Gestão de Big Data: Novos Paradigmas

Big Data Management: New Paradigms

Elisabete Paulo Morais, Unidade de Investigação Aplicada à Gestão (UNIAG), Instituto Politécnico de Bragança, Mirandela, Portugal, beta@ipb.pt

Carlos R. Cunha, INESC TEC - Instituto de Engenharia de Sistemas e Computadores, Tecnologia e Ciência, Instituto Politécnico de Bragança, Mirandela, Portugal, crc@ipb.pt

Vítor Mendonça, Instituto Politécnico de Bragança, Mirandela, Portugal, mendonca@ipb.pt

Resumo

A popularidade crescente de aplicações Web de acesso massivo que armazenam e analisam grandes quantidades de dados, sendo o Facebook, o Twitter, a Amazon e a Google alguns exemplos proeminentes de tais aplicações, apresentam novas exigências que desafiam os tradicionais SGBDR. Motivados principalmente por questões de escalabilidade, uma nova geração de bases de dados, apelidadas de NoSQL, tem vindo a ganhar alguma força. Neste artigo serão apresentadas as principais características dessas bases de dados. As bases de dados NoSQL são comparadas com os tradicionais SGBDR e conceitos importantes serão explanados.

Palavras-chave: (Big Data; NoSQL; Bases de Dados Relacionais; Web)

Abstract

The growing popularity of massively accessed web applications that store and analyze large amounts of data, being Facebook, twitter and google search some prominent examples of such applications, have posed new requirements that greatly challenge traditional RDBMS. Driven primarily by scalability issues, a new generation of databases, called NoSQL, has gained some strength. This paper presents the main characteristics of these databases. NoSQL databases are compared with traditional RDBMS and important concepts are explained.

Keywords: *(Big Data; NoSQL; Ralational Databases; Webs)*

1. INTRODUÇÃO

As bases de dados relacionais têm sido usadas em grande escala desde a sua criação, no início dos anos 1970, e pode, como tal, ser considerada uma tecnologia com um elevado grau de maturidade para armazenar dados e os seus relacionamentos.

Tendo surgido como sucessor dos modelos hierárquico e de rede, o modelo relacional tornou-se padrão para a maioria dos SGBDs (Sistemas Gestores de Bases de Dados), tais como o SQL Server,

ISSN 2183-489X
DOI http://dx.doi.org/10.18803/capsi.v17.00-00

Oracle, PostgreSQL, MySQL, etc. Os seus elementos básicos são as tabelas, as quais são compostas por linhas e colunas (ou atributos) (Cood, 1970).

Outra característica fundamental deste modelo é a utilização de restrições de integridade. Esses elementos são utilizados para garantir que a integridade dos dados seja mantida. As restrições de integridade mais comuns são as chaves: primárias e estrangeiras.

No entanto, problemas de armazenamento em sistemas orientados para a Web ultrapassam os limites das bases de dados relacionais, levando investigadores e empresas a procurarem formas não tradicionais de armazenamento de dados (Stonebraker & Cattell, 2011). Os dados atuais podem ser escalados para terabytes por dia e devem estar disponíveis para milhões de utilizadores em todo o mundo sob baixos requisitos de latência.

Outro especto crítico, para além da quantidade de dados, é a variedade de dados na área da Multimédia.

2. BIG DATA

O conceito Big Data não possui uma definição formal única adotada por todos. De uma forma simples e pragmática, Big Data pode ser definido formalmente como um grande volume de dados, que são disponibilizados com diferentes graus de complexidade, gerados a diferentes velocidades e que possuem diferentes graus de ambiguidade; o que resulta numa complexidade que está para além da suportada pelas tecnologias, métodos de processamento e algoritmos "tradicionais" (Krishnan, 2013).

Existem duas grandes fontes de dados que podem ser considerados sob o paradigma Big Data. A primeira, os dados estruturados, semiestruturados e não estruturados que existem no seio das organizações como: correio eletrónico, documentos em formato PDF, folhas de cálculo, registos de servidores e outros dados decorrentes da própria atividade da organização. A segunda fonte de dados é o conjunto de dados disponíveis fora das organizações, alguns disponíveis livremente, outros mediante o pagamento de uma subscrição ou disponíveis para grupos restritos de parceiros e/ou clientes selecionados (Sathi, 2012).

Os dados Big Data possuem 3 características que, em conjunto, os diferenciam de todos os outros tipos de dados. Essas características são conhecidas como os 3 Vs dos dados Big Data e são: volume, velocidade e variedade (Singh & Singh, 2012).

Existem autores que advogam a adição de mais Vs aos 3 normalmente considerados como característicos dos dados Big Data. Estes autores consideram que o Valor e a Veracidade dos dados

também são duas características que distinguem os dados Big Data de todos os outros tipos de dados (Demchenko Grosso, Laat & Membrey, 2913), (Sathi, 2012).

3. BASES DE DADOS NOSQL

As mudanças ocorridas na tentativa de se propor alternativas ao uso do Modelo Relacional levaram os investigadores a pensar num modo alternativo de se modelar bases de dados. A estrutura pouco flexível utilizada até então passou a ser um problema a ser contornado e as soluções propostas tinham como base a eliminação ou minimização dessa estruturação.

O objetivo dos projetistas de bases de dados de organizações de grande porte passou a ser desenvolver uma nova estratégia de armazenamento na qual pudessem estar livres de certas estruturas e regras presentes no Modelo Relacional. Assim, foram surgindo soluções que pareciam voltar no tempo, retornando aos simples sistemas de gestão de ficheiros.

Se, por um lado, tais soluções perdiam todo o arcabouço de regras de consistência presentes no Modelo Relacional, por outro lado, poderiam ganhar em desempenho, através da flexibilização dos sistemas de bases de dados para as características particulares de cada organização.

Como a integridade e a consistência são fatores críticos de sucesso, as bases de dados relacionais não apresentam os requisitos de escalabilidade necessários para suportar grandes volumes de dados transacionais (Krishnan, 2013). Surgiu então o movimento NoSQL, cuja ascensão está intimamente ligada ao crescimento das grandes empresas web (Moniruzzaman & Hossain, 2013). Sendo que grande parte dos projetos principais no âmbito NoSQL, têm a sua base em projetos criados pela Google (BigTable), Facebook (Cassandra) e Amazon (Dynamo).

O termo NoSQL surgiu em 1998, a partir de uma solução de bases de dados que não oferecia uma interface SQL, mas esse sistema ainda era baseado no Modelo Relacional.

Posteriormente, o termo passou a representar soluções que promoviam uma alternativa ao Modelo Relacional, tornando-se uma abreviação de Not Only SQL (não apenas SQL), sendo utilizado principalmente em casos em que o Modelo Relacional não apresentava o desempenho adequado.

O propósito das soluções NoSQL não é substituir o Modelo Relacional como um todo, mas apenas nos casos em que seja necessária uma maior flexibilidade na estruturação da base de dados.

3.1. Classificação de Bases de Dados NoSQL

Apesar de possuírem certas características em comum, tais como serem livres de esquema, promoverem alta disponibilidade e maior escalabilidade, s sistemas de bases de dados NoSQL existentes possuem diversas singularidades.

As bases de dados NoSQL podem ser divididas em várias categorias de acordo com a classificação proposta em Tudorica e Bucur (2011), Cattell (2011), Leavitt (2010), Hecht (2011), cada uma prescrevendo um determinado modelo de dados:

Chave-valor: Os dados armazenados neste tipo de modelo, são constituídos por duas partes: uma string que representa a chave, e os dados a serem armazenados, que representam o valor, assim criando um par "chave-valor" (Nayak, Poriya & Poojary, 2013). Não existe nenhum requerimento específico para os dados a utilizar, estes podem possuir o tamanho desejado e podem ser representados por qualquer tipo de ficheiro (Celko, 2014), o que inclui poderem também representar outros conjuntos de chaves (Bernardino & Abramova, 2013). As chaves, por sua vez, podem possuir nomenclaturas bastante flexíveis, de acordo com as necessidades do sistema. Estes sistemas de base de dados, apresentada agora na era Big Data, tem a sua grande influência no sistema Dynamo apresentado pela Amazon (Decandia et al., 2007). Estes sistemas estão vocacionados para garantir grandes quantidades de leituras, o que se traduz num grande poder de escalabilidade por parte destes sistemas, ao mesmo tempo que nunca colocam em causa a sua disponibilidade.

Orientados a documento: Este tipo de bases de dados é considerado como o mais generalista, flexível, poderoso e popular de todos os que integram o movimento NoSQL (McCreary & Kelly, 2014). Como o próprio nome indica, este tipo de bases de dados recorre a documentos como método de armazenamento de dados, quase como que um arquivo, destinado a armazenar ficheiros da era digital. Este tipo de bases de dados fornece um bom desempenho ao mesmo tempo que possibilita uma grande escalabilidade horizontal (Nayak, Poriya & Poojary, 2013). Os documentos armazenados são normalmente de tipos bastante comuns como XML (eXtensible Markup Language), JSON (JavaScript Object Notation) ou BSON (Binary JSON) (Nayak, Poriya & Poojary, 2013).

Neste tipo de sistemas a cada documento é atribuída uma chave única. Estas chaves podem ser representadas por uma simples string, por um caminho de ficheiro (Nayak, Poriya & Poojary, 2013) ou um outro tipo de URL ou URI. De uma maneira geral estes sistemas baseiam-se em índices para tornar mais fácil o acesso a documentos (Robinson, Webber & Eifrem, 2013) Uma consequência da utilização de documentos como armazenamento é que, sempre que um novo documento é guardado, todo o conteúdo desse documento tem de ser indexado (McCreary & Kelly, 2014).

As bases de dados NoSQL do tipo armazenamento de documentos são as mais indicadas para aplicações web que necessitem de armazenar grande quantidade de dados semiestruturados, onde também é necessário executar várias consultas dinâmicas (Kaur & Rani, 2013), sendo os sistemas mais populares o MongoDB e o CouchDB.

Orientados a coluna: Os sistemas NoSQL orientados a colunas são conhecidos por possuírem uma capacidade notável de escalar horizontalmente, de modo a conseguirem albergar grandes quantidades de dados (McCreary & Kelly, 2014). Estes sistemas também são conhecidos por estarem intimamente relacionados com vários sistemas que recorrem a funções MapReduce. Quase todos os sistemas deste tipo em existência são altamente influenciados pelo artigo (Chang et al.,2006), onde em 2006 a Google apresentava o BigTable: um sistema de armazenamento distribuído para gerir dados estruturados, desenhado com o intuito de conseguir escalar horizontalmente de forma natural, suportando quantidades de dados muito elevadas (Hecht, 2011). O Facebook utiliza o sistema Cassandra que é orientado a coluna.

Para além destes modelos de dados, alguns autores também concebem uma quarta categoria de bases de dados NoSQL (Hecht, 2011):

Baseados em grafos: Neste tipo de bases de dados, o modelo de dados representa uma rede que contém nós de um grafo, arestas e propriedades (McCreary & Kelly, 2014). As arestas são utilizadas para ligar dois nós e representam uma relação, as relações podem possuir uma direção, conferindo assim um significado à relação. Os nós podem possuir propriedades, estas descrevem com variáveis graus de profundidade os dados contidos nesses nós; arestas também podem possui os seus próprios conjuntos de propriedades. As relações são identificadas pelos seus nomes e podem ser atravessadas em ambas as direções (Kaur & Rani, 2013).

3.2. Comparação de Bases de Dados NoSQL

Existem uma série de estudos que listam e comparam Bases de Dados NoSQL, expondo as suas virtudes e fraquezas.

Catell (2011) analisou e comparou várias bases de dados relativamente aos métodos de controlo de concorrência, armazenamento de dados (isto é, na memória principal ou disco), mecanismo de replicação utilizado (síncrono ou assíncrono) e método de transação. Esta comparação inclui Bases de Dados Comerciais e não comerciais, mas não inclui Bases de dados baseados em grafos, que são consideradas parte do NoSQL. Os gráficos são o modelo de dados essenciais de aplicações da Web, como Redes sociais (Krepska, Kielmann, Fokkink & Bal, 2011). De modo semelhante, Padhy, Patra e Satapathy (2011) apresentam uma comparação de seis bases de dados NoSQL relevantes, incluindo Bases de Dados de diferentes tipos ou esquemas, excluindo também Bases de Dados orientadas a grafos.e outras implementações relevantes de Bases de Dados NoSQL. Hecht e Jablonski (2011) apresentam outra comparação entre Bases de Dados NoSQL, incluindo as orientadas a gráficos. Os autores focam aspetos de particionamento e replicação de dados através da comparação de 14 Bases de Dados NoSQL.

Existem outros estudos que analisam bases de dados NoSQL usando um determinado conjunto de dados ou aplicações. Por exemplo, Sakr, Liu, Batista e Alomari (2011) realizaram uma análise completa de armazéns de dados adequados para ambientes de computação em nuvem, onde incluíam bases de dados NoSQL. Os autores apresentam um conjunto de metas que uma aplicação de dados intensiva deve realizar na nuvem. Descrevem também estruturas essenciais e algoritmos de bases de dados bem conhecidos como o BigTable e o Dynamo. Além disso, eles comparam várias APIs relacionadas com a consulta e manipulação de dados massivos.

Outra comparação de bases de dados NoSQL foi efetuada por Orend (2010). O objetivo final do estudo era selecionar uma solução NoSQL para software de Colaboração Web e Gestão de Conhecimento. MongoDB, uma base de dados orientada a documentos, foi selecionada a partir das bases de dados disponíveis devido ao seu suporte para consultas em vários campos. O estudo faz uma comparação do desempenho do MongoDB relativamente ao MySQL e ao HyperSQL.

Todurica e Bucur (2011) analisaram uma extensa lista de bases de dados NoSQL disponíveis, e utilizaram duas delas - Cassandra e HBase - versus MySQL e Sherpa, uma variação do MySQL. Os resultados indicaram que, em alta carga, o Cassandra e HBase mantêm o tempo de resposta relativamente constante, enquanto o MySQL e o Sherpa aumentam o tempo de resposta. Por outro lado, Lith e Mattson (2010) apresentaram um estudo baseado numa aplicação própria onde uma abordagem baseada em MySQL oferece um melhor desempenho do que usar uma solução NoSQL. No estudo, cinco bases de dados NoSQL foram consideradas. Os autores alegam que a diferença desempenho é devido à estrutura de dados da aplicação e à forma como ela é acedida.

4. RELACIONAL VS NOSQL

Quando se analisa a possibilidade de se optar uma estratégia NoSQL em detrimento de um SGBD relacional, devemos ter em consideração questões básicas como, por exemplo, os critérios de escalabilidade, a consistência dos dados e a disponibilidade.

A questão da escalabilidade é essencial, porque é justamente neste ponto que as bases de dados NoSQL apresentam vantagens relativamente às bases de dados relacionais, principalmente porque foram criadas com esse objetivo, enquanto os SGBD relacionais possuem uma estruturação menos flexível e menos adaptada para cenários em que a escalabilidade se torna um fator necessário.

A escalabilidade toma importância quando o número de utilizadores que acedem à base de dados aumenta significativamente. As soluções passam por aumentar a capacidade do servidor, conhecida por escalabilidade vertical (scale up) ou pelo aumento do número de servidores (scale out).

A escalabilidade vertical, que se caracteriza pela sua simplicidade, tem sido mais utilizada para a camada da base de dados, enquanto a escalabilidade horizontal tem sido mais utilizada na camada de aplicação, principalmente para a Web.

Quando falamos em bases de dados distribuídas, a ideia é distribuir a base de dados por várias máquinas, fazendo-se o particionamento dos dados. Este processo é também conhecido por fragmentação. Aplicar fragmentação em bases de dados relacionais, apesar de possível, não é fácil. Primeiro, os SGBD relacionais obedecem ao critério de normalização, enquanto o processo de fragmentação se caracteriza justamente pelo inverso: a desnormalização dos dados. Segundo, existe uma mudança de paradigma em relação ao processo de escalabilidade. Os SGBDs relacionais aplicam a estratégia de escalabilidade vertical, ou seja, reforçar o servidor, o processo de fragmentação visa trabalhar com a escalabilidade horizontal, paralelizando os dados em vários servidores. Terceiro, o volume de dados por máquina é minimizado devido ao processo de distribuição. Conjuntos de dados menores são mais fáceis de serem acedidos, atualizados e geridos. Por último, o grau de disponibilidade do sistema é otimizado.

Os principais benefícios da escalabilidade horizontal são: maior disponibilidade, menor tempo de resposta para efetuar consultas, paralelismo na atualização dos dados e maior grau de concorrência.

As bases de dados NoSQL foram especialmente projetadas para dar resposta a estas características de forma mais natural.

Ao se substituir um SGBD relacional por uma solução NoSQL, a arquitetura perde em consistência, mas pode ganhar em flexibilidade, disponibilidade e desempenho. Esta ideia baseia-se no Teorema CAP (Consistency, Availability e Partition Tolerance) ou teorema de Brewer, o qual afirma que num sistema distribuído é impossível garantir de forma simultânea, consistência, disponibilidade e tolerância ao particionamento. Segundo este teorema, um sistema distribuído pode garantir somente duas as três características simultaneamente (Gilbert & Ahmed, 2002).

O tipo de consistência utilizada nas bases de dados NoSQL é chamada de Consistência Eventual: significa que todas as operações de leitura podem retornar dados diferentes da última operação de gravação concluída, mas como o passar do tempo passa: "A um estado estável", o sistema acabará por retornar o último valor escrito. Portanto, os clientes podem enfrentar um estado inconsistente de dados à medida que as atualizações estão a ser efetuadas (Ayman, Ahmed, Haitham & Hesham, 2016).

Este conceito é estendido para o paradigma BASE (Basically Available, Soft state, Eventual consistency) que se caracteriza por ser basicamente disponível, ou seja, o sistema deve estar disponível sempre, mas não precisa de estar sempre consistente.

Este modelo entra em contraste com o paradigma ACID (Atomicity, Consistency, Isolation, Durability) comumente associado aos SGBDs relacionais. Enquanto o modelo ACID força a consistência no final de cada operação, o modelo BASE permite que a base de dados esteja eventualmente em estado inconsistente.

5. CONCLUSÃO

A decisão de optar por uma abordagem NoSQL por oposição ao modelo relacional, tem que ter em consideração as necessidades do problema.

Devemos ter em consideração diversos aspetos, desde as questões de escalabilidade do sistema, consistência dos dados até à própria facilidade de utilização.

Os SGBDR são soluções com um maior grau de maturidade e de maior consistência de dados, no entanto as aplicações Web e os dados Multimédia, poderão necessitar de soluções menos flexíveis, como o NoSQL, principalmente por questões de escalabilidade.

A própria simplicidade de consulta nos SGBDR, através da utilização da linguagem SQL, em sistemas NoSQL ainda nada se aproxima desta simplicidade.

O NoSQL deverá ser visto como uma nova possibilidade para dados com determinadas características e não algo que vem substituir os SGBDR. Deverão existir como tecnologias complementares em vez de competidoras entre si.

ACKNOWLEDGMENTS

UNIAG, R&D unit funded by the FCT – Portuguese Foundation for the Development of Science and Technology, Ministry of Science, Technology and Higher Education.

REFERÊNCIAS

Ayman, E., Ahmed, I., Haitham, A. & Hesham, A. (2016). A middle layer solution to support ACID properties for NoSQL databases", Journal of King Saud University – Computer and Information Sciences, 28, 133–145.

Bernardino, J. & Abramova, V. (2013). NoSQL Databases: MongoDB vs Cassandra". C3S2E '13 - International C* Conference on Computer Science and Software Engineering, Julho 10 - 12, Porto, Portugal. ACM.

Cattell, R. (2011). Scalable SQL and NoSQL datastores. ACM SIGMOD, (39: 4), 12–27.

Chang et. al. (2016). Bigtable: a distributed storage system for structured data", Proceedings of the 7th USENIX Symposium on Operating Systems Design and Implementation, Seattle, EUA, USENIX Association, 7, 15-15.

Celko, J. (2014). Joe Celko's Complete Guide to NoSQL: 1ª Edição. EUA, Morgan Kaufmann / Elsevier.

Codd, E. F. (1970). A Relational Model of Data for Large Shared Data Banks, Communications of the ACM, (13: 6), 377-387.

Decandia et. al. (2017). "Dynamo: amazon's highly available key-value store". SIGOPS Oper. Syst. Rev., 41, 205-220.

Demchenko, Y. & Grosso, P., Laat, C. D. & Membrey, P. (2013). Addressing Big Data Issues in Scientific Data Infrastructure. Collaboration Technologies and Systems (CTS), International Conference on, 20-24 Maio 2013 San Diego, California, EUA. IEEE Computer Society, 48 – 55.

Gilbert, S. & Lynch, N. (2002). Brewer's conjecture and the feasibility of consistent, available, partition-tolerant web services, ACM SIGACT News (33:2), 51.

Hecht, R. & Jablonski, S. (2011). NoSQL evaluation: Ause case oriented survey. In: IEEE International Conference on Cloud and Service Computing (CSC2011), 336–341.

Kaur, K. & Rani, R. (2013). Modeling and querying data in NoSQL databases. Big Data, IEEE International Conference, 6-9 Oct. Silicon Valley, California, EUA. IEEE Computer Society, 1-7.

Krepska, E., Kielmann, T., Fokkink, W. & Bal, H. (2011). HipG: parallel processing of large-scale graphs", ACM SIGOPS Oper.Syst. (45: 2), 3–13.

Krishnan, K. (2013). Data Warehousing in the Age of Big Data: 1ª Edição. Massachusetts, Estados Unidos da América, Elsevier.

Leavitt, N. (2010). Will NoSQL databases live up to their promise? Computer (43: 2), 12–14.

Lith, A. & Mattsson, J. (2010). Investigating storage solutions for large data (Ph.D.thesis), Department of Computer Science and Engineering Chalmers University of Technology.

McCreary, D. & Kelly, A. (2014). Making Sense of NoSQL: A Guide for Managers and the Rest of Us: 1ª Edição. Shelter Island, Nova York, EUA, Manning Publications.

McCreary, D. & Kelly, A. (2014). Making Sense of NoSQL: A Guide for Managers and the Rest of Us: 1ª Edição. Shelter Island, Nova York, EUA, Manning Publications.

Moniruzzaman, A. & Hossain, S. A. (2013). NoSQL Database: New Era of Databases for Big data Analytics - Classification, Characteristics and Comparison. In International Journal of Database Theory and Application, (6: 14).

Nayak, A., Poriya, A. & Poojary, D. (2013). Type of NOSQL Databases and its Comparison with Relational Databases. International Journal of Applied Information Systems, 5, 16-19.

Orend, K. (2010). Analysis and classification of NoSQL databases and evaluation of their ability to replace an object-relational persistence layer (Master'sthesis), Technische Universität München.

Padhy, R., Patra, M. & Satapathy, S. (2011). RDBMS to NoSQL: Reviewing some next-generation non-relational database's", Int. J. Adv. Eng. Sci. Technol. (11: 1), 15–30.

Robinson, I., Webber, J. & Eifrem, E. (2013). Graph Databases: 1ª Edição. Sebastopol, California, EUA, O'Reilly Media.

Sakr, S., Liu, A., Batista, D. & Alomari, M. (2011). A survey of large scale data management approaches in cloud environments, IEEE Commun. Surv. Tutorials, (13: 3), 311–336.

Sathi, A. (2012). Big Data Analytics: Disruptive Technologies for Changing the Game: 1ª Edição. Canadá, MC Press Online.

Singh, S. & Singh, N. (2012). Big Data Analytics. Communication, Information & Computing Technology (ICCICT), [International Conference on, Outubro 19-20 2012 Mumbai, India. IEEE Computer Society].

Stonebraker, M., & Cattell, R. (2011). 10 rules for scalable performance in 'simple operation' datastores, Communications of the ACM, (54: 6), 72–80.

Tudorica, B. & Bucur, C. (2011). A comparison between several NoSQL databases with comments and notes. In: Proceedings of the 10th Roedunet International Conference (RoE- duNet2011), 1–5.