação com os demais modelos, no contexto estudado.

Oliveira, Lóscio e Gama (2015) e Tavares, Oliveira e Lóscio (2016) têm como objeto de estudo a utilização de SGBDs NoSQL como forma de armazenamento de catálogos de produtores e consumidores de dados de IoT. O primeiro trabalho realizou experimentos de modo a verificar a viabilidade no uso do SensorML e do MongoDB com esta finalidade. O segundo desenvolveu diferentes protótipos de catálogos para adequação aos SGBDs OrientDB e MongoDB, de forma a verificar se esses são opções válidas para o contexto tratado, por meio do uso do SensorML e do WoTDataSchema. Ambos os estudos puderam demonstrar, por meio dos experimentos realizados com base em várias simulações, que há viabilidade no uso destes SGBDs nos contextos estudados. O presente estudo se assemelha com esses trabalhos, uma vez que trata de testes de desempenho aplicados a contextos bastante similares.

A partir do exposto, percebe-se uma lacuna de trabalhos focados no uso de SGBDs NoSQL dentro do contexto de IoT. Portanto, este artigo objetiva identificar dados comumente utilizados em aplicações de IoT e povoar BDs NoSQL com esses dados, possibilitando análises focadas no desempenho, integridade e consumo de recursos, de maneira que se possa traçar um comparativo. Para a realização do experimento proposto, foi desenvolvida uma ferramenta específica para *benchmarking*, a qual é descrita neste trabalho. O artigo está estruturado da seguinte forma: Na Seção 2, é apresentada a fundamentação teórica necessária para o entendimento deste trabalho; na Seção 3, são descritas as definições do ambiente de execução dos testes, bem como a ferramenta para *benchmarking* e a base de dados utilizada; Na Seção 4, são descritos os resultados sucintos alcançados com o experimento; e na Seção 5, são apresentadas a Conclusão e os Trabalhos Futuros.

2. Fundamentação Teórica

O embasamento teórico deste trabalho fundamenta-se, principalmente, nos conceitos de BDs NoSQL, IoT e técnicas de *benchmarking*.

2.1. Sistemas de Bancos de Dados NoSQL

Os BDs relacionais podem ser empregados na maioria das aplicações, tendo muitas vantagens em sua utilização. No entanto, um problema recorrente em seu uso é a incompatibilidade de impedância. Segundo Sadalage e Fowler (2013), a incompatibilidade de impedância ocorre quando há uma diferença entre o modelo de dados relacional e as estruturas dos dados na memória. Os BDs NoSQL são, então, mais adequados a aplicações e sistemas que utilizam dados não estruturados e que necessitam de grande desempenho em escalabilidade e no acesso aos dados, como sistemas que funcionam em tempo real. Nesse modelo, os bancos priorizam a disponibilidade dos dados em detrimento à consistência absoluta desses.

Vieira *et al.* (2012, p. 3) resumem as características dos BDs NoSQL em quatro propriedades básicas: “(i) dados na ordem de dezenas ou centenas de Terabytes; (ii) poder de

crescimento elástico horizontal; (iii) fácil distribuição dos dados e/ou processamento; e (iv) tipos de dados variados, complexos e/ou semiestruturados”. Essas características tornam os BDs NoSQL recomendáveis para cenários como o da Internet das Coisas, que demanda de alto poder de processamento de dados não estruturados e de grande disponibilidade.

2.2. Internet das Coisas

Segundo Almeida (2015, p.7), a IoT “refere-se à integração de objetos físicos e virtuais em redes conectadas à Internet, permitindo que “coisas” colem, troquem e armazenem uma enorme quantidade de dados numa nuvem, em que uma vez processados e analisados esses dados, gerem informações e serviços em escala inimaginável”.

Sobre os dados das aplicações de IoT, Li *et al.* (2012) evidenciam que essas têm seus dados como ativos de valor. Tendo essas as seguintes características: (i) provêm de várias fontes e são heterogêneos; (ii) têm grande escala; (iii) têm correlação temporal e espacial; (iv) necessitam de interoperabilidade; e (v) são de multidimensionalidade. Resumidamente, essas características mostram que os dados destas aplicações surgem de forma contínua e expansiva de diversas fontes (sensores, câmeras, leitores, dentre outros), precisam de referências de tempo, já que as suas informações são geralmente tratadas em tempo real e devem trabalhar independentemente de plataformas, sendo seus dados acessados por diferentes aplicações.

2.3. Testes de Desempenho

Segundo Santos e Neto (2008), os testes de desempenho verificam em um sistema: latência, vazão, escalabilidade e uso de recursos de máquina. Para a realização de diversos tipos de testes, comumente são utilizadas várias ferramentas, focadas ou especializadas em uma característica específica a ser testada em um sistema computacional. Tais ferramentas são conhecidas como ferramentas de *benchmarking*.

Existem diversas ferramentas disponíveis para avaliar *softwares*, SGBDs, elementos de *hardware*, dentre outros. No entanto, como citado por Gray (1993), nenhuma das ferramentas poderia testar o desempenho de um sistema por completo, havendo então a necessidade de se testar um sistema utilizando ferramentas adequadas ao seu domínio de aplicação.

3. Experimento

Para a realização do experimento foram analisadas diversas características de SGBDs e ferramentas para testes, com o intuito de realizar uma simulação tão próxima quanto possível da realidade quanto ao uso de aplicações de IoT integradas a BDs NoSQL. Com o intuito de prezar ao máximo pelo correto funcionamento dos SGBDs durante a realização dos testes, bem como garantir que todos os SGBDs testados estivessem sob o mesmo ambiente com a mínima variação possível, optou-se pela utilização de uma partição de disco do computador (com 17,7 GB de capacidade), com um Sistema Operacional (SO) dedicado apenas aos testes, abrigando tanto o *benchmarking* utilizado, quanto os SGBDs testados. O computador utilizado na realização dos testes foi um Notebook Acer Aspire V5 - 472-6_BR826, contando com 8 GB de memória RAM, 500 GB de armazenamento de disco rígido e processador Intel i3-3217U, que dispõe de 2 núcleos físicos e 4 *threads* com frequência base de 1,8 GHz. O SO utilizado foi o Ubuntu 16.04.1. Tal escolha se deu por ter esse SO livre disponibilidade, por ser compatível com todos os SGBDs testados e por sua grande comunidade, que auxilia na resolução de problemas. Durante a realização dos testes, optou-se pelo uso do SO apenas via linha de comando de modo a economizar recursos que seriam utilizados por sua interface gráfica.

Dada a necessidade da automatização da tarefa de submeter objeto testado a diversas cargas e condições de testes, fez-se necessário o uso de ferramentas de *benchmarking*. Após a análise das ferramentas elencadas com tal finalidade, percebeu-se uma dificuldade em utilizá-las para um contexto específico, nesse caso, o de IoT. Optou-se então pelo desenvolvimento de uma ferramenta específica para o foco deste trabalho e moldada de forma que fosse o mais simples e eficiente possível. O *software* desenvolvido se chama Inobench, uma referência à sua

finalidade (IoT-NoSQL-Benchmarking), e está disponível para livre uso sob a licença GNU General Public License V.3 em seu repositório na Internet¹. A ferramenta em questão é composta de alguns módulos: *core* (controle geral da aplicação), *CSVReader* (leitura de arquivos externos), *files* (armazenamento e manipulação de arquivos externos), além de um módulo específico para cada um dos SGBDs abrangidos pela solução (implementação de operações para a manipulação de dados em cada SGBD).

Os BDs utilizados no experimento foram de documentos e de chave-valor. Tal escolha se deu pelo entendimento de que esses modelos de BDs permitem uma manipulação mais simples dos dados utilizados, tornando seu armazenamento mais fácil. Foram escolhidos três SGBDs: MongoDB e Couchbase para os testes com SGBDs de documentos, e Redis para os testes com SGBDs de chave-valor. Os critérios que levaram à escolha desses foram sua compatibilidade com o ambiente de teste, a possibilidade de sua utilização de forma gratuita, sua adequação à proposta deste trabalho e a sua facilidade de uso a partir da experiência do agente realizador do experimento. Durante a realização dos testes, buscou-se utilizar a versão mais recente de cada SGBD. Sendo assim, o MongoDB foi utilizado em sua versão 3.2.12, o Couchbase em sua versão 4.6.0 e o Redis em sua versão 3.2.8. É importante salientar que, em todos os SGBDs, não foram utilizadas técnicas de replicação. Sendo assim, os resultados alcançados são baseados no desempenho relativo a apenas uma instância de cada SGBD.

Para que o experimento proposto simulasse de forma fiel o ambiente real, optou-se pela utilização de uma base de dados com informações colhidas de aplicações de IoT. Assim, foi utilizada a base de dados *Air Quality Data Set*, disponibilizada publicamente no repositório² de aprendizagem de máquina da Universidade da Califórnia, Irvine. A base de dados em questão contém 9.358 instâncias de dados colhidos de diversos sensores que medem quimicamente a qualidade do ar. Cada instância corresponde à média dos dados colhidos dentro de uma hora, durante os anos de 2004 e 2005, em ambientes de grande concentração de poluição em uma cidade na Itália. Cada instância da base de dados contém 15 dados diferentes. Os dados são relativos à data da coleta, hora da coleta, concentração de monóxido de carbono, concentração de dióxido de estanho, concentração de benzeno, temperatura, umidade, dentre outros.

No que tange à metodologia de teste empregada, foram realizadas múltiplas operações de inserção e consulta, utilizando-se de um número crescente de usuários, com o intuito de verificar os limites máximos de operação de cada um dos SGBDs. Cada ciclo de teste foi iniciado com 1 usuário realizando 100.000 operações, subindo ao nível de 10 usuários simultâneos realizando 10.000 operações, 100 usuários realizando 5.000 operações, 500 usuários realizando 1.000 operações e finalizando-se com 750 usuários realizando 100 operações. A quantidade máxima foi definida após a observação de limites que pudessem garantir a confiabilidade dos resultados, visto que os resultados foram comprometidos pela sobrecarga em um cenário com cerca de 1.000 usuários. Nos ciclos de teste, avaliou-se: tempo de resposta; vazão; taxa de erros; consumo de CPU e memória; e espaço de armazenamento utilizado (para operações de inserção).

4. Resultados

A análise dos resultados obtidos no experimento realizado possibilita perceber a superioridade de um SGBD no cenário testado para cada uma das métricas definidas. Assim sendo, as conclusões tomadas neste trabalho são baseadas inicialmente nos fatores individuais que fizeram com que cada um dos SGBDs testados se mostrasse superior, fatores que devem, portanto, ser analisados para a escolha de um desses SGBDs para uso em ambiente real de IoT.

Tratando-se de vazão e tempo de resposta, os resultados demonstraram que os SGBDs testados têm resultados parecidos na inserção de registros, tendo o Redis alcançado resultados

¹ Inobench: <https://github.com/allexandresampaio/inobench>

² Air Quality Data Set: <http://archive.ics.uci.edu/ml/datasets/Air+Quality>

levemente superiores aos demais. Para a leitura de dados, o Couchbase foi mais rápido. Sobre as taxas de erros, verifica-se que estes ocorrem com maior frequência quando se inserem mais usuários simultâneos acessando os SGBDs. No entanto, a ocorrência de erros representa valores muito pequenos dentro o total de operações realizadas. Nesses cenários, o MongoDB obteve resultados exitosos nos testes de inserção e consulta, e o Redis nos testes de consultas.

A análise do consumo de CPU e memória RAM demonstra que, na maioria dos casos, os SGBDs testados não demandam de muitos recursos para realizar as operações com eficiência. Sobre o consumo de CPU, nos testes de inserção de dados, tanto o Couchbase quanto o Redis têm resultados semelhantes, sendo os mais econômicos. Para leitura de dados, o Couchbase consumiu menos processamento. Sobre o consumo de memória RAM, nenhum dos SGBDs chegou a usar sequer 10% do total de memória RAM disponível. Tanto nas inserções, quanto nas consultas de registros, o Redis se mostrou superior em relação aos demais SGBDs.

A análise de consumo de memória física para guardar dados nos SGBDs é importante visto que, no ambiente de IoT, a quantidade de dados armazenados é consideravelmente grande e crescente. Neste quesito, os testes evidenciaram superioridade do MongoDB, tendo os demais SGBDs consumido várias vezes mais espaço do que o utilizado por esse.

Na Figura 1 é exibida em formato de gráfico de radar a adequação dos SGBDs testados às métricas avaliadas. Cada vértice das linhas que representam os SGBDs está posicionado de acordo com uma escala que, do centro para fora do gráfico, vai de zero a cem. O uso deste gráfico tem como objetivo demonstrar visualmente qual dos SGBDs testados mostra-se mais recomendado ao uso de acordo com apenas uma ou por combinação de métricas. Desta forma, é fácil perceber e possível inferir que: o MongoDB é recomendado para aplicações onde há restrições de confiabilidade e de espaço para armazenamento em disco, além de ter um resultado mediano nas demais métricas; o Couchbase é recomendado para aplicações que demandam de alta velocidade; e o Redis é recomendado para aplicações onde há restrições de consumo de processamento, tendo um resultado satisfatório nas demais métricas avaliadas.

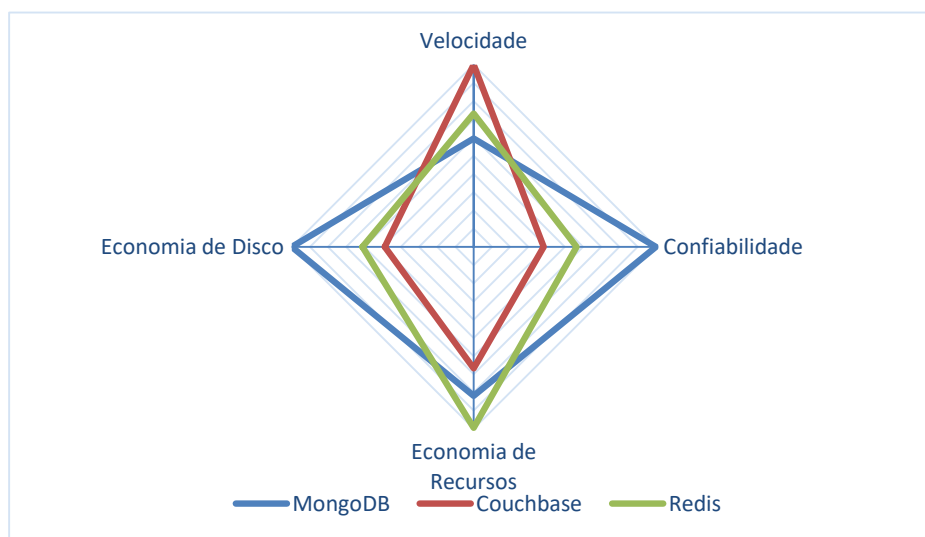


Figura 1. Gráfico Radar de métricas avaliadas. Fonte: Próprio autor

Sobre a realização dos testes, faz-se necessário frisar que, como a aplicação utilizada foi desenvolvida em Java, houve um consumo de recursos pela máquina virtual da aplicação que pode ter influenciado nos resultados. No entanto, considera-se que os resultados obtidos são válidos, uma vez que a possível influência da aplicação ocorreu para todos os testes, igualmente, independentemente do SGBD utilizado ou cenário testado.

5. Conclusão

O objetivo proposto por este trabalho foi de desenvolver uma análise comparativa entre SGBDs NoSQL quando inseridos no contexto da IoT. Para tanto, desenvolveu-se uma ferramenta capaz de realizar testes em SGBDs NoSQL utilizando bases de dados predefinidas. Durante os testes, as principais variáveis que definem o bom desempenho ou não de um determinado SGBD foram colhidas. Isso possibilitou identificar a superioridade de um determinado SGBD em situações específicas no ambiente da IoT, sendo essa superioridade condicionada ao contexto da aplicação que fará uso do SGBD. Portanto, as necessidades da aplicação devem ser avaliadas para a escolha do SGBD que melhor se adapte a elas.

Apesar de existirem alguns trabalhos correlatos que compararam SGBDs relacionais e não-relacionais, observa-se que este trabalho tem como diferencial a realização de um *benchmarking* aplicado a um contexto específico, por meio de testes de desempenho. Pode-se destacar como principais contribuições deste estudo o desenvolvimento de um *software* voltado para o *benchmarking* de SGBDs NoSQL, sendo possível contextualizar tais testes a partir da utilização de uma base de dados específica. Além disso, considera-se de grande contribuição os resultados obtidos no experimento, podendo ser utilizados como auxílio na análise da utilização de um dos SGBDs NoSQL testados em uma aplicação no contexto da IoT. Por fim, recomenda-se como trabalhos futuros a realização de testes de escalabilidade dos SGBDs NoSQL, bem como a utilização de SGBDs NoSQL temporais (*time-series*).

Referências

- Almeida, H. (2015) Tudo Conectado. *Computação Brasil: Revista da Sociedade Brasileira de Computação*, Porto Alegre, v. 29, 14-18.
- Carniel, A. C., Sá, A. A., Ribeiro, M. X., Bueno, R., Ciferri, C. D. A. e Ciferri, R. R. (2012) Análise Experimental de Bases de Dados Relacionais e NoSQL no Processamento de Consultas sobre Data Warehouse. In: *XXVII SBBB*, pages 113-120.
- Gray, J. (1993) *The Benchmark Handbook for Database and Transaction Systems*. 2ª ed. Morgan Kaufmann Publishers.
- Li, T., Liu, Y., Tian, Y., Shen, S. e Mao, W. (2012) A storage solution for massive IoT data based on NoSQL In: *GreenCom*, pages 50-57. IEEE.
- Oliveira, M. I. S., Lóscio, B. F. e Gama, K. S. (2015) "Análise de Desempenho de Catálogo de Produtores de Dados para Internet das Coisas baseado em SensorML e NoSQL." In: XIV Workshop em Desempenho de Sistemas Computacionais e de Comunicação.
- Pinto, A. P., Malagoli, F., Maciel, R. M., Soares, H. R. e Santana, S. A. (2013) "Testes de Performance Utilizando o DB4O e MongoDB" In: *e-RAC*, v. 3, n. 1.
- Porto, F., Ziviani, A. e Ogasawara, E. (2015) BIG DATA: Um Novo Desafio à Nossa Porta. *Computação Brasil: RSBC*, Porto Alegre, v. 29, 23-27.
- Sadalage, P. e Fowler, M. (2013) *NoSQL Essencial: Um guia conciso para o mundo emergente da persistência poliglota*. São Paulo: Novatec.
- Santos, I. e Neto, P. (2008) Automação de Testes de Desempenho e Estresse com o JMeter. In: *II ERCEMAPI*.
- Strauch, C. (2011) "NoSQL databases" In: Lecture Notes, Stuttgart Media University.
- Tavares, A. T., Oliveira, M. I. S. e Lóscio, B. F. (2016) "Data producer catalogs for the web of things: a study on NoSQL solutions." In: 31st Annual ACM Symposium on Applied Computing.
- Vieira, M., Figueiredo, J., Liberatti, G. e Viebrantz, A. (2012) Bancos de Dados NoSQL: Conceitos, Ferramentas, Linguagens e Estudos de Casos no Contexto de Big Data. In: *XXVII SBBB*, pages 439-468.