

Migração de um banco de dados relacional, para o não relacional utilizando Azure Cosmos DB.	Kevin S. de Siqueira; Mateus P. de Oliveira; William Z. da S. Rosa; Mariângela F. F. Molina
---	---

MIGRAÇÃO DE UM BANCO DE DADOS RELACIONAL PARA O NÃO RELACIONAL UTILIZANDO AZURE COSMOS DB

KEVIN SANTOS DE SIQUEIRA¹
MATEUS PAZ DE OLIVEIRA²
WILLIAM ZACARIAS DA SILVA ROSA³
MARIANGELA FERREIRA FUENTES MOLINA⁴

RESUMO

Este artigo apresenta as técnicas necessárias para realizar a migração de um banco de dados relacional para um não relacional Cosmos DB em ambiente *Cloud*. Foram realizadas seis etapas, sendo elas: levantamento de requisitos, análise da base de dados de origem, modelagem base de dados Cosmos DB, migração dos dados, validação do processo de migração e otimização dos dados após a migração. Também foram realizados processos para comparar o tempo de busca na manipulação e o tamanho final de cada base de dados. Com os resultados dos comparativos nota-se que as técnicas adotadas para migração utilizando Cosmos DB apresentaram eficiência, performance e modernização dos dados de um modelo relacional para não-relacional.

Palavras-chave: Banco de dados; Migração; Cosmos DB.

ABSTRACT

This paper presents the techniques necessary to perform the migration from a relational database to a non-relational Cosmos DB in a Cloud environment. Six steps were carried out, namely: requirements gathering, analysis of the source database, Cosmos DB database modeling, data migration, validation of the migration process and data optimization after migration. Processes were also carried out to compare the manipulation search time and the final size of each database. With the results of the comparisons, it is noted that the techniques adopted for migration using Cosmos DB showed efficiency, performance and data modernization from a relational to non-relational model.

Key words: Databases; Migration; Cosmos DB.

¹Graduando em Tecnologia em Análise e Desenvolvimento de Sistemas, Faculdade de Tecnologia de Mogi das Cruzes - FATEC-MC. E-mail: kevin.siqueira@fatec.sp.gov.br.

²Graduando em Tecnologia em Análise e Desenvolvimento de Sistemas, Faculdade de Tecnologia de Mogi das Cruzes - FATEC-MC.

³Graduando em Tecnologia em Análise e Desenvolvimento de Sistemas, Faculdade de Tecnologia de Mogi das Cruzes - FATEC-MC.

⁴Docente, Faculdade de Tecnologia de Mogi das Cruzes – FATEC-MC.

Revista Eletrônica Anima Terra, Faculdade de Tecnologia de Mogi das Cruzes – FATEC-MC.

Mogi das Cruzes-SP., nº16, ano VIII, p.118-130, 1º semestre, 2023. ISSN 2526-1940.

Migração de um banco de dados relacional, para o não relacional utilizando Azure Cosmos DB.	Kevin S. de Siqueira; Mateus P. de Oliveira; William Z. da S. Rosa; Mariângela F. F. Molina
---	---

INTRODUÇÃO

Com a crescente diversidade de origens de dados que muitas vezes possuem informações correlacionadas, surge também a necessidade de as organizar para estudos. Para tanto são necessárias identificá-las, extraí-las, transformá-las e carregá-las em uma nova camada para que sejam consumidas por cientistas de dados, analistas de dados e aplicativos.

Em aplicações que envolvam um grande volume de dados e necessidade de orquestração da ingestão destes dados, é de responsabilidade de Engenheiros de Dados a definição de estratégias e melhores práticas para que estas atividades sejam executadas e a entrega dos dados transformados sejam feitas na camada de consumo. Todavia, obter dados, documentos e informações por meio de consultas nos bancos de dados relacionais em grandes escalas, estão se tornando bastante complexas, gerando um baixo desempenho e insatisfação na qualidade do serviço em alguns cenários específicos.

Segundo Leite (2010), a utilização de bancos de dados relacionais tem apresentado algumas limitações, principalmente porque sua natureza estrutural não permite muita flexibilidade na estruturação desses dados. Como resultado, surgiram novas soluções no mercado com o objetivo de preencher essas lacunas, principalmente relacionadas a questões de escalabilidade e disponibilidade do sistema. Dentre essas soluções, destacam-se os bancos de dados *NoSQL*.

É necessário complementar o discurso de Leite (2010), acrescentando o aumento da necessidade de distribuição dos dados de maneira global. Ou seja, grandes corporações que possuam filiais em vários países e com necessidade de escrita e leitura dos dados de maneira descentralizada afim de diminuir a latência em sua taxa de transferência (*throughput*).

Neste contexto, este artigo tem como objetivo apresentar as etapas e técnicas necessárias para a migração de um banco dados relacionais para o banco de dados não relacional Cosmos DB. As etapas realizadas foram: levantamento de requisitos,

Migração de um banco de dados relacional, para o não relacional utilizando Azure Cosmos DB.	Kevin S. de Siqueira; Mateus P. de Oliveira; William Z. da S. Rosa; Mariângela F. F. Molina
---	---

análise da base de dados, modelagem da base de dados, migração dos dados, validação do processo de migração e, finalmente, a otimização dos dados.

MATERIAL E MÉTODOS

Para alcançar o objetivo de identificar vantagens e desvantagens de determinadas ferramentas durante o processo de migração de dados de um banco relacional para um banco não relacional, foram utilizados métodos de pesquisa qualitativa e quantitativa, exploratória e bibliográfica. Foram abordados tópicos a respeito das áreas envolvidas para que fosse possível o desenvolvimento do software proposto.

Big Data

Nos últimos anos os volumes de dados vêm crescendo exponencialmente, em virtude da grande difusão de computadores e da adesão cada vez maior de empresas e usuários a diferentes dispositivos que geram informações. Segundo a IBM (2022), as fontes de dados atuais estão se tornando cada vez mais complexas porque estão sendo impulsionadas por inteligência artificial (IA), dispositivos móveis, mídias sociais e Internet das Coisas (IoT), sendo gerados em grande escala e em tempo real.

Hoje, é possível imaginar o porquê desse aumento, já que todos os dias milhões de e-mails são trocados, novas fotos são armazenadas e milhares de novos vídeos aparecem. Trazendo para o âmbito corporativo, a maioria das empresas e setores do governo armazenam e analisam informações de seus bancos de dados, buscando melhorar o atendimento prestado aos clientes, usuários e público em geral. Diante desse cenário, foi percebido que as ferramentas convencionais já não seriam mais o suficiente e então nos deparamos com o *Big Data* (MURARI; CUNHA, 2014).

De acordo com Vieira et al. (2012), o *Big Data* pode ser definido como um grande conjunto de dados que requer ferramentas específicas para armazenar, recuperar e manipular dados e documentos. Em complemento a isso Murari e Cunha

Migração de um banco de dados relacional, para o não relacional utilizando Azure Cosmos DB.	Kevin S. de Siqueira; Mateus P. de Oliveira; William Z. da S. Rosa; Mariângela F. F. Molina
---	---

(2014) citam em seu artigo que Gonsowski (2012), diz que o *Big Data* é baseado em cinco “Vs”, que são:

- Velocidade – Independentemente do tamanho do banco de dados, os tempos de resposta para armazenamento e recuperação de dados devem ser mais rápidos do que com outros tipos de banco de dados;
- Volume – As ferramentas utilizadas em *Big Data* devem ser capazes de suportar o volume cada vez maior de dados e documentos;
- Variedade – É possível localizar as informações de forma estruturada, porém existem várias outras formas de armazenar os dados, como arquivos de texto, imagens, vídeos, som, entre outros;
- Veracidade – Enfatiza a importância de todos os dados armazenados serem precisos e consistentes;
- Valor – Todos os elementos destacados acima são irrelevantes se não gerarem as informações corretas para o negócio.

Na visão de alguns especialistas, não é necessário que todos os fatores existam ao mesmo tempo para que um ambiente seja considerado *Big Data*, pois há casos em que mais ênfase é colocada em uma característica ou outra, e em outros casos, não é totalmente definido (RAUTEMBERG; CUNHA, 2019). Contudo, com as características apresentadas é possível ter um maior entendimento a respeito do termo.

Banco de Dados Relacional

O modelo de dados relacional foi o primeiro modelo existente, criado por volta de 1970 por Edgar Frank Codd. Seguindo este modelo, todos os dados são armazenados em tabelas, cada coluna da tabela é conhecida como um atributo e todas as linhas são chamadas de tuplas. De acordo com Costa (2011), cada atributo possui um domínio que define o tipo e/ou número de caracteres que o atributo suportará.

Migração de um banco de dados relacional, para o não relacional utilizando Azure Cosmos DB.	Kevin S. de Siqueira; Mateus P. de Oliveira; William Z. da S. Rosa; Mariângela F. F. Molina
---	---

Na maioria dos modelos, existem chaves primárias para identificar tuplas de forma exclusiva. Esta chave envolve uma ou mais colunas cujos valores distinguem uma linha da outra na base de dados. Todavia, ainda temos a possibilidade de criar um relacionamento por meio do uso de chaves estrangeiras, criando uma relação entre duas tuplas de tabelas diferentes. Para chaves estrangeiras, também podem ser utilizadas uma ou mais colunas, que devem estar relacionadas às chaves primárias de outras tabelas de modo a estabelecer relacionamentos entre elas (MURARI; CUNHA apud VASCONCELOS, C., 2014).

Bancos de dados relacionais, em geral, são baseados na arquitetura ACID, para garantir sua integridade. A sigla ACID significa Atomicidade, Consistência, Isolamento e Durabilidade e suas características têm os seguintes significados (SOUZA; PEREIRA 2015):

- Atomicidade: caso haja uma transação com falha, o efeito de sua operação não será aplicado no banco de dados;
- Consistência: os dados devem ser tratados a cada solicitação de transação do banco de dados;
- Isolamento: diferentes usuários conseguem acessar a mesma tupla em paralelo, por meio do isolamento entre as transações.
- Durabilidade: garantir que os dados estarão sempre acessíveis, após a confirmação de uma transação.

Além de seguir essa arquitetura, outro fato comum dos bancos de dados relacionais é a utilização do SQL (*Structured Query Language*) como padrão. Essa linguagem foi criada em 1970 pela IBM e utiliza conceitos de DDL (*Data Definition Language*) e DML (*Data Manipulation Language*). O DDL trata das especificações de relacionamentos, domínios, regras de integridade, entre outros aspectos estruturais, enquanto o DML nos permite realizar operações de controle (COSTA, 2011).

Ainda assim, segundo Leite (2010), a utilização de bancos de dados relacionais tem apresentado algumas limitações, principalmente porque sua

Migração de um banco de dados relacional, para o não relacional utilizando Azure Cosmos DB.	Kevin S. de Siqueira; Mateus P. de Oliveira; William Z. da S. Rosa; Mariângela F. F. Molina
---	---

natureza estrutural não permite muita flexibilidade na estruturação desses dados. Como resultado, surgiram novas soluções no mercado com o objetivo de preencher essas lacunas, principalmente relacionadas a questões de escalabilidade e disponibilidade do sistema. Dentre essas soluções, destacam-se os bancos de dados *NoSQL (Not Just SQL)*.

Banco de Dados NoSQL

Por volta de 2009, os bancos de dados *NoSQL* começaram a ganhar popularidade, vindo como uma possível solução para problemas como escalabilidade e manipulação de grandes volumes de dados (MURARI; CUNHA, 2014). Em complemento a isso, Leite (2010) diz que isto deve-se, principalmente, à vasta dificuldade relacionada ao ordenamento de dados em diversos servidores enquanto empregado um banco de dados relacional.

Segundo Nascimento (2010), os bancos de dados *NoSQL* são divididos com base em como armazenam e gerenciam dados, que podem ser os seguintes:

- *Document Store* – Este modelo é baseado em arquivos *JSON (JavaScript Object Notation)* ou *XML (Extensible Markup Language)*, que por sua vez podem ser identificados pelo seu ID único ou por qualquer registro existente no documento;
- *Graph Store* – Este modelo armazena os dados na forma de objetos e as pesquisas são realizadas nesses objetos;
- *Wide Columns Store* – Este modelo suporta várias colunas e linhas, como os bancos de dados *BigTable* e *Cassandra*;
- *Key/Value Store* – Esse modelo é o que suporta o maior número de dados, usando uma chave e um valor para essa chave;
- *Column Oriented Store* – Este modelo armazena a partir de colunas e não linhas, como normalmente acontece em outros bancos de dados.

O Cosmos DB possui alguns benefícios em relação a alguns outros bancos de dados *NoSQL* por oferecer seu serviço em nuvem no modelo *SaaS – Software*

Migração de um banco de dados relacional, para o não relacional utilizando Azure Cosmos DB.	Kevin S. de Siqueira; Mateus P. de Oliveira; William Z. da S. Rosa; Mariângela F. F. Molina
---	---

as a *Service* (Software como Serviço), o permitindo ser auto gerenciável pela plataforma de nuvem do ponto de vista da infraestrutura, o serviço permite também configurar a distribuição do banco de dados de maneira global, e por fim, mas não menos importante, é possível modelar dados em quatro famílias de *NoSQL* das citadas acima, que são *Document Store*, *Graph Store*, *Wide Columns Store* e *Key/Value Store*.

Grandes empresas começaram a criar suas próprias soluções para problemas que envolviam grandes volumes de dados e documentos, passaram a publicar artigos científicos descrevendo essas soluções encontradas para gerenciar o gerenciamento de dados distribuídos em larga escala, sem utilizar a terminologia *NoSQL* (DIANA; GEROSAL, 2010).

Migração de Dados

A migração de dados é de grande importância para projetos de software, só que muitas vezes passam despercebidos. A migração de dados pode atrasar e até travar um projeto de software, por isso o processo de migração de dados deve ser cuidadosamente aplicado e validado, para evitar perda ou inconsistência de informações (GUEDES, 2013).

Durante a fase de planejamento deve ser criada uma estratégia que vise determinar quais informações devem ser movidas em uma perspectiva mais ampla, e esse processo geralmente inclui todos os stakeholders do projeto. Após a criação da estratégia, a base de dados deve ser analisada e buscada as informações exigidas pelos *stakeholders* (GUEDES, 2013).

Durante a implementação do software de migração, os bancos de dados de origem e destino devem ser modelados usando diagramas. Em seguida, o software deve ser desenvolvido para mover os dados de um banco de dados para outro (GUEDES, 2013).

No processo de validação devem ser feitos testes que para responder questões como, quantos registros foram migrados, se os dados foram para as

Migração de um banco de dados relacional, para o não relacional utilizando Azure Cosmos DB.	Kevin S. de Siqueira; Mateus P. de Oliveira; William Z. da S. Rosa; Mariângela F. F. Molina
---	---

colunas corretas e até mesmo se os dados estão em formatação correta. Com essas respostas em mãos, um relatório indicará se a análise, o projeto, o desenvolvimento e os testes precisam ser refeitos (GUEDES, 2013).

Phyton

O sistema foi implementado por meio da linguagem de programação Python. Python é uma linguagem de programação criada por Guido van Rossum em 1991. A linguagem tinha como principal objetivo a produtividade e legibilidade. Seu código fonte é aberto e funciona em qualquer plataforma (PYSCIENCE BRASIL, 2014).

Python é uma linguagem de programação considerada de fácil aprendizagem. Existem diversas extensões que podem ser adicionadas ao Python com o objetivo de adaptar a linguagem para cada necessidade. Neste trabalho foram utilizadas as bibliotecas externas *azure.cosmos*, criada e mantida pela Microsoft, e *pandas*, que permitirá o uso do objeto *DataFrame* para manipulação de dados (PYSCIENCE BRASIL, 2014).

Azure Migrate

O Azure Migrate fornece um hub centralizado para avaliar e migrar servidores, infraestrutura, aplicativos e dados locais para o Azure. Ele fornece o seguinte (MICROSOFT, 2022):

- Plataforma de migração unificada - Por meio de um único portal somos capazes de iniciar, executar e acompanhar sua migração para o Azure;
- Gama de ferramentas - As ferramentas de migração incluem: descoberta e avaliação, migração de servidor e se integram a outros serviços e ferramentas do Azure e a ofertas de fornecedores de software independentes (ISV).
- Avaliação e migração - Estando no hub do Azure Migrate, você é possibilitado a avaliar e migrar servidores, bancos de dados, aplicativos web e áreas de trabalho virtuais.

Migração de um banco de dados relacional, para o não relacional utilizando Azure Cosmos DB.	Kevin S. de Siqueira; Mateus P. de Oliveira; William Z. da S. Rosa; Mariângela F. F. Molina
---	---

Azure Data Factory

O Azure Data Factory (ADF) é um serviço de integração de dados baseado em nuvem para automatizar a transformação e migração de dados. O conceito base do modelo ADF é o pipeline. Um pipeline é um grupo lógico de operações, cada uma das quais define as ações a serem executadas nos dados contidos no conjunto de dados. Os serviços vinculados definem as informações necessárias para que o Data Factory se conecte a recursos de dados (MICROSOFT, 2022).

Considere usar o ADF quando dados precisarem ser migrados continuamente em um cenário híbrido que acessa recursos locais e de nuvem e quando os dados precisam de transformações ou têm lógica de negócios adicionada a eles ao serem migrados (MICROSOFT, 2022).

RESULTADOS E DISCUSSÃO

Foi traçado um estudo comparativo entre os sistemas encontrados na área de migração de dados para chegamos na solução implementada. Durante esse estudo foi verificado que o Azure Migration possui uma interface mais amigável e oferece migração facilitada para as bases de dados que possui suporte, porém sua lista de bases suportadas é limitada o que pode não atender a necessidades de migração em base de dados específicas.

Já no caso do Azure Data Factory, a quantidade de bases de dados suportadas pelo provedor da ferramenta é muito abrangente e em casos específicos de bases de dados proprietárias, ou seja, que não possuem uma conexão nativa com a ferramenta, ainda é possível utilizar o recurso Azure Function suportado pelo ADF, que permite consumir API's que atendam a necessidade de conectividade. Por outro lado, utilizar o processo de migração com *ADF* passa a ser oneroso para pequenas demandas ou necessidade de carga com data de expiração curta.

Pensando em um cenário para necessidades de mapeamento de origens específicas para pequenas demandas, foi elencada a criação de aplicativos para conexão e migração dos dados utilizando a linguagem de programação Python,

Migração de um banco de dados relacional, para o não relacional utilizando Azure Cosmos DB.	Kevin S. de Siqueira; Mateus P. de Oliveira; William Z. da S. Rosa; Mariângela F. F. Molina
---	---

como alternativa às ferramentas de migração existentes no mercado. Sendo possível desta forma a conectividade com todas as origens que possuam *drivers* disponíveis para a linguagem e o tratamento e escrita destes dados no destino. Esta comparação é apresentada no quadro 1.

Quadro 1: Comparativo entre as ferramentas estudadas.

CARACTERÍSTICAS	FERRAMENTA		
	AZURE MIGRATE	ADF	PYTHON
Complexidade de migração baixa	SIM		
Suporte a muitos Bancos de dados relacionais		SIM	SIM
Suporte a muitos Bancos de dados não-relacionais		SIM	SIM
Ideal para pequenas demandas de migração			SIM

Fonte: Os autores (2022).

Como visto no Quadro 1, a solução desenvolvida realiza uma atividade que as demais ferramentas estudadas neste artigo não, que é realizar a migração para o banco de dados Cosmos DB em um cenário de baixa complexidade e até mesmo de custos otimizados de operação.

A implementação da solução foi desenvolvida em 6 etapas, conforme abaixo:

- Levantamento de requisitos e análise da base de dados relacional – Têm por objetivo demonstrar a migração de dados relacionais para um banco de dados não-relacional;
- Modelagem da base de dados não-relacional - Para a modelagem do banco de dados Cosmos DB foram transcritas as tabelas de origem com as características e estruturas necessárias para o armazenamento em banco de dados *NoSQL*;
- Configuração do software – Foram configuradas as dependências de software, sendo elas do banco de dados de origem, Cosmos DB, Python e ambiente Azure;
- Migração – Processo que visa buscar todas as informações da base de dados de origem, e transferi-las para o Cosmos DB;

Migração de um banco de dados relacional, para o não relacional utilizando Azure Cosmos DB.	Kevin S. de Siqueira; Mateus P. de Oliveira; William Z. da S. Rosa; Mariângela F. F. Molina
---	---

- Validação da migração - O processo de validação tem o objetivo de garantir que todo o conteúdo das tabelas de origem foi migrado, isso ocorre por validação quantitativa e qualitativa dos dados;
- Otimização dos dados - Visa remover colunas sem conteúdo, ou com conteúdo idêntico em todos os registros, assim como melhorar a tipagem dos dados na base de dados de destino (Cosmos DB).

CONCLUSÃO

Durante o processo de migração foi notado, com base no desenvolvimento e demonstração do software apresentado, que é possível realizar a migração de uma base de dados relacional para uma base de dados não relacional, sem precisar fazer alterações na base de dados e perder informações durante o processo. Durante o desenvolvimento do software foram realizados estudos sobre *BigData*, Azure Cosmos DB e Python. Estes tópicos não haviam sido abordados durante o curso na graduação em Análise e Desenvolvimento de Sistemas, sendo que os conhecimentos práticos aqui aplicados se baseiam em sua maior parte na experiência dos autores com as ferramentas e tecnologias, os quais em sua maioria já atuam no mercado de trabalho com este tipo de atividade. O software não possui interface gráfica, o que para usuários com menor conhecimento técnico, pode trazer alguma dificuldade em seu manuseio. Fica como sugestão de trabalho futuro, desenvolver uma interface gráfica para configurar as conexões com o banco de dados, quais tabelas devem ser processadas, quais processos realizar e criar relatórios para facilitar a leitura de cada processo realizado pelo software. Outra melhoria que poderia ser implementada é permitir que sejam feitas outras migrações, utilizando outros bancos de dados, tais como a migração de MySQL para Cosmos DB. A estrutura do software está dinâmica e adicionar suporte a novos bancos de dados seria simples. Por fim, o código fonte ficará disponível para toda a comunidade acadêmica, possibilitando assim, futuras atualizações e correções, conforme a

Migração de um banco de dados relacional, para o não relacional utilizando Azure Cosmos DB.	Kevin S. de Siqueira; Mateus P. de Oliveira; William Z. da S. Rosa; Mariângela F. F. Molina
---	---

necessidade. O software implementado realiza o que foi proposto de forma eficaz, suas rotinas são de média complexidade, porém ainda há muito que pode ser feito para tornar o software mais fácil de utilizar e realizar o mesmo processo entre outros bancos de dados.

REFERÊNCIAS BIBLIOGRÁFICAS

AUTORES. **Código** **Fonte.** 2022. Disponível em: <https://github.com/WiliamRosa/CosmosDB/tree/main>. Acesso em: 30 mai. 2022.

COSTA, Elisângela Rocha. **Banco de Dados Relacionais**. Faculdade de Tecnologia de São Paulo, São Paulo, 2011. Disponível em: <http://www.fatecsp.br/dti/tcc/tcc0025.pdf>. Acesso em: 30 mai. 2022.

DIANA, M.; GEROSAL, M. A. **NOSQL na Web 2.0: Um Estudo Comparativo de Bancos Não Relacionais para Armazenamento de Dados na Web 2.0**. Departamento de Ciência da Computação – Universidade de São Paulo, 2010. Disponível em: https://www.ime.usp.br/~mdeidiana/nosql_wtdbd10.pdf. Acesso em: 30 mai. 2022.

GUEDES, Anselmo. **Migração de dados: não deixe essa atividade tornar seu projeto um fracasso**. 2013. Disponível em: <https://imasters.com.br/desenvolvimento/migracao-de-dados-nao-deixe-essa-atividade-tornar-seu-projeto-um-fracasso>. Acesso em: 30 mai. 2022.

IBM. **Big data analytics**. 2022. Disponível em: <https://www.ibm.com/analytics/big-data-analytics>. Acesso em: 30 mai. 2022.

LEITE, Gleidson Sobreia. **Análise Comparativa do Teorema CAP Entre Bancos de Dados NoSQL e Bancos de Dados Relacionais**. Monografia (Graduação)-Curso de Ciência da Computação, Faculdade Farias Brito, Fortaleza, 2010. Disponível em: <https://fbuni.edu.br/sites/default/files/tcc-20102-gleidson-sobreia-leite.pdf>. Acesso em: 30 mai. 2022.

MICROSOFT. **Azure Cosmos DB documentation**. 2022. Disponível em: <https://docs.microsoft.com/en-us/azure/cosmos-db/>. Acesso em: 30 mai. 2022

MICROSOFT. **Sobre as Migrações para Azure**. 2022. Disponível em: <https://docs.microsoft.com/pt-br/azure/migrate/migrate-services-overview>. Acesso em: 30 mai. 2022.

Migração de um banco de dados relacional, para o não relacional utilizando Azure Cosmos DB.	Kevin S. de Siqueira; Mateus P. de Oliveira; William Z. da S. Rosa; Mariângela F. F. Molina
---	---

MURARI, M. A.; Cunha, B. **Desenvolvimento de um software para migração de um banco de dados relacional Firebird, para o não relacional MongoDB.** Universidade Federal de Santa Maria, Rio Grande do Sul, 2014. Disponível em: https://repositorio.ufsm.br/bitstream/handle/1/12904/TCCG_SIFW_2014_MURARI_MAURO.pdf?sequence=1&isAllowed=y. Acesso em: 20 mai. 2022.

NASCIMENTO, Jean. **NoSQL – você realmente sabe do que estamos falando?** 2010. Disponível em: <https://imasters.com.br/banco-de-dados/nosql-voce-realmente-sabe-do-que-estamos-falando>. Acesso em: 30 mai. 2022

PYSCIENCE BRASIL. **Python: O que é? Por que usar?** 2014. Disponível em: <http://pysciencebrasil.wikidot.com/python:pythonoqepq>. Acesso em: 30 mai. 2022.

SOUZA, G.; PEREIRA, J. L. **Document-Based Databases:** Estudo exploratório no âmbito das Bases de Dados NoSQL. 15ª Conferência da Associação Portuguesa de Sistemas de Informação (CAPSI 2015), Lisboa, Portugal, v. 15, p. 475–491, 2015. DOI: <http://dx.doi.org/10.18803/capsi.v15.475-491>. Disponível em: https://repositorium.sdum.uminho.pt/bitstream/1822/51744/3/CAPSI2015_GS_JLP.pdf. Acesso em: 30 mai. 2022.

VIEIRA, M. R.; FIGUEIREDO, J. M.; LIBERATTI, G.; VIEBRANTZ, A. F. **Bancos de Dados NoSQL:** Conceitos, Ferramentas, Linguagens e Estudos de Casos no Contexto de Big Data. Simpósio Brasileiro de Bancos de Dados. 2012. Disponível em: https://sbbd.org.br/2012/SBBD2012/artigos/pdfs/sbbd_min_01.pdf. Acesso em: 20 mai. 2022.