SEMANA ACADÊMICA
REVISTA CIENTÍFICA
ISSN 2236-6717

MODELAGEM DE DADOS EM DATA WAREHOUSES MODERNOS
COMPARAÇÃO ENTRE MODELOS NO CONTEXTO NAS NOVAS PLATAFORMAS ANALÍTICAS

Raissa Silva Menezes de Santana[1]
https://orcid.org/0009-0004-8180-5763

RESUMO

Após a segunda metade da década de 1980, as empresas armazenavam apenas dados provenientes de seus sistemas transacionais. Porém, naturalmente, surgiu a necessidade de se obter métricas baseadas em dados que pudessem apoiar os tomadores de decisão em suas atividades gerenciais. Nessa esteira, foram sendo desenvolvidos diversos tipos de Sistemas de Suporte à Decisão, a exemplo dos Data Warehouses. Trata-se de uma tecnologia em que os dados são extraídos dos sistemas transacionais, subsequentemente transformados e carregados em um banco de dados. Dessa forma, os usuários finais conseguiam realizar análises sob diversas perspectivas através de uma fonte única e integrada de dados. Esse foi um modelo bem-sucedido por muitos tempo, até que, nos anos 2000, presenciou-se um crescimento exponencial na quantidade e variedade de dados gerados pelas organizações. Isso impulsionou o desenvolvimento de tecnologias para armazenamento e processamento distribuído, como Hadoop, e, em seguida, as plataformas de computação em nuvem, como Azure, AWS e Google Cloud. Esse novo contexto dos ambientes analíticos proporcionou mudanças relevantes, como a queda expressiva nos custos de armazenamento de dados e o desacoplamento entre processamento e armazenamento. Diante disso, é natural surgirem questionamentos como: os modelos de dados tradicionais como Star Schema ainda fazem sentido nos tempos atuais ou e a melhor opção é abraçar propostas mais ousadas, como One Big Table? Ao se investigar o que os profissionais de dados estão pensando a respeito do assunto, percebe-se que não há consenso em torno do tema. Isso ocorre porque cada caso concreto apresenta suas peculiaridades, de forma que nenhum modelo irá atender às necessidades de todas as situações. Porém, apesar dessas limitações, é possível obter um resultado equilibrado entre armazenamento, manutenção e desempenho através do conhecimento das vantagens e desvantagens apresentadas por cada um deles.

Palavras-chave

Data lake; Data lakehouse; Modelagem de dados; Star schema; Data vault

1 Auditora-Fiscal da Receita Federal do Brasil, Secretaria Especial da Receita Federal do Brasil, MG, raissa.santana@rfb.gov.br.

http://dx.doi.org/10.35265/2236-6717-234-12613
FORTALEZA-CE. EDIÇÃO 234. V.11. ANO 2023.

Data Modeling in modern Data Warehouses

Model comparison in the context of new analytics platforms

ABSTRACT

After the second half of the 1980s, companies only stored data from their transactional systems. However, naturally, the need arose to obtain metrics based on data that could support decision makers in their management activities. As a result, several types of Decision Support Systems were developed, such as Data Warehouses. This is a technology in which data is extracted from transactional systems, subsequently transformed and loaded into a database. In this way, end users were able to perform analyses from multiple perspectives through a single, integrated source of data. This was a successful model for a long time, until the 2000s saw an exponential growth in the amount and variety of data generated by organizations. This spurred the development of technologies for distributed storage and processing such as Hadoop, and then cloud computing platforms such as Azure, AWS, and Google Cloud. This new context of analytical environments has brought about important changes, such as a significant decrease in data storage costs and the decoupling of processing and storage. In view of this, it is natural to ask questions such as: do traditional data models like Star Schema still make sense nowadays or is the best option to embrace bolder proposals like One Big Table? When investigating what data professionals are thinking about the subject, one realizes that there is no consensus around the topic. This is because each specific case presents its own peculiarities, so that no model will meet the needs of all situations. However, despite these limitations, it is possible to achieve a balanced result between storage, maintenance, and performance by knowing the advantages and disadvantages presented by each of them.

Keywords

Data lake; Data lakehouse; Data modeling; Star schema; Data vault

Submetido em: 30/04/2023 – **Aprovado em**: 17/05/2023 – **Publicado em**: 05/06/2023

1 INTRODUÇÃO

Os anos 2000 marcaram o início de uma era em que a geração de dados se tornou massiva e variada. Com o advento da era digital, as empresas passaram a ter que considerar não apenas os dados provenientes de seus sistemas transacionais, como *Enterprise Resouce Planning* (*ERP)* e *Costumer Relationship Management* (CRM), mas também de redes sociais, sensores, imagens, vídeos, entre outros. Para lidar com esse grande volume de dados, os tradicionais *Data Warehouses* foram perdendo lugar para tecnologias de *Data Lake* e, posteriormente plataformas de nuvem. Com isso, novos tipos de modelagens de dados passam a ser considerados a fim de se aproveitar as evoluções decorrentes das novas tecnologias. Enquanto isso, os tipos de modelagens de tradicionais em alguns momentos passaram a ser classificados como obsoletos e, portanto, deixados de lado. Mas será que existe o modelo mais adequado, considerando as transformações ocorridas nos últimos anos? Para tentar responder a essa questão, o presente trabalho relaciona alguns argumentos apresentados por profissionais da área de dados em relação à três modelos: *Star Schema, Data Vault* e *One Big Table.* O objetivo é avaliar quais as principais vantagens e desvantagens apresentadas por cada um e, a partir disso, refletir sobre qual deles seria o mais pertinente para o cenário atual.

2 SISTEMAS DE APOIO À DECISÃO

Até a metade da década de 1980, as empresas costumavam armazenar apenas dados gerados pelos sistemas transacionais que eram organicamente gerados pelas atividades do negócio. Acessar esses dados de forma a gerar insights sobre diversos aspectos que impactam as companhias se mostrou cada vez mais fundamental, sendo computadores a única ferramenta capaz de viabilizar um adequado proveito dessas informações. Foi nesse contexto que surgiram *os Decision Support Systems* (DSS), ou Sistemas de Apoio à Decisão (GOLFARELLI; RIZZI, 2009).

Golfarelli e Rizzi (2009, p.3) definem DSS como:

> A decision support system (DSS) is a set of expandable, interactive IT techniques and tools designed for processing and analyzing data and for supporting managers in decision making. To do this, the system matches individual resources of managers with computer resources to improve the quality of the decisions made.

Em outras palavras, Sistemas de Suporte à Decisão são um conjunto de técnicas e ferramentas que possuem o objetivo específico de auxiliar gestores na tomada de decisões. E, ao contrário do que se pode intuir, esse suporte não acontece apenas em funcionalidades relacionadas à manipulação de dados, as quais permitem sumarizações sob diversas perspectivas. Ocorre também por meio de recursos que viabilizam o trabalho colaborativo, o

contato entre funcionários, o compartilhamento de arquivos diversos e a recuperação de documentos. Ou seja, qualquer tecnologia que estruture o fluxo de dados com o intuito promover a obtenção de conhecimento pode ser considerada um Sistemas de Suporte à Decisão.

Segundo Power (2002), os Sistemas de Suporte à Decisão podem ser classificados da seguinte forma:

a) **Communications driven DSS:** esse tipo de DSS é baseado no compartilhamento de arquivos e interação entre os usuários através da rede de computadores. A tomada de decisões também exige que pessoas troquem informações e trabalhe de forma colaborativa para se chegar à melhor decisão possível, por isso tais tecnologias são classificadas como DSS. Sistemas como *Google Docs* e *Microsoft SharePoint Workspace*, são exemplos que se encaixam nessa categoria (MOSTAFA;ROJA; NASIBEH; ABDELWAHED, 2022) .

b) **Data-driven DSS:** São sistemas que permitem a manipulação de dados estruturados, como a execução de consultas em diversos níveis de agregação. Como exemplo, temos os *Data Warehouses*.

c) **Document-driven DSS:** São ferramentas que auxiliam a recuperação, classificação e gerenciamento de documentos não estruturados, como arquivos HTML, imagens, sons e vídeos. Podemos citar os motores de busca (*search engine*) como exemplo de *Documet-driven* DSS.

d) **Knowledge-driven DSS:** Esse tipo de DSS fornece "recomendações" de tomadas de decisão utilizando as bases de dados e regras de negócio da empresa. Para exemplificar, pode-se citar as ferramentas de *Data Mining*, as quais encontram padrões de relacionamento em grandes conjuntos de dados que podem gerar insights a seus usuários.

e) **Model-driven DSS:** Aqui são classificados os sistemas que propiciam a obtenção de informações através do fornecimento de parâmetros a modelos estatísticos. Geralmente, esses sistemas não utilizam grandes bases de dados. Como exemplo, temos sistemas com modelos contábeis e financeiros, modelos representacionais e de otimização.

No contexto das tecnologias que permitem a realização de consultas multidimensionais e a construção de relatórios (*Data-driven* DSS), a forma mais comum de implementação é a associação de um *Data Warehouse* à alguma ferramenta de geração de consultas e relatórios (POWER, 2008).

3 DATA WAREHOUSES

Os *Data Warehouses* (DW's) foram desenvolvidos no final da década de 1980 por dois pesquisadores da IBM, Barry Devlin e Paul Murphy, com o objetivo estruturar o fluxo de dados para os sistemas de apoio a decisões (NAMBIAR; MUNDRA, 2022). Os motivos que impulsionaram o surgimento dos DW's manifestaram-se nos corredores das organizações,

através de queixas relacionadas à (KIMBALL, 2002): Falta de acesso à enorme quantidade de dados da companhia;

a) Necessidade constante de manipular dados para analisá-los através de diferentes perspectivas;
b) Dificuldade na localização dos dados mais relevantes para a tomada de decisão;
c) Obtenção de resultados diferentes para o mesmo tipo de análise;
d) Necessidade de tomar decisões baseadas em fatos.

Um conceito de *Data Warehouse* bastante aceito desde o princípio é que se trata de uma base para processamento de informações com as seguintes características: focado em um tema específico, integrado, não volátil, expansível ao longo do tempo e detentor de um conjunto de dados para apoio às decisões gerenciais (INMON;STRAUSS; NEUSHLOSS,2010).

Há vários tipos de arquiteturas de Data Warehouses, porém, a mais comum é a denominada arquitetura de três camadas (*three-tier archtecture*). A primeira camada seria o banco de dados do DW, no qual os dados são carregados após serem limpos, transformados e integrados. Já a camada do meio seria uma abstração dos dados carregados na camada anterior, e atuaria como camada intermediária entre o banco de dados e os usuários finais. Por fim, a terceira camada seria a responsável pela interação do usuário com os dados, consistindo em ferramentas para a realização de consultas analíticas, construção de relatórios e *data mining* (NAMBIAR; MUNDRA, 2002) .

O processo responsável pela disponibilização dos dados nos DW's é denominado ETL (*Extract, Transform and Load* ou Extração, Transformação e Carregamento). Com o aumento da popularidade das bases de dados na década de 1970, o processo de ETL foi desenvolvido como uma forma de integrar e carregar dados para processamento analítico, se tornando, posteriormente, o método preferencial dos projetos de *Data Warehouse* (ETL, 2023) .

A etapa de Extração consiste na exportação de dados provenientes de diversas fontes (como sistemas do tipo CRM, SRM, ERP e arquivos simples) para uma área de preparação (*staging area*). Já a Transformação é o passo em que os dados brutos sofrem uma série de processamentos para serem organizados e integrados. São exemplos de tais operações: filtragens, eliminação de duplicidades, sumarizações, entre outros. Por fim, no Carregamento, os dados transformados são movidos para a base de dados do *Data Warehouse* (ETL, 2023).

Porém, o cenário mudou bastante nas últimas duas décadas em relação ao período de surgimento dos DWs. Em virtude do uso massivo de redes sociais (MOHD, 2020) , bem como dos avanços tecnológicos que permitiram a obtenção de dados oriundos de aparelhos de comunicação e sensores digitais (THE BIG DATA-IOT RELATIONSHIP: HOW THEY HELP EACH OTHER, 2023), a quantidade de dados gerados aumentou substancialmente, dando início à era do *big data*. Como consequência disso, os DWs tradicionais começaram revelar suas limitações, como (WHAT IS A DATA WAREHOUSE ? , [s.d.]):

a) **Dados não estruturados:** Os DWs tradicionais apenas dão suporte a dados altamente estruturados, não comportando arquivos como imagens, vídeos, dados

provenientes de sensores, JSON e XML.

b) **Aprisionamento tecnológico:** Como a maioria das empresas de DW utilizam seus próprios formatos de arquivo, ao invés de formatos baseados em código aberto, torna-se extremamente custoso migrar os dados para outras ferramentas.
c) **Alto custo:** As empresas fornecedoras de DW cobram tanto pelo armazenamento quanto pelo processamento dos dados. Em um contexto de big data, isso se torna bastante oneroso.

4 DATA LAKES

Por volta de 2005 iniciou-se a *Web 2.0* (HOW ORGANIZATIONS USE LAKEHOUSES TO GET MORE VALUE FROM DATA, 2022). Ao contrário da *Web 1.0,* cuja geração de conteúdo era feita por poucas pessoas, na *Web 2.0* tem-se um número crescente de indivíduos criando comunidades, colaborando entre si e se conectando através de mídias sociais (TERRA, 2023). Isso significa uma enorme quantidade de textos, vídeos, imagens, áudios, dados estruturados e semiestruturados, todos disponíveis para serem coletados, armazenados e processados.

Nesse contexto, surgiu o Hadoop, a tecnologia de distribuição e processamento de grande volume de dados que viabilizou os *Data Lakes* (na verdade, essa expressão foi cunhada por James Dixon, *Chief Technology Officer* do Pentaho, apenas em 2010 (HOW ORGANIZATIONS USE LAKEHOUSES TO GET MORE VALUE FROM DATA, 2022)). O Hadoop foi criado por Doug Cunting, a partir do projeto Apache Nutch, uma *web search engine* de código aberto que, por sua vez, era parte de um projeto maior chamado Lucene. O Apache Nutch foi criado em 2002 e se desenvolveu com base em dois artigos publicados pelo Google: "*The Google File System*", em 2003, e "*MapReduce: Simplified Data Processing on Large Clusters*" em 2004 (WHITE, 2015).

O Hadoop rapidamente se expandiu através da oferta de ferramentas para gerenciamento de *clusters*, agendamento de processos, execução de consultas, entre outros. Porém, muitas empresas que adotaram essas tecnologias descobriram que gerenciar seus próprios *clusters* e encontrar um equilíbrio entre a quantidade de nós era um grande desafio. Isso porque enquanto alguns processos demandavam centenas ou milhares de nós, em outros momentos esses nós ficavam ociosos, impactando economicamente as empresas (HOW ORGANIZATIONS USE LAKEHOUSES TO GET MORE VALUE FROM DATA, 2022).

Nesse sentido, o surgimento das nuvens públicas (*public cloud*) em 2010 ajudou as empresas a lidar com esses desafios. A partir de então, as companhias puderam ingerir e armazenar seus dados brutos de forma barata e escalável. Além disso, um outro benefício decorrente da delegação do gerenciamento da infraestrutura do *cluster* aos provedores de armazenamento em nuvem foi permitir que as empresas focassem mais na gestão dos dados em si, o que viabilizou o surgimento de um novo paradigma: os *Data Lakehouses* (HOW ORGANIZATIONS USE LAKEHOUSES TO GET MORE VALUE FROM DATA, 2022).

5 DATA LAKEHOUSES

Quando os *Data Lakes* foram concebidos, pensava-se que bastava carregar os dados brutos no *cluster* para que o usuário final pudesse realizar suas explorações e obter *insights*. Porém, as empresas rapidamente perceberam que utilizar os dados do *Data Lakes* era muito mais complexo do que simplesmente depositá-los lá. Isso porque os dados armazenados no lago não eram submetidos a nenhum tipo de controle de qualidade e governança de dados, o que fazia o *Data Lakes* assemelhar-se mais a um *data swamp* (pântano de dados) (INMON; LEVINS, 2021).

Dessa forma, para que os usuários conseguissem utilizar esses dados em suas tomadas de decisões, as empresas passaram a fazer ETL de uma parte deles para um *Data Warehouse* separado, configurando uma arquitetura de duas camadas. Nesse sentido, com a substituição do HDFS em *clusters on-premise* por *Data Lakes* na nuvem (como *Amazon S3*, *Microsoft Azure e Google Cloud Plataform*) a partir de 2015, a arquitetura de duas camadas se tornou dominante, sendo utilizada por todas as 500 maiores empresas dos Estados Unidos listadas relacionadas pela revista Forbes (ARMBRUST; GHODSI;XIN; ZAHARIA, 2021).

Apesar de o *cloud Data Lakes* possuir um custo baixo em virtude da separação do armazenamento e processamento, a arquitetura de duas camadas é cara e complexa. Isso porque nesse tipo de organização, primeiro há a ETL dos sistemas transacionais para o *Data Lake* e, em seguida, a ELT, do *Data Lake* para o *Data Warehouse*. Cada vez que os dados passam por esse tipo de operação, há maiores chances de falhas, comprometendo a qualidade dos dados. Além disso, mantê-los consistentes e atualizados torna-se mais desafiador. Por fim, acaba-se pagando por dois armazenamentos, o do *Data Lake* e do *Data Warehouse*, sendo que no último caso os dados não podem ser utilizados em outras ferramentas analíticas por possuírem um formato específico do fornecedor do *Data Warehouse* (ARMBRUST; GHODSI;XIN; ZAHARIA, 2021).

Nesse cenário, os *Data Lakehouses* foram concebidos para lidar com esses desafios. *Data Lakehouse* é um tipo de arquitetura em que ferramentas de gerenciamento de dados típicas de *Data Warehouses* são implementadas em uma camada diretamente acima do armazenamento em nuvem (ARMBRUST; GHODSI; LORICA; XIN; ZAHARIA, 2020). Com isso, é possível depositar os dados em um único local, aproveitando o baixo custo de armazenagem dos *Data Lakes* e, ao mesmo tempo, promover dados governados e uma infraestrutura analítica adequada.

Uma das características dos *Data Lakehouses* que proporcionam confiabilidade dos dados é o suporte a transações ACID. ACID é um acrônimo para Atomicidade, Consistência, Isolamento e Durabilidade. Atomicidade significa que qualquer alteração nos dados só é efetivada se for bem-sucedida, caso contrário não há mudanças. Já Consistência expressa o respeito às restrições do banco de dados e integridade referencial, garantindo a correção dos dados. Quanto ao Isolamento, serve para assegurar a independência de transações

concorrentes. Por fim, a Durabilidade garante que as alterações feitas sejam definitivas, permanecendo mesmo quando há falhas no sistema (ACID transactions, *[s.d]*).

Outra vantagem reside nos aspectos relacionados à segurança e à governança de dados. Por exemplo, com os *Data Lakehouses* é possível ter registros contendo detalhes sobre todas as alterações às quais os dados foram submetidos, proporcionando rastreabilidade, o que é essencial em auditorias. O versionamento dos dados também é viabilizado nesse tipo de arquitetura, permitindo o retorno dos dados às suas versões antigas para a realização de auditorias e experimentos. Por fim, controles de acesso baseados em papéis também podem ser implementados, tanto nos níveis de linha quanto de coluna, contribuindo para a segurança do ambiente (INMON; LEVINS; SRIVASTAVA, 2021).

Por último, como os dados do *Data Lakehouse* são armazenados em *Data Lakes*, é possível a utilização de arquivos de formato aberto, como o *Apache Parquet* e ORC. Um dos principais benefícios que se obtém na utilização desses tipos de formatos de arquivo é a flexibilidade na utilização de ferramentas analíticas, tanto de *Business Intelligence*, como *Power BI* e *Tableau*, quanto *Machine Learning*, através de bibliotecas como *Tensor Flow* e *Spark MLlib* (ARMBRUST; GHODSI; XIN; ZAHARIA, 2021).

Diante deste novo paradigma de armazenamento e processamento de grande volume de dados para fins analíticos, um questionamento natural ocorre: como os tipos de modelagens de dados existentes se encaixam nesse ecossistema emergente? Para tentar responder a essa pergunta, nas próximas seções serão analisados três tipos de modelagens de dados: *Star Schema, Data Vault* e *One Big Table* (OBT). A ideia é descrever suas respectivas arquiteturas, bem como apresentar argumentos de profissionais em relação a como eles se inserem nessa novo contexto.

6 STAR SCHEMA

Em 1996, Ralph Kimball divulgou o livro *The Data Warehouse Toolkit*, uma guia para a elaboração de projetos e construção de *Data Warehouses*, utilizando a modelagem dimensional *Star Schema* como base. Essa técnica de modelagem divide as tabelas entre fato e dimensões.

Nesse sentido, as tabelas fato apresentam informações numéricas de uma empresa, ou seja, medidas, decorrentes de suas operações. Por exemplo, uma tabela de vendas apresenta o valor das vendas sob diversas perspectivas: produto, loja, mês, ano, e assim sucessivamente. Todas as medidas da tabela fato devem estar no mesmo nível de granularidade.

Segundo Kimball (2002) existem três tipos de tabela fato. O primeiro deles é o mais útil de todos, as tabelas aditividas. Trata-se de tabelas que podem ser sumarizadas através da soma das métricas por diversas dimensões. Além disso, há as tabelas semiaditivas, cujas métricas podem ser agregadas apenas para determinadas dimensões. Por fim, existem as

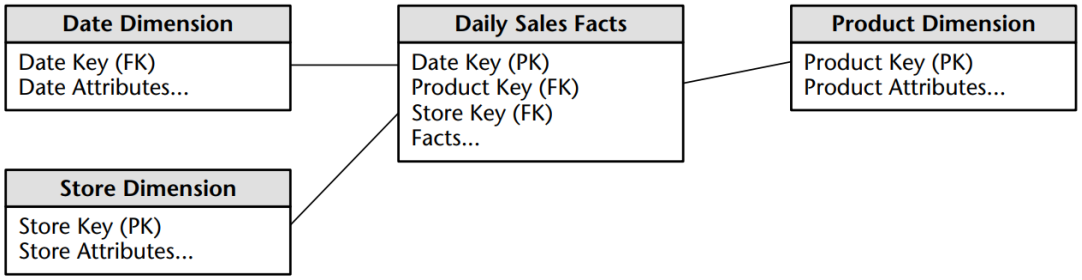
tabelas não aditivas , as quais não permitem somas, sendo necessário utilizar contagens e médias na sumarização dos dados.

Outra característica das tabelas fato é que nelas não são armazenadas informações textuais, as quais estarão presentes em alguma tabela dimensão. Além disso, as tabelas fato devem ser preenchidas apenas com eventos que realmente aconteceram, e não com zeros, no caso da não ocorrência do fato para alguma combinação de dimensões. Nesse sentido, as tabelas fato conseguem ocupar menos espaço, considerando que naturalmente terão um grande número de linhas.

Como consequência do não armazenamento de informações textuais, as tabelas fato contém chaves estrangeiras que as vinculam a diversas tabelas dimensão. Além disso, elas também possuem geralmente uma chave primária composta. O motivo disso é que as tabelas fato expressam os relacionamentos muitos para muitos com as dimensões. Dessa forma, para que uma linha seja identificada de forma única, sua chave primária precisa ser composta pelas chaves estrangeiras de algumas dimensões. Poderia haver uma coluna com o id da linha, mas, de acordo com Kimball (2002), na maioria dos casos isso não é vantajoso, por aumentar ainda mais o espaço ocupado pela tabela.

Por sua vez, as tabelas dimensão apresentam atributos, ou seja, informações descritivas do negócio. Geralmente possuem diversas colunas, contendo o máximo de informações textuais relevantes, as quais são utilizadas para filtragens e agrupamentos. Porém, esse tipo de tabela geralmente dispõe de poucas linhas. Como consequência, elas são altamente desnormalizadas, pois apesar de possuírem valores repetidos, decorrentes do relacionamento hierárquico dos atributos, por serem tabelas enxutas, isso não impacta muito o armazenamento e, além disso, melhora a performance das consultas por evitar a necessidade de mais junções. Caso essas tabelas fossem normalizadas, estar-se-ia diante de outro tipo de arquitetura, a *Snowflake*.

Figura 1. Star Schema



Fonte: KIMBALL, The Data Warehouse Toolkit, 2002, p. 22

No contexto das novas ferramentas analíticas as opiniões são divergentes. Nair (2022) argumenta que o *Star Schema* não é mais necessário para garantir a performance e otimizações das consultas, pois atualmente os serviços de nuvem possuem recursos abundantes que conseguem entregar bons resultados nesses aspectos. Por outro lado, segundo o autor, esse tipo de modelagem ainda é muito útil para integrar dados de diferentes fontes e obter uma visão mais ampla e organizada do negócio, o que facilita a elaboração e compreensão de KPI's (*Key Performance Indicator*), relatórios e dashboards. Além disso, ele argumenta que fazer transformações nos dados para adequá-los a esse schema é bastante facilitado por plataformas de nuvem como *Snowflake,Synapse e Google Big Query*. Por fim, ele ressalta que os dados modelados não precisam ser transferidos para outro banco de dados, podendo permanecer armazenados na nuvem e serem lidos por ferramentas de *Cloud Data Warehouse*.

Nesse mesmo sentido, Ali (2022) declara que as os benefícios de modelagens dimensionais como *Star Schema* ainda podem ser verificados no contexto das ferramentas analíticas modernas. Segundo o autor, tais benefícios, como integração dos dados e escalabilidade, ainda podem ser aproveitados na atualidade.

Ainda nessa linha, verifica-se que empresas como Databricks não excluem o *Star Schema* da arquitetura analítica moderna. No artigo *Data Warehousing Modeling Techniques and Their Implementation on the Databricks Lakehouse Platform*, Bhatt e Sekar (2022) mencionam o referido modelo como uma forma fácil e intuiva de organização dos dados para a execução de consultas analíticas. Além disso, mencionam que esse tipo de modelagem é bastante comum em tabelas *Gold* (Ouro), que são tabelas otimizadas para leitura por necessitarem de poucas junções. Porém, alertam que diferentes casos de uso podem exigir modelagens diferentes.

Por outro lado, alguns profissionais não compartilham dessa visão. Brooks (2022a) explica que as vantagens do *Star Schema*, mencionadas pelo próprio Ralph Kimball no primeiro capítulo do livro *The Data Warehouse Toolkit*, como redução de custos, melhora da performance e melhor entendimento dos dados pelo usuário, embora tenham sido válidas no passado, não são mais nos dias de hoje.

Como exemplo, em relação à redução de custos, Brooks (2022b) argumenta que enquanto na década de 1990 o preço do armazenamento era próximo a US$ 12.000 por Gigabyte, atualmente, com o armazenamento em nuvem, esse valor caiu para cerca de US$ 0,23 mensais por Gigabyte. Além disso, o autor afirma que transformar um banco relacional em *Star Schema* demanda tempo dos profissionais (os quais, ao contrário do armazenamento, ainda são caros), além de exigir muito mais junções na execução de consultas. A consequência disso seria aumento de custos e queda da performance.

Com relação ao melhor entendimento dos dados por parte do usuário, Brooks (2022b) defende que um modelo mais desnormalizado, como *One Big Table* (OBT) ,cumpriria melhor essa função, considerando que na época atual muitos trabalhadores do conhecimento não conhecem linguaguens de programação, mas sim planilhas. Dessa forma, juntar dados organizados segundo o *Star Schema* pode ser um desafio para esse público, não contribuindo assim para o entendimento dos dados.

Nessa perspectiva, Fowler (2022), ressalta o alto custo do armazenamento de dados no passado, mas também aponta diferenças no tipo de armazenamento. Enquanto antigamente o tipo de armazenamento era orientado a linhas, atualmente existe o armazenamento orientado a colunas, através de formatos como Apache Parquet. Como esses formatos utilizam algoritmos de compressão para evitar o armazenamento de dados repetidos, o *Star Schema* se tornaria desnecessário para a redução dos custos de armazenamento.

Por fim, no que toca à performance, Fowler (2022) destaca um estudo da Fivetran em que a performance do *Star Schema* e OBT são comparadas através de três *Data Warehouses: Redshift, Snowflake e Big Query*. Os resultados são favoráveis ao schema OBT, o qual apresenta uma vantagem de desempenho de 25% a 30% para o primeiro, 25% para o segundo e 50% para o terceiro.

7 DATA VAULT

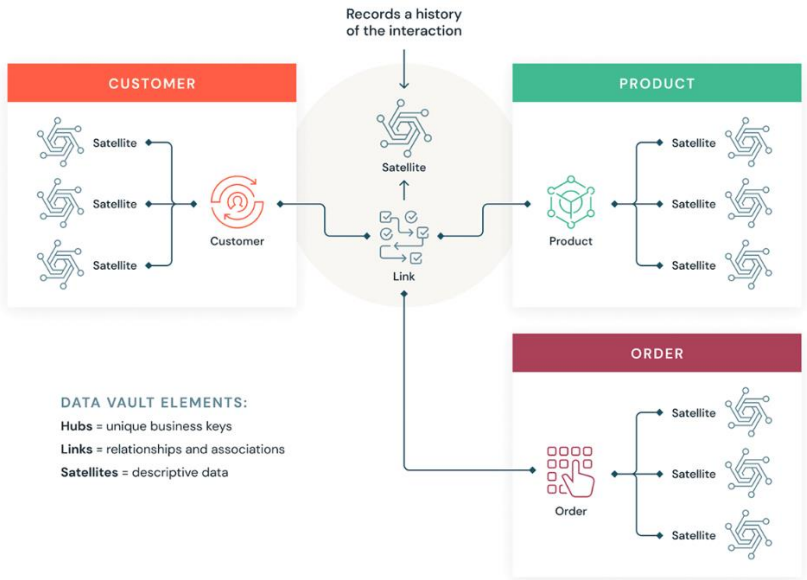
Data Vault é um sistema de Business Intelligence, criado por Dan Linstedt, que engloba temas como modelagem de dados, metodologia, implementação e boas práticas. Seu desenvolvido levou 10 anos, tendo sido finalizado e divulgado ao público nos anos 2000. Nesse sentido, em um primeiro momento, esse sistema foi desenvolvido para uso do Departamento de Defesa dos Estados Unidos, da Agência de Segurança Nacional e da NASA, visando superar questões relacionadas à integração de centenas de fontes de dados, à segurança e ao desempenho (LINSTEDT, 2018).

A primeira versão do modelo, denominada "*Data Vault 1.0*", era mais voltada para bancos de dados relacionais, de forma que a modernização do aparato analítico levou à atualização da técnica através do "*Data Vault 2.0*", o qual é mais focado em plataformas de Big Data (LINSTEDT, 2018). Dessa forma, embora esse seja um tema extenso por abranger diversos aspectos relacionados ao fluxo de grande volume de dados, nessa seção, o escopo recairá sobre a modelagem de dados.

Data Vault possui três conceitos principais: *hubs,links* e *satellites*. Os *hubs* representam os temas centrais de um negócio, como Cliente, Produto e Ordem. Já as tabelas *link* expressam as relações entre os *hubs*. Por fim, *sattelites* são tabelas que contém atributos descritivos dos *hubs* e dos *links* (WHAT IS A DATA VAULT?, [s.d]).

Dessa forma, a ideia é que os links tenham apenas as chaves dos *hubs*, expressando essencialmente o relacionamento entre eles. Esse tipo de configuração dá flexibilidade em relação às mudanças nos requisitos de negócio, especificamente no que toca ao vínculo entre as entidades, uma vez que a mudança do tipo de relacionamento não impacta na arquitetura das tabelas. Além disso, os *hubs* também possuem apenas chaves estrangeiras dos atributos que caracterizam aquele conceito de negócio. Isso significa que tanto os atributos das tabelas *hub* quanto os atributos das tabelas *link* residem nas tabelas *satellites* (FINGER; DE KEUSTER; 2022)

Figura 2. Data Vault



Fonte: What is Data Vault, Databricks, [s.d.]

Samuel (2021) argumenta que esse tipo de arquitetura, além de promover flexibilidade em relação à lógica de funcionamento do negócio, resolve conflitos relacionados a múltiplas fontes de informações para o mesmo item. Como exemplo, ele cita o caso em que o mesmo produto foi registrado com três nomes diferentes por três vendedores. Essa situação poderia ser contornada através de uma tabela *link*, que vincularia os três nomes ao mesmo id.

Porém, ele ressalta que a desvantagem do modelo é que ele não é otimizado nem para escrita, como ocorre em bancos de dados transacionais, e nem para a leitura. Em relação ao último, isso ocorreria porque tal arquitetura exige muitas junções, considerando que todos os atributos são armazenados em tabelas *satellites*, o que geraria lentidão nas consultas. Dessa forma, segundo o autor, o *Data Vault* não seria adequado para fins analíticos, sendo necessário construir uma camada superior com dados organizados de forma mais otimizadas para esse tipo de atividade.

A empresa Databricks também segue nessa mesma linha. Em seu site, ela afirma que a modelagem *Data Vault* se encaixa perfeitamente no paradigma de *lakehouse*, podendo ser

utilizada na arquitetura das tabelas Silver. Essas tabelas seriam o resultado da transformação e limpeza dos dados brutos (chamados de tabelas bronze) de forma a organizá-los conforme os conceitos do negócio. Porém, esse tratamento seria apenas o suficiente para promover uma organização mínima, uma vez que a preparação definitiva para que os dados sejam acessados por uma camada analítica ocorreria nas tabelas *gold*. Sendo assim, a empresa declara que uma das maiores vantagens da técnica reside no processo de ETL, que passa a ser mais flexível e exigir menos refatorações.

Por fim, Finger e De Keuster (2022) declaram que o *Data Vault* contém todos os elementos necessários criar uma solução analítica moderna, implementando adequadamente a visão do negócio em *Data Warehouses* e *Data Marts* (ao qual chamam de Information Marts). Em relação ao modelo de dados, os autores ressaltam sua adaptabilidade, uma vez sua adoção viabiliza o acréscimo de dados de diferentes origens, de novos requisitos de negócio e modelos de informação. Dessa forma, no que se refere ao último, afirmam que comumente utilizam modelagens dimensionais, como *Star Schema* e *Snowflake*. Disso, infere-se que, na visão dos autores, é uma prática comum e adequada as empresas utilizarem o *Data Vault* em camadas intermediárias de processamento de dados analítcos.

8 ONE BIG TABLE

One Big Table (OBT) nada mais é do que o mais alto nível de desnormalização. Nesse tipo de modelagem, junta-se todas as tabelas em uma única. Khan (2022) afirma que, embora tal modelo ofereça uma performance superior em consultas analíticas, principalmente quando os dados são armazenados em formato colunar, sua arquitetura apresenta problemas, os quais estão vinculados à redundância dos dados, o que torna a manutenção complexa.

Rivera (2023), por sua vez, aconselha o leitor a evitar essa técnica o máximo possível, e relaciona diversos desafios dela decorrentes. Um deles seria escalabilidade, ou seja, o aumento do número de registros, uma vez que isso afetaria o desempenho das consultas em virtude do grande número de colunas que essas tabelas tipicamente apresentam. Além disso, a junção de uma grande quantidade de tabelas em uma única aumenta a chances de haver duplicidade de registros, presença excessiva de dados nulos e perda da granularidade da tabela (o que afeta o resultado de agregações). Por fim, ele afirma que as plataformas de nuvem são otimizadas para lidar com grande número de registros, mas perdem performance à medida que o número de colunas cresce demasiadamente.

Por outro, Kaminsky (2022) realizou um experimento em que compara como os modelos *Star Schema* e OBT performam em consultas simples utilizando três ferramentas de warehouse: *RedShift, Snowflake* e *BigQuery*. O resutado foi que o OBT teve um desempenho de 25% a 50% (dependendo da ferramenta) mais rápido do que o *Star Schema*. Porém, apesar do desempenho superior, ele destaca que o modelo vencedor ocupou uma quantidade significativa de espaço no *RedShift*. Enquanto o *Star Schema* ocupou pouco mais de 30 GB,

OBT ocupou cerca de 90 GB. Outro ponto levantado pelo autor seria em relação ao *Star Schema* proporcionar melhor arquitetura do código feito para ETL/ELT e ser mais fácil para os usuários navegarem. Nesse sentido, ele argumenta que os modelos *Star Schema* e OBT podem ser combinados, sendo que o primeiro poderia ser aplicado em uma etapa intermediária (*staging*) à implementação do segundo.

Por fim, para Whiteley (2023), é inegável que as *Wide Tables* (nome alternativo a OBT) possuem um desempenho superior a outros modelos, como *Star Schema* e *Data Vault*. Isso porque, apesar de possuírem muitas colunas, isso não afeta a performance em virtude do armazenamento colunar, no qual apenas as colunas de interesse para a consulta são lidas. Porém, para ele, os modelos dimensionais não devem ser abandonados, uma vez que apesar de as *Wide Tables* possuirem uma performance vantajosa, elas são difíceis de manter. Como exemplo ele cita a mudança na classificação de um produto. Ao invés de necessitar atualizar apenas um registro, será necessário atualizar todos aqueles que possuírem esse produto. Dessa forma, uma alternativa seria utilizar modelagens dimensionais em *Data Warehouses* e *Wide Tables* em *Data Marts*.

9 CONCLUSÃO

A escolha da melhor modelagem de dados para a realização de consultas analíticas no contexto das ferramentas modernas de armazenamento e processamento de dados é desafiadora. Há diversas questões a serem consideradas. Uma delas é o público-alvo dessas informações. Serão apenas colaboradores especialistas do negócio e gerentes, ambos sem expertise tecnológica? Serão analistas de dados? Talvez cientistas de dados? Ou quem sabe todos eles? Nesse caso, será necessária uma modelagem para cada tipo de usuário ou é possível utilizar apenas uma? Qual tipo de modelagem utilizar para cada situação? Qual modelagem é melhor, de forma geral?

O objetivo do presente trabalho não é chegar a uma conclusão definitiva capaz de responder a esses questionamentos de forma categórica. Isso porque cada organização possui suas particularidades, as quais devem ser consideradas para que se consiga encontrar um caminho que consiga equilibrar as vantagens e desvantagens entre as opções escolhidas e, principalmente, gerar valor para os usuários dos dados.

Diante disso, percebe-se que naturalmente não há um consenso no mercado em relação a como os modelos abordados se encaixam no novo paradigma analítico. No entanto, conhecer os diferentes argumentos em relação a seus prós e contras fornece uma percepção mais clara sobre as possibilidades de uso de cada técnica e aprimora a capacidade de elaboração da solução mais equilibrada para o caso concreto.

Dessa forma, pode-se perceber que o *Star Schema*, apesar de ter sido classificado como desnecessário em alguns casos em função da evolução e queda nos custos de

tecnologias de armazenamento de grandes volumes de dados, ainda é reconhecido pelos benefícios relacionados à organização e integração dos dados. Porém, deve-se atentar para questões relacionadas à performance, pois, apesar de necessitar de menos junções em relação a outros modelos, como *Snowflake*, ainda assim não as elimina, o que, dependendo do caso, pode prejudicar o desempenho das consultas.

No que se refere ao *Data Vault*, apesar de já existir não ser uma proposta nova, vem ganhando força atualmente pelo fato de representar bem os conceitos das empresas, bem como proporcionar grande flexibilidade no que se refere à mudança de requisitos do negócio. Porém, nota-se que há uma tendência no sentido de utilizá-lo em camadas intermediárias da arquitetura analítica moderna. Isso porque trata-se de um modelo que não é exatamente otimizado para a leitura, por necessitar de muitas junções, até mais do que o *Star Schema*, cujas tabelas dimensão são desnormalizadas.

Já no que tange ao modelo OBT, o fato de formatos de arquivos colunares, como Parquet, eliminarem a necessidade de leitura de colunas não utilizadas nas consultas e possuírem algoritmos de compressão para evitar o armazenamento de dados repetidos, em combinação com a dispensa de junções, pode dar a impressão de que se trata do modelo ideal para as os *Data Warehouses* dos dias de hoje. Apesar de sua performance em consultas analítica apresentar vantagens em relação às demais metodologias, não se deve desconsiderar a complexidade de manutenção decorrente de sua implementação.

Sendo assim, verifica-se que não há um modelo completo que resolva todas as questões enfrentadas pelos engenheiros de dados nos tempos atuais. Nesse sentido, conclui-se que não se deve interpretar os modelos como soluções excludentes, mas sim como técnicas que podem ser aplicadas em intensidades variáveis conforme a questão a ser enfrentada. Para isso, é fundamental conhecer quais as vantagens e desvantagens de cada um, a fim de se obter um resultado equilibrado entre performance, armazenamento e manutenção.

REFERÊNCIAS

ACID transactions. **REDIS**.Disponível em: https://redis.com/glossary/acid-transactions/ .Acesso em: 2 abr. 2023

ALI, A. Dimensional modeling is relevant... and that's a fact! **Astera**, 2 ago.2022. Disponível em: https://www.astera.com/type/blog/dimensional-modeling-is-relevant/ . Acesso em: 16 abr.2023.

ARMBRUST, M.; GHODSI, A.; XIN, R.; ZAHARIA, M. **Lakehouse: A New Generation of Open Platforms that Unify Data Warehousing and Advanced Analytics**. *In*: 11th Conference on Innovative Data Systems Research, 11 (CIDR 2021), 2021, Evento Virtual, Disponível em: https://cs.stanford.edu/~matei/papers/2021/cidr_lakehouse.pdf. Acesso em 20.abr.2023.

ARMBRUST,M.; GHODSI, A.; LORICA,B.; XIN, R., ZAHARIA, M. What is a lakehouse? **Databricks**. 30 jan.2020. Disponível em: https://www.databricks.com/blog/2020/01/30/what-is-a-data-lakehouse.html . Acesso em 2 abr.2023.

BHATT, S.; SEKAR, D. Data warehousing modeling techniques and their implementation on the databricks lakehouse platform. **Databricks**. 24 jun.2022. Disponível em: https://bit.ly/41QKRom . Acesso em: 16 abr. 2023.

BROOKS, C. The star schema is obsolete, reduced cost, and better performance. **Perficient**, 13 set. 2022b. Disponível em: https://bit.ly/44g3vb6 . Acesso em: 16 abr. 2023.

BROOKS, C. The star schema is obsolete. **Perficient**, 13 set.2022. Disponível em: https://blogs.perficient.com/2022/09/06/the-star-schema-is-obsolete/ . Acesso em: 16 abr. 2023a.

ETL. IBM, 2023. Disponível em: https://www.ibm.com/topics/etl . Acesso em: 26 mar. 2023.

FINGER,M. , DE KEUSTER, J. **A Modern Data Analytics Platform on Azure with Data Vault 2.0**. Microsoft, 23 nov. 2022. Disponível em: https://bit.ly/3LGJkvF .Acesso em: 21 abr. 2023.

FOWLER, D. Kimball in the context of the modern data warehouse: what's worth keeping, and what's not [vídeo] . **dbt**, 15 dez. 2020. Disponível em: https://www.youtube.com/watch?v=3OcS2TMXELU&t=383s . Acesso em: 16 abr. 2023.

GOLFARELLI, M.; RIZZI, S.. Introdution to data warehousing. **Data Warehouse Design: Modern Principles and Methodologies**. Primeira edição.Nova York. McGraw-Hill Companies. 2009. p.1-29.

HOW organizations use lakehouses to get more value from data. **Oracle**, 2022. Disponível em: https://www.oracle.com/a/ocom/docs/big-data/big-data-evolution.pdf. Acesso em 29 abr.2023.

INMON, B.; LEVINS, M. Evolution to the data lakehouse**. Databricks**, 19 mai.2021. Disponível em: https://www.databricks.com/blog/2021/05/19/evolution-to-the-data-lakehouse.html .Acesso em: 30 mar. 2023.

INMON, B.; LEVINS, M..; SRIVASTAVA, R. Evolution to the data lakehouse. *In*: INMON, B.; LEVINS, M..; SRIVASTAVA, R. **Building the Data Lakehouse**. Primeira edição. .Technics Publications: Basking Ridge, NJ, 2021, p 5-28.

INMON, W. H.; STRAUSS, D.; NEUSHLOSS, G. A brief history of data warehousing and fi rst-generation data warehouses . *In*: INMON, W. H.; STRAUSS, D.; NEUSHLOSS, G. **DW 2.0 : The Architecture for the Next Generation of Data Warehousing**. San Francisco: Elsevier Science, 2010, p.3-21.

KAMINSKY, M. Star schema vs. obt for data warehouse performance**. Fivetran**, 16 ago. 2022. Disponível em: https://www.fivetran.com/blog/star-schema-vs-obt . Acesso em: 16 abr. 2023.

KHAN, F. A. What's the best data warehouse architecture for reporting? Astera, 17 nov. 2022. Disponível em: https://shorturl.at/rIJX7 . Acesso em: 21 abr. 2023.

KIMBALL, R., ROSS, M. Dimensional Modeling Primer. *In:* KIMBALL, R. , ROSS, M. **The data warehouse toolkit:** The Complete Guide to Dimensional Modeling ; Second Edition., Canada, John Wiley & Sons, 2002, p.1-24.

LIN, P. THE big data-iot relationship: how they help each other. **Spiceworks**, 2023. Disponível em: https://shorturl.at/mxyM6 . Acesso em 29 de abr.2023.

LINSTEDT, D. Defining data vault 1.0 and 2.0 for business. **Linkedin**, 19 jul. 2018. Disponível em: https://shorturl.at/eow37 . Acesso em: 17 abr. 2023.

MOHD, A. Evolution of big data and tools for big data analytics. **Journal of Interdisciplinary Cycle Research,** *[s.l]*, v.12. n.10, p.309-316, out.2020. ISSN 0022-1945. Disponível em https://bit.ly/420afI6. Acesso em 29 abr.2023.

NAIR, V. C. Do we need dimensional data warehouses in the modern data stack? **Medium**. 14 abr.2022**.** Disponível em: https://bit.ly/3HtwsGz . Acesso em: 16 abr. 2023.

NAMBIAR, A.; MUNDRA, D. An Overview of Data Warehouse and Data Lake in Modern Enterprise Data Management**. Big Data and Cognitive Computing**. India, v.6 ,n.4, p.132, nov. 2022. Disponível em https://www.mdpi.com/2504-2289/6/4/132 .Acesso em 26/04/2023. DOI 10.3390/bdcc6040132.

POWER, D. J. Supporting Business Decision Making .**Decision support systems : concepts and resources for managers**. Primeira edição. Westport. Quorum Books. 2002. p.13-16

POWER, D. Understanding Data-Driven Decision Support Systems. **Information Systems Management,** [s.l.].v. 25, n.2, p. 149-154, mar.2008. Disponível em: https://bit.ly/3AuNmkp. Acesso em 27.abr.2023. DOI 10.1080/10580530801941124.

RIVERA, S. Data modeling best practices for data & analytics engineers. **ThoughSpot**, 23 mar.2023. Disponível em: https://bit.ly/3HkZrwo . Acesso em: 21 abr. 2023.

SAMUEL, R. Data vault modeling concepts through real-world examples. **Medium**, 10 fev.2021. Disponível em: https://shorturl.at/DIT89 . Acesso em: 18 abr. 2023.

TERRA, J.What is web 1.0, web 2.0, and web 3.0? definition, difference & similarities. **Simplilearn**, 2023. Disponível em: https://shorturl.at/dglFH . Acesso em 29 abr.2023.

WHAT is a data vault? **Databricks.** Disponível em: https://www.databricks.com/glossary/data-vault .Acesso em 17 de abr. 2023.

WHAT is a data warehouse? . Databricks. Disponível em: https://www.databricks.com/glossary/data-warehouse . Acesso em: 26 mar. 2023.

WHITE, T. Meet Hadoop. **Hadoop: the definitive guide: storage and analysis at internet scale.** 4 ed rev e atual. Sebastopol, CA : O'Reilly Media, 2015, p.12-14.

WHITELEY, S. Is kimball still relevant in the modern data warehouse? advancing analytics**, 17 jun. 2019.** Disponível em: https://bit.ly/41GE9kR . Acesso em: 22 abr. 2023.

ZAMAN, M. ; EINI, R.; ZOHRABI, N.; ABDELWAHED, S. A Decision Support System for Cyber Physical Systems under Disruptive Events: Smart Building Application. *In:* **IEEE INTERNATIONAL SMART CITIES CONFERENCE (ISC2),**8,2022,Paphos. IEEE International Smart Cities Conference (ISC2). Paphos,2022. Disponível em https://bit.ly/3AvGt26. Acesso em 27.abr.2023. DOI 10.1109/ISC255366.2022.9922493.