Migração de Dados Utilizando o Cluster Hadoop

Marcus Norberto[1], Maicon Bernardino[1], Rodrigo Machado[1], Robson Gonçalves[2]

[1]Laboratory of Empirical Studies in Software Engineering (LESSE)
Universidade Federal do Pampa (UNIPAMPA)
Av. Tiaraju, 810 – Alegrete, RS – Brasil

[2]Diretoria de Tecnologia da Informação e Comunicação (DTIC)
Universidade Federal do Pampa (UNIPAMPA)
Av. Tiaraju, 810 – Alegrete, RS – Brasil

ocramdf@hotmail.com, bernardino@acm.org, rodrigo.blizzard92@gmail.com,
robsongoncalves@unipampa.edu.br

Abstract. *This article presents a solution for performing data migration between a relational and nonrelational database. It also presents its development, implementation, and verification of its performance in a real environment.*

Resumo. *Este artigo apresenta uma solução para a realização de migração de dados entre um banco de dados relacional e não relacional. Também apresenta seu desenvolvimento, implementação, e uma verificação do seu desempenho em um ambiente real.*

1. Introdução

Com o avanço tecnológico as informações estão cada vez mais acessível, dispersando-se quase que de maneira instantânea. Desta forma, é crucial que as organizações se mantenham em constante adaptação para estarem capacitadas em fornecer informações desejadas o mais rápido possível. Assim, conseguem se manter competitivas no mercado.

Muitas dessas organizações coletam grandes volumes de dados, as quais frequentemente são armazenados em bancos de dados relacionais. Os banco relacionais são amplamente utilizados na indústria, porém eles não são capazes de armazenar e processar dados grandes de forma eficaz e não são muito eficientes para fazer transações e participar de operações [Abramova and Bernardino 2013]. Com o intuito de resolver alguns desses problemas, surgiu o um novo paradigma, bancos de dados não relacionais (NoSQL).

Os dados acumulados por essas organizações são um dos ativos mais importantes para elas. Pressionadas pela demanda do mercado e as mudanças tecnológicas, as organizações, de tempos em tempos, migram seus dados para outro tipo de sistema de informação para se manterem competitivas. Portanto, os dados do sistema legado precisam ser migrados para o novo sistema [Karnitis and Arnicans 2015].

Este estudo apresenta uma solução que realiza a migração de dados de um sistema de gerenciamento de banco de dados (SGBD) relacional para um SGBD não relacional. Porém, esta solução é apenas uma parte de uma projeto *open source* proposto pela Diretoria de Tecnologia da Informação e Comunicação[1] (DTIC). Este pedaço da solução foi

[1]DTIC: https://dtic.unipampa.edu.br

desenvolvida por um estagiário do curso de Engenharia de Software dentro de um setor do DTIC.

A proposta geral é um sistema/componente genérico capaz de apresentar um relatório em formato de gráficos e tabelas de quaisquer tipos de dados recebidos como entrada.

Para realizar este projeto a DTIC convocou uma equipe de 5 estagiários e discutiu possíveis soluções para elaborar o mesmo. Ao final foi criado uma solução que foi dividida em 4 partes: *View*, *API*, e duas soluções para migração de dados entre um SGBD relacional e não relacional. A Figura 1 apresenta uma visão geral do planejamento do projeto.

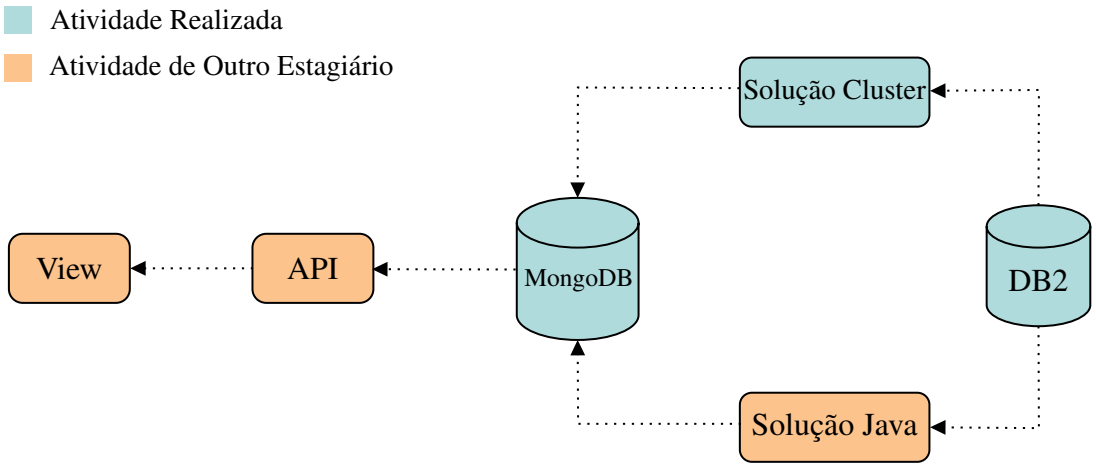

Figura 1. Proposta geral da DTIC.

2. Referencial Teórico

A presente seção apresenta uma série de informações associadas ao ferramental utilizado para desenvolver e aplicar a solução.

2.1. Hadoop

O Hadoop é uma plataforma de *software* de código aberto para computação distribuída confiável e escalável que faz parte do "universo" Apache. A biblioteca de *software* Apache Hadoop é uma estrutura que permite o processamento distribuído de grandes conjuntos de dados em *clusters* de computadores usando modelos de programação simples. Foi projeto para escalar um único servidor para milhares de máquinas, em que cada uma das maquinas oferecem computação e armazenamento local [The Apache Software Foundation 2019a].

2.2. Sqoop

O Apache Sqoop é um projeto de *software* de interface de linha de comandos para transferir dados entre SGBDs relacionais e o Apache Hadoop [The Apache Software Foundation 2019d].

2.3. Hive

O Apache Hive é um projeto de software de *data warehouse* construído sobre o Apache Hadoop para fornecer consulta e análise de dados. O Hive fornece uma interface semelhante a SQL para consultar dados armazenados em vários bancos de dados e sistemas de arquivos que se integram ao Apache Hadoop [The Apache Software Foundation 2019b].

2.4. Nifi

O Apache NiFi também é um projeto de *software* do Apache *Software Foundation*, projetado para automatizar o fluxo de dados entre sistemas de software. O Nifi é uma ferramenta do tipo extração, transformação e carregamento (ETL), ou seja, é capaz de extrair dados de diversos sistemas, transformar os dados de acordo com as regras de negocios e por fim carregar os dados para um *Data Mart* ou *Data Warehouse* [The Apache Software Foundation 2019c].

3. Metodologia

Um dos principais objetivos do projeto desenvolvido pelo DTIC é o desacoplamento da camada de apresentação (*front-end*) da camada de lógica do negocio (*back-end*) e transformá-las em componentes independentes. Para isto, foi adotado uma estratégia com base em *Application Programming Interface* (APIs). A camada de apresentação será responsável por apresentar os relatórios e realizará requisições para a API dos dados que devem ser consumidos. Por sua vez, a API consumirá os dados disponíveis no SGBD não relacional MongoDB, para que os dados sejam tratados de maneira correta. Porém, os dados estão originalmente no SGBD relacional DB2 da *IBM*. Logo, esses dados devem ser migrados para o MongoDB, por meio de uma solução utilizando o *cluster* Hadoop e outra solução escrita na linguagem de programação Java.

A migração dos dados para o MongoDB se deve pelo fato do DB2 estar sendo executado em um sistema operacional em disco (DOS) em uma versão de 32 *bits*, em uma máquina que dispunha de 4 *gigabytes* de memória principal. Devido a esta limitação, frequentemente muitos serviços da UNIPAMPA que dependem desta base de dados sofrem constantes gargalos, em consequência do número elevado de requisições que recebe durantes algumas datas específicas do ano.

Vale ressaltar que as ferramentas mencionadas neste estudo fazem parte apenas da solução que envolve o *cluster* Hadoop. A Figura 1 apresenta uma visão geral do funcionamento da solução.

3.1. Empresa na Qual o Estudo Foi Conduzido

A DTIC é um órgão diretamente vinculado à administração superior da UNIPAMPA, localizado nas cidades de Alegrete e Bagé no Rio Grande do Sul. As atividades foram realizadas no setor da Coordenação de Desenvolvimento de Sistemas (CODEV), responsável pelo desenvolvimento e manutenção dos sistemas institucionais.

3.2. Organização dos Integrantes

A equipe de estagiários responsáveis pelo projeto foram divididos para cada modulo do projeto. A Figura 2 apresenta a comunicação geral dos envolvidos e a responsabilidade de cada atividade. Os estagiários 1 e 2 foram responsáveis pela camada de apresentação,

o estagiário 3 pela API. Enquanto que os estagiários 4 e 5 ficaram responsáveis pela migração dos dados. Porém, cada um dos estagiários 4 e 5 ficou responsável por uma solução diferente.

Todos os estagiários tiveram que manter uma comunicação constante para que todos componentes se comunicassem adequadamente. Além disso, recebiam auxílio de especialista de cada área para ajudar a resolver problemas e discutir ideias durante todo o processo de desenvolvimento de suas atividades. Os mediadores também agiam como ponte para comunicação entre os estagiários e o diretor da DTIC, o qual requisitou o projeto.

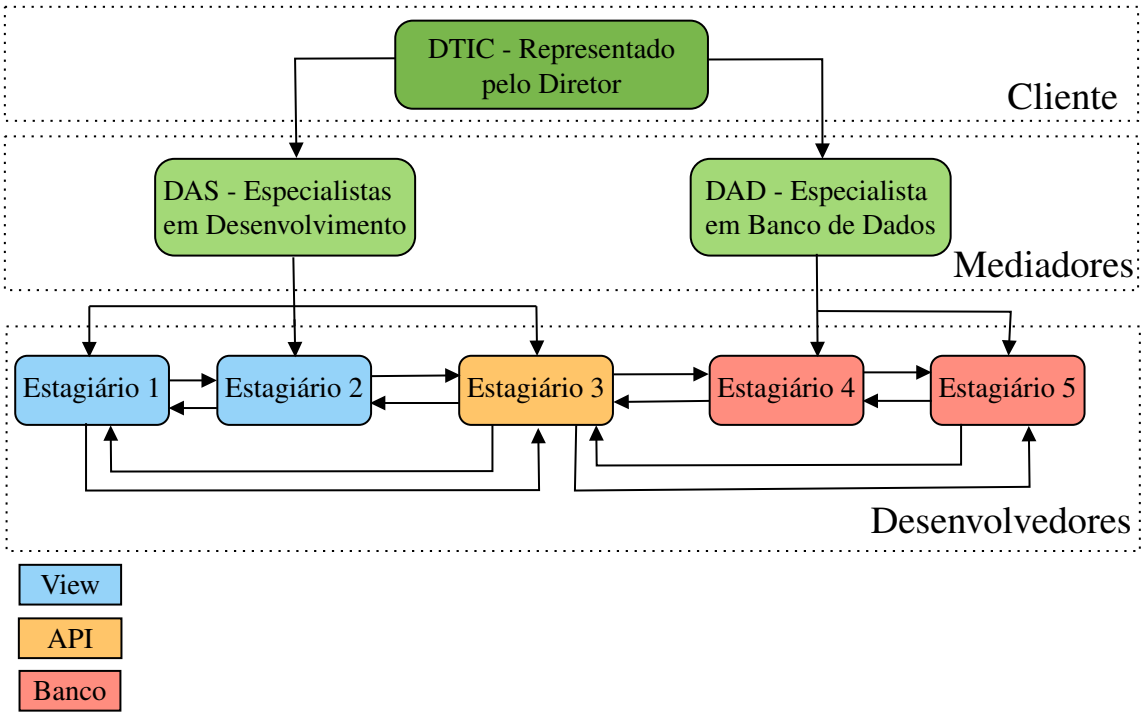

Figura 2. Comunicação geral dos envolvidos e suas responsabilidades.

3.3. Descrição do Processo

O período de elaboração, desenvolvimento e implementação da solução ocorreu entre 15/04/2019 a 11/06/2019.

Inicialmente, a DTIC havia especificou o uso de duas ferramentas para realizar a migração de dados, sendo elas o Hadoop e o Sqoop. Contudo, somente essas ferramentas não foram suficientes para resolver o problema. Uma vez que o Sqoop opera apenas com SGBDs relacionais, e o MongoDB é um SGBD não relacional.

Então, uma nova solução foi elaborada, na qual foi acrescentada mais duas ferramentas, sendo elas o Hivi e o Nifi. A Figura 3 apresenta uma visão geral da comunicação e funcionamento da nova versão da solução de migração de dados.

Primeiramente, todas as ferramentas foram instaladas e configuradas. Em seguida foi realizado a importação dos dados do DB2 por meio do Sqoop para o HDFS do Hadoop, o qual é o local onde o Hadoop armazena os dados. Durante esse processo o Sqoop mantém a mesma estrutura do relacionamento das tabelas da base de dados de origem.

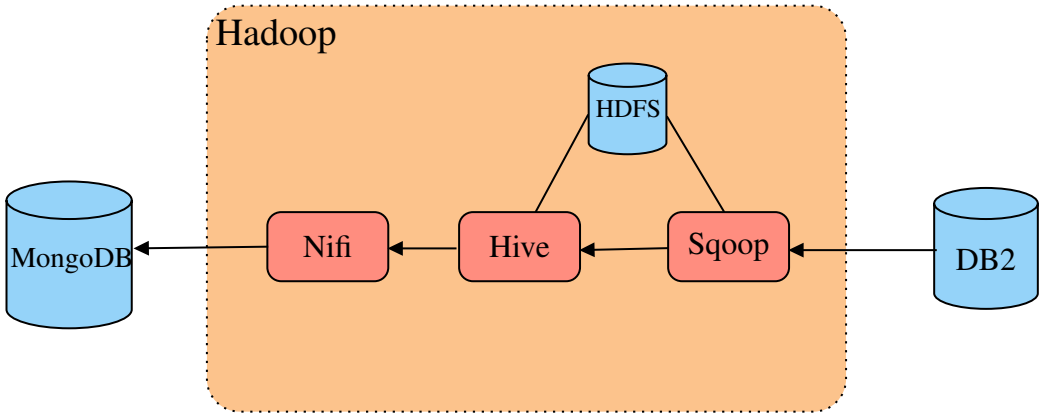

Figura 3. Nova proposta.

Com isso, o Hive pode consultar, atualizar, deletar e criar dados a parti do dados importados para o HDFS por meio de consultas SQL.

Por fim, foi criado um processo de tratamento dos dados no Nifi, com o objetivo de realizar a importação dos dados do HDFS do Hadoop para o MongoDB. Para isso, foi criado 4 componentes que são chamados de processadores pela ferramenta. Esses processadores são componentes internos do Nifi que podem ser configuradas para realizar uma atividade conforme definido pelo usuário. Também podem ser programados para que sejam executados em um tempo determinado. A Figura 4 apresenta uma visão geral do *template*.

O primeiro processador SelectHiveQl é responsável por realizar uma consulta SQL para o Hive retornar os dados armazenados no HDFS do Hadoop. Como resultado o processador retorna um arquivo no formato AVRO. Em seguida, o processador SplitAvro recebe o arquivo e transforma cada ocorrência recuperada em um único arquivo do mesmo tipo. Logo após o processador SplitAvro, o processador ConvertAvroToJson reebe os arquivos e os transforma em um arquivo do tipo JSON. Por fim, o último processador PutMongo recebe cada bloco de arquivo e os insere em uma coleção do MongoDB definida.

Vale salientar que todas as ferramentas apresentadas trabalham em conjunto com o *cluster* Hadoop. Também, todas as atividades que são executadas pelo Sqoop, Nifi e Hive são gerenciados pelo *cluster* Hadoop, o qual as atividades podem ser divididas em vários pedaços para que possam ser executadas em cada nó do *cluster*.

Com o objetivo de automatizar todo o processo, foi criado um *Shell script* que recebe um arquivo de configuração em JSON como entrada. Este arquivo contém informações das tabelas que se deseja importar para o MongoDB. Por meio deste *script* o Sqoop executa os comandos necessários para importar as tabelas desejadas do DB2 e armazena de forma correta no HDFS. O *script* foi definido para ser executado de forma automatizada por meio do agendador de tarefas do Linux (crontab). Em seguida foi definida um horário especifico para o Nifi realizar suas atividades. Contudo, todo o processo de migração de dados é executado de forma automatizada.

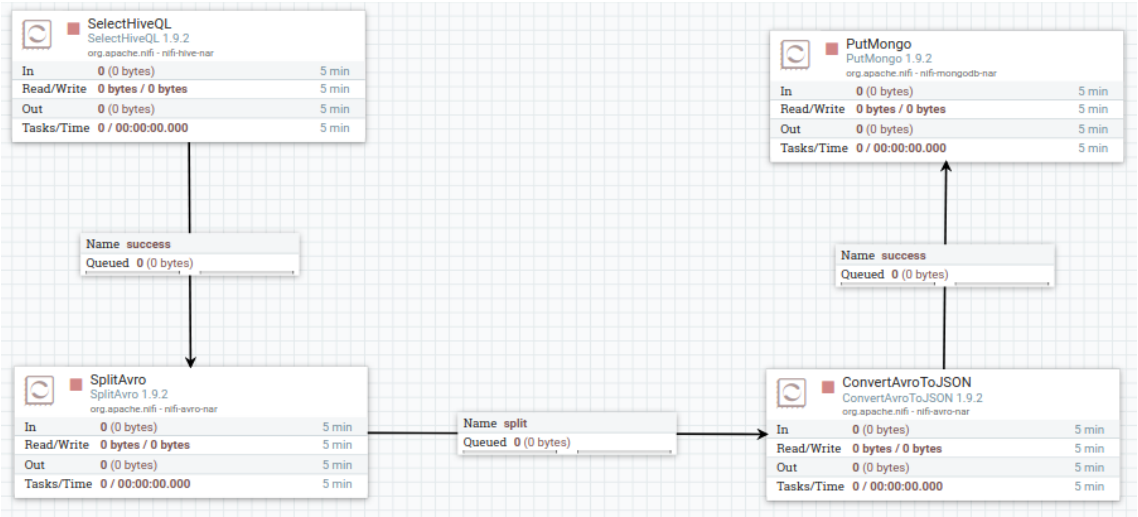

Figura 4. *Template* genérico.

4. Resultados

Para averiguar o desempenho da solução como um todo foi realizado um teste em ambiante real, na qual foi importado uma tabela que continha as informações gerais dos funcionários da Unipampa. Essa tabela possui aproximadamente 200000 registros, com tamanho aproximado de 2,5 *gigabytes*.

O teste foi realizado em um servidor cedido pela DTIC, em que dispunha de 2 *gigabytes* de memória principal e um núcleo de processamento de 1 *gigahertz*. Todo esse recurso foi cedido ao *cluster* Hadoop que foi configurado para ter um nó gerenciador do *cluster* e um nó filho para realizar as atividades.

Apesar da pouca disponibilidade de recurso de *hardware*, a solução obteve um desempenho satisfatório para a DTIC. O tempo estimado de todo o processo de migração levou aproximadamente 12 minutos.

5. Considerações Finais

Apesar da solução ter sido desenvolvida especificamente para o projeto da organização, ela não se limita somente a esse contexto. Pode ser usada em qualquer outro contexto que tenha como objetivo migrar dados de SGBDs relacionais para não relacionais.

Devido a complexidade da implementação da solução, recomenda-se apenas sua utilização em grandes projetos que trabalhem de preferência com *Big Data*. Em contextos que se trabalham com pouco volume de dados é recomendável utilizar soluções mais simples. Apesar de sua complexidade, ela se demonstrou eficiente na migração dos dados com relação ao tempo. Porem foi comparada apenas com a solução java desenvolvida pelo outro estagiário, não foi realizado nenhum outro tipo de experimento para comprovar sua eficiência com demais soluções que tenham o mesmo objetivo.

A solução ainda não esta completa, este trabalho apresenta apenas a migração dos dados. Caso uma tabela que já tenha sido importada para o MongoDB e essa mesma tabela tenha tido seus registros atualizados no DB2, a solução não é capaz de atualizar os dados no MongoDB automaticamente. A DTIC pretende futuramente disponibilizar todo

o projeto, contendo a solução descrita neste estudo, uma vez que trata-se de um projeto *open source*.

Este trabalho apresenta uma proposta de solução automatizada de migração de dados do DB2 para o MongoDB utilizando um *cluster*. Podendo ser facilmente adaptada para outros SGBDs relacionais e não relacionais.

Por fim, foi possível utilizar esta solução para o contexto da DTIC, preenchendo uma lacuna existente dentro da organização para a realização da migração de dados do DB2 da IBM para uma tecnologia que a organização citada tem interesse em adicionar em projetos futuros.

Referências

Abramova, V. and Bernardino, J. (2013). Nosql databases: Mongodb vs cassandra. In *Proceedings of the International C* Conference on Computer Science and Software Engineering*, C3S2E '13, pages 14–22, New York, NY, USA. ACM.

Karnitis, G. and Arnicans, G. (2015). Migration of relational database to document-oriented database: Structure denormalization and data transformation. In *2015 7th International Conference on Computational Intelligence, Communication Systems and Networks*, pages 113–118. IEEE.

The Apache Software Foundation (2019a). Apache hadoop. `https://hadoop.apache.org/`. accessed in 06/09/2019.

The Apache Software Foundation (2019b). Apache hive. `http://precog.iiitd.edu.in/people/anupama`. accessed in 06/09/2019.

The Apache Software Foundation (2019c). Apache nifi. `https://nifi.apache.org/`. accessed in 06/09/2019.

The Apache Software Foundation (2019d). Apache sqoop. `https://sqoop.apache.org/`. accessed in 06/09/2019.