CENTRO UNIVERSITÁRIO DO PARÁ - CESUPA
ESCOLA DE NEGÓCIOS, TECNOLOGIA E INOVAÇÃO - ARGO
CURSO DE ENGENHARIA DA COMPUTAÇÃO

ARTHUR FERREIRA BESSA
PEDRO BEZERRA DOS SANTOS

COMPARAÇÃO DE PRÁTICAS E FERRAMENTAS DE INTEGRAÇÃO DE DADOS UTILIZANDO O BANCO DE DADOS AMOSTRAL ADVENTUREWORKS

BELÉM
2023

ARTHUR FERREIRA BESSA
PEDRO BEZERRA DOS SANTOS

COMPARAÇÃO DE PRÁTICAS E FERRAMENTAS DE INTEGRAÇÃO DE DADOS UTILIZANDO O BANCO DE DADOS AMOSTRAL ADVENTUREWORKS

Trabalho de conclusão de curso apresentado à Escola de Negócios, Tecnologia e Inovação do Centro Universitário do Estado do Pará como requisito para obtenção do título de Engenheiro da Computação na modalidade ARTIGO.

Orientador(a): Dr. Isaac Souza Elgrably

BELÉM
2023

ARTHUR FERREIRA BESSA
PEDRO BEZERRA DOS SANTOS

COMPARAÇÃO DE PRÁTICAS E FERRAMENTAS DE INTEGRAÇÃO DE DADOS UTILIZANDO O BANCO DE DADOS AMOSTRAL ADVENTURE WORKS

Trabalho de conclusão de curso apresentado à Escola de Negócios, Tecnologia e Inovação do Centro Universitário do Estado do Pará como requisito para obtenção do título de Engenheiro da Computação na modalidade ARTIGO.

Data da aprovação: 15/ 12 /2023
Nota final aluno(a) I: 9,0
Nota final aluno(a) II: 8,4

Banca examinadora

gov.br Documento assinado digitalmente
ISAAC SOUZA ELGRABLY
Data: 15/12/2023 09:57:57-0300
Verifique em https://validar.iti.gov.br

Prof. Isaac Souza Elgrably
Orientador(a) e Presidente da banca

gov.br Documento assinado digitalmente
PEDRO HENRIQUE SALES GIROTTO
Data: 15/12/2023 10:00:00-0300
Verifique em https://validar.iti.gov.br

Prof. Pedro Henrique Sales Girotto
Examinador interno

gov.br Documento assinado digitalmente
FABIO ROCHA DE ARAUJO
Data: 15/12/2023 10:07:56-0300
Verifique em https://validar.iti.gov.br

Prof. Fábio Rocha de Araújo
Examinador interno

Dados Internacionais de Catalogação-na-publicação (CIP)
Biblioteca do CESUPA, Belém – PA

Bessa, Arthur Ferreira.

Comparação de práticas e ferramentas de integração de dados utilizando o banco de dados amostral adventureworks / Arthur Ferreira Bessa, Pedro Bezerra dos Santos; orientador Isaac Souza Elgrably. — 2023.

Trabalho de Conclusão de Curso (Bacharelado em Engenharia de Computação) – Centro Universitário do Estado do Pará, Belém, 2023.

1. Integração de dados. 2. Banco de dados. I. Santos, Pedro bezerra dos. II. Elgrably, Isaac Souza, orient. III. Título.

CDD 23ª ed. 005.7

RESUMO

Na "Era da Informação", empresas lidam com grandes volumes de dados. A sobrecarga de informações dificulta a tomada de decisão. A transformação digital é crucial, envolvendo a integração de dados por meio de data pipelines. Esses processos convertem dados brutos em informações úteis para análises de negócios. Este estudo visa comparar a performance de diferentes tipos de pipelines, a ETL e a ELT, analisando o tempo de transformação dos dados para garantir disponibilidade e eficiência na tomada de decisão. A comparação entre ELT e ETL revela a importância da aplicação específica de cada método. No estudo com AdventureWorks, o ETL mostrou maior desempenho ao transformar dados antes do carregamento, em contraste com o ELT. A diferença entre uso de código e ferramentas como o *Pentaho* destaca um trade-off entre desempenho e acessibilidade. A escolha entre ELT e ETL deve considerar as necessidades do projeto, enquanto estudos futuros podem ampliar a compreensão desses processos.

Palavras-chave: Integração de dados; *pipeline*; transformação de dados; *schema*.

ABSTRACT/RESUMEN/RÉSUMÉ

In the “Information Era”, companies deal with vast volumes of data. Information overload hampers decision-making. Digital transformation is pivotal, involving data integration through data pipelines. These processes convert raw data into actionable insights for business analysis. This study aims to compare the performance of different pipelines, ETL and ELT, analyzing data transformation time to ensure availability and decision-making efficiency. The comparison between ELT and ETL shows the significance of each method's specific application. In this study using AdventureWorks, ETL outperformed by transforming data before loading, unlike ELT. The difference between code usage and tools like Pentaho highlights a trade-off between performance and accessibility. Choosing between ELT and ETL should account for project needs, while future studies may deepen the understanding of these processes.

Palavras-chave: Data integration; pipeline; data transformation; schema.

SUMÁRIO

1 CONTEXTUALIZAÇÃO

1.1 Revisão Bibliográfica

A integração de dados é o processo de obter acesso e entrega consistentes para todos os tipos de dados em uma empresa. Todos os departamentos de uma organização coletam grandes volumes de dados com estruturas, formatos e funções variados. Ela inclui técnicas de arquitetura, ferramentas e práticas que unificam esses dados díspares para análise. Assim, as organizações podem visualizar totalmente os dados para informações e business intelligence de alto valor. Além disso, o processo é especialmente importante à medida que uma empresa busca estratégias de transformação digital, já que sua capacidade de melhorar as operações, aumentar a satisfação do cliente e competir em um mundo cada vez mais digital exige a visualização de todos os seus dados.

Existem diversas práticas que aceleram esses processos, além de ferramentas que facilitam os mesmos, permitindo que os dados sejam reunidos e gerenciados de uma ampla variedade de sistemas de origem sem a necessidade de uma capacitação técnica para tal (HAIDER, 2020).

1.1.1 – Práticas

1.1.1.1 – ETL (Extrair, Transformar e Carregar)

ETL é um processo de três etapas que envolve a extração de dados de vários sistemas de origem, transformando-os em um formato útil e carregando-os para banco de dados de destino para inteligência de negócios e relatórios, permitindo que os dados fluem para um sistema unificado, resultando em tomadas de decisão mais informadas.

Ter os dados certos, coletá-los e armazená-los de maneira segura e organizada é crucial para obter informações oportunas baseadas em dados, à medida que as empresas adotam a transformação digital (NAEEM, 2020). Assim, os processos ETL são uma das práticas essenciais para organizações que visam o futuro.

A prática ETL é dividida em três etapas:

- **Extração:** É a parte inicial do processo que trata principalmente da obtenção dos dados, esses podendo ser oriundo, por exemplo, de relatórios de gestão de relacionamento com clientes, sistemas de arquivos, bancos de dados, etc. Mais de

80% desses dados não são estruturados (NAEEM, 2020), o que é um desafio para organizações que usam sistemas legado, pois esses tipos de dados são mais difíceis de se processar e analisar. No entanto, soluções ETL mais modernas permitem extrair dados estruturados, semi-estruturados e não estruturados de múltiplas fontes sem grandes complicações.

- **Transformação:** Os dados normalmente são extraídos de diferentes fontes, que não costumam ser padronizados e sofrem frequentemente com problemas de qualidade, o que prejudica o desempenho da infraestrutura dos bancos de dados (NAEEM, 2020) . Assim, acaba por haver a necessidade com que exista alguma etapa transformação de dados, que costuma envolver a limpeza, a padronização e a validação dos dados, o que normalmente melhora sua qualidade. Esta etapa garante que os dados consolidados tendem a ser precisos e completos.
- **Carga:** É a última etapa da prática ETL, onde os dados transformados são carregados em um banco de dados destino ou em um *data warehouse*, um repositório central onde informações são armazenadas, onde os dados fluem de forma regular. O carregamento pode ser feito de forma "bruta", conhecida como *full load*, onde os dados são inteiramente carregados no destino (NAEEM, 2020). Em contrapartida, há o chamado *incremental load*, em que os conjuntos de dados do destino são atualizados com os dados novos de forma controlada, minimizando recursos computacionais e o tempo necessário para carregar os dados (NAEEM, 2020).

1.1.1.2 – ELT (Extrair, Carregar e Transformar)

ELT é uma prática mais recente para integração de dados, que vem ganhando popularidade pelas suas vantagens na escalabilidade e aumento de performance (HAIDER, 2020). O ELT é similar ao ETL, com a diferença de que a etapa de transformação de dados ocorre depois da etapa de carregamento no sistema destino. Isso implica que os dados são carregados diretamente em sua forma bruta e nativa, sem qualquer tipo de tratamento. Uma vez carregados no sistema alvo, os dados são transformados e processados usando o poder computacional do sistema destino, que geralmente são *Data Warehouses* ou algum tipo de armazenamento em nuvem. Por conta disso, o processo ELT é geralmente mais rápido que o ETL, uma vez que a etapa de transformação é feita em um servidor separado, antes de carregar os dados no sistema destino (GARG, 2023).

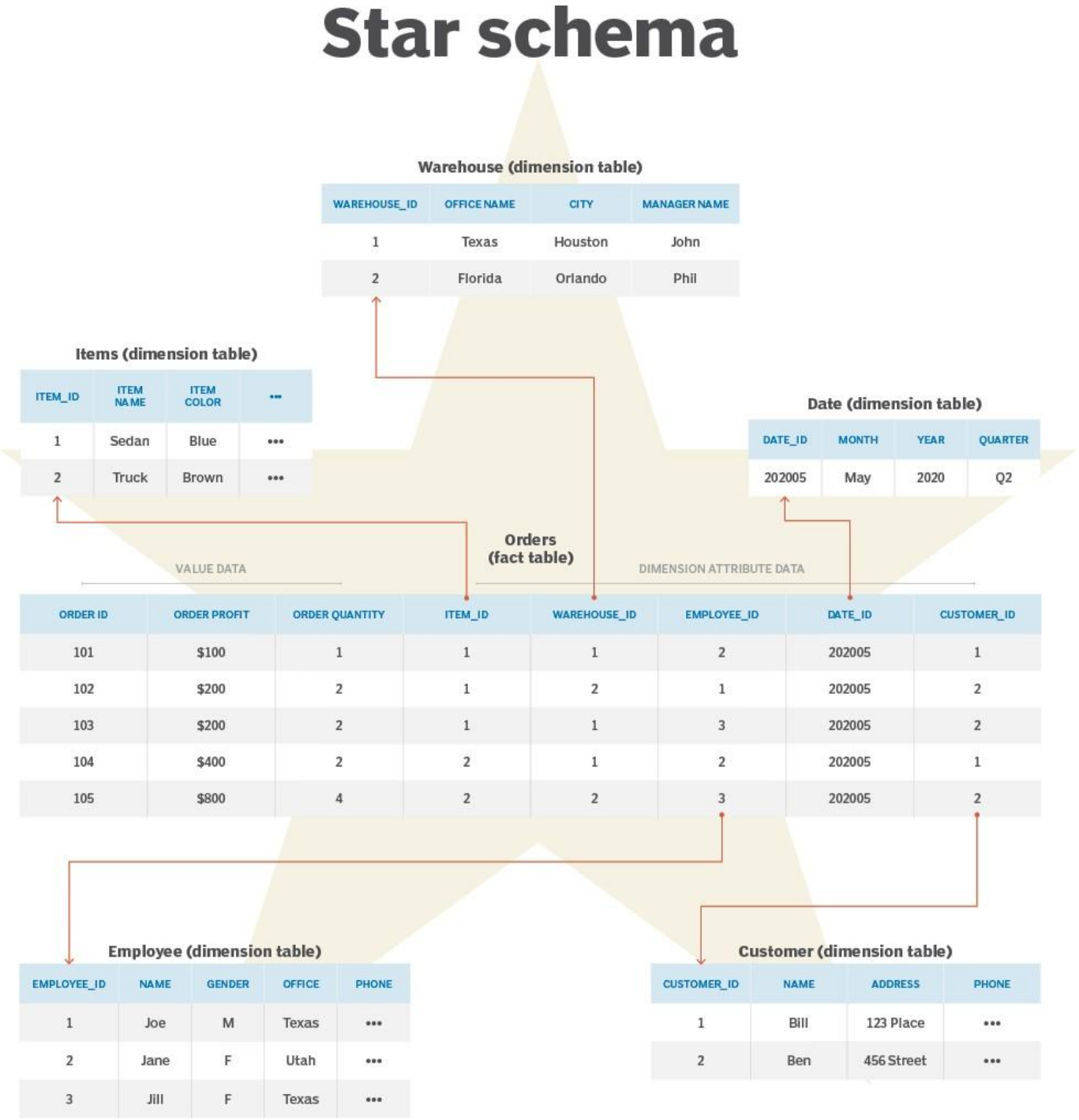
O ELT é mais usado ao se tratar de grandes quantidades de dados que precisam ser processados de forma mais rápida e eficiente, além disso ele também garante mais flexibilidade no banco de dados, de acordo com as necessidades do modelo de negócios (GARG, 2023).

1.1.2 – Star Schema

Um *schema* é um modelo lógico de dados que representa todo um banco de dados (ZOLA, 2021). São geralmente representados em forma de diagramas e são tipicamente usados para reunir fatores importantes para uma organização e analisar de que forma eles se relacionam. Após serem aprovadas, esses modelos se tornam a base de um modelo físico de dados e de um banco de dados.

Nesse contexto, o *Star Schema* (Modelo em Estrela) é o estilo mais simples de *schema* de banco de dados, e é o estilo mais utilizado ao desenvolver *data warehouses* (DEDIĆ; STANIER, 2016). É um *schema* que se assemelha a uma estrela, e consiste em uma ou mais tabelas-fatos relacionadas a inúmeras tabelas-dimensão. A figura abaixo mostra a disposição visual da tabela-fatos e como ela se relaciona com as tabelas-dimensão representadas em um *star schema*.

Figura 1. Exemplo de *star schema*.



Fonte: TechTarget.

Como pode-se observar, a tabela-fatos é a tabela central de um *star schema* de um banco de dados. Ela reúne informações quantitativas para análise e é normalmente desnormalizada, ou seja, são adicionados dados redundantes, normalmente uma cópia para cada informação, para

melhorar a performance de leitura do banco (WRIGHT; VAUGHAN, 2021). A tabela-fatos se relaciona às tabelas-dimensão, que armazena os atributos que descrevem os fatos da tabela-fatos. Os fatos são um conjunto de eventos que são armazenados na tabela-fatos e cuja as informações são descritas como dimensões. Essas informações descritivas nas dimensões permitem filtrar e categorizar os fatos e suas medidas, de forma a extrair informações importantes para questões empresariais (SHELDON, 2023). Um exemplo de banco de dados que utiliza o *star schema* é o banco de dados amostral utilizado para este trabalho, o *AdventureWorksDW*, desenvolvido pela *Microsoft*.

1.1.3 – Data Warehouse

Data Warehouse é um sistema de gerenciamento de dados que guarda dados estruturados, ou seja, informação que foi transformada e formatada em um modelo de dados definido. Esse modelo de dados minimiza redundância, o que significa que os dados estruturados são interdependentes e pouco flexíveis (NAEEM, 2020). Um *Data Warehouse* usa uma arquitetura de múltiplas camadas, onde a camada mais baixa engloba um banco de dados que armazena todos os dados, a camada intermediária contém as ferramentas para realizar o processo analítico desses dados, e a camada superior é o *front end*, que inclui os relatórios e análises completas feitas a partir dos dados armazenados.

As vantagens de se utilizar *Data Warehousing* incluem a facilidade de uso, pois os usuários podem analisar os dados e tomar decisões estratégicas a partir dessa análise sem a necessidade de um conhecimento técnico. Além disso, a organização dos dados garante uma alta consistência, precisão e qualidade dos dados armazenados (ASTERA, 2023).

1.2 Problema da Pesquisa

O processo de integração de dados envolve a combinação de dados oriundos de diferentes fontes, de forma que o usuário possa vê-los de forma unificada (LENZERINI, 2002). Esse processo é de muita importância em diversos casos. É recorrente empresas utilizarem diversos sistemas diferentes, que geram dados que não “conversam” entre si. Nos últimos 20 anos, uma empresa contábil sediada em Porto Alegre incorporou várias inovações tecnológicas. No entanto, mesmo com essas atualizações, ainda enfrentava a lacuna de um processo de integração para unificar os dados provenientes de seus diversos sistemas.

(BICCA, 2020). No entanto, quando se escolhe algum processo de integração de dados de forma equivocada, perde-se muito em performance, disponibilidade, tempo de execução e outros parâmetros computacionais. Portanto, faz-se importante realizar uma análise comparativa para verificar as práticas mais otimizadas para processos de integração de dados, para que se possa melhorar esses parâmetros.

1.3 Justificativa

Integração de dados é de grande importância para empresas que trabalham com coleta de dados para fazer análises de mercado. Um dado é uma informação bruta, que inicialmente não tem valor de uso, mas guarda um potencial para isso. Já a informação é o resultado do processamento dos dados, que ganham sentido e valor, dentro de um contexto e de acordo com algum objetivo. O processo de conversão de dados não tratados para informação, chamado de transformação digital, é interessante para essas empresas, pois permitem automatizar tarefas e embasar decisões, que são atividades essenciais na era digital, cada vez mais ágil e competitiva. (CASAROTTO, 2021), e para cada contexto é preciso reavaliar o melhor processo para reduzir custos, melhorar desempenho, diminuir tempo e auxiliar na tomada de decisão com o intuito de beneficiar o cliente.

1.4 Objetivos

1.4.1 – Objetivo Geral: Realizar uma análise comparativa entre processos de integração de dados e analisar os resultados para verificar qual foi o mais eficiente para o contexto aplicado.

1.4.2 – Objetivos específicos:

- Apresentar cada um dos processos demonstrando suas vantagens e desvantagens em cada um dos contextos.
- Analisar casos específicos e avaliar qual seria o melhor processo levando em consideração diferentes tipos de dados.

1.5 Estrutura do Trabalho

O trabalho se divide em três capítulos. O primeiro é a contextualização, que contém a revisão bibliográfica, o problema de pesquisa, a justificativa do trabalho e os objetivos. No segundo capítulo, abordou-se a introdução do tema do trabalho, a metodologia com a qual o trabalho foi feito, é apresentado os resultados obtidos e os mesmos são discutidos. Após isso, é descrito as conclusões que foram tiradas ao longo do trabalho. O terceiro capítulo envolve as referências bibliográficas utilizadas ao longo do trabalho.

2. COMPARAÇÃO DE PRÁTICAS E FERRAMENTAS DE INTEGRAÇÃO DE DADOS UTILIZANDO O BANCO DE DADOS AMOSTRAL ADVENTURE WORKS.

2.1 Introdução

Hodiernamente vivemos na "Era da Informação", onde grandes volumes de dados são produzidos pela maioria das empresas e órgãos públicos de forma cada vez mais crescente (MCAFEE, 2012) e, por conta disso, mais e mais empresas têm feito substanciais investimentos para descobrir formas de usar seus dados para beneficiar seus negócios (KALLINIKOS, 2015). No entanto, essa quantidade massiva de dados desorganizados faz com que os *data consumers* dessas empresas, ou seja, as entidades que utilizam esses dados para algum fim, enfrentem um problema conhecido como sobrecarga de informações, o que dificulta o processo de tomada de decisão (WANG *et al*, 2020).

Assim, o processo de coleta de dados e de transformação digital, que consiste na conversão de dados brutos em informações utilizáveis em análises de negócios é de suma importância para qualquer empresa situada nesse contexto. Para a integração de dados, é necessário extrair e carregar grandes quantidades de dados a partir de fontes massivas. Exemplos de arquiteturas de armazenamento utilizadas para esses dados distribuídos são as *data warehouses* e os *data lakes*, arquiteturas importantes para as grandes empresas, pois permitem garantir a consistência dos dados, o que facilita seu uso para a tomada de decisões por parte dos executivos dessas empresas, e permitem também a capacidade de armazenar enormes quantidades de dados por um baixo custo.

Existem diversas práticas de implementação para essas operações de integração de dados e transformação digital, um processo conhecido como *data pipeline.* Esse processo é um conjunto de etapas que visam levar dados brutos de um ponto de origem para que cheguem a um destino, tratados e prontos para serem utilizados, virando informação (DEARMER, 2020), e é bastante relevante para empresas que trabalham com diversas ferramentas diferentes, gerando dados fragmentados que não "conversam" entre si. Assim, *data pipelines*, ao consolidar dados de diversas fontes diferentes em um destino comum, permite uma rápida análise de dados para um bom entendimento de negócios, além de garantir a qualidade consistente desses dados (DEARMER, 2020).

Neste trabalho, será feito uma análise comparativa de performance para os diferentes tipos de pipelines, tendo como objetivo principal analisar e comparar os processos envolvidos na coleta, transformação e análise de dados em diferentes contextos, para que não haja prejuízos nos âmbitos de performance e disponibilidade dos dados sendo tratados, o que é afetado pelo tempo de execução desses processos, ou seja, quanto tempo levaria até os dados brutos passarem pela transformação digital e se tornarem informação utilizável em análises computacionais.

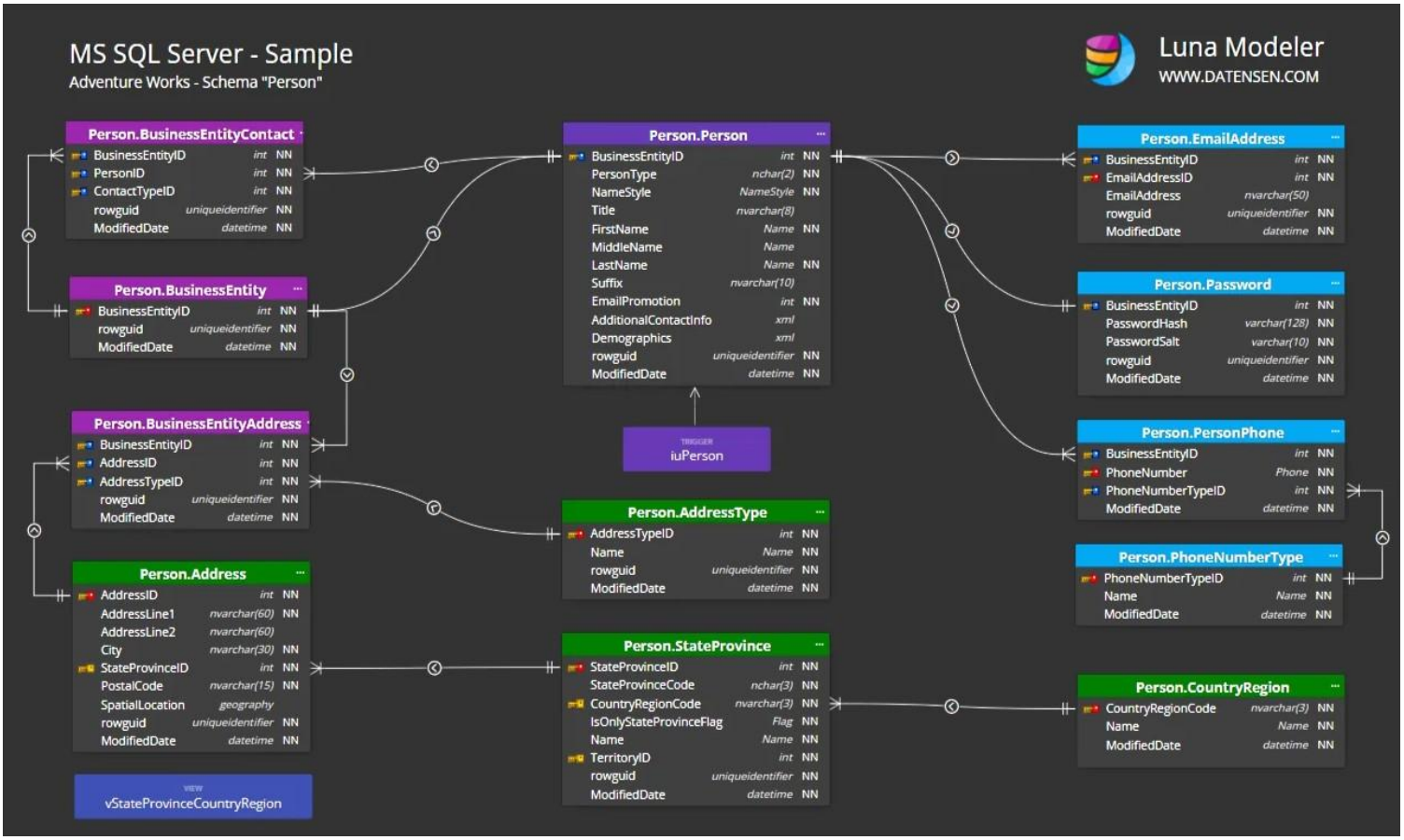
2.2 Metodologia

Em um primeiro momento, foi planejado de que forma seria organizado o procedimento comparativo e foi definido o cenário base, que seria a utilização do *AdventureWorks* como base de dados. Além disso, foram definidos os diferentes tipos de cenários montados para os processos de integração de dados.

O *AdventureWorks* é um conjunto de banco de dados amostral que simula dados de uma empresa de venda de bicicletas, desenvolvido pela *Microsoft*, ou seja, são bancos feitos especificamente para armazenar e tratar de dados de exemplo, com o intuito de estudar e testar diferentes funcionalidades de um SGBD (Sistema de Gerenciamento de Banco de Dados). Ele foi projetado para ser usado em ambientes de treinamento e aprendizado para que estudantes possam desenvolver suas competências de manipulação de dados livremente.

O *AdventureWorks* agrega dois bancos de dados, o *AdventureWorks*, que é o banco OLTP (Processamento de Transações Online) amostral, e o *AdventureWorksDW*, que é o *data warehouse* amostral do conjunto. Um banco OLTP é um banco que trata especificamente de dados de transações online. Trata-se de um tipo de processamento de dados que envolve a ocorrência de transações simultaneamente, sejam essas transações oriundas de bancos digitais, compras, etc. Abaixo segue o modelo ER de um dos *schemas* presentes no AdventureWorks, onde é possível visualizar de que forma as tabelas se relacionam no banco.

Figura 2. Modelo ER de um dos *schemas* do AdventureWorks.



Fonte: Datensen.

Essas transações são normalmente chamadas de transações financeiras ou econômicas, e esses dados precisam estar guardados e seguros para que a empresa possa acessá-los a qualquer momento, seja para a produção de relatórios ou para contabilidade. Um sistema OLTP se destaca pelo processamento rápido de muitas operações simples, como inserção e exclusão de dados de transações, garantindo acesso multiusuário para preservar a integridade dos dados e evitar conflitos. A ordem precisa das transações é essencial para manter a consistência dos dados. Além disso, a disponibilidade contínua é crucial: esses sistemas devem estar acessíveis a qualquer hora, todos os dias. Qualquer perda de dados ou inatividade pode ter consequências significativas para a empresa que depende desse sistema.

Foram utilizados os computadores pessoais dos membros do grupo com as seguintes especificações:

Quadro 1. Máquinas utilizadas e especificações.

Máquinas	**Especificações**
Máquina 1	Processador *AMD Ryzen 5 5600* 8GB de Memória RAM
Máquina 2	Processador *AMD Ryzen 5 5600G* 32GB de Memória RAM

Fonte: Autores (2023).

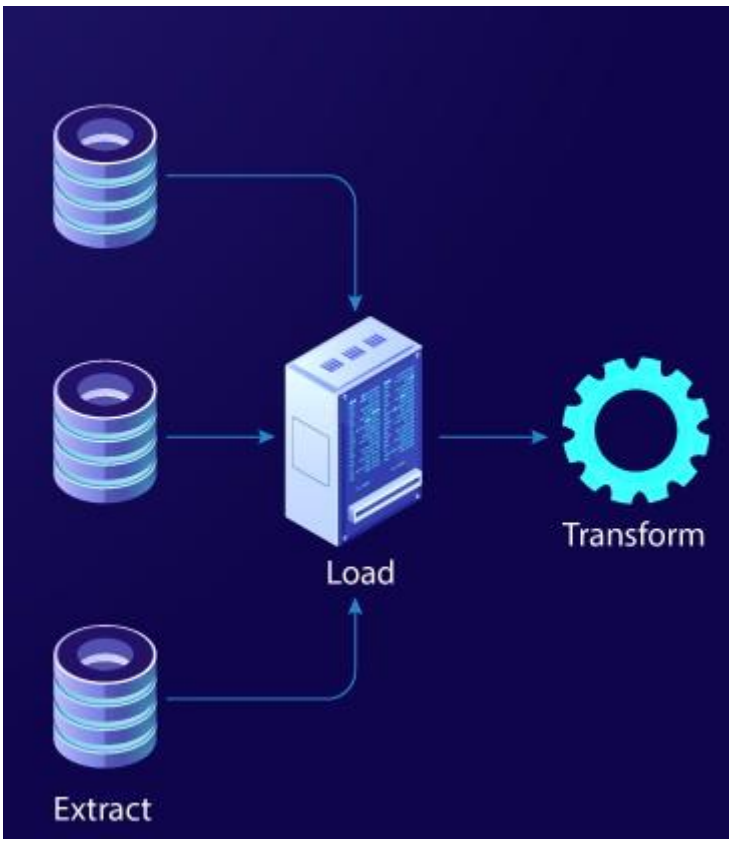
Essas máquinas foram usadas para executar as instâncias dos bancos de dados, definindo o banco de dados origem como sendo o *AdventureWorks2022*, onde foi possível encontrar um banco com dados OLTP, que são oriundos de cargas de trabalho típicas de processamento de transações online. Além disso, também foi utilizado a modelagem dimensional do *AdventureWorksDW* como base para a aplicação de uma engenharia reversa.

Esse processo foi feito com o objetivo de identificar as tabelas necessárias do banco de dados *AdventureWorks* na versão OLTP para recriar essas dimensões da versão DW desse banco. O processo foi iniciado examinando detalhadamente os dados presentes nas tabelas de dimensões do *AdventureWorksDW*. Esta análise envolveu entender a estrutura de dados, os tipos de dados armazenados e as relações entre eles. Com base nos insights obtidos, foram identificadas as tabelas correspondentes no banco de dados *AdventureWorks* na versão OLTP. A chave para essa etapa foi a compreensão da nomenclatura utilizada nas dimensões e nos campos, permitindo correlacionar esses elementos às tabelas OLTP equivalentes. Através deste processo, foi possível estabelecer um mapeamento entre as dimensões do DW e as tabelas do OLTP. Com isso, foram criados os scripts SQL de extração para as tabelas base e os scripts de transformação desses dados das tabelas base para as tabelas dimensão.

Para a realização dos processos de integração, foi estabelecida uma conexão entre as máquinas que foram usadas como servidores de bancos de dados por meio de um script de pipeline na linguagem de programação *Python*. O script estabelece uma conexão ao SGBD *SQL Server*, o qual também foi desenvolvido pela *Microsoft*, e acessa o banco de dados *AdventureWorks2022*. Com esse acesso, foi realizado o que está descrito nos dois primeiros cenários abaixo.

2.2.1 - Cenário 1 (ELT - Código)

Figura 3. Processo de ELT.



Fonte: SoftwebSolutions.

Neste cenário, ao realizar o processo de engenharia reversa mencionado anteriormente, foi possível desenvolver um *script* de *pipeline* para realizar o processo ELT (Extrair, Carregar e Transformar). Foi utilizado um banco *PostgreSQL*, devido à familiaridade de um dos membros da equipe com o mesmo e por ser uma ferramenta *open source*, como o banco destino para o processo acima. A execução do *script* se sucedeu da seguinte forma: uma conexão é estabelecida entre a Máquina 2 com o banco *SQL Server* sendo executado na Máquina 1, e então é feita uma varredura que retorna o *schema*, uma estrutura que descreve a forma dos dados e de que maneira eles se relacionam com outras tabelas (ZOLA, 2021), e as tabelas presentes no banco.

Após isso, é feito uma *query* que utiliza esses parâmetros (o *schema* e as tabelas) para criar um dataframe para cada tabela consultada. Após a criação do dataframe, o script estabelece uma conexão com um banco de dados *PostgreSQL* que está sendo executado na Máquina 2. Nesse banco de dados, é criada uma tabela que replica os dados existentes no *SQL Server*.

Esse procedimento é repetido para todas as tabelas do banco de dados *AdventureWorks*, com exceção das tabelas relacionadas ao sistema do banco, como *logs* e diagramas.

Após a conclusão da extração e carregamento dos dados, é iniciado o processo de transformação desses dados por meio de um *script Python*. Esse *script* utiliza os comandos *SQL* que foram gerados a partir da análise reversa do banco de dados *AdventureWorksDW*. Esses comandos *SQL* são usados para carregar os dados nas tabelas de dimensão do data warehouse hospedado no *PostgreSQL*. O *data warehouse* no *PostgreSQL* segue a mesma modelagem de tabelas utilizada no *AdventureWorksDW*. Segue abaixo o pseudo-código de todo esse processo.

Configurar Conexões com Bancos de Dados
Conexão com *SQL Server*: Definir *string* de conexão com
    driver,
    host,
    banco de dados,
    usuário
    e senha.

Conexão com *PostgreSQL*: Definir parâmetros de conexão incluindo
    host,
    banco de dados,
    usuário
    e senha.

Definir uma lista contendo caminhos para vários *scripts SQL*, esses arquivos *SQL* são os responsáveis por fazer a extração das tabelas bases.

Executar os comandos *SQL* para extração no banco *SQL Server*.
Inserir os resultados no banco *PostgresSQL*.

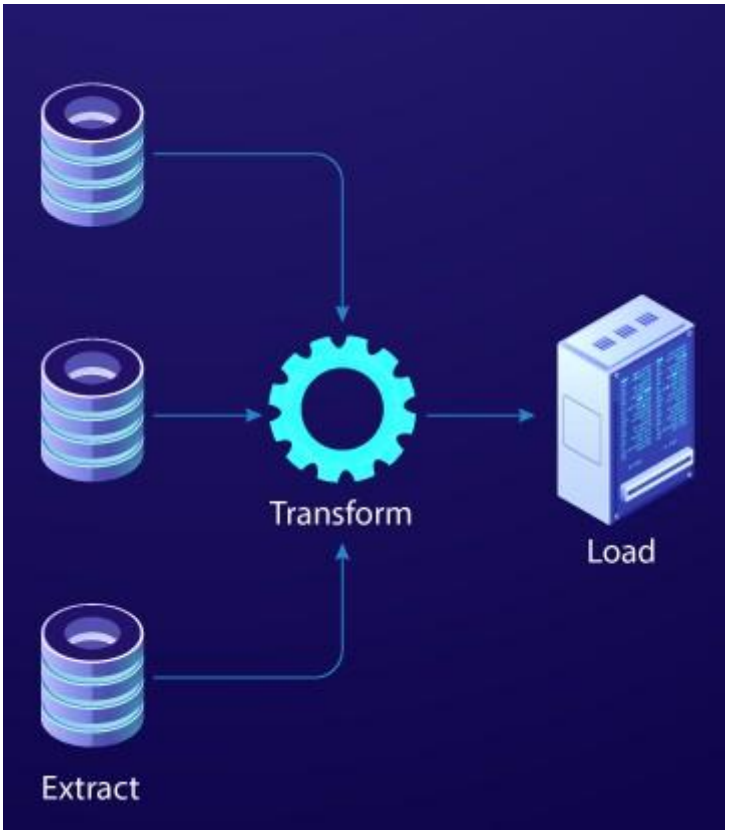
Conectar ao banco de dados *PostgreSQL*
Definir uma lista contendo caminhos para vários *scripts SQL* (esses arquivos *SQL* são os responsáveis por fazer a transformação dos dados para as dimensões.)

Executar *script SQL*.
Buscar resultados e inseri-los nas tabelas dimensões do *PostgreSQL*

Esse processo de extração, carga e transformação dos dados visa garantir que os dados estejam disponíveis para análises e consultas no *PostgreSQL* de forma consistente com o modelo de dados original do *AdventureWorksDW*.

2.2.2 - Cenário 2 (ETL - Código)

Figura 4. Processo de ETL.



Fonte: SoftwebSolutions.

Neste cenário o *script* foi modificado para realizar o processo de ETL (Extrair, Transformar e Carregar). O *script* funciona relativamente da mesma forma que no processo anterior. Uma conexão é estabelecida entre a Máquina 2 e o *SQL Server* executado na Máquina 1. A diferença entre os processos está na realização da *query*, que além de consultar as tabelas do *SQL Server* para a criação dos *dataframes*, ela também vai realizar o processo de transformação dos dados, antes dos mesmos serem carregados no *PostgreSQL*. Após esse processo, são criados os *dataframes* oriundos dos

dados transformados. É estabelecida uma conexão com o *PostgreSQL* e então é feita a carga para as tabelas-dimensões que têm a mesma modelagem de tabelas do *AdventureWorksDW*. Abaixo segue o pseudo-código deste processo.

Configurar Conexões com Bancos de Dados
Conexão com *SQL Server*: Definir *string* de conexão com
driver,
host,
banco de dados,
usuário
e senha.

Conexão com *PostgreSQL*: Definir parâmetros de conexão incluindo
host,
banco de dados,
usuário
e senha.

Definir uma lista contendo caminhos para vários *scripts SQL* (esses arquivos *SQL* são os responsáveis por fazer a extração e transformação dos dados das tabelas bases para as tabelas dimensões.)

Executar os comandos *SQL* para extração no banco *SQL Server*.

Inserir os resultados no banco *PostgresSQL*.

2.2.3 - Cenário 3 - ETL (*Pentaho*)

Para os cenários seguintes, foi considerado o uso de alguma ferramenta desenvolvida para realizar esses processos de transformação de dados. A ferramenta que foi utilizada para este trabalho foi o *Pentaho*, devido à familiaridade de um dos membros da equipe com a ferramenta. O *Pentaho* é um software de *business intelligence* que possui diversas funções, tais como mineração de dados, produção de

dashboards e ferramentas de integração de dados, que foi a função essencial para seu uso neste trabalho.

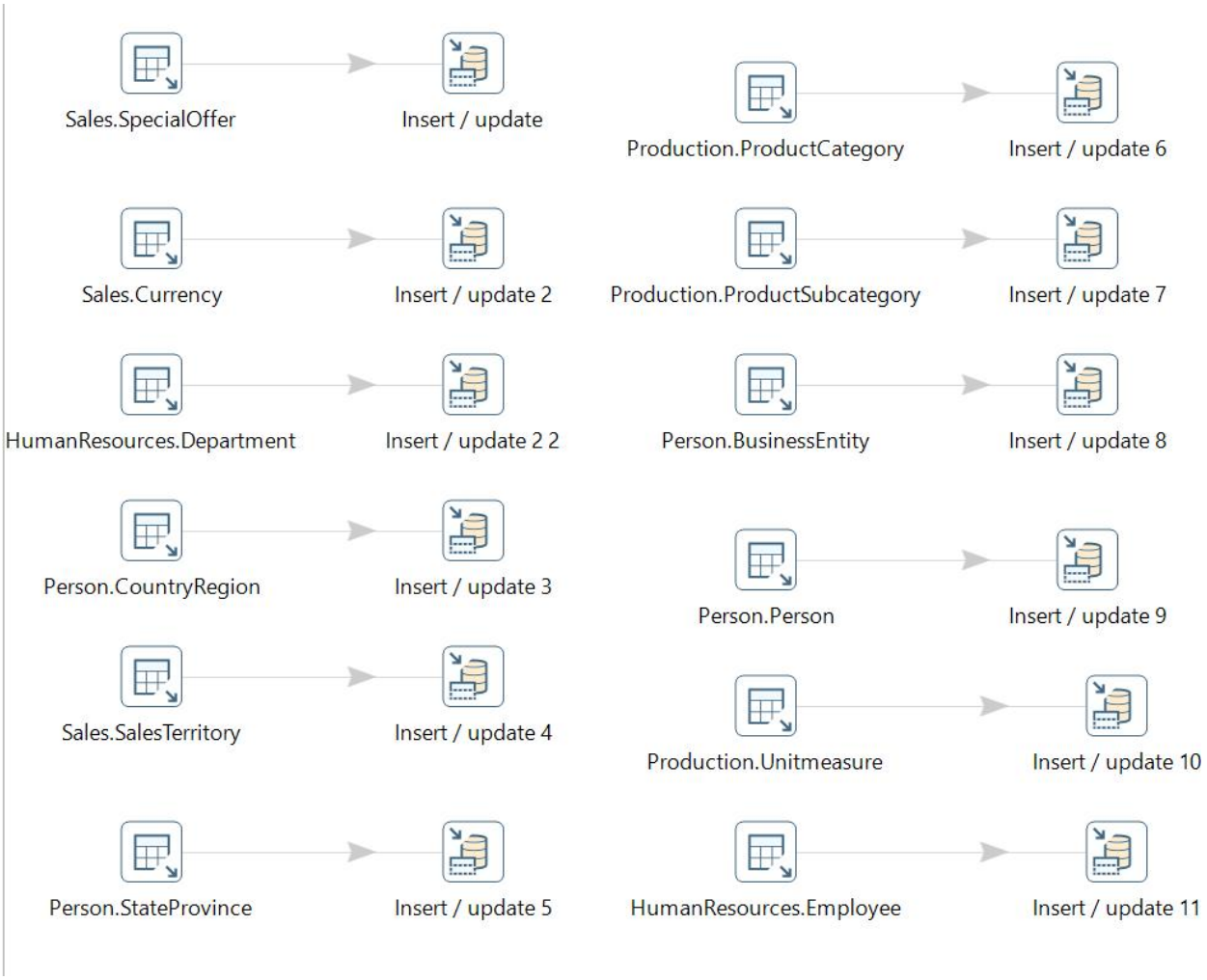
Para realizar o processo ETL, foi necessário recriar as tabelas-dimensão do *SQL Server* no *PostgreSQL*. Em seguida, utilizou-se o *Pentaho* para executar uma etapa de extração e transformação (ET) dos dados. Durante essa fase, uma consulta foi realizada para selecionar as tabelas do *SQL Server* e transformar os dados nelas contidos.

Posteriormente, o *Pentaho* foi novamente empregado para executar outra função que inseriu os dados já transformados para as tabelas-dimensões criadas no *PostgreSQL*. Essas tabelas-dimensões são semelhantes às existentes no *SQL Server*, concluindo assim o processo de carga no destino *PostgreSQL* após as etapas de extração e transformação.

2.2.4 - Cenário 4 - ELT (*Pentaho*)

Para o processo ELT, a metodologia foi um pouco diferente. Para extrair e carregar os dados do *SQL Server* para o *PostgreSQL*, foi necessário, em um primeiro momento, criar as mesmas tabelas e suas correspondentes colunas do *SQL Server* no banco *PostgreSQL*. Após isso, o *Pentaho* foi utilizado para criar os processos de integração de dados. Com ele, foi possível criar um query que seleciona a tabela e suas colunas presentes no *SQL Server* e uma função de inserção, que insere o conteúdo das mesmas nas tabelas correspondentes do *PostgreSQL*. Esse processo, que engloba as etapas de extração e carga do ELT, foi repetido em 35 tabelas, que foram as tabelas necessárias para compor as tabelas-dimensões do *AdventureWorksDW* no *SQL Server*.

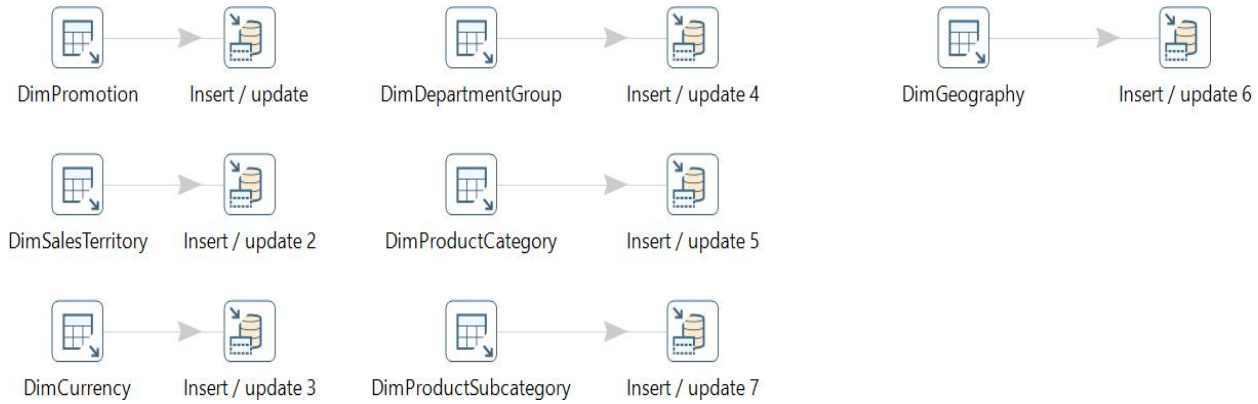
Figura 5. Tabelas extraídas do *SQL Server* para o *PostgreSQL*.



Fonte: Autores (2023).

Com as etapas de extração e carga concluídas, resta a etapa de transformação. Para isso, foi necessário recriar as tabelas-dimensões do *SQL Server* no *PostgreSQL* e então os scripts de cargas de dimensões obtidos através da engenharia reversa mencionada anteriormente foram utilizados para fazer a carga dessas tabelas, selecionando elas e suas colunas e transformando os dados contidos nelas para então inseri-los nas tabelas-dimensões criadas.

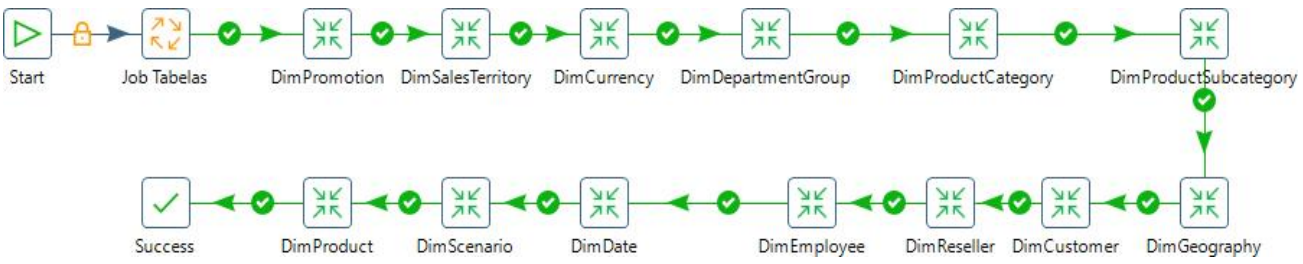
Figura 6. Tabelas sendo carregadas nas tabelas-dimensões.



Fonte: Autores (2023)

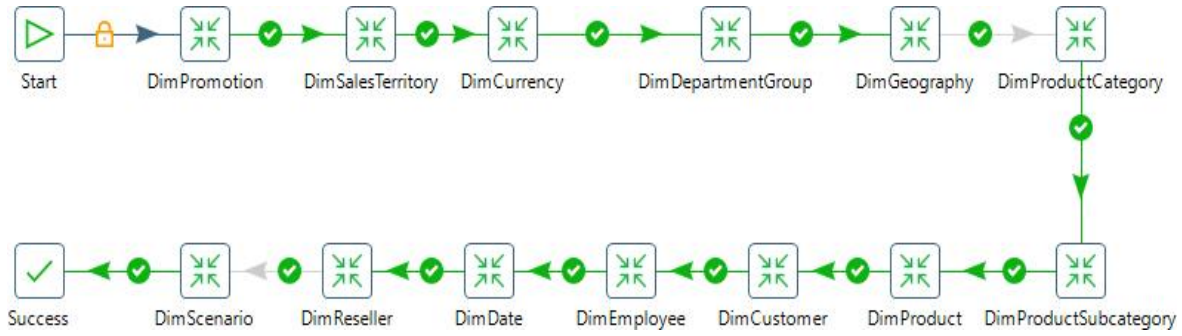
Cada um dos cenários foi submetido a execução e monitoramento meticulosos. Esses procedimentos foram repetidos um total de 10 vezes para garantir consistência nos resultados e permitir uma análise comparativa entre os cenários. Durante cada execução, foi registrado o tempo necessário para a conclusão de cada processo, visando obter uma visão abrangente e detalhada das diferenças de desempenho entre eles.

Figura 7. Pipeline do ELT no *Pentaho.*



Fonte: Autores (2023).

Figura 8. Pipeline do ETL no *Pentaho.*



Fonte: Autores (2023).

2.3 Resultados

Após a conclusão da montagem dos cenários descritos na seção anterior, realizou-se uma análise meticulosa para comparar e avaliar os resultados provenientes dos distintos processos de integração de dados que foram investigados.

Essa fase de análise envolveu a compilação e organização dos dados coletados durante as 10 repetições de cada cenário. Cada execução foi minuciosamente registrada, detalhando o tempo necessário para a conclusão de cada processo de integração de dados. O registro do tempo de cada cenário está descrito nos quadros abaixo.

Quadro 2. Tempo de execução do ETL em código.

Tempo ETL Tabelas
0:00:25:300
0:00:25:200
0:00:25:500
0:00:25:200
0:00:25:100
0:00:25:100
0:00:25:100
0:00:25:500
0:00:24:900
0:00:25:200

Fonte: Autores (2023).

Com esses resultados, foi possível calcular a média do tempo de execução desse cenário, que foi de 25,2 segundos. Em seguida, o quadro com os resultados do cenário do processo ELT em código *Python*.

Quadro 3. Tempo de execução do ELT em código.

Tempo ELT Tabelas-base Código	Tempo ELT Dimensões Código	Tempo ELT Total Código
0:01:10	0:00:19	0:01:29
0:01:10	0:00:19	0:01:29
0:01:14	0:00:19	0:01:33
0:01:12	0:00:19	0:01:31
0:01:14	0:00:19	0:01:33
0:01:13	0:00:19	0:01:32
0:01:14	0:00:19	0:01:33
0:01:12	0:00:19	0:01:31
0:01:16	0:00:19	0:01:35
0:01:12	0:00:58	0:02:10

Fonte: Autores (2023).

Neste quadro, foi registrado o tempo que levou para o código extrair as tabelas-base do banco *SQL Server* de origem e carregá-las no banco *PostgreSQL* destino. Em seguida, o tempo de transformação dessas tabelas para as tabelas-dimensão. A média do tempo de extrações e cargas foi de 1 minuto e 13 segundos, enquanto que a média do tempo das transformações de cada teste foi de 19 segundos, com uma discrepância de tempo no último teste, que durou quase 1 minuto. A média do tempo total de cada teste foi de 1 minuto e 33 segundos. Em seguida, o quadro com os resultados dos testes do processo ETL fazendo uso da ferramenta *Pentaho*.

Quadro 4. Tempo de execução do ETL no *Pentaho*.

Tempo ETL *Pentaho*
0:00:28
0:00:53
0:00:28
0:00:28
0:00:54
0:00:29
0:00:28
0:00:28
0:00:28
0:00:28

Fonte: Autores (2023).

Neste quadro, o tempo médio encontrado para os resultados do processo de ETL utilizando o *Pentaho* foi de 28 segundos, sendo possível observar uma discrepância em um dos testes, onde o tempo durou 54 segundos. A seguir, o quadro do último cenário avaliado, onde o processo ELT foi feito também na ferramenta.

Quadro 5. Tempo de execução do ELT no *Pentaho*.

Tempo ELT Tabelas-Base	Tempo ELT Tabelas-Dimensões	Tempo ELT Total
0:02:44	0:00:26	0:03:10
0:02:51	0:00:26	0:03:17
0:02:43	0:00:25	0:03:08
0:02:53	0:00:27	0:03:20
0:02:49	0:00:50	0:03:39
0:02:46	0:00:25	0:03:11
0:02:48	0:00:25	0:03:13
0:02:48	0:00:26	0:03:14
0:02:45	0:00:26	0:03:11
0:02:45	0:00:26	0:03:11

Fonte: Autores (2023).

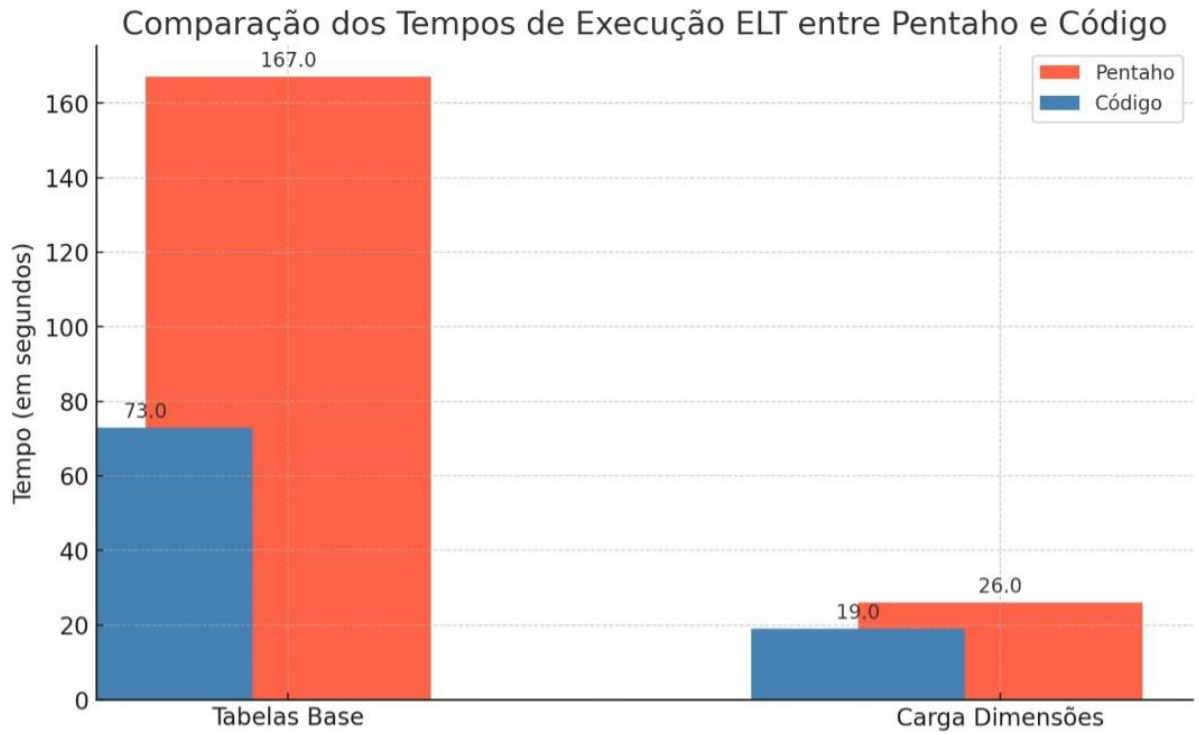
Neste quadro, foi registrado o tempo de extração das tabelas-base do banco de origem e o tempo de carga no banco destino, assim como no cenário ELT em código. Foi encontrada, então, a média do tempo de carga e extração, que foi de 2 minutos e 47 segundos. Já a média do tempo de transformação para as tabelas dimensão foi de 26 segundos, sendo possível

observar uma discrepância em um dos testes, onde o tempo durou 50 segundos. Por fim, a média da duração total dos testes foi de 3 minutos e 12 segundos.

2.4 Discussão

Após a obtenção dos resultados obtidos das análises dos diferentes cenários, foi realizada uma representação visual dos resultados por meio da criação de dois gráficos distintos. Esses gráficos foram construídos para oferecer uma comparação visual e detalhada das médias dos tempos de execução de cada cenário. Abaixo está o gráfico que compara as médias do tempo de extração, carga e transformação do processo ELT tanto em código quanto no *Pentaho*.

Imagem 9.Comparação dos tempos de execução ELT.



Fonte: Autores (2023).

No gráfico acima, observou-se que a maior parte do tempo de execução do ELT concentrou-se no estágio de extração e carga das tabelas-base do banco de dados de origem para o banco de destino. Isso pode ser observado tanto no teste em código quanto usando o *Pentaho*. Em contrapartida, o processo de transformação exibiu uma duração significativamente mais breve.

De modo geral, um dos principais fatores que afetam a velocidade desse tipo de processamento é o volume de dados sendo tratados. A etapa de extração pode consumir mais

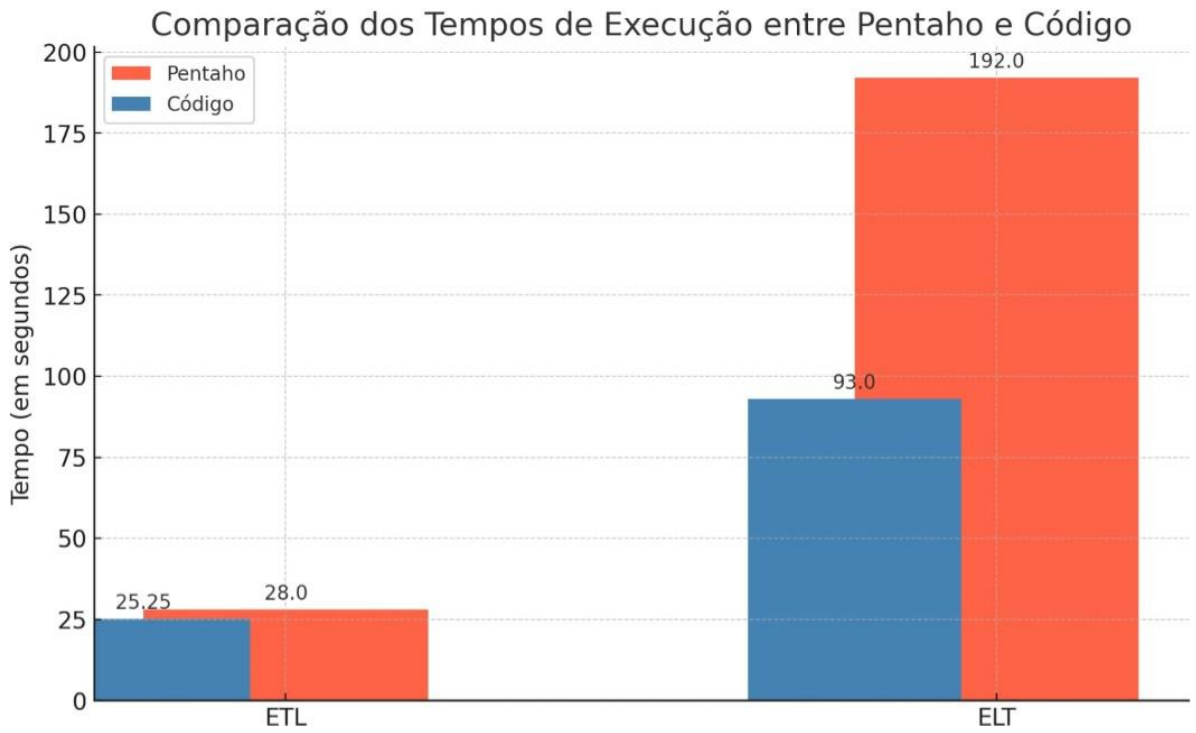
tempo, pois todo o volume de dados brutos é primeiro movido da fonte para o banco de destino, sem muita transformação inicial. Além disso, uma observação importante é que o tempo de execução em código é menor em relação ao tempo de execução no *Pentaho*, o que evidencia que, apesar da eficiência da ferramenta, que possui funcionalidades específicas para as etapas do processo ELT, o tempo de execução é menor quando se trabalha diretamente com código, ressaltando a superioridade do desempenho em termos de tempo de execução na abordagem por código principalmente devido ao fato de que o código permite uma personalização mais aprofundada e flexível. Isso possibilita a otimização de algoritmos e lógicas de processamento para uma eficiência superior, maximizando o aproveitamento dos recursos disponíveis na máquina ou ambiente de execução. Em contrapartida, as ferramentas gráficas podem ser menos eficientes na utilização de memória e processamento, já que tendem a ser mais genéricas para atender a uma variedade de casos de uso.

Entretanto, é válido considerar que a implementação por código pode ser mais desafiadora para profissionais que não possuem familiaridade com programação em geral. Em contrapartida, a utilização da ferramenta pode oferecer uma curva de aprendizado mais suave, possibilitando que o profissional adquira habilidades para manipular a ferramenta e realizar os processos de integração de maneira mais acessível.

Outro principal fator é o poder computacional das máquinas que estavam realizando os testes. Em determinado momento, as máquinas foram utilizadas para realizar outros serviços que demandam poder de processamento, o que acabou aumentando o tempo de alguns testes, em algumas situações aumentando mais que o triplo do tempo médio, como foi possível observar nos quadros de resultados.

Abaixo está o gráfico que representa a comparação entre os tempos de execução dos processos ETL e ELT.

Imagem 10. Comparação entre os tempos de execução dos dois processos.



Fonte: Autores (2023).

No gráfico, é notável a discrepância significativa entre os tempos de execução dos dois processos, ETL e ELT. O ETL demonstrou um intervalo de tempo consideravelmente mais curto em comparação com o ELT. Em ambas as abordagens, seja por meio do código ou utilizando o *Pentaho*, a média do tempo de execução do ETL permaneceu abaixo dos 30 segundos. Por outro lado, o ELT exibiu um tempo de execução mais prolongado. Tanto na implementação por código quanto no uso do *Pentaho*, a média do tempo de execução do ELT ultrapassou um minuto, sendo mais que o dobro da média observada no ETL. Essas observações evidenciam que o ETL foi bem mais otimizado para realizar a integração de dados. É importante ressaltar que esses resultados foram obtidos ao utilizar o banco de dados de amostra *AdventureWorks 2022*, e que os tempos de execução não são necessariamente universais e podem variar consideravelmente ao serem aplicadas essas técnicas em diferentes conjuntos de bancos de dados.

É interessante ressaltar algumas diferenças entre os resultados dos testes e informações encontradas na literatura. É possível encontrar algumas fontes descrevendo o processo ELT tendo uma performance melhor do que o processo ETL, em quesitos como flexibilidade (LALEYE, 2022) e tempo (GARG, 2023). Apesar disso, como pontuado anteriormente, o processo ETL foi consideravelmente mais rápido que o ELT, dentro do contexto específico

trabalho nesta pesquisa, com o banco de dados amostral *AdventureWorks*, onde o tempo de execução do ETL foi metade do que o tempo de execução do ELT. O principal motivo para isso se dá pelo fato do processo ELT precisar carregar todas as tabelas-base para então transformar apenas os dados necessários para o desenvolvimento das tabelas-dimensão. Já no processo ETL, apenas os dados necessários são transformados e então carregados, gerando um teste mais otimizado para o *AdventureWorks*.

2.5 Conclusão/Considerações Finais

Diante da análise comparativa entre os processos ELT e ETL, evidencia-se a relevância de considerar a aplicação específica de cada metodologia. A literatura aponta vantagens do ELT em termos de flexibilidade, velocidade e redundância, porém, no contexto do estudo com o banco de dados AdventureWorks, o processo ETL demonstrou uma performance temporal superior. Este resultado é justificado pela estratégia do ETL em realizar a transformação dos dados essenciais antes do carregamento, em contraposição ao ELT, que demanda o carregamento prévio de todas as tabelas-base. Além disso, é interessante também perceber as diferenças entre o uso de código e o da ferramenta *Pentaho* para a aplicação desses processos. Essa diferenciação entre o uso do código e da ferramenta destaca um trade-off significativo: enquanto o código oferece melhor desempenho em tempo de execução, a ferramenta proporciona uma abordagem mais acessível e amigável para profissionais sem experiência extensiva em programação, permitindo a realização dos processos de integração com relativa facilidade após um período de aprendizado.

Nesse sentido, fica evidente a importância de avaliar as particularidades de cada abordagem em relação ao contexto de aplicação, reconhecendo que a escolha entre ELT e ETL deve ser baseada nas necessidades específicas de cada projeto e nas características do banco de dados em questão. Futuros estudos podem aprofundar essa análise, considerando diferentes conjuntos de dados e cenários, como por exemplo investigar como a evolução das tecnologias, como computação em nuvem, big data e automação, influenciam a escolha entre ELT e ETL, examinando seu impacto na eficácia e eficiência desses processos, a fim de oferecer insights mais abrangentes sobre a eficácia e a eficiência desses processos na prática.

3 REFERÊNCIAS BIBLIOGRÁFICAS

AdventureWorks Sample Database. Datensen. Disponível em: <https://www.datensen.com/blog/sql-server/adventureworks-sample-database/>. Acesso em 12 dez. 2023.

BICCA, Daniela. TECNOLOGIA APLICADA À CONTABILIDADE: ESTUDO DE CASO EM UMA ORGANIZAÇÃO CONTÁBIL. Revista Contabilidade em Foco. Porto Alegre. v. 2., n.2 p. 3-30. 2020.

CASAROTTO, Camila. Transformar dados em informação é essencial: saiba como fazer isso. Rockcontent, 28 jun. 2021. Disponível em: <https://rockcontent.com/br/blog/transformar-dados-em-informacao/>. Acesso em: 29 mai. 2023.

CONSTANTIOU, Ioanna D.; KALLINIKOS, Jannis. New games, new rules: big data and the changing context of strategy. Journal of Information Technology. Frederiksberg. v. 30, n. 1, p. 44-57. Março. 2015.

Data Lake vs Data Warehouse: Which Is Right for You? Astera. 27 abr. 2023. Disponível em: <https://www.astera.com/knowledge-center/data-lake-vs-data-warehouse-which-is-right-for-you/>. Acesso em: 14 set. 2023

Data mining in business analytics. Western Governors University. 8 set. 2023. Disponível em: <https://www.wgu.edu/blog/data-mining-business-analytics2005.html#:~:text=Simply%20put%2C%20data%20mining%20is,raw%20data%20into%20useful%20information.&text=It%20pulls%20out%20information%20from,%2C%20market%20effectively%2C%20and%20more.>. Acesso em: 29 mai. 2023.

DEARMER, Abe. The Importance and Benefits of a Data Pipeline. Integrate.io, 22 set. 2020. Disponível em: <https://www.integrate.io/blog/what-is-a-data-pipeline/#:~:text=Data%20pipelines%2C%20by%20consolidating%20data,crucial%20for%20reliable%20business%20insights>. Acesso em: 30 mai. 2023.

DEDIĆ, Nedim; STANIER, Clare. **An Evaluation of the Challenges of Multilingualism in Data Warehouse Development**. *In:* International Conference on Enterprise Information Systems - ICEIS. nº 18. 2016. Roma, Itália. Artigo. Roma: 2016. p. 196 - 206.

Definition of Fact Table. TechTarget. abr. 2012. Disponível em: <https://www.techtarget.com/searchdatamanagement/definition/fact-table>. Acesso em: 20 set. 2023.

Definition of Star Schema. TechTarget. mai. 2021. Disponível em: <https://www.techtarget.com/searchdatamanagement/definition/star-schema#:~:text=A%20star%20schema%20is%20a,store%20attributes%20about%20the%20data>. Acesso em: 20 set. 2023.

ETL vs ELT.- Understanding the key differences. SoftwebSolutions. Disponível em: <https://www.softwebsolutions.com/resources/key-differences-etl-vs-elt.html>. Acesso em: 20 set. 2023.

GARG, Aryan. ETL vs ELT: Which One is Right for Your Data Pipeline? KDnuggets. 31 mar. 2023. Disponível em: <https://www.kdnuggets.com/2023/03/etl-elt-one-right-data-pipeline.html#:~:text=ETL%20and%20ELT%20are%20data,transforms%20the%20data%20after%20loading>. Acesso em: 04 set. 2023.

HAIDER, Aelia. Data Integration Tools and Solutions: Top 10 for 2023 and Beyond. Astera. 28 Ago. 2020. Disponível em: <https://www.astera.com/pt/type/blog/data-integration-tools-for-businesses/>. Acesso em: 28 ago. 2023.

LENZERINI, Maurizio. **Data Integration: A theoretical perspective**. 2002. *In:* Symposium on Principles of Database Systems. nº 21. 2002. Madison, Wisconsin. Artigo. New York: Association for Computing Machinery 2002. p. 233-246.

LEONG, Nicholas. How I Redesigned over 100 ETL into ELT Data Pipelines. 4 out. 2021. Medium. Disponível em: <https://towardsdatascience.com/how-i-redesigned-over-100-etl-into-elt-data-pipelines-c58d3a3cb3c>. Acesso em: 15 set. 2023.

MCAFEE, Andrew *et al*. **Big data: the management revolution.** Harvard business review, v. 90, n. 10, p. 60-68, Outubro 2012.

MIKALEF, Patrick et al. **Big data analytics and firm performance: Findings from a mixed-method approach.** Journal of Business Research. vol. 98. pág 261-276. Fevereiro 2019.

NAEEM, Tehreem. Understanding Structured, Semi-Structured, and Unstructured Data. Astera. 1 nov. 2020. Disponível em: <https://www.astera.com/type/blog/structured-semi-structured-and-unstructured-data/>. Acesso em: 14 set. 2023.

NAEEM, Tehreem. Unstructured Data Management: Challenges & Opportunities For 2023. Astera. 23 abr. 2020. Disponível em: <https://www.astera.com/type/blog/unstructured-data-management/>. Acesso em: 29 ago. 2023.

NAEEM, Tehreem. What is an ETL Tool: Types, Features, and Use Cases. Astera. 28 abr. 2020. Disponível em: <https://www.astera.com/knowledge-center/what-is-etl-tool/>. Acesso em: 29 ago. 2023.

SHELDON, Robert. Definition of Dimension Table. jul. 2023. TechTarget. Disponível em: <https://www.techtarget.com/searchdatamanagement/definition/dimension-table>. Acessoem: 20 set. 2023.

SURANI, Ibrahim. Data Integration for Business Intelligence: Best Practices. Dataversity, 30 mar. 2020. Disponível em: <https://www.dataversity.net/data-integration-for-business-intelligence-best-practices/>. Acesso em: 29 mai. 2023.

WANG, Jin, et al. **Big Data Service Architecture: A Survey**. Journal of Internet Technology, v. 21, p. 393-405, 2020.

WAMBA, Samuel Fosso *et al*. **Big data analytics and firm performance: Effects of dynamic capabilities.** Journal of Business Research, v. 70, p. 356-365, 2017.

What is Online Transaction Processing? Oracle. Disponível em: <https://www.oracle.com/database/what-is-oltp/>. Acesso em: 20 set. 2023.

What is OLTP? IBM. Disponível em: <https://www.ibm.com/topics/oltp>. Acesso em: 20 set. 2023.

WRIGHT, Gavin; VAUGHAN, Jack. Definition of Denormalization. TechTarget. mai. 2021. Disponível em: <https://www.techtarget.com/searchdatamanagement/definition/denormalization#:~:text=Denormalization%20is%20the%20process%20of,of%20each%20piece%20of%20information> Acesso em: 20 set. 2023.

ZOLA, Andrew. Definition of Schema. TechTarget. mai. 2021. Disponível em: <https://www.techtarget.com/searchdatamanagement/definition/schema>. Acesso em: 15 set. 2023.

APÊNDICE A

https://github.com/AnorakBs/TCC