

Data Warehouse (Fundamentos, Arquitetura e Modelagem)

Fundamentos de um Data Warehouse

Data Warehouse é uma *“coleção de dados orientada por assunto, não volátil e variável com o tempo, utilizada para tomadas de decisões”*(**William H. Inmon**), além disso podemos definir Data Warehouse como *“Uma cópia de dados transacionais estruturada especificamente para consultas e análise”* (**Ralph Kimball**).

- **Orientado a assunto:** é orientado a assunto porque ele é projetado para armazenar dados organizados em torno de temas específicos ou assuntos de interesse para uma organização. Em vez de organizar os dados com base em como são utilizados operacionalmente (como em um banco de dados transacional), um Data Warehouse organiza os dados em torno de assuntos ou temas relevantes para análises e tomada de decisão.

Essa abordagem facilita a análise de dados, pois os dados relacionados a um determinado assunto ou área de interesse são armazenados juntos, independentemente de sua origem operacional. Isso permite que os usuários realizem consultas complexas que abrangem várias fontes de dados de maneira mais eficiente e eficaz. Além disso, a orientação a assunto facilita a compreensão e interpretação dos dados para os usuários finais, pois os dados são organizados de acordo com os conceitos e processos que são familiares para eles em seu contexto de negócios.

- **Não volátil:** Porque os dados armazenados nele geralmente não são atualizados com a mesma frequência que em um sistema transacional. Em um ambiente transacional, os dados estão em constante mudança devido às operações de inserção, atualização e exclusão realizadas pelas transações.

No entanto, em um Data Warehouse, os dados tendem a ser carregados em lotes em intervalos regulares, geralmente de fontes operacionais, e depois são armazenados de forma estática para fins de análise. Isso significa que uma vez que os dados são carregados no Data Warehouse, eles geralmente não são atualizados ou modificados com a mesma frequência que em um sistema transacional.

Essa natureza não volátil dos dados em um Data Warehouse facilita a análise histórica e a geração de relatórios ao longo do tempo, pois os dados permanecem consistentes e estáveis, permitindo que os usuários façam comparações e análises históricas de forma confiável. Além disso, essa característica ajuda a preservar a integridade dos dados, pois eles

não estão sujeitos a mudanças frequentes que poderiam causar inconsistências ou erros nos relatórios e análises.

- **Variável com o tempo:** porque ele armazena dados históricos e permite análises ao longo do tempo. Essa característica é fundamental para permitir que as organizações analisem tendências, padrões e comportamentos ao longo de diferentes períodos de tempo.

Existem várias razões pelas quais um Data Warehouse é variável com o tempo:

1. **Histórico de dados:** Um Data Warehouse geralmente mantém um histórico de dados, permitindo que os usuários analisem como os dados evoluíram ao longo do tempo. Isso é crucial para entender tendências passadas e prever padrões futuros.
2. **Dados temporais:** Muitos Data Warehouses incluem informações de tempo, como carimbos de data/hora, para permitir análises temporais. Isso permite que os usuários realizem análises comparativas e identifiquem mudanças ao longo do tempo.
3. **Atualizações incrementais:** Embora os Data Warehouses sejam geralmente não voláteis no sentido de que os dados não são atualizados com a mesma frequência que em sistemas transacionais, eles podem ser atualizados periodicamente com novos dados. Isso permite que o Data Warehouse reflita mudanças contínuas nos dados operacionais.
4. **Adição de novas fontes de dados:** À medida que uma organização evolui e novas fontes de dados se tornam disponíveis, um Data Warehouse pode ser expandido para incluir essas novas fontes. Isso significa que o escopo do Data Warehouse pode mudar ao longo do tempo para acomodar as necessidades de análise em evolução da organização.

Portanto, a variabilidade com o tempo é uma característica essencial de um Data Warehouse, permitindo análises históricas e longitudinais que são críticas para a tomada de decisões informadas e o planejamento estratégico

- **Cópia de dados (Integrado):** Um Data Warehouse é integrado porque ele combina dados de várias fontes diferentes e os organiza de forma coesa e unificada para facilitar a análise e a tomada de decisões. Existem algumas razões pelas quais um Data Warehouse é considerado integrado:
 1. **Consolidação de dados:** Um Data Warehouse integra dados de várias fontes, como sistemas transacionais, bancos de dados operacionais, planilhas e arquivos de texto, em um único local. Isso elimina a

necessidade de os usuários acessarem várias fontes de dados dispersas e simplifica o processo de análise.

2. **Padronização de formatos e estruturas:** Os dados de diferentes fontes podem estar em formatos e estruturas diferentes. Um Data Warehouse integra esses dados e os transforma em um formato consistente e uniforme, facilitando a análise e a comparação.
3. **Eliminação de redundância:** Ao integrar dados de várias fontes, um Data Warehouse ajuda a eliminar redundâncias e inconsistências nos dados. Isso garante que os usuários tenham uma única fonte confiável de dados para análise e relatórios.
4. **Acesso centralizado:** Um Data Warehouse fornece um ponto centralizado de acesso aos dados para os usuários finais. Isso simplifica o processo de obtenção de dados para análise e permite que os usuários acessem informações relevantes de forma rápida e eficiente.
5. **Metadados:** Um Data Warehouse geralmente inclui metadados detalhados sobre os dados armazenados, incluindo sua origem, significado e estrutura. Isso ajuda os usuários a entenderem melhor os dados e a garantir sua integridade e qualidade.

Em resumo, a integração de dados em um Data Warehouse é fundamental para garantir que os usuários tenham acesso a uma fonte única e confiável de dados para análise e tomada de decisões. Isso simplifica o processo de análise e permite que as organizações aproveitem ao máximo seus dados para obter insights valiosos.

Arquitetura de um Data Warehouse

- **Estrutura independente:** A arquitetura de um Data Warehouse com uma estrutura independente refere-se a um design que separa claramente as camadas de extração, transformação e carregamento (ETL) dos dados das camadas de armazenamento e acesso aos dados. Esse tipo de arquitetura é comumente conhecido como arquitetura de Data Warehouse independente de estrutura ou "Data Warehouse com abordagem de armazenamento puro".

Aqui estão os principais componentes dessa arquitetura:

1. **Camada de extração (Extract):** Nesta camada, os dados são extraídos de várias fontes de dados, como bancos de dados transacionais, sistemas legados, arquivos de texto, planilhas, etc. Esses dados são extraídos em sua forma bruta, sem qualquer modificação.
2. **Camada de transformação (Transform):** Os dados extraídos são então transformados nesta camada. Isso inclui limpeza de dados, padronização de formatos, enriquecimento com dados adicionais, agregação e outras manipulações necessárias para preparar os dados para análise.

3. **Camada de carregamento (Load):** Após a transformação, os dados são carregados na camada de armazenamento do Data Warehouse. Existem diferentes abordagens para o carregamento de dados, como carregamento em lote (batch) ou carregamento em tempo real (real-time), dependendo dos requisitos e das necessidades da organização.
4. **Camada de armazenamento (Storage):** Nesta camada, os dados transformados são armazenados de forma organizada e estruturada. Geralmente, os dados são armazenados em um esquema dimensional, como estrela ou floco de neve, otimizado para consultas analíticas. Esta camada pode incluir tabelas de fatos (contendo métricas) e tabelas de dimensão (contendo atributos descritivos).
5. **Camada de acesso (Access):** Por fim, os usuários finais acessam os dados armazenados na camada de armazenamento para realizar análises, criar relatórios e obter insights. Isso pode ser feito por meio de ferramentas de Business Intelligence (BI), consultas SQL, dashboards interativos e outras interfaces de análise.

A arquitetura de Data Warehouse com estrutura independente oferece flexibilidade e escalabilidade, pois separa claramente as preocupações de extração, transformação e carregamento dos dados das preocupações de armazenamento e acesso aos dados. Isso permite que as organizações alterem ou atualizem as fontes de dados, os processos de ETL ou as ferramentas de acesso aos dados sem afetar as outras partes do sistema.

- **Estrutura de Data Marte dependente:** A estrutura de Data Mart dependente refere-se a uma abordagem na qual os Data Marts são criados para atender a necessidades específicas de departamentos ou áreas funcionais da organização, e esses Data Marts dependem diretamente de um Data Warehouse centralizado.

Aqui estão os principais componentes dessa arquitetura:

1. **Data Warehouse Centralizado:** Na estrutura de Data Mart dependente, existe um Data Warehouse centralizado que serve como repositório principal para todos os dados corporativos. Esse Data Warehouse é responsável por integrar dados de diversas fontes e manter uma única fonte de verdade para toda a organização.
2. **Data Marts Departamentais:** Os Data Marts são subconjuntos do Data Warehouse e são criados para atender às necessidades específicas de departamentos ou áreas funcionais da organização, como vendas, marketing, finanças, recursos humanos, etc. Cada Data Mart contém um conjunto de dados otimizado e modelado de acordo com os requisitos analíticos da área funcional correspondente.
3. **Camada de Extração e Transformação (ETL):** A camada de ETL é responsável por extrair dados do Data Warehouse centralizado e

transformá-los de acordo com os requisitos do Data Mart específico. Isso pode envolver agregação de dados, filtragem, limpeza e outras transformações necessárias para atender às necessidades analíticas do departamento ou área funcional.

4. **Camada de Armazenamento do Data Mart:** Cada Data Mart possui sua própria camada de armazenamento, onde os dados transformados são armazenados de forma otimizada para consultas analíticas. Os dados podem ser organizados em um esquema dimensional, como estrela ou floco de neve, para facilitar a análise e o acesso aos dados.
5. **Camada de Acesso:** Os usuários finais acessam os dados armazenados nos Data Marts por meio de ferramentas de análise, consultas SQL, relatórios e dashboards. Essas ferramentas permitem que os usuários realizem análises específicas para sua área funcional, utilizando os dados do Data Mart correspondente.

A estrutura de Data Mart dependente oferece benefícios, como maior agilidade e capacidade de resposta às necessidades específicas de cada área funcional da organização. No entanto, também pode resultar em redundância de dados e gerenciamento mais complexo, pois cada Data Mart requer sua própria camada de armazenamento e processo de ETL.

- **Estrutura de barramento:** A estrutura de barramento em um ambiente de Data Warehouse refere-se a uma arquitetura que utiliza um barramento de serviços para facilitar a integração de sistemas e a comunicação entre os componentes do Data Warehouse. Essa abordagem é conhecida como arquitetura orientada a serviços (SOA) e é comumente aplicada em ambientes corporativos para promover a flexibilidade, reutilização e interoperabilidade dos sistemas.

Aqui estão os principais componentes e características de uma estrutura de barramento em um ambiente de Data Warehouse:

1. **Barramento de Serviços (Enterprise Service Bus - ESB):** O barramento de serviços é uma infraestrutura de software que atua como um intermediário entre os sistemas e aplicativos, facilitando a comunicação e a troca de dados entre eles. Ele fornece recursos para roteamento, transformação, segurança e monitoramento das mensagens que fluem através dele.
2. **Serviços de Integração:** Os serviços de integração são componentes modulares que oferecem funcionalidades específicas para facilitar a integração entre os sistemas. Isso pode incluir serviços para integração de dados (ETL), transformação de mensagens, orquestração de processos, enriquecimento de dados e muito mais.

3. **Camadas de Acesso aos Dados:** As camadas de acesso aos dados fornecem interfaces para acessar e manipular os dados armazenados no Data Warehouse. Isso pode incluir serviços para consulta, inserção, atualização e exclusão de dados, bem como serviços para gerenciamento de metadados e segurança.
4. **Serviços de Análise e Relatórios:** Além dos serviços de integração e acesso aos dados, a estrutura de barramento também pode incluir serviços especializados para análise e geração de relatórios. Isso pode incluir serviços para processamento analítico online (OLAP), geração de relatórios ad hoc, visualização de dados e muito mais.
5. **Segurança e Monitoramento:** A estrutura de barramento geralmente inclui recursos para garantir a segurança e o monitoramento das operações realizadas nos sistemas. Isso pode incluir serviços para autenticação, autorização, criptografia, auditoria e monitoramento em tempo real do desempenho e da integridade dos sistemas.

Em resumo, uma estrutura de barramento em um ambiente de Data Warehouse oferece uma abordagem flexível e modular para integrar sistemas, acessar dados e fornecer serviços de análise e relatórios. Isso permite que as organizações construam soluções de Data Warehouse altamente interoperáveis, escaláveis e seguras.

Modelagem de um Data Warehouse

- **Tabela Fato:** Uma tabela fato em um Data Warehouse é uma tabela central que contém métricas e medidas quantitativas sobre um processo de negócios, evento ou atividade específica. Essas tabelas são essenciais para análises multidimensionais e são uma parte fundamental da modelagem dimensional, uma técnica amplamente utilizada para organizar dados em um Data Warehouse.

Aqui estão algumas características importantes de uma tabela fato:

1. **Métricas Quantitativas:** Uma tabela fato contém as métricas numéricas que são medidas e analisadas no contexto do processo de negócios. Essas métricas podem incluir valores monetários, quantidades físicas, contagens, tempos, etc.
2. **Chaves Estrangeiras:** Além das métricas, uma tabela fato contém chaves estrangeiras que se relacionam às tabelas de dimensão. Essas chaves estrangeiras são usadas para vincular a tabela fato às dimensões relevantes, permitindo a análise dos dados sob diferentes perspectivas.
3. **Grãos de Granularidade:** Uma tabela fato pode ter diferentes granularidades, dependendo dos requisitos analíticos. Por exemplo, uma tabela fato de vendas pode ter uma granularidade diária, onde cada linha

representa as vendas de um determinado produto em um determinado dia, ou pode ter uma granularidade mais fina, como por hora ou por transação.

4. **Registros Aditivos:** Os dados em uma tabela fato são normalmente aditivos, o que significa que eles podem ser somados ou agregados ao longo de diferentes dimensões. Isso permite que os usuários analisem e comparem as métricas em diferentes níveis de detalhe.
5. **Dados de Tempo:** Muitas vezes, uma tabela fato inclui colunas de tempo, como data de venda, data de entrega, data de pedido, etc. Isso permite que os usuários analisem as métricas ao longo do tempo e identifiquem tendências e padrões temporais.

Exemplos comuns de tabelas fato incluem tabelas de vendas, tabelas de pedidos, tabelas de estoque, tabelas de desempenho de campanha, entre outros. Essas tabelas são projetadas para armazenar dados de transações ou eventos que são relevantes para a análise de negócios e são a espinha dorsal de muitos modelos dimensionais em Data Warehouses.

- **Tabela dimensão:** Uma tabela de dimensão em um Data Warehouse é uma tabela que contém atributos descritivos que fornecem contexto e detalhes adicionais sobre os dados armazenados na tabela fato. As tabelas de dimensão são essenciais para a análise multidimensional e fazem parte fundamental da modelagem dimensional, que é uma técnica comum para organizar dados em um Data Warehouse.

Aqui estão algumas características importantes de uma tabela de dimensão:

1. **Atributos Descritivos:** Uma tabela de dimensão contém atributos que descrevem as características dos dados na tabela fato. Esses atributos podem incluir informações como produtos, clientes, localizações geográficas, datas, categorias, vendedores, entre outros.
2. **Chave Primária:** Cada linha em uma tabela de dimensão geralmente possui uma chave primária que a identifica exclusivamente. Essa chave é frequentemente referenciada na tabela fato por meio de chaves estrangeiras para estabelecer a relação entre os dados na tabela fato e os atributos na tabela de dimensão.
3. **Grãos de Granularidade:** Assim como as tabelas fato, as tabelas de dimensão também podem ter diferentes granularidades, dependendo dos requisitos analíticos. Por exemplo, uma tabela de tempo pode ter granularidade diária, mensal ou anual, enquanto uma tabela de produto pode ter granularidade por SKU (Stock Keeping Unit) ou por categoria.
4. **Atributos Hierárquicos:** Em algumas situações, os atributos em uma tabela de dimensão podem ser hierárquicos, o que significa que eles têm

uma estrutura de pai-filho. Por exemplo, uma dimensão de localização pode ter uma hierarquia que vai de país para estado, estado para cidade, etc.

5. **Dados Estáticos:** Os dados em uma tabela de dimensão geralmente são estáticos e não mudam com tanta frequência quanto os dados em uma tabela fato. No entanto, é possível que os atributos em uma tabela de dimensão sejam atualizados periodicamente para refletir alterações no ambiente de negócios.

Exemplos comuns de tabelas de dimensão incluem tabelas de produto, tabelas de cliente, tabelas de tempo (calendário), tabelas de localização geográfica, tabelas de categoria, entre outras. Essas tabelas fornecem o contexto necessário para a análise dos dados na tabela fato e permitem que os usuários explorem e compreendam os dados de diferentes perspectivas.

- **Granularidade:** Granularidade em um contexto de Data Warehouse refere-se ao nível de detalhe ou refinamento dos dados armazenados em uma tabela fato ou dimensão. É essencial entender e definir a granularidade ao modelar um Data Warehouse, pois isso afeta diretamente a precisão das análises e o desempenho das consultas.

Aqui estão alguns pontos importantes sobre granularidade:

1. **Nível de Detalhe:** A granularidade determina o quão detalhados são os dados em uma tabela fato. Por exemplo, se estamos lidando com vendas, a granularidade pode ser por transação individual, por dia, por mês ou por ano. Quanto mais granular for a granularidade, mais detalhados serão os dados, mas também podem ser mais numerosos e exigir mais recursos para armazenamento e processamento.
2. **Nível de Sumarização:** A granularidade também afeta o nível de sumarização ou agregação dos dados. Se os dados são altamente granulares, é possível obter insights detalhados sobre transações individuais ou eventos específicos. Por outro lado, se os dados são agregados em níveis mais altos (por exemplo, por mês ou por ano), os insights serão mais abrangentes e gerais.
3. **Requisitos de Negócios:** A escolha da granularidade deve ser guiada pelos requisitos analíticos e de negócios da organização. Por exemplo, se os analistas precisam de insights detalhados sobre o desempenho de vendas diárias para tomar decisões operacionais, uma granularidade diária seria apropriada. Se a alta administração está interessada apenas em tendências de vendas trimestrais para fins estratégicos, uma granularidade trimestral pode ser suficiente.
4. **Desempenho das Consultas:** A granularidade afeta o desempenho das consultas no Data Warehouse. Consultas que lidam com dados altamente

granulares podem exigir mais recursos de processamento e tempo de resposta do que consultas que lidam com dados agregados em níveis mais altos. Portanto, é importante encontrar um equilíbrio entre granularidade e desempenho.

5. **Flexibilidade:** Uma modelagem flexível pode envolver a criação de tabelas fato e dimensões em diferentes granularidades para atender a uma variedade de requisitos analíticos. Isso permite que os usuários realizem análises em diferentes níveis de detalhe, dependendo das necessidades específicas de suas análises.

Em resumo, a granularidade desempenha um papel crucial na modelagem de um Data Warehouse, pois determina o nível de detalhe dos dados e influencia diretamente a precisão das análises e o desempenho das consultas. É importante escolher a granularidade adequada com base nos requisitos analíticos e de negócios da organização.

- **Esquema Estrela:** O esquema estrela é um dos esquemas mais comuns usados na modelagem de dados em um Data Warehouse. Ele é projetado para fornecer uma estrutura simples e eficiente para consultas analíticas, facilitando a análise multidimensional e o acesso rápido aos dados. Aqui estão os principais componentes de um esquema estrela:

1. **Tabela Fato (Fact Table):** A tabela fato é o ponto central do esquema estrela. Ela contém métricas e medidas quantitativas que representam os eventos de negócios ou transações que estão sendo analisados. Cada linha na tabela fato corresponde a uma instância de evento ou transação e contém chaves estrangeiras que se relacionam às tabelas de dimensão.
2. **Tabelas de Dimensão (Dimension Tables):** As tabelas de dimensão contêm atributos descritivos que fornecem contexto e detalhes adicionais sobre os dados na tabela fato. Esses atributos podem incluir informações sobre produtos, clientes, tempo, localização, etc. Cada tabela de dimensão é ligada à tabela fato através de chaves estrangeiras.
3. **Chaves Estrangeiras (Foreign Keys):** As chaves estrangeiras são colunas nas tabelas de dimensão que são referenciadas pelas tabelas fato para estabelecer a relação entre os dados na tabela fato e os atributos na tabela de dimensão. Geralmente, uma tabela fato contém várias chaves estrangeiras que se relacionam com diferentes dimensões.
4. **Relacionamentos Estrela:** No esquema estrela, o relacionamento entre a tabela fato e as tabelas de dimensão forma uma estrutura de estrela, onde a tabela fato é o "fato central" e as tabelas de dimensão são os "braços" da estrela. Esse design simplificado facilita a navegação e a análise dos dados, permitindo que os usuários explorem as métricas em diferentes dimensões.

5. **Simplicidade e Desempenho:** Uma das principais vantagens do esquema estrela é sua simplicidade e facilidade de compreensão. Além disso, ele é altamente otimizado para consultas analíticas, o que resulta em um melhor desempenho de consulta em comparação com outros esquemas mais complexos, como o esquema floco de neve.

O esquema estrela é amplamente utilizado em ambientes de Data Warehouse devido à sua simplicidade, desempenho e capacidade de suportar análises multidimensionais. Ele fornece uma estrutura flexível e eficiente para armazenar e acessar dados analíticos, tornando-o uma escolha popular para implementações de Data Warehouse.

- **Esquema Floco de Neve:** O esquema floco de neve é uma extensão do esquema estrela utilizado na modelagem de dados em um Data Warehouse. Enquanto o esquema estrela possui uma estrutura simples com uma tabela fato central e tabelas de dimensão conectadas diretamente a ela, o esquema floco de neve adiciona uma camada de normalização às tabelas de dimensão.

Aqui estão os principais componentes e características do esquema floco de neve:

1. **Tabela Fato (Fact Table):** Assim como no esquema estrela, a tabela fato é o ponto central do esquema floco de neve e contém métricas e medidas quantitativas sobre eventos de negócios ou transações.
2. **Tabelas de Dimensão (Dimension Tables):** As tabelas de dimensão contêm atributos descritivos que fornecem contexto e detalhes adicionais sobre os dados na tabela fato. No esquema floco de neve, essas tabelas de dimensão são normalizadas, o que significa que são divididas em várias tabelas relacionadas.
3. **Relacionamentos Normalizados:** No esquema floco de neve, as tabelas de dimensão são normalizadas para reduzir a redundância de dados e melhorar a eficiência de armazenamento. Isso significa que os atributos em uma tabela de dimensão podem ser divididos em subgrupos lógicos e armazenados em tabelas separadas, com chaves estrangeiras estabelecendo relacionamentos entre elas.
4. **Normalização Hierárquica:** Uma característica comum do esquema floco de neve é a normalização hierárquica, na qual os atributos hierárquicos em uma tabela de dimensão são divididos em tabelas separadas para cada nível da hierarquia. Por exemplo, em uma dimensão de tempo, os atributos de ano, trimestre, mês e dia podem ser armazenados em tabelas separadas.
5. **Complexidade e Flexibilidade:** Enquanto a normalização no esquema floco de neve reduz a redundância de dados e melhora a eficiência de armazenamento, ela também adiciona complexidade ao modelo de

dados. No entanto, essa abordagem pode ser benéfica em cenários onde a hierarquia dos atributos é importante e quando há necessidade de compartilhamento de dimensões entre múltiplas tabelas fato.

Em resumo, o esquema floco de neve é uma extensão do esquema estrela que adiciona normalização às tabelas de dimensão, dividindo-as em subgrupos lógicos de atributos relacionados. Essa abordagem pode oferecer vantagens em termos de eficiência de armazenamento e hierarquia de dados, mas também adiciona complexidade ao modelo de dados. A escolha entre o esquema estrela e o esquema floco de neve depende dos requisitos específicos do projeto e das necessidades analíticas da organização.

EXEMPLO

Exemplo simplificado de um diagrama de um Data Warehouse com dois Data Marts: um para vendas e outro para marketing. Vou usar a notação básica de modelagem dimensional para representar as tabelas fato e dimensão.

Data Warehouse Centralizado:

Tabela Fato de Vendas (Sales_Fact):

Chave da Data (Date_Key)

Chave do Produto (Product_Key)

Chave do Cliente (Customer_Key)

Quantidade Vendida (Quantity Sold)

Valor Total de Vendas (Total Sales Value)

Tabela de Dimensão de Data (Date_Dimension):

Data (Date)

Ano (Year)

Mês (Month)

Dia (Day)

Tabela de Dimensão de Produto (Product_Dimension):

Produto ID (Product_ID)

Nome do Produto (Product Name)

Categoria do Produto (Product Category)

Preço Unitário (Unit Price)

Tabela de Dimensão de Cliente (Customer_Dimension):

Cliente ID (Customer_ID)

Nome do Cliente (Customer Name)

Localização (Location)

Categoria de Cliente (Customer Category)

Data Mart de Vendas:

Tabela Fato de Vendas (Sales_Fact):

(Mesmas colunas da tabela fato no Data Warehouse centralizado)

Tabela de Dimensão de Produto (Product_Dimension):

(Mesmas colunas da tabela de dimensão de produto no Data Warehouse centralizado)

Tabela de Dimensão de Data (Date_Dimension):

(Mesmas colunas da tabela de dimensão de data no Data Warehouse centralizado)

Tabela de Dimensão de Cliente (Customer_Dimension):

(Mesmas colunas da tabela de dimensão de cliente no Data Warehouse centralizado)

Data Mart de Marketing:

Tabela Fato de Campanha de Marketing (Marketing_Campaign_Fact):

Chave da Data (Date_Key)

Chave da Campanha (Campaign_Key)

Custo da Campanha (Campaign Cost)

Número de Leads (Number of Leads)

Número de Conversões (Number of Conversions)

Tabela de Dimensão de Campanha de Marketing (Campaign_Dimension):

Campanha ID (Campaign_ID)

Nome da Campanha (Campaign Name)

Tipo de Campanha (Campaign Type)

Canal de Marketing (Marketing Channel)

Tabela de Dimensão de Data (Date_Dimension):

(Mesmas colunas da tabela de dimensão de data no Data Warehouse centralizado)

Este é um exemplo simplificado que ilustra como um Data Warehouse centralizado pode ser conectado a dois Data Marts separados, cada um atendendo a necessidades analíticas específicas. Cada Data Mart contém suas próprias tabelas fato e dimensão, que são uma subconjunto das tabelas do Data Warehouse centralizado.