

# Gerenciamento de Migração de Dados: Um Workflow Eficiente para Empresas (Versão Estendida)

Gustavo Moraes<sup>1</sup>, Victor Misael<sup>1</sup>, Angelo Brayner<sup>1</sup>

<sup>1</sup>Departamento de Computação – Universidade Federal do Ceará (UFC)  
Campus do Pici – Fortaleza – CE – Brasil

{gustavo.moraes, victor.misael}@alu.ufc.br, brayner@dc.ufc.br

**Abstract.** *Data migration between different database systems (DBMS) is crucial for companies to keep up with technological advancements and optimize their processes. This article proposes a workflow to assist in conducting migrations, from planning to post-migration. The objective is to provide a roadmap for migrations, reducing execution time, mitigating risks, and ensuring data integrity and security. The proposed approach was validated in a real company scenario, demonstrating its feasibility and effectiveness. This article offers a practical contribution to the data migration area, providing a validated roadmap that assists companies in conducting complex migrations efficiently, safely, and with high quality.*

**Resumo.** *A migração de dados entre diferentes sistemas de bancos de dados (SBDs) é crucial para empresas acompanharem a evolução tecnológica e otimizar seus processos. Este artigo propõe um workflow para auxiliar a condução de migrações, desde o planejamento até a pós-migração. O objetivo é fornecer um roteiro para migrações, reduzindo o tempo de execução, mitigando riscos e garantindo a integridade e segurança dos dados. A abordagem proposta foi validada em um cenário real de empresa, demonstrando sua viabilidade e efetividade. Este artigo oferece uma contribuição prática para a área de migração de dados, fornecendo um roteiro validado que auxilia empresas na condução de migrações complexas de forma eficiente, segura e com alta qualidade.*

## 1. Introdução

Diversos fatores podem levar as organizações a recorrer ao processo de migração de dados, como a atualização de tecnologias, a economia de despesas e a correção de problemas de segurança. Os benefícios de uma migração bem-sucedida são significativos, incluindo a melhoria de desempenho e da escalabilidade dos sistemas, a consolidação de dados legados e a garantia da continuidade dos negócios. No entanto, a migração de dados pode apresentar diversos desafios, requerendo uma metodologia mais sólida [Hussein et al. 2021, Thalheim and Wang 2013].

A migração de banco de dados, embora crucial, muitas vezes é subestimada, resultando em riscos significativos para a integridade e segurança dos dados. Entre os principais desafios enfrentados durante esse processo, destaca-se a perda ou comprometimento desses dados, provocada, por exemplo, pela incompatibilidade de tipos de dados. Para

ilustrar este fenômeno, pode-se citar o tipo *int unsigned* do Mysql com o *int* do PostgreSQL. No PostgreSQL, não há compatibilidade com tipos *unsigned*, exigindo, assim, intervenções como a escolha de um tipo de dado que suportasse o mesmo intervalo de valores. Claramente, tais fenômenos podem acarretar efeitos negativos do ponto de vista organizacional (bancos de dados inconsistentes) bem como legais. Além disso, problemas de desempenho são frequentemente observados, após processos de migração, afetando a eficiência operacional.

A escolha do processo de migração adequado é uma atividade fundamental para o sucesso do processo. Por exemplo, o processo de migração para a nuvem pode oferecer benefícios como escalabilidade, custos reduzidos e maior disponibilidade. No entanto, é importante avaliar os riscos e desafios antes de iniciar a migração. Problemas de segurança, compatibilidade e desempenho podem surgir durante esse processo. Além disso, existem diferentes tipos de migração a considerar: entre diferentes Sistemas Gerenciadores de Banco de Dados (SGBDs), entre SGBDs iguais porém com versões diferentes, e entre SGBDs com diferentes modelos de dados (por exemplo, de um banco de dados relacional para NoSQL) [Ellison et al. 2018] e, por fim, entre SGBDs residentes em diferentes ambientes computacionais (por exemplo, de ambiente local para uma nuvem).

Neste contexto, este artigo apresenta uma metodologia genérica para garantir um processo de migração eficiente, tanto de ponto de vista de correção, como do aspecto tempo de execução. O objetivo é fornecer uma estratégia abrangente e sistemática, que oriente as organizações durante todo o processo de migração, desde a fase de planejamento até a fase de pós-migração. O *workflow* descrito neste trabalho forneça às organizações uma base sólida para realizar qualquer tipo de migração de bancos de dados, que mitigue riscos e assegure a integridade dos dados.

A estrutura deste trabalho é organizada da seguinte forma: na Seção 2, são apresentados diversos estudos relacionados ao processo de migração de banco de dados. Na Seção 3, detalhamos o *workflow* proposto, dividido em etapas distintas, com o intuito de estabelecer uma base teórica sólida que possa servir como referência para a execução de migrações de dados. Na Seção 4, demonstramos a aplicação prática do *workflow* em um cenário empresarial real. Por fim, na Seção 5, apresentamos as conclusões do artigo.

## 2. Trabalhos relacionados

Os estudos de migração de banco de dados vêm evoluindo juntamente com o surgimento de novas tecnologias utilizadas pelas empresas. Em [Elamparithi and Anuratha 2015] é apresentada uma análise das estratégias de migração de bancos de dados relacionais entre 1987 e 2008, descrevendo as principais limitações de cada estratégia e uma vasta coleção de diferentes tipos de ferramentas de migração. Um aspecto crucial desse processo é a modelagem conceitual, como discutido em [Mayr and Thalheim 2021], que desempenha um papel fundamental na compreensão e representação de dados. A capacidade de transformar modelos conceituais é essencial para garantir a preservação da semântica e a integridade dos dados durante a migração.

Com o constante desenvolvimento dos bancos de dados NoSQL, novas perspectivas e estratégias foram desenvolvidas, permitindo que a indústria não se limite apenas a sistemas de banco de dados relacionais, mas também explore a transição para modelos NoSQL. O estudo de [Nan and Bai 2019] foca na migração de dados de bancos de

dados relacionais para bancos de dados de grafos, propondo um algoritmo que transforma modelos de Entidade-Relacionamento (ER) em modelos de grafos, utilizando regras de transformação para manter a integridade e as restrições dos dados. Similarmente, [Karnitis and Arnicans 2015] apresenta uma solução para a migração de bancos de dados relacionais para bancos de dados orientados a documentos, permitindo a conversão semiautomática dos dados, a criação de modelos de dados lógicos refinados e *templates* de migração personalizados.

Além do aspecto da migração envolvendo o tipo de banco de dados de origem (legado) ou destino com suas características específicas, é importante considerar os métodos de armazenamento. No contexto atual, muitas empresas optam por realizar uma transição para sistemas em nuvem. O estudo de [Ellison et al. 2018] aborda a migração de bancos de dados para a nuvem, destacando a importância de compreender a duração, os custos de migração e os custos operacionais futuros.

Esses estudos contribuem significativamente para o campo da migração de dados, relatando problemas específicos enfrentados devido às características dos bancos de dados de origem ou destino e ao uso de armazenamento em nuvem. Em todos esses casos, existe um padrão de etapas a serem seguidas como boas práticas. Por exemplo, o trabalho de [Wang et al. 2014] apresenta um *workflow* de migração para bancos de dados heterogêneos, que consiste em três procedimentos principais: extração dos dados do sistema legado, conversão e carregamento dos dados no destino. Durante esses procedimentos, vários fatores podem complicar o processo de migração de dados em ambientes heterogêneos, principalmente problemas relacionados aos tipos de dados. Esses três procedimentos principais se conciliam ao conceito de ETL (*Extract, Transform, Load*), sendo uma prática comum e explorada no contexto de migração de dados [Thalheim and Wang 2013].

No entanto, é importante destacar que a migração de dados vai além de um simples procedimento ETL. O artigo de [Hussein et al. 2021] demonstra que a migração de dados é um processo multifacetado que começa com a análise dos dados legados e culmina na reconciliação dos dados carregados em novas aplicações. Este processo pode ser complexo, exigindo testes para garantir a qualidade dos dados e pode ser muito custoso se as melhores práticas não forem seguidas. O artigo propõe uma metodologia de três fases: planejamento, migração e validação.

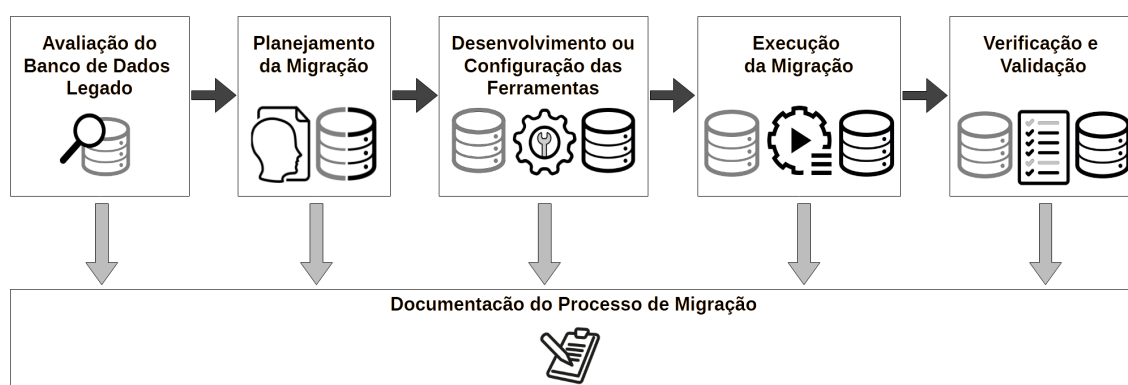
Em síntese, os estudos sobre migração de banco de dados oferecem metodologias práticas para enfrentar os desafios na transição entre sistemas, sejam eles relacionais, NoSQL ou armazenados em nuvem. Cada abordagem visa garantir a integridade dos dados e a eficiência do processo de migração. Baseando-nos em trabalhos como o de [Hussein et al. 2021] e em um caso de uso real de migração de dados na indústria, propomos, na próxima seção, uma metodologia expandida em etapas que vai além do simples processo ETL, incorporando perspectivas teóricas [Thalheim and Wang 2013] e uma ênfase na documentação do processo de migração.

### **3. DB-SMigra: Um *Workflow* para migração inteligente de bancos de dados**

Como já afirmado anteriormente, vários cenários podem levar empresas a recorrer ao processo de migração de bancos de dados. Contudo, esse processo apresenta riscos desafiadores. No intuito de minimizar riscos durante o processo de migração, descreveremos

nesta seção o DB-SMigra, um *workflow* para conduzir o processo de migração, independente do tipo de migração.

A Figura 1 apresenta o *workflow* proposto. Observe que ele é dividido em seis etapas. Cada etapa é delineada para proporcionar uma compreensão clara e uma execução eficaz do processo. Começando com a avaliação do banco de dados legado, passando pelo planejamento da migração, desenvolvimento ou configuração das ferramentas necessárias, execução da migração, até a verificação e validação final. É importante destacar que a etapa de documentação é realizada continuamente ao longo de todo o processo de migração, à medida que as atividades são desenvolvidas. Isso proporciona uma visão consolidada e contínua do *workflow*.



**Figura 1. Workflow para o Processo de Migração de Bancos de Dados.**

Além das opções de migração parcial ou completa dos dados legados e da compatibilidade dos sistemas de gerenciamento de banco de dados origem ou destino (migração heterogênea ou homogênea), é importante considerar o número de bancos de dados envolvidos no processo. Em muitos casos, apenas um banco de dados de origem e destino está presente (1:1), mas outras possibilidades incluem a consolidação (n:1), onde vários bancos de dados legados são migrados para um único, a distribuição (1:n), onde um banco de dados legado é distribuído em vários, e a redistribuição (n:m), onde vários bancos de dados legados são redistribuídos em vários destinos [Google Cloud 2024]. Compreender qual cenário de migração se aplica facilita a adaptação das etapas (ver Figura 1) para se adequarem ao número de bancos de dados envolvidos.

Nas Seções a seguir descrevemos as etapas junto com as suas atividades, perspectivas e documentação sugerida. Vale ressaltar que as atividades não necessariamente precisam ser executadas em ordem sequencial, diversas atividades podem ser feitas em paralelo deixando o processo mais dinâmico.

### 3.1. Etapa 1: Avaliação do banco de dados legado

Esta etapa visa compreender a estrutura e funcionamento do banco de dados legado. Inclui análise da modelagem, configuração física, capacidade de armazenamento, carga de trabalho, aplicações conectadas, dados históricos ou redundantes e detecção de inconsistências. Podemos assim identificar áreas de melhoria e gargalos no sistema legado.

**Análise inicial da modelagem do banco de dados legado.** Esta atividade concentra-se na análise da modelagem do banco de dados legado, visando compreender sua estrutura

e organização. É essencial realizar uma avaliação abrangente das relações diretas ou indiretas dos dados, independentemente da natureza relacional ou não do banco de dados. Ao final dessa atividade, deve ser gerado um diagrama do modelo de dados legado. Essa representação visual é fundamental para que a equipe e outros interessados compreendam rapidamente a complexidade e interconexão dos dados.

**Análise da configuração física do banco de dados legado.** Esta atividade concentra-se na análise das configurações físicas específicas do banco de dados legado, incluindo esquemas, tabelas, tipos de dados, índices, visões, gatilhos, funções e o tamanho do banco de dados em gigabytes, proporcionando uma visão abrangente do volume de dados a ser migrado. Ao final, é crucial gerar um relatório detalhado dessas configurações identificadas, oferecendo uma compreensão clara da infraestrutura atual do banco de dados e servindo como base para decisões futuras de migração.

**Análise da carga de trabalho do banco de dados legado.** Essa atividade tem como objetivo mapear quais aplicações estão conectadas ao banco de dados legado, entendendo sua finalidade e frequência de uso. Além disso, consiste em avaliar a quantidade de leituras, escritas e o nível de isolamento das transações executadas sobre o banco de dados, permitindo identificar gargalos que possam comprometer a escalabilidade futura do sistema. Ao final dessa atividade, é recomendado gerar um documento que contenha um diagrama ou o nome de cada aplicação conectada ao banco de dados, juntamente com os detalhes de acesso e suas dependências. Ademais, uma análise da vazão de transações atual do sistema pode ser valiosa, oferecendo uma compreensão abrangente do volume de trabalho e contribuindo para o planejamento de ações futuras de otimização e migração.

**Dados históricos ou redundantes.** Essa atividade visa identificar dados obsoletos, redundantes ou históricos que possam ser removidos ou armazenados em outra mídia. Essa ação não apenas pode otimizar o espaço de armazenamento no novo ambiente, mas também contribui para a eficiência geral do sistema. Ao final dessa atividade, é recomendado gerar um documento detalhando os dados considerados históricos ou redundantes, juntamente com recomendações para seu tratamento. Esse documento serve como um guia para a equipe de migração, auxiliando na tomada de decisões sobre quais dados podem ser eliminados ou arquivados.

**Deteção de inconsistências nos dados.** Esta atividade visa identificar inconsistências, erros ou falhas nos dados armazenados no banco de dados legado. A qualidade dos dados é fundamental para o sucesso de qualquer projeto de migração. Dados completos, corretos e limpos não apenas reduzem custos, complexidade e riscos, mas também fornecem uma base sólida para decisões estratégicas rápidas e eficazes. Sem uma qualidade de dados aceitável, a migração de dados pode ser inviável [Azeroual and Jha 2021]. Ao final desta atividade, é recomendado gerar um relatório detalhado destacando as inconsistências encontradas e possíveis ações corretivas. Este relatório servirá como um guia para a limpeza e correção dos dados antes da migração, garantindo que todos os dados relevantes sejam transferidos com sucesso para o sistema de destino.

### **3.2. Etapa 2: Planejamento da migração**

Nesta etapa, o foco é a elaboração de um plano de migração. Trata-se de um processo estratégico que visa definir passos específicos para a transição do banco de dados legado para um novo ambiente. Durante essa fase, são estabelecidas estratégias para a escolha do

modelo de dados, definição de otimizações, permissões de acesso, estratégias de backup e procedimentos para minimizar riscos, garantindo uma migração bem-sucedida. Além disso, é fundamental criar um canal de comunicação eficaz com as partes interessadas relevantes, como administradores de banco de dados (DBAs) e desenvolvedores, para alinhar objetivos e requisitos. Este plano servirá como um guia detalhado, proporcionando suporte em todas as etapas subsequentes do processo de migração.

**Definição da configuração física e otimizações do novo banco de dados.** Nesta atividade deve-se definir e documentar a área de armazenamento(arquivos) na qual os dados do novo banco de dados serão armazenados, com especial atenção à área de armazenamento do arquivo de log, caso o banco de dados ofereça suporte para tal. Em seguida, é recomendado realizar uma avaliação da viabilidade de utilizar funcionalidades como particionamento de tabelas ou arquivos, bem como materialização de visões. Dentro desse contexto, é importante avaliar a necessidade de implementar estratégias para otimizar o desempenho do banco de dados. A utilização de índices, particionamento e aprimoramentos nas consultas são essenciais para maximizar a eficiência do sistema. A otimização das consultas assegura respostas mais rápidas e eficazes, tendo um impacto direto na performance e produtividade das aplicações [Monteiro Filho and Brayner 2013].

**Mapeamento entre dados de origem e destino.** Elabore um documento detalhando o mapeamento entre os dados do banco legado e o novo banco de dados, incluindo os procedimentos para transferência e organização no novo sistema. É crucial considerar a criação de um novo modelo de dados otimizado, especialmente ao lidar com migração entre bancos de dados heterogêneos. Priorize também a definição da ordem de migração dos dados. Por exemplo, determine quais tabelas ou arquivos devem ser migrados primeiro. Isso pode ajudar a minimizar interrupções e tornar o novo ambiente parcialmente disponível para as aplicações.

**Escolha da abordagem de migração (única ou incremental).** Na decisão sobre a estratégia de migração mais adequada para o contexto específico, deve-se considerar se migrar tudo de uma vez ou de forma incremental. A migração única oferece rapidez, porém pode resultar em maior tempo de inatividade. Já a migração incremental proporciona controle e menos interrupção, embora possa demandar mais tempo e acarretar em riscos de inconsistências.

**Métodos ETL ou ELT para extração de dados.** Escolha entre utilizar métodos tradicionais de ETL (*Extract, Transform, Load*) ou abordagens de ELT (*Extract, Load, Transform*) para transferir os dados. Com o ETL, a extração dos dados legado tem sua transformação conforme os requisitos do novo banco de dados e a carga no novo banco são feitas de maneira estruturada e sequencial, ideal para volumes menores e transformações complexas durante a transferência. Já o ELT, ao priorizar a extração e carga inicial dos dados, seguida pela transformação no banco de dados novo, é mais adequado para volumes grandes de dados e quando se deseja explorar a capacidade de processamento direta do destino para transformações complexas após a migração. A escolha dependerá da complexidade das transformações, dos volumes de dados e da infraestrutura disponível no ambiente de destino [Singhal and Aggarwal 2022].

**Avaliação de requisitos específicos de arquitetura e funcionalidades.** Essa atividade envolve a análise detalhada das necessidades específicas do novo ambiente de banco de

dados, considerando diversos fatores técnicos e operacionais que impactam a eficiência, escalabilidade e robustez do sistema.

Um fator crítico é a definição de prioridades entre o desempenho do banco de dados e a disponibilidade dos dados. Em alguns casos, a performance pode ser mais importante, exigindo otimizações que acelerem as operações de leitura e escrita. Em outros, a alta disponibilidade pode ser prioritária, necessitando de estratégias robustas de *failover* e recuperação de falhas. Nesse contexto, a análise da necessidade de implementar técnicas de clusterização ou replicação se torna essencial. A clusterização distribui a carga entre vários servidores, aumentando a capacidade de processamento e melhorando a escalabilidade do sistema, enquanto a replicação garante a alta disponibilidade dos dados, mantendo cópias em diferentes locais para evitar perda de informação em caso de falhas. Identificar onde a ênfase deve ser colocada ajudará a determinar a configuração ideal do sistema e as tecnologias a serem adotadas [Fuaad et al. 2018].

Diversos outros requisitos específicos podem surgir durante o planejamento da migração. Entre eles estão a implementação de técnicas de busca por similaridade e/ou fonéticas, o uso de estratégias ou bancos de dados de cache para melhorar o desempenho, técnicas de gerenciamento de dados geográficos, e métodos de criptografia para proteção de dados confidenciais.

Também é importante considerar requisitos como segurança dos dados, conformidade com regulamentos, integração com sistemas existentes e suporte a transações distribuídas se necessário. Avaliar todos esses fatores é essencial para garantir uma migração bem-sucedida e que o novo ambiente de banco de dados atenda a todas as necessidades operacionais e de negócios. Por fim, é fundamental incluir em um relatório detalhado que explique a necessidade de cada requisito e descreva, de forma incremental, como será sua implementação.

**Estabelecimento de metas de desempenho para o novo banco de dados.** O objetivo desta atividade é estabelecer métricas e objetivos de desempenho para assegurar que o novo banco de dados atenda às expectativas dos usuários. Defina parâmetros mensuráveis, como tempo de resposta, vazão do sistema, disponibilidade do banco de dados, escalabilidade, integridade dos dados e segurança. No final desta atividade, elabore um documento com recomendações sobre as metas de desempenho o novo ambiente.

**Permissões de acesso ao novo bancos de dados.** Para aprimorar a segurança do banco de dados, é fundamental estabelecer permissões e níveis de acesso para cada usuário. Se o novo ambiente oferecer suporte, defina quais partes do banco de dados eles podem interagir, incluindo quais operações (leitura ou escrita) podem ser executadas. Se for o caso, baseie-se nos usuários e restrições de acesso do banco de dados legado, caso ele também ofereça essa funcionalidade.

**Backup e plano de reversão em caso de erros.** Elaborar estratégias de backup e planos de contingência é essencial para garantir a segurança e resiliência do processo de migração. Essas medidas asseguram a prontidão para lidar com possíveis falhas ou erros críticos, minimizando impactos adversos e permitindo a reversão segura da migração, mantendo a integridade dos dados e a continuidade dos sistemas [Magalhaes et al. 2021]. Anexe um relatório detalhando quais serão as estratégias de backup de dados caso seja necessário e qual o plano de contingência para reverter a migração em caso de falhas críticas,

pensando principalmente em sistemas que exigem alta disponibilidade e um tempo baixo do processo de migração.

**Avaliação do custo e impacto da migração.** Uma migração, tanto para uma infraestrutura local quanto para a nuvem, envolve a análise de custos financeiros e recursos. Elabore um relatório com uma análise dos custos associados à migração. Avalie a disponibilidade das partes interessadas e faça uma avaliação do impacto da migração. Por exemplo, considere se haverá mudanças nas aplicações que acessam o banco de dados legado e, em caso de falhas ou demora no processo, o quanto o tempo de inatividade é prejudicial aos negócios. Em caso de uma migração para a nuvem, é importante considerar o preço em diferentes plataformas e quais serviços elas oferecem.

**Planos de testes.** Diversas metodologias podem ser utilizadas para verificar a corretude da migração. Uma delas é a verificação técnica para assegurar que as informações essenciais do banco de dados legado estão disponíveis no novo ambiente [Haller 2009]. Todavia, nem sempre os dados migrados estão organizados da mesma forma que no banco de dados legado. Mesmo com as devidas alterações nas aplicações para suportarem a migração, testes nessas aplicações também se tornam um requisito. A ideia é elaborar um plano de testes com cenários detalhados em cada aplicação para validar os dados migrados. Esse plano deve servir de base para orientar os testes realizados na Etapa 5 (ver Seção 3.5).

### **3.3. Etapa 3: Desenvolvimento ou configuração das ferramentas de migração**

Essa etapa consiste no desenvolvimento ou configuração das ferramentas essenciais para realizar a migração. Defina as tecnologias utilizadas, escolha as ferramentas de migração, estabeleça testes de validação e realize o monitoramento de erros e métricas para assegurar que o processo de migração seja capaz de lidar com diferentes cenários.

**Configuração ou desenvolvimento da ferramenta de migração.** Com base nas avaliações das etapas anteriores, é o momento de configurar uma ferramenta existente ou desenvolver uma nova. A ferramenta de migração escolhida deve atender a diversos requisitos, como a migração entre bancos de dados heterogêneos, migração paralela, migração incremental, reengenharia de dados, testabilidade e boa usabilidade com uma baixa curva de aprendizado [Neto et al. 2013]. Se o problema envolve um banco de dados legado simples ou não há uma ferramenta adequada para atender a todos os requisitos, pode ser necessário implementar uma ferramenta própria. Embora isso ofereça vantagens em termos de autonomia, também implica em custos de implementação. Outra opção é configurar uma ferramenta já existente que atenda aos requisitos ou dividir o processo de migração em fases, dependendo das escolhas anteriores, como as abordagens e mapeamentos (ver Seção 3.2). Ou seja, podemos migrar partes dos dados usando uma ferramenta existente e migrar as partes mais sensíveis por meio de uma ferramenta própria, garantindo o devido controle e requisitos. É importante também estabelecer uma documentação mínima com os detalhes de configuração ou implementação da(s) ferramenta(s).

**Desenvolvimento de rotinas de testes para verificação e validação.** Esta atividade consiste na implementação ou configuração do plano de testes criado na atividade da Seção 3.2. Da mesma forma que na seção anterior de configuração ou desenvolvimento da ferramenta, pode ser necessário implementar uma ferramenta de testes ou utilizar outras ferramentas já existentes.

**Monitoramento de erros e métricas de desempenho.** O objetivo desta atividade é con-



figurar e assegurar que as ferramentas de migração sejam capazes de monitorar e registrar possíveis erros, métricas de desempenho (discutidas na Seção 3.2) e outras informações relevantes. Este monitoramento é importante porque permite a identificação de problemas e a avaliação da eficácia do resultado da migração.

**Simulação de migração para avaliar a eficácia da ferramenta.** Uma das formas de avaliar a eficácia das ferramentas e dos testes desenvolvidos ou configurados é a realização de simulações, que permitem a identificação de possíveis áreas de melhoria e a resolução de problemas antes da migração real dos dados, além de ajudar a avaliar a capacidade das ferramentas em lidar com grandes volumes de dados. Durante essa atividade, é crucial criar cenários de migração que reproduzam as condições reais. Isso possibilita uma análise detalhada do desempenho das ferramentas, permitindo uma melhor compreensão de sua eficácia e identificando oportunidades de otimização.

### 3.4. Etapa 4: Execução da migração

Esta etapa visa realizar a execução controlada do plano de migração. Garantimos permissões e segurança adequadas, verificamos a capacidade do sistema de destino, isolamos o ambiente de migração e monitoramos em tempo real o processo de migração para identificar e resolver possíveis problemas.

**Permissões de acesso e segurança necessárias.** Esta atividade tem como objetivo garantir que todas as permissões de acesso e medidas de segurança estejam configuradas de maneira adequada tanto no sistema de banco de dados legado quanto no sistema de destino, preparando assim o ambiente para o início do processo físico de migração. Para tanto, é interessante elaborar uma *checklist* abrangente que inclua aspectos como permissões de acesso, políticas de senhas, criptografia, entre outros protocolos de segurança.

**Verificação da capacidade do sistema de destino.** Esta atividade visa garantir que o sistema de destino disponha dos recursos computacionais necessários para suportar o processo de migração. É fundamental assegurar que o sistema tenha capacidade de armazenamento suficiente, poder de processamento adequado e conectividade de rede estável.

**Execução controlada da migração com monitoramento em tempo real.** Esta atividade implica na realização da migração dos dados em si, em fases planejadas, que incluem a extração, transformação e armazenamento dos dados legados utilizando as estratégias escolhidas nas etapas e atividades anteriores. Durante todo o processo, é essencial monitorar em tempo real para identificar possíveis problemas ou erros. Este monitoramento contínuo permite uma intervenção imediata em casos de erros.

### 3.5. Etapa 5: Verificação e validação

Esta etapa é dedicada à verificação e validação dos dados migrados. Serão realizados testes de integração com as aplicações que acessam o novo ambiente, simulações de carga de trabalho e comparações detalhadas entre os dados migrados e os dados originais. Esses processos garantem a precisão, integridade e funcionalidade dos dados no novo ambiente, assegurando que estejam prontos e operacionais para uso contínuo.

**Comparação entre dados migrados e dados legados.** Nesta atividade, busca-se documentar uma análise comparativa entre os dados migrados e os dados originais, visando garantir a precisão e integridade da migração. As metodologias de verificação da correção

da migração, propostas por [Haller 2009], incluem a comparação dos dados usando medidas estatísticas, como verificar se o número de tuplas de uma tabela ou instâncias legadas é igual no novo ambiente. Outra opção, mais detalhada porém mais demorada, é a comparação linha a linha ou instância a instância, dependendo do modelo de dados.

**Simulações e testes de integração com outros sistemas.** Esta atividade visa verificar o desempenho por meio da execução e documentação dos resultados de simulações (conforme projetado na atividade da Seção 3.2) que representem a carga de trabalho das aplicações esperada no novo ambiente. Além disso, é recomendável executar e documentar testes para verificar a integração dos dados migrados com outros sistemas ou aplicações, garantindo o funcionamento adequado em conjunto.

**Documentação do resultado da migração.** Após a conclusão da migração, é importante documentar os resultados obtidos. Isso inclui os registros de *log* gerado pelas ferramentas de migração, bem como os resultados alcançados nos testes realizados após a migração. Esta documentação abrange a identificação de discrepâncias encontradas e as soluções aplicadas, garantindo assim a rastreabilidade e a transparência do processo.

#### 4. Instanciação do DB-SMigra em uma migração real

Nesta Seção, apresentamos a uso bem-sucedido do *workflow* de migração em uma empresa real. A migração envolveu a transição de um banco de dados legado MySQL para um PostgreSQL com replicação (um mestre e vários escravos com transparência total) [Ozsu and Valduriez 2001], além de incluir o gerenciamento de dados espaciais. Devido à natureza confidencial dos dados, omitiremos alguns detalhes específicos. O banco de dados legado possuía 120 tabelas e cerca de 460 índices, contendo aproximadamente 30 milhões de linhas com 5,60 GB. A maior parte do volume de dados (cerca de 85%) estava concentrada em uma única tabela, predominantemente utilizada para operações de inserção. Após uma limpeza pré-migração, não foram detectadas inconsistências nos dados. Duas tabelas foram identificadas como obsoletas e excluídas do processo.

O novo ambiente de armazenamento foi configurado em um servidor local dedicado. A migração entre os dois modelos de dados foi facilitada pelo fato de ambos serem banco de dados relacionais, não houve a necessidade de alterações drásticas no modelo de dados ou fragmentações de tabelas. Apenas algumas transformações por conta dos tipos de dados foram necessárias. A migração foi dividida em duas abordagens: uma migração única para a maioria das tabelas e uma migração incremental para a tabela com maior volume de dados.

A prioridade era garantir a alta disponibilidade do sistema. Para isso, foi usado uma solução do Patroni<sup>1</sup>, que gerencia um banco de dados primário com duas réplicas. Em caso de falha do banco de dados primário, uma das réplicas pode assumir o controle. Também havia requisitos específicos para armazenamento de dados geográficos, que foram atendidos com a extensão PostGIS<sup>2</sup>.

Apesar da disponibilidade de várias ferramentas de migração de MySQL para PostgreSQL, optamos por desenvolver uma ferramenta própria, para automatizar várias das atividades da metodologia de migração proposta. Destacam-se as atividades relacio-

---

<sup>1</sup><https://patroni.readthedocs.io/en/latest/>

<sup>2</sup><https://postgis.net/>

nadas à detecção de inconsistências, mapeamento entre origem e destino, monitoramento de erros, reversão em caso de falhas, transferência dos dados e verificações de integridade. Embora tenha exigido mais tempo, essa escolha proporcionou maior controle do código, facilitando a implementação de conversores de tipos de dados, especialmente para dados geográficos. O código da ferramenta, juntamente com diversos experimentos de desempenho, está disponível em um repositório online<sup>3</sup>.

A ferramenta desenvolvida é capaz de realizar a leitura e conversão das definições das tabelas legadas para o novo ambiente. Em seguida a migração das tuplas de cada tabela, chaves primárias, regras de integridade, chaves estrangeiras, índices e sequências. A ferramenta também possui um sistema de *log* que registrava todo o processo, além de um sistema de tolerância a falhas, capaz de reverter as últimas transações (*rollback*) e continuar a partir de um ponto estável em caso de falhas. Durante a execução, primeiro passo é migrar todas as tabelas com exceção da tabela com maior volume, que deve ser migrada de forma incremental até que o sistema legado pudesse ser desligado. Diversas simulações de migração com dados de teste foram realizadas para garantir a eficácia do processo. O tempo de migração total do banco de dados legado pela ferramenta foi de aproximadamente 45 minutos. Em seguida, as aplicações também foram atualizadas para operarem com o novo banco de dados. Já o processo como um todo, desde do planejamento até a pós-migração seguindo o *workflow* proposto foi de aproximadamente de 35 dias úteis.

Um dos pontos-chave para o sucesso da migração foi a abrangência e o nível de detalhes do *workflow*, aliados ao forte incentivo à documentação de todo o processo. Isso permitiu que todas as partes interessadas no projeto contribuíssem de maneira mais eficiente, garantindo a transparência necessária para uma operação crítica e minimizando os riscos aos negócios da empresa.

## 5. Conclusão

Este trabalho apresentou o DB-SMigra, um *workflow* genérico para a migração de bancos de dados, dividido em etapas com boas práticas para garantir um processo eficiente e seguro. A migração de dados é um processo complexo que exige planejamento meticuloso e execução rigorosa. O DB-SMigra destaca a importância de cada etapa do processo, desde a análise inicial dos dados legados até a validação final no novo ambiente. Através da instanciamento do DB-SMigra foi possível demonstrar que as diversas estratégias adaptativas para diferentes cenários e volumes de dados sugeridas no *workflow*, juntamente com a documentação, conduziram a bons resultados durante a migração de um banco de dados MySQL para PostgreSQL.

Por fim, o sucesso de uma migração de dados depende do planejamento adequado, do uso de ferramentas apropriadas e do monitoramento contínuo para identificar e resolver possíveis problemas de forma ágil. A documentação das etapas e dos resultados da migração proporciona maior transparência e rastreabilidade do processo, permitindo que futuras migrações na empresa tirem proveito das lições aprendidas. A experiência adquirida e os métodos aplicados neste estudo servem como um guia para profissionais da área, reforçando a importância de uma abordagem estruturada.

---

<sup>3</sup><https://github.com/VicMisael/IDataMig>

## Referências

- Azeroual, O. and Jha, M. (2021). Without data quality, there is no data migration. *Big Data and Cognitive Computing*, 5(2):24.
- Elamparithi, M. and Anuratha, V. (2015). A review on database migration strategies, techniques and tools. *World Journal of Computer Application and Technology*, 3(3):41–48.
- Ellison, M., Calinescu, R., and Paige, R. F. (2018). Evaluating cloud database migration options using workload models. *Journal of Cloud Computing*, 7:1–18.
- Fuaad, H., Ibrahim, A., Majed, A., and Asem, A. (2018). A survey on distributed database fragmentation allocation and replication algorithms. *Current Journal of Applied Science and Technology*, 27(2):1–12.
- Google Cloud (2024). Migração de banco de dados: conceitos e princípios (parte 1). <https://cloud.google.com/architecture/database-migration-concepts-principles-part-1?hl=pt-br>. Acessado em: 2024-05-29.
- Haller, K. (2009). Towards the industrialization of data migration: concepts and patterns for standard software implementation projects. In *Advanced Information Systems Engineering: 21st International Conference, CAiSE 2009, Amsterdam, The Netherlands, June 8-12, 2009. Proceedings 21*, pages 63–78. Springer.
- Hussein, A. A. et al. (2021). Data migration need, strategy, challenges, methodology, categories, risks, uses with cloud computing, and improvements in its using with cloud using suggested proposed model (dmig 1). *Journal of Information Security*, 12(01):79.
- 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.
- Magalhaes, A., Monteiro, J. M., and Brayner, A. (2021). Main memory database recovery: A survey. *ACM Comput. Surv.*, 54(2).
- Mayr, H. C. and Thalheim, B. (2021). The triptych of conceptual modeling: A framework for a better understanding of conceptual modeling. *Software and Systems Modeling*, 20(1):7–24.
- Monteiro Filho, J. M. S. and Brayner, A. R. A. (2013). Sintonia e auto-sintonia de bancos de dados. In Medeiros, C. B., editor, *Atualizações em Informática*, volume 1, pages 141–206. Ed. Edufal, Maceió.
- Nan, Z. and Bai, X. (2019). The study on data migration from relational database to graph database. In *Journal of Physics: Conference Series*, volume 1345, page 022061. IOP Publishing.
- Neto, P. S., Neto, J. R., Júnior, F. R., and Oliveira, P. (2013). Requisitos para ferramentas de migração de dados. In *Anais do IX Simpósio Brasileiro de Sistemas de Informação*, pages 887–898, Porto Alegre, RS, Brasil. SBC.
- Ozsu, M. T. and Valduriez, P. (2001). *Principios de sistemas de bancos de dados distribuídos*. Campus.

- Singhal, B. and Aggarwal, A. (2022). Etl, elt and reverse etl: a business case study. In *2022 Second International Conference on Advanced Technologies in Intelligent Control, Environment, Computing & Communication Engineering (ICATIECE)*, pages 1–4. IEEE.
- Thalheim, B. and Wang, Q. (2013). Data migration: A theoretical perspective. *Data & Knowledge Engineering*, 87:260–278.
- Wang, G., Jia, Z., and Xue, M. (2014). Data migration model and algorithm between heterogeneous databases based on web service. *Journal of Networks*, 9(11):3127.