Legacy-DB: Uma ferramenta para validação da migração de dados em bancos de dados heterogêneos

Marlon Duarte[1], **Gustavo Moraes**[2], **Amanda D. P. Venceslau**[1], **Wellington Franco**[1], **Paulo A. L. Rego**[2], **Paulo Honório** [2]

[1]Campus de Crateús – Universidade Federal do Ceará (UFC)
Crateús – CE – Brazil

[2]Campus do Pici – Universidade Federal do Ceará (UFC)
Fortaleza – CE – Brazil

marlongduarte@alu.ufc.br, gustavo.moraes@alu.ufc.br, amanda.pires@ufc.br

wellington@crateus.ufc.br, paulo@dc.ufc.br, paulo.honorio@alu.ufc.br

Abstract. *This article presents the Legacy-DB tool to support data migration validation in heterogeneous databases. Legacy-DB provides post-migration validation strategies through a graphical and interactive interface, which projects attributes from the legacy and migrated database, providing a mapping mechanism that enables the adoption of strategies to mitigate inconsistencies between the data. Legacy-DB allows the user to verify and validate the migration using its attributes and the mapping between them, allowing conclusions about visual perception of the nonconformities that occurred during the migration process. In this way, the tool facilitates the recognition of inconsistencies and changes necessary for a successful migration.*

Resumo. *Este artigo apresenta a ferramenta Legacy-DB [1] [2] para apoiar a validação da migração de dados em banco de dados heterogêneos. O Legacy-DB fornece estratégias de validação pós migração através de uma interface gráfica e interativa, que projeta atributos do banco legado e migrado, fornecendo um mecanismo de mapeamento que possibilita a adoção de estratégias para mitigar inconsistências entre os dados. O Legacy-DB possibilita que o usuário verifique e valide a migração utilizando seus atributos e mapeamento entre eles, permitindo conclusões sobre uma percepção visual das inconformidades ocorridas durante o processo de migração. Desta forma, a ferramenta facilita o reconhecimento de inconsistências e alterações necessárias para uma migração bem sucedida.*

1. Introdução

Fatores como a fusão de grandes empresas, melhorias dos sistemas ou necessidade de gerar determinado volume de dados para aplicação de técnicas em Ciência de Dados, implicam na demanda por realizar migração dos dados para novos bancos de dados.

De acordo com [Heleno Ramos de Mendonça 2009], o processo de migração de dados é complexo, e deve ser iniciado por uma avaliação que tem objetivo de migrar

[1]Demonstração da ferramenta - https://youtu.be/6yFrWkbfxb4
[2]Repositório do projeto - https://github.com/eletromarlon/Projeto-GeradorDeGraficos.git

uma quantia de dados razoável para que a aplicação de destino permaneça funcional. Em um cenário que essa migração atende os requisitos, todos os dados devem ser migrados, implicando em maior complexidade de planejamento e esforço. Como solução para essa problemática surge o processo de migração separado em extração e carga de dados [Heleno Ramos de Mendonça 2009].

Existem ferramentas no mercado, onde podemos destacar a MySQL Migration Toolkit[3], SwisSQL[4] e DBConvert[5] para auxiliar o processo de migração visando torná-lo ágil, porém, mantendo a integridade das informações. Além dessas, o *Pentaho Data Integration*[6], é uma ferramenta de código aberto para extração, transformação e carga (ETL) de dados.

Apesar de existir diversas ferramentas de migração de dados, ainda existe a necessidade de verificar a correção funcional dos dados migrados, ou seja, verificar se os dados migrados foram migrados de forma consistente, correta e completa. De acordo com [Haller 2009], um método padrão para realizar essa tarefa é usar casos de teste. Isso significa verificar se todos os atributos dos objetos selecionados foram migrados corretamente e verificar se nenhum dado foi perdido durante o processo de migração. Essa é uma tarefa que deve ser automatizada devido à sua complexidade e importância no processo de migração. O foco da verificação são os atributos relevantes dos dados migrados, e como resultado, um relatório técnico de migração que aponta os principais problemas, se houver, deve ser entregue para a equipe envolvida no processo.

Com objetivo de automatizar o processo de validação da migração baseado em correção funcional, desenvolvemos o Legacy-DB, uma ferramenta que fornece suporte à validação de migração de dados entre bancos de dados heterogêneos e foi construída conforme a metodologia proposta por [Haller 2009]. O diferencial da nossa ferramenta é a realização de uma análise completa e de fácil utilização, focada no processo de pós migração.

Este artigo possui a seguinte estrutura. Na Seção 2, abordamos os trabalhos relacionados. Na Seção 3, é detalhado a ferramenta proposta (Legacy-DB), sua arquitetura, componentes e usos, juntamente com exemplos da interface gráfica. E por fim, apresentamos a conclusão e melhorias futuras da ferramenta.

2. Trabalhos Relacionados

Ao longo dos anos, pesquisadores realizam reflexões sobre o processo de migração com ênfase na pós migração de banco de dados. Nesta seção, discutiremos os principais trabalhos nesta área.

No trabalho [Rodrigues and Vieira 2019], os autores realizaram um estudo de caso em migração de dados sobre bancos de dados heterogêneos, adotando como origem um banco de dados SQLServer e como destino o Oracle. Para fornecer suporte ao processo de ETL, os autores utilizam a ferramenta *Pentaho Data Integration*. Para criar as tabelas e campos entre origem e destino, os autores utilizam uma ferramenta proprietária, que

[3]https://downloads.mysql.com/archives/migration/

[4]http://www.swissql.com/

[5]https://dbconvert.com/

[6]https://pentaho-community.atlassian.net/wiki/spaces/COM/overview?mode=global

se conecta com o banco de origem e destino para preparar as tabelas para migração. Assim, não é possível customizar os mapeamentos quando necessário, como também, não é possível mensurar através de métricas a consistência das instâncias migradas de uma tabela origem para uma destino. Como os próprios autores identificam em suas conclusões, apenas a migração dos dados não garante a integridade dos dados, ficando a cargo de testes posteriores as validações necessárias.

Em [Malacrida et al. 2014], os autores propõem uma ferramenta de suporte para migração de dados que possibilita o mapeamento entre tabelas, atributos e relacionamentos conforme os elementos da nova modelagem, de forma semi-automatizada. O mapeamento proposto pela ferramenta relaciona conceitos similares a partir de diferentes bancos por uma relação de equivalência. Os autores definem um plano de migração composto de etapas de mapeamento dos tipos de dados (tamanho e atributos obrigatórios e opcionais), equivalência de atributos e gerenciamento de conflitos (chaves primárias e estrangeiras). Contudo, a ferramenta não analisa a qualidade da migração, cujos resultados são importantes critérios de decisão.

No trabalho [Haller 2009], os autores definem duas técnicas de verificação dos dados migrados que produzem o *feedback* necessário para a equipe de migração. A primeira técnica utiliza casos de teste que selecionam uma amostra para verificação manual dos atributos. A segunda técnica, de reconciliação, é uma verificação automática, que valida se todos os objetos foram migrados. Os autores adotam uma planilha que pode apresentar estatísticas e erros de migração. Nossa abordagem explora diferentes estatísticas em uma abordagem visual que facilita a verificação de correspondência entre os atributos do banco legado e banco migrado.

Por fim, em [Patel et al. 2014], os autores apresentam um sistema de migração de dados em bancos de dados heterogêneos. A ferramenta fornece bancos de dados de origem como *MS-Access* e bancos de dados de destino como SQL Server e Oracle. Os autores disponibilizam um sistema de verificação da qualidade da migração e pós migração, que inclui: verificação de integridade; consistência; contagem de registros e integridade do sistema, como, velocidade da CPU, capacidade de memória e tempo de migração. Contudo, a ferramenta não dispõe de métricas estatísticas sobre os registros, apoiando a verificação de consistência entre atributos mapeados entre a origem e o destino.

3. Legacy-DB

A ferramenta Legacy-DB foi projetada para atuar na validação dos dados na etapa de pós migração. Ela fornece aos admistrador de banco de dados um ambiente interativo, simples e funcional, além de ser independente do Sistema de Gerênciamento de Banco de Dados (SGBD) adotado no processo de migração. A seguir, vamos descrever como foi a concepção e construção da ferramenta.

3.1. Arquitetura do Legacy-DB

Para construção do Legacy-DB usamos a metodologia de verificação de migração proposta por [Haller 2009]. Baseado na sua experiência, o autor propõe três padrões para verificação da corretude na pós migração de dados. Esses três padrões levam em consideração duas premissas básicas: análise estatística e mitigação de erros. Assim, os padrões são definidos da seguinte forma: *Top-down*, *Bottom-up equivalence* e *Bottom-up fingerprint*.

Essa metodologia foi adotada na perspectiva da nossa ferramenta, fazemos uso das abordagens *Top-down* e *Bottom-up equivalence*. A abordagem *Top-down* tem como principal estratégia utilizar métricas estatísticas, baseadas em instâncias numéricas, com objetivo de validar os dados migrados entre o banco legado e o banco migrado. Na *Bottom-up equivalence*, a ideia é definir uma comparação linha a linha usando uma chave candidata em comum entre o banco legado e o banco migrado.

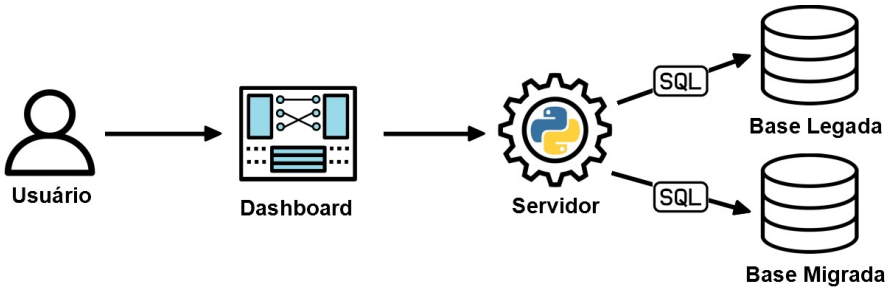

Figura 1. Arquitetura do Legacy-DB

Na Figura 1, apresentamos a arquitetura do Legacy-DB. Através de um *dashboard*, o usuário pode acessar de forma fácil e visual o banco legado, banco migrado e as estratégias de validação. Um servidor atua sobre os bancos permitindo a conexão entre elas, consultas SQL e escolha dos atributos utilizados na validação. A validação ocorre através das estratégias de validação *Top-down*, que implementa métricas estatísticas, como quantidade, média, mínimo e máximo, e *Bottom-up equivalence*, onde é possível verificar atributos nominais, linha por linha.

Durante a execução da estratégia *Top-down*, Legacy-DB reescreve o SQL informado pelo usuário como uma subconsulta na cláusula *from* e adiciona funções de agregação simples (*count()*, *max()*, *min()*, *avg()*) nas colunas selecionadas pelo usuário para ambos os bancos, legado e migrado. Essa consulta tende a ser rápida, pois é executada no próprio SGBD de origem, no qual tiramos proveito dos benefícios de gerenciamento de memória, otimização de consulta e índices das tabelas. Embora a estratégia *Top-down* traga uma visão rápida do estado da migração, nem sempre é capaz de garantir a conformidade dos dados, principalmente para tipos *string*. Por conta disso, usamos a estratégia *Bottom-up equivalence* para analisar as linhas, inicialmente executamos um procedimento de *full outer join* implementado dentro do servidor do Legacy-DB por meio de uma chave candidata em comum entre as duas tabelas comparadas. Em seguida, para cada linha correspondentes do resultado da junção é verificado se os dados de cada coluna correspondente são iguais. Caso contrário, essa linha deve ser sinalizada como incorreta. As linhas que não obtiveram correspondências, ou seja, estão presentes em apenas uma das tabelas, também devem ser sinalizadas.

3.2. Funcionamento do Legacy-DB

Para exemplificar a atuação da ferramenta, definimos um exemplo de validação desde a modelagem (Figura 2) até o uso do Legacy-DB pós migração.

Na Figura 2, a modelagem usada para gerar os *scripts* de migração utiliza uma tabela que identifica colaboradores de uma empresa. Contudo, para o banco migrado, mudanças no nome dos atributos, criação de novos atributos e de nova tabela, são fatores que podem implicar na necessidade de validação pós migração. Visualmente, é necessário mapear campos como **nome completo** em campos **primeiro nome** e **último nome**.

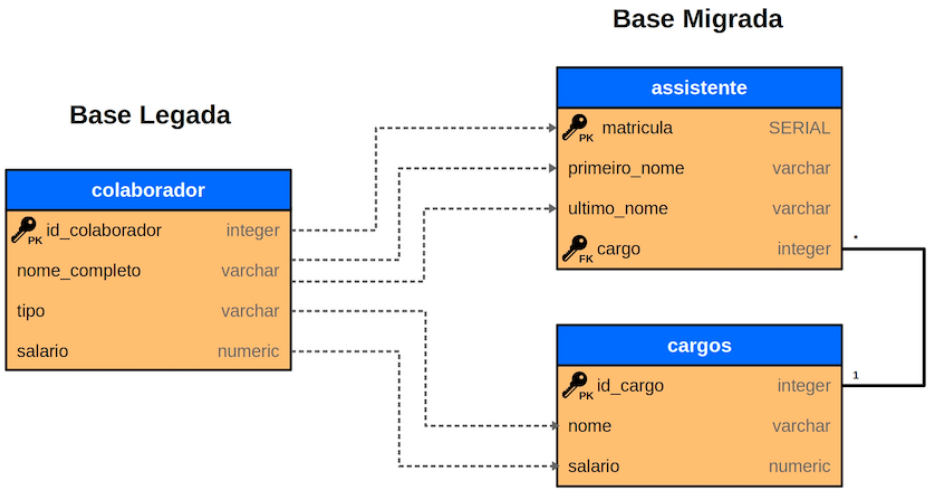

Figura 2. Modelagem exemplo Legacy-DB

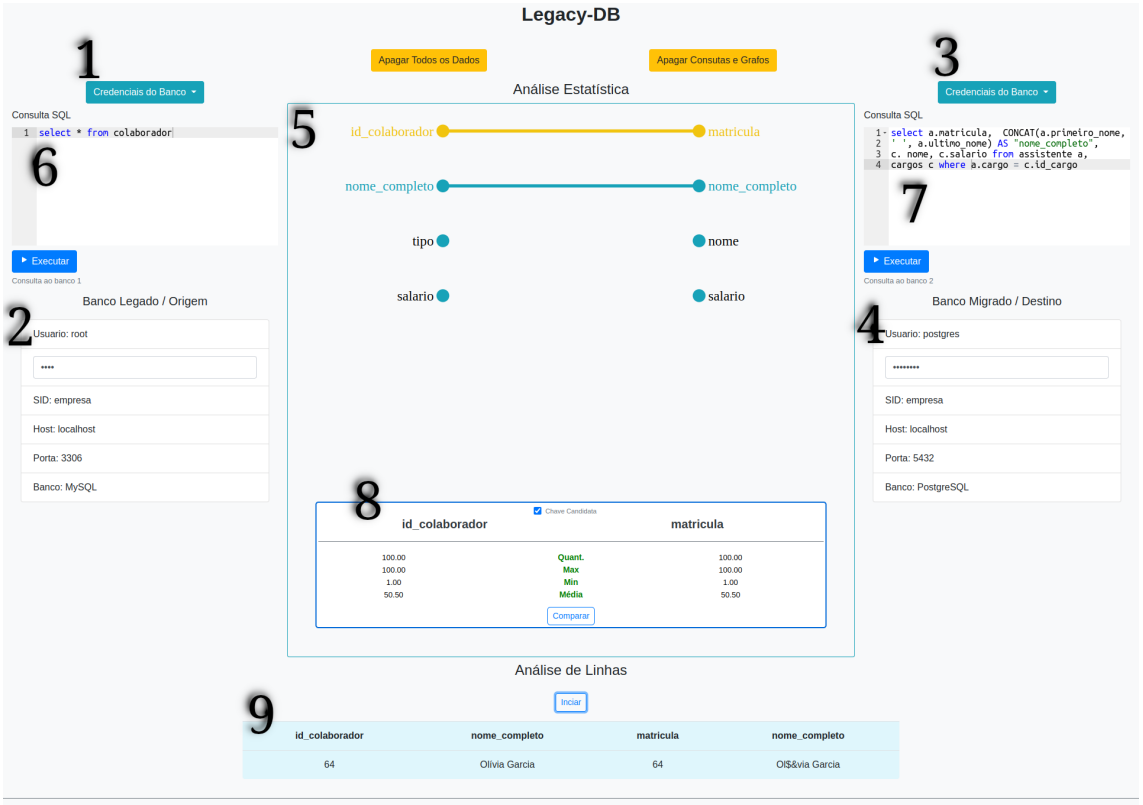

Figura 3. Interface da ferramenta e suas funcionalidades

Na figura 3 é possível visualizar a interface do Legacy-DB e suas funcionalidades. Na parte superior, 1 e 3, a ferramenta dispõe de botões como, *Credenciais do Banco*, que promovem acesso ao formulário de cadastro e conexão entre os bancos de dados legado (origem) e migrado (destino).

Após inserir as informações dos bancos, o status da conexão será exibido em 2 e 4. Consultas SQL manuais podem ser inseridas (6 e 7). Esse recurso pode ficar em destaque para facilitar o uso. Em *Análise Estatística*, 5, é possível visualizar nomes de

colunas como vértices, conforme consulta realizada pelo usuário. Os vértices deverão ser conectados de acordo com a necessidade de análise.

Ao conectar vértices e clicar na aresta, será exibida a planilha, em 8. Utilizando a estratégia *Top-down*, é possível visualizar os atributos dos bancos legado e migrado, comparando os atributos por métricas de quantidade, mínimo, máximo e média. Escolhemos uma dessas arestas como chave candidata em comum (em amarelo) necessário no procedimento seguinte de *Análise de Linhas*. Em 9, *Análise de Linhas*, utiliza-se a estratégia *Bottom-up equivalence*. Após o clique do usuário no botão iniciar, executa-se uma comparação de valores nas linhas das colunas escolhidas.

Exemplificando a *Análise de Linhas*, percebe-se uma inconformidade com o **nome completo** de uma pessoa entre o banco legado e o banco migrado. Inconformidade causada por um erro de acentuação nos nomes. Em um cenário de pós migração, esse erro é visualmente detectável, alertando o usuário para alguma falha no processo de migração em relação à palavras que contenham acentuação gráfica. Além disso, quaisquer outros erros que resultem em valores divergentes nos dois bancos podem ser analisados.

4. Conclusão

Neste artigo apresentamos a ferramenta Legacy-DB para apoiar a validação da migração de dados em bancos de dados heterogêneos. Legacy-DB apresenta estratégias para pós migração juntamente com uma interface gráfica interativa. Um dos pontos positivos do Legacy-DB é prover um ambiente visual para o mapeamento dos atributos entre o banco legado e migrado, permitindo uso de estatísticas sobre os atributos e posterior identificação de instâncias linha a linha. A ferramenta proporciona o reconhecimento das inconformidades que afetam a integridade do processo de migração. Como trabalhos futuros, pretendemos adicionar um método de detecção de *outliers* e comparar o desempenho computacional nas operações de validação usando *benchmarks*. Ainda como perspectiva de trabalhos futuros, devemos provê suporte aos diferentes bancos de dados, inclusive, não-relacionais.

Referências

Haller, K. (2009). Towards the industrialization of data migration: concepts and patterns for standard software implementation projects. In *International Conference on Advanced Information Systems Engineering*, pages 63–78. Springer.

Heleno Ramos de Mendonça, M. (2009). Metodologia de migração de dados em um contexto de migração de sistemas legados. Master's thesis, Universidade Federal de Pernambuco.

Malacrida, T. F., Zaupa, A. P., and Pazoti, M. A. (2014). Desenvolvimento de uma ferramenta para migração de dados entre bancos de dados relacionais. In *Colloquium Exactarum. ISSN: 2178-8332*, volume 6, pages 20–36.

Patel, S., Wakchaure, S., Pingale, M., and Siraj, S. (2014). Data migration system in heterogeneous database. *International journal of Research in Engineering and Technology*, 3(02):2319–1163.

Rodrigues, T. P. and Vieira, R. B. (2019). Estudo de caso de migração entre banco de dados heterogêneos utilizando pentaho. *DIVERSITÀ: Revista Multidisciplinar do Centro Universitário Cidade Verde*, 5(2):14–28.