

An abstract graphic of a circuit board pattern in a lighter blue shade, located in the top-left corner of the slide.

Banco de Dados Aplicado

Cristiano Santos

cristiano.santos@amf.edu.br

An abstract graphic of a circuit board pattern in a lighter blue shade, located in the bottom-right corner of the slide.

The image features a white background with decorative blue lines in the corners. The top-left corner has a vertical line that steps to the right. The top-right corner has a horizontal line that steps to the left. The bottom-left corner has a horizontal line that steps to the right. The bottom-right corner has a horizontal line that steps to the left.

Data Warehouse (DW)

Perspectiva histórica e motivação

- Anos 1970: criação do modelo relacional e adoção da linguagem SQL como padrão para os SGBDs
 - Facilitou a manipulação, manutenção e recuperação dos dados
 - Popularizou uso de BDs em organizações do mundo todo

Perspectiva histórica e motivação

- Os SGBDs são amplamente usados para documentar operações diárias, ou seja, **transações rotineiras**
- Essas **transações, geralmente**, fazem pequenas **alterações nos dados**
- Os **SGBDs precisam lidar** com um **grande número de transações** desse tipo de **forma eficiente**

Perspectiva histórica e motivação

- Os SGBDs têm sido extensivamente otimizados para lidar com as aplicações de **processamento de transações online** (OLTP) \Rightarrow **bancos de dados transacionais**

Perspectiva histórica e motivação

Com o aumento da competitividade entre empresas e da busca por melhorias nos processos de produção, intensificou-se a necessidade de se obter **visões analíticas e gerenciais** sobre os dados armazenados nos BDs

Perspectiva histórica e motivação

- **Problema:** os BDs transacionais sozinhos não se mostravam adequados para prover visões de apoio à tomada de decisões
- **Consequência:** desenvolvimento de **aplicações de apoio à decisão** que **analisam e exploram dados atuais e históricos, identificando tendências e criando resumos**
- Fabricantes de SGBD tradicionais já vêm adicionando a seus produtos funcionalidades especiais de apoio à tomada de decisão
- A SQL vem sendo estendida para suportar consultas complexas

Apoio à tomada de decisão

- **Problemas do uso de sistemas SQL tradicionais para a realização de consultas de apoio à tomada de decisão:**
 - Cláusula **WHERE** de consultas com muitos **ANDs** e **ORs** ⇒
 - **processamento ineficiente**
- Aplicações exigem o uso de funções estatísticas não suportadas pelo padrão SQL
- **Muitas consultas envolvem condições ao longo do tempo ou exigem agregação ao longo do tempo ⇒ padrão SQL oferece suporte deficiente para isso**

Apoio à tomada de decisão

- ▶ **A tomada de decisão organizacional exige uma visão mais abrangente** de todos os aspectos de uma empresa (CRM, ERP ...)
- ▶ **Essa visão geralmente é construída a partir de dados extraídos de vários BDs mantidos por diferentes unidades empresariais**, com informações históricas e de resumo ⇒ **data warehouse**

Data Warehouse

- ▶ Um ***Data Warehouse (DW)***, assim como um sistema de banco de dados, **pode ser compreendido como uma coleção de dados + um sistema de apoio**
- ▶ DWs se diferem bastante dos BDs tradicionais – em sua estrutura, funcionalidade, desempenho e propósito
- ▶ O uso inicial do termo *Data Warehouse* é creditado a **William (Bill) Inmon (1992)**

Data Warehouse

- Definição de DW, segundo Inmon:

“Um *Data Warehouse* é uma coleção de dados orientada a assunto, integrada, não volátil, variável no tempo, que dá apoio às decisões de gerência”

Data Warehouse – Outras definições

- ▶ A **definição de Inmon restringe o DW** a uma “**coleção de dados**”
- ▶ **Entretanto, textos mais recentes empregam o termo com um significado mais abrangente**
- ▶ DW – “**sistema projetado com o propósito de dar apoio à extração, processamento e apresentação eficiente** (dos dados) para fins analíticos e de tomada de decisão”

Data Warehouse

- Características importantes
 - ▶ Geralmente **integram quantidades muito grandes de dados**, provenientes de múltiplas fontes [heterogêneas]
 - ▶ **São otimizados para a recuperação de dados (e não para o processamento rotineiro de transações, como os BDs “tradicionais”)**

Data Warehouse

- Levam em consideração o armazenamento, a manutenção e a recuperação eficiente de **dados históricos**
- DWs tipicamente **dão apoio a análises de séries temporais e tendências**, ambas as quais **requerem mais dados históricos** do que **costuma-se manter em BDs transacionais**

Data Warehouse

- Mais características importantes
 - A informação em um DW muda com pouca frequência (não-volátil), e pode ser considerada como não sendo de tempo real e com atualização periódica.
 - A política de atualização, que geralmente é incremental, define também a periodicidade das cargas de dados

Data Warehouse

- Mais características importantes
 - ▶ Apoiam diferentes tipos de aplicações de análise.
Exemplos:
 - OLAP (*online analytical processing*)
 - Mineração de dados

Data Warehouse

- Data Warehouse **é como** um banco de dados relacional projetado e **otimizado para necessidades analíticas (OLAP)**.
- ***Data Warehousing*** é o ato de organizar e armazenar dados de forma a tornar sua recuperação eficiente e perspicaz. **Também é chamado de processo de transformação de dados em informações.**

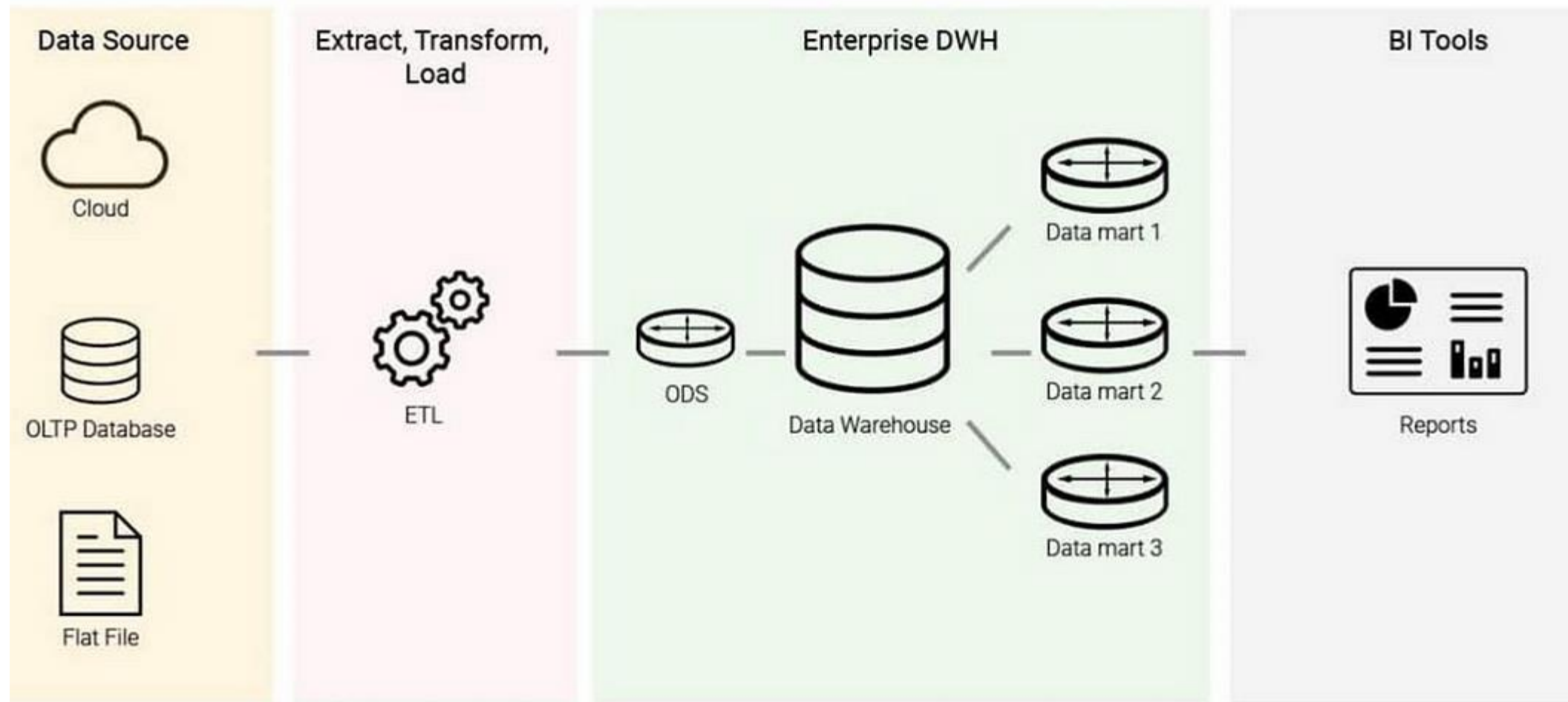
Data Warehouse – Sintetizando

	Banco de Dados Transacional	Data Warehouse
Características	<ul style="list-style-type: none">• Alto volume de Transações• Pequenas Transações• Normalizado• Consultas Rápidas	<ul style="list-style-type: none">• Alto volume de Dados• Transações Grandes• Denormalizado• Consultas Agregadas• Colunar
Tipo de pergunta	<i>Qual o saldo do usuário com CPF xyz?</i>	<i>Quantos usuários possuem saldo maior que R\$500,00?</i>
Aplicações	SQL Server, MySQL, Oracle SQL, PostgreSQL	Amazon Redshift, Google Big Query, Snowflake

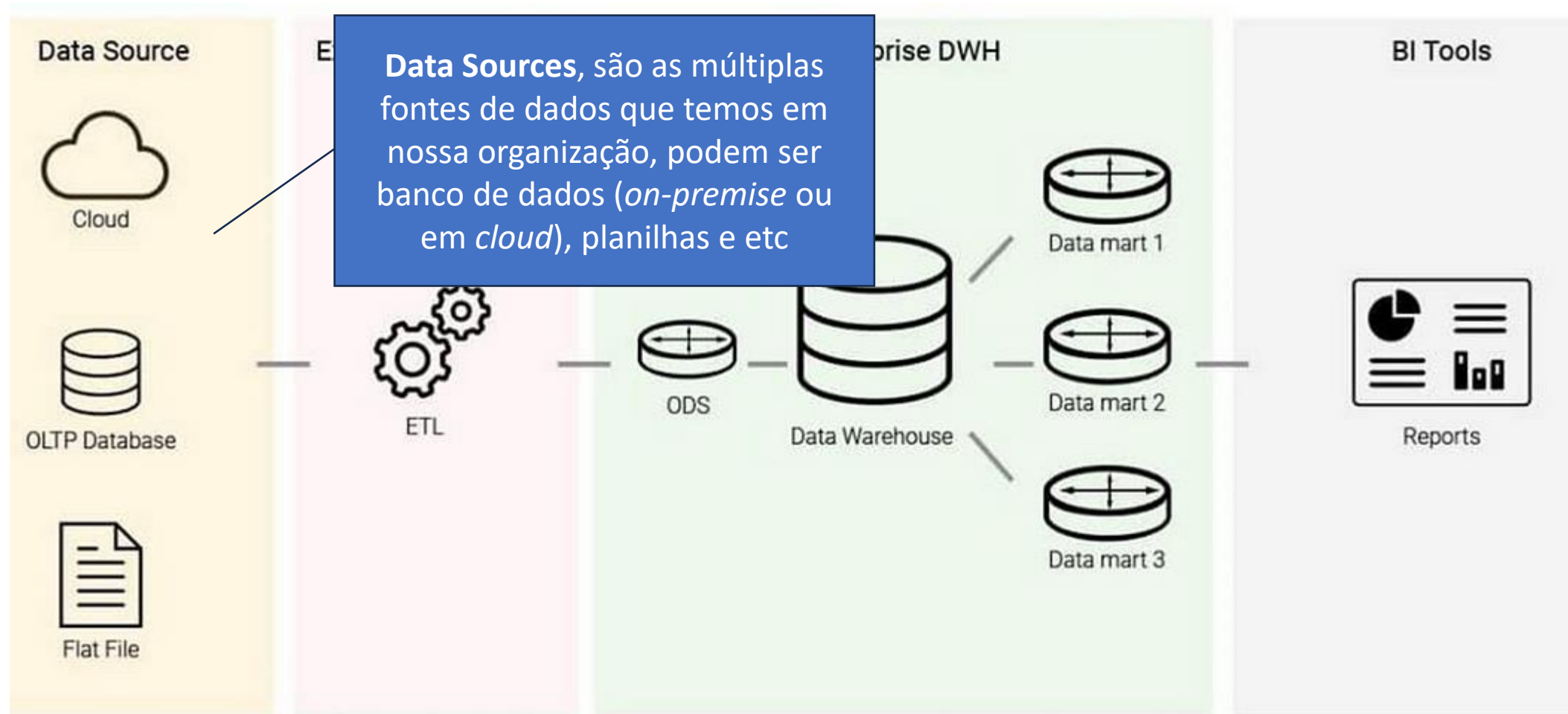
Data Warehouse – Vantagens

- Permite centralizar dados de diferentes fontes e sistemas em um único repositório, gerando “uma única fonte de verdade”
- Otimizando consultas
- Qualidade dos dados
- Governança (acessos) mais fácil
- Armazena a historia dos dados

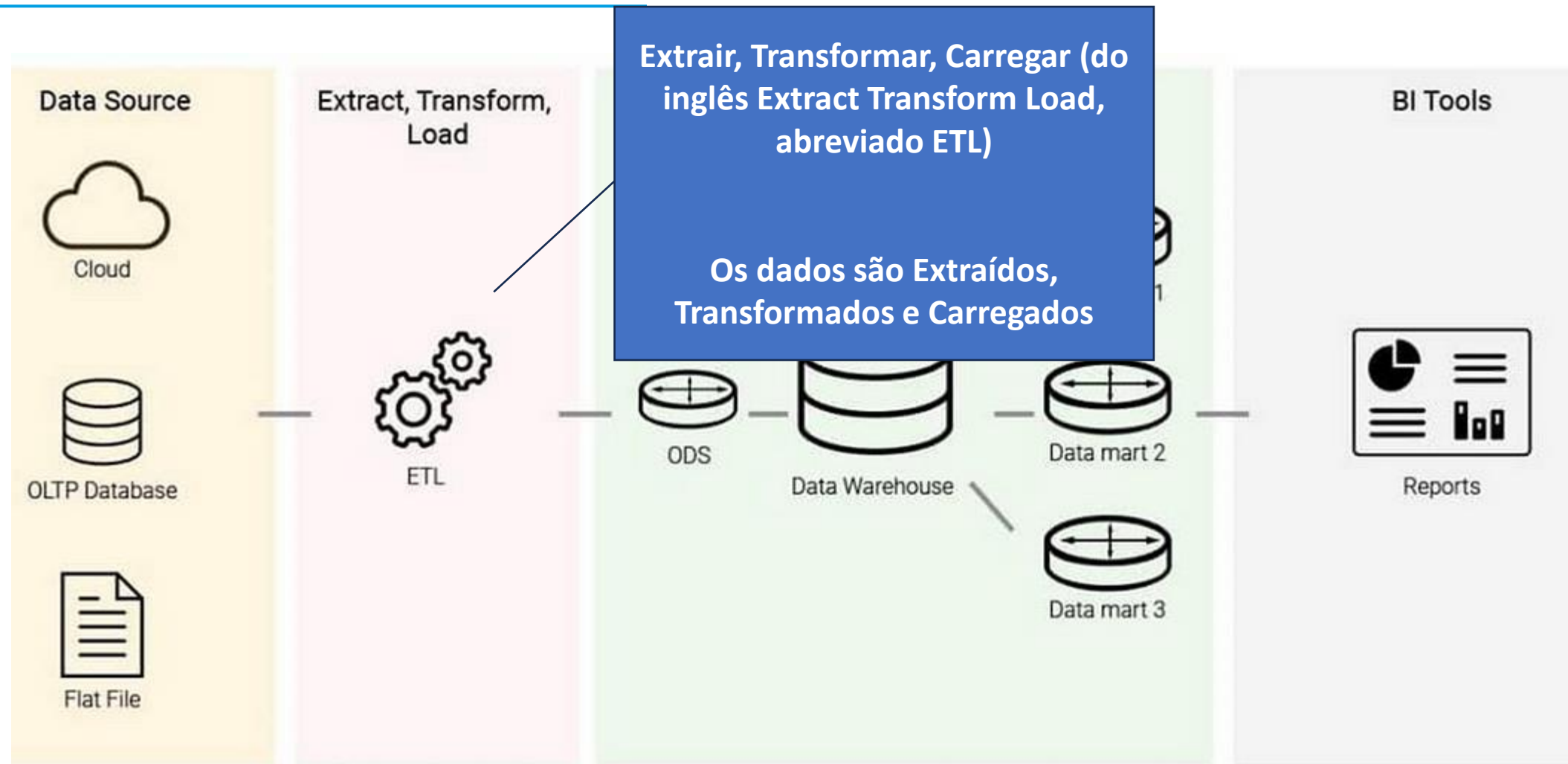
Data Warehouse



Data Warehouse



Data Warehouse



Data Warehouse

Data Source

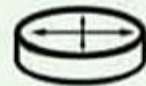


Cloud

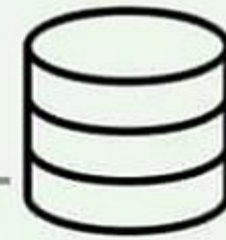
Extract, Transform,
Load

Enterprise DWH

BI Tools



ODS



Data Warehouse



Data mart 1



Data mart 2



Data mart 3

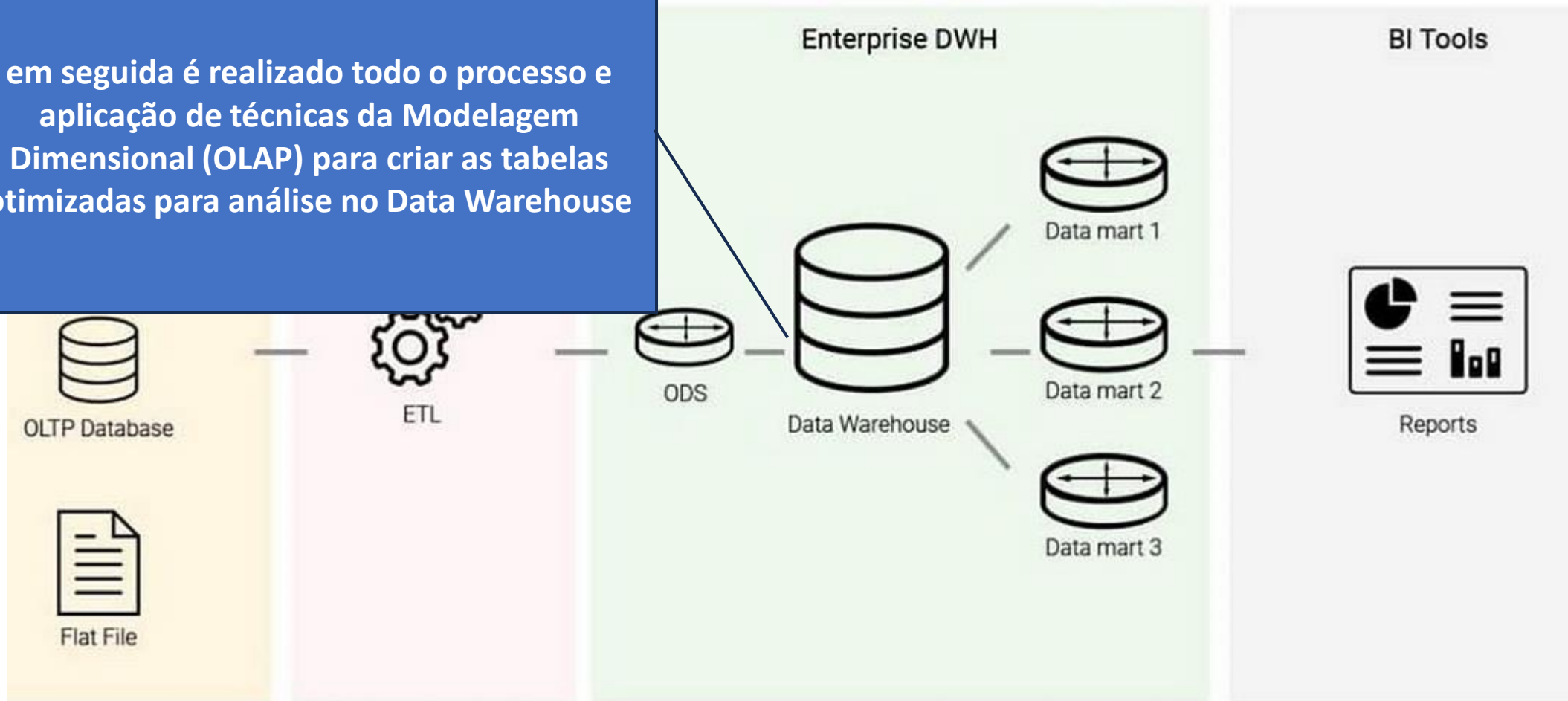


Reports

Os dados vão para um repositório temporário chamado ODS (Operational Data Store, por muitos também chamado de Stage area)

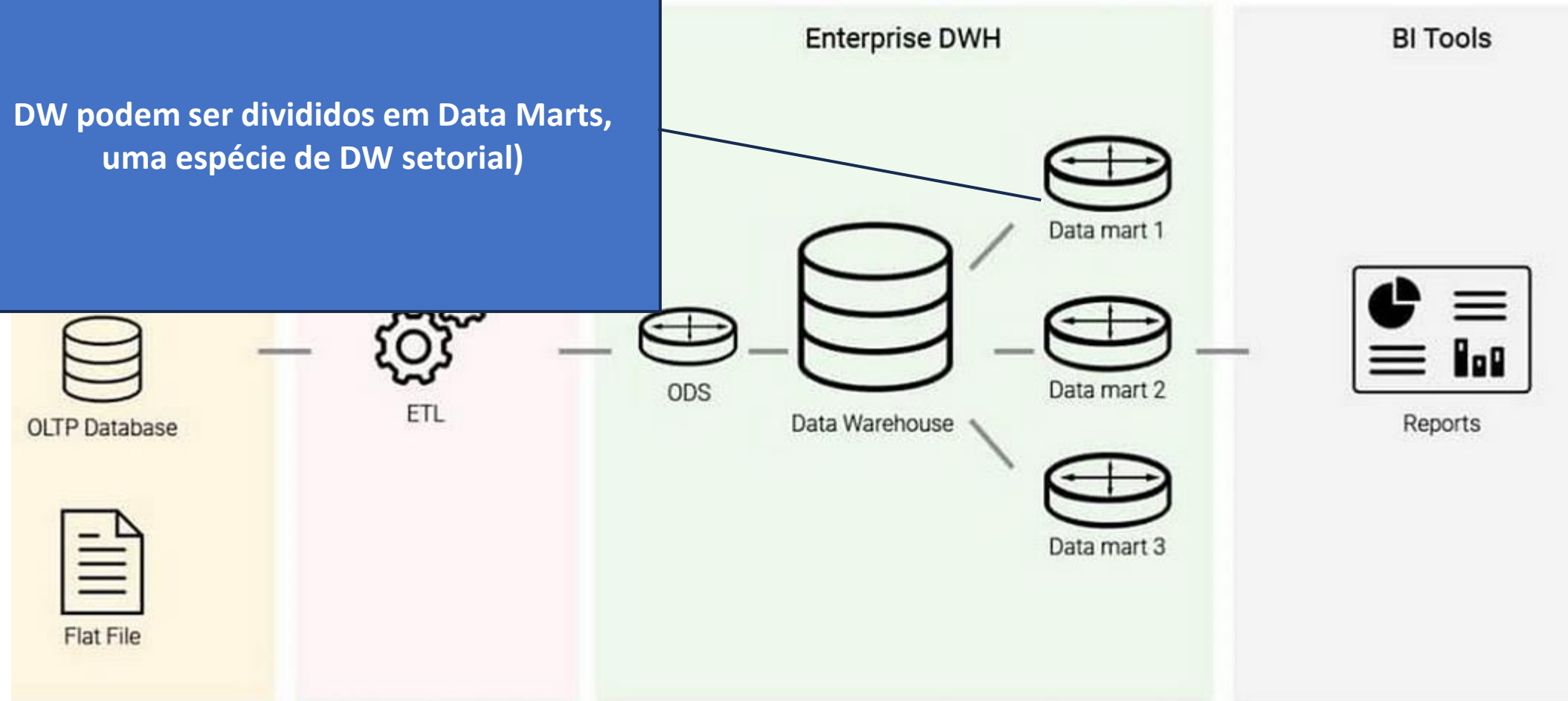
Data Warehouse

em seguida é realizado todo o processo e aplicação de técnicas da Modelagem Dimensional (OLAP) para criar as tabelas otimizadas para análise no Data Warehouse



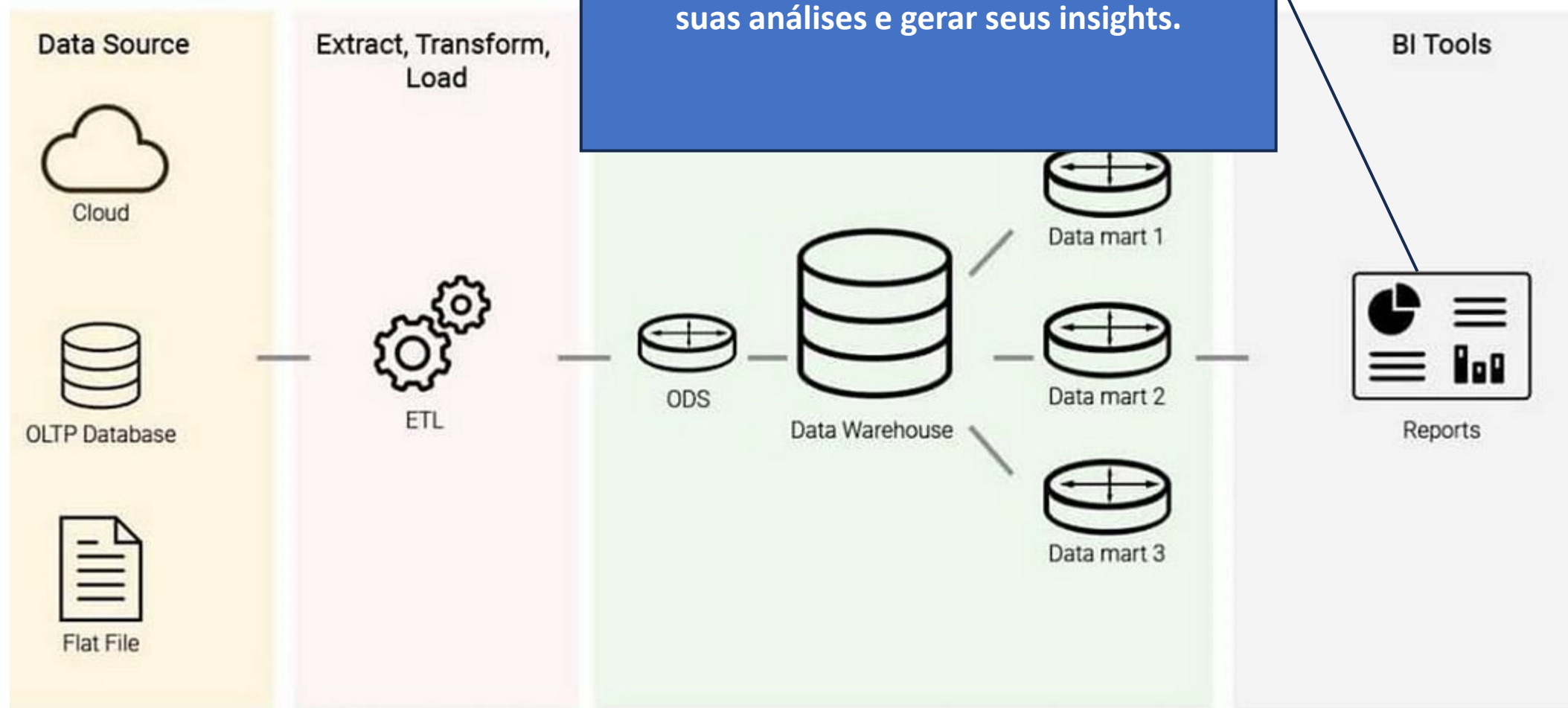
Data Warehouse

DW podem ser divididos em Data Marts, uma espécie de DW setorial)



Data Warehouse

Por último, os usuários utilizam esse Data Warehouse ou Data Marts para realizarem suas análises e gerar seus insights.



OLAP x OLTP

- **OLAP** — *Online analytical processing*

É a capacidade para manipular e analisar um grande volume de dados sob múltiplas perspectivas (cubos) (**Multidimensional**).

OLAP x OLTP

- Um **Data Warehouse** é modelado no conceito de OLAP,
- Enquanto os **bancos de dados** tradicionais (**relacionais**) são modelados no conceito de OLTP (*Online Transaction Processing* — esse é otimizado para inserções, atualizações, deleções, ou seja, para as necessidades do dia a dia da operação dos seus sistemas: ERP, CRM, Contábil, Financeiro e etc).

Elementos da **Modelagem Dimensional**:

- Tabelas de Fatos
- Tabelas Dimensões

O que são fatos?

- Objetivo do DW é encontrar os fatos que façam sentido dentro de uma empresa
 - E posteriormente mapear quais são as **dimensões** que detalham esse fatos
- Dimensões: Características dos fatos, elas detalham os fatos

O que são fatos?

- Processos de negócios fundamentais dentro de uma empresa
 - Ordem de compra
 - Venda de um e-commerce
 - Ordem de produção
 - Saldo de estoque

A tabela de fatos

- Pensando em uma tabela de pedidos:
 - Uma dimensão pode ser o dia em que foi feito o pedido
 - A cidade de onde foi feito esse pedido
 - Cliente que fez o pedido
 - Produto

A tabela de fatos

FK_Dia	FK_Cidade	FK_Cliente	FK_Produto
2019-01-01	10	5	2
2019-01-01	30	15	2
2019-01-02	10	5	2

A tabela de fatos

Chaves Estrangeiras

FK_Dia	FK_Cidade	FK_Cliente	FK_Produto	Unid. Vendidas	Valor
2019-01-01	10	5	2	1	200
2019-01-01	30	15	2	2	400
2019-01-02	10	5	2	1	200

Medidas

A tabela de fatos

Chaves Estrangeiras

FK_Dia	FK_Cidade	FK_Cliente	FK_Produto	Unid. Vendidas	Valor
2019-01-01	10	5	2	1	200
2019-01-01	30	15	2	2	400
2019-01-02	10	5 Grão	2	1	200

Medidas

Granularidade

- *A granularidade dos dados em um **modelo dimensional** representa o que cada fato pode nos dizer.*
- *Para melhor entender este conceito, precisamos recorrer à forma como sistemas transacionais armazenam dados durante a execução dos processos de negócio.*

Granularidade

- ***Sistema transacional*** (Online Transaction Processing — **OLTP**) *é todo aquele que dá suporte à execução dos processos de negócio, registrando todo o detalhamento necessário a sua correta consecução (normalmente utilizando modelos relacionais).*
- *Desta forma, ao consultar os dados armazenados, teremos o menor detalhamento possível, ou seja, o nível atômico daquele fato registrado (granularidade atômica).*

Granularidade

- *É o que ocorre, por exemplo, quando registramos uma Venda em um Sistema ERP.*
- *Tal Venda traz todo o detalhamento necessário para que o produto possa chegar até o endereço de entrega cumprindo todos os requisitos legais, de qualidade e de segurança definidos.*

Granularidade

- **Mas se mudarmos a perspectiva** e o objetivo atribuído aos dados transacionais, agora olhando em um **viés analítico (OLAP)**, considerando responder a perguntas como “**Quantas entregas foram realizadas para a cidade X aos sábados?**”
- **Podemos pensar na possibilidade de agregar dados de ordens de compra a partir das chaves de cidade e dia da semana** em que a entrega ocorreu.
- **Mudamos então a granularidade dos dados**, pois se assim forem agregados, **não iremos mais conseguir responder qual foi o horário** da entrega de uma ordem específica.

Granularidade

- *Mas afinal, por que o nível de granularidade (detalhamento) dos dados importa?*
- *Bem, importa por duas razões principais: **custo** e **velocidade**.*
- *Quando maior o detalhamento dos dados, maior será o volume armazenado e, por conseguinte, maior será o tempo necessário que **consultas** sejam processadas.*
- *Ou seja, o **custo computacional** é diretamente impactado pela granularidade dos dados. Logo, a escolha equilibrada do nível de detalhamento é uma decisão chave para o sucesso de qualquer projeto de análise de dados.*

Métricas/Medidas

- Métricas são **quantificações atribuídas** a um registro de **fato**.
- É o que representam as colunas “*QuantidadeVenda * ValorUnitarioVenda*” na tabela fato.
- E sobre essas quantificações, aplicam-se funções agregadores, que permitem consolidar as métricas **sobre o olhar de uma ou mais dimensões**.

Métricas

- Imagine como poderíamos responder perguntas como:

Qual foi o faturamento de vendas no ano de 2018 via o vendedor X? Quais são os clientes com o maior número de compras em 2022?

Métricas

- As principais características de métricas são:
 - São valores quantitativos (numéricos)
 - Estão sempre associados com um agregador, orientado pelas dimensões utilizadas

Métricas

- Podem ser de três tipos:
 - Aditivas
 - Não-aditivas
 - Semi-aditivas

Métricas

- **Aditivas:** podem ser sumarizadas(somadas) através de todas as dimensões, sem restrição.

Um **exemplo** poderia ser o **valor de venda de um e-commerce**.

Métricas

- **Não-aditivas:** não podem ser sumarizadas através de nenhuma dimensão.

Um exemplo seria percentual de desconto de venda de produtos, por exemplo.

Exemplos: médias, total de unidades distintas (mudar o agrupamento)

Métricas

- **Semi-aditivas:** podem ser sumarizadas através de algumas dimensões, já outras não.

É o caso do saldo bancário, por exemplo.

A soma de saldos de 2 dias distintos (Problemas)

Métricas

- Quando pensamos em métricas em uma perspectiva visual, costumamos representar através de um **cubo**
- Nele, temos várias métricas obtidas na intersecção entre as dimensões de produto e tempo

Tipos de Tabelas de Fatos

Tabela de Fatos - Transacional

- Grão é o registro individual – Cada linha um registro de venda

FK_Dia	FK_Cidade	FK_Cliente	FK_Produto	ID_Contrato	Unid. Vendidas	Valor	Desconto	Subtotal
2019-01-01	10	5	2	A1052	1	200	10	190
2019-01-01	30	15	2	A30152	2	400	10%	360
2019-01-02	10	5	2	B1052	1	200	10%	180

Tabela de Fatos – Snapshot Periódico

- Grão – Cada linha um período de tempo (dia, semana, mês)

FK_Dia	FK_Cidade	FK_Cliente	FK_Produto	unidades	valor_total	media_desconto
2019-01-01	10	5	2	1	800	7.75%
2019-01-01	30	15	2	2	2000	10%
2019-01-02	10	5	2	1	1000	11%

Tabela de Fatos – Snapshot Periódico

- Problemas
 - Não podemos voltar atrás
- Melhor do menor grau para o maior

Tabela de Fatos – Snapshot Acumulativo

- Grão = registro com o início e fim claro (Exemplo: dia)

Ex: Saldo de estoque, chamada de suporte

FK_Dia_inicio	FK_dia_fim	FK_Cliente	FK_Motivo	tempo_dias	valor_total
2019-01-01	10	5	2	1	800
2019-01-01	30	15	2	2	2000
2019-01-02	10	5	2	1	1000

Dimensões

- Dimensões são tabelas que oferecem um contexto descritivo ao evento de negócio registrado (no fato).
- Respondem a questões como *quando, quem, o que, por que, como e onde*, por exemplo [1].

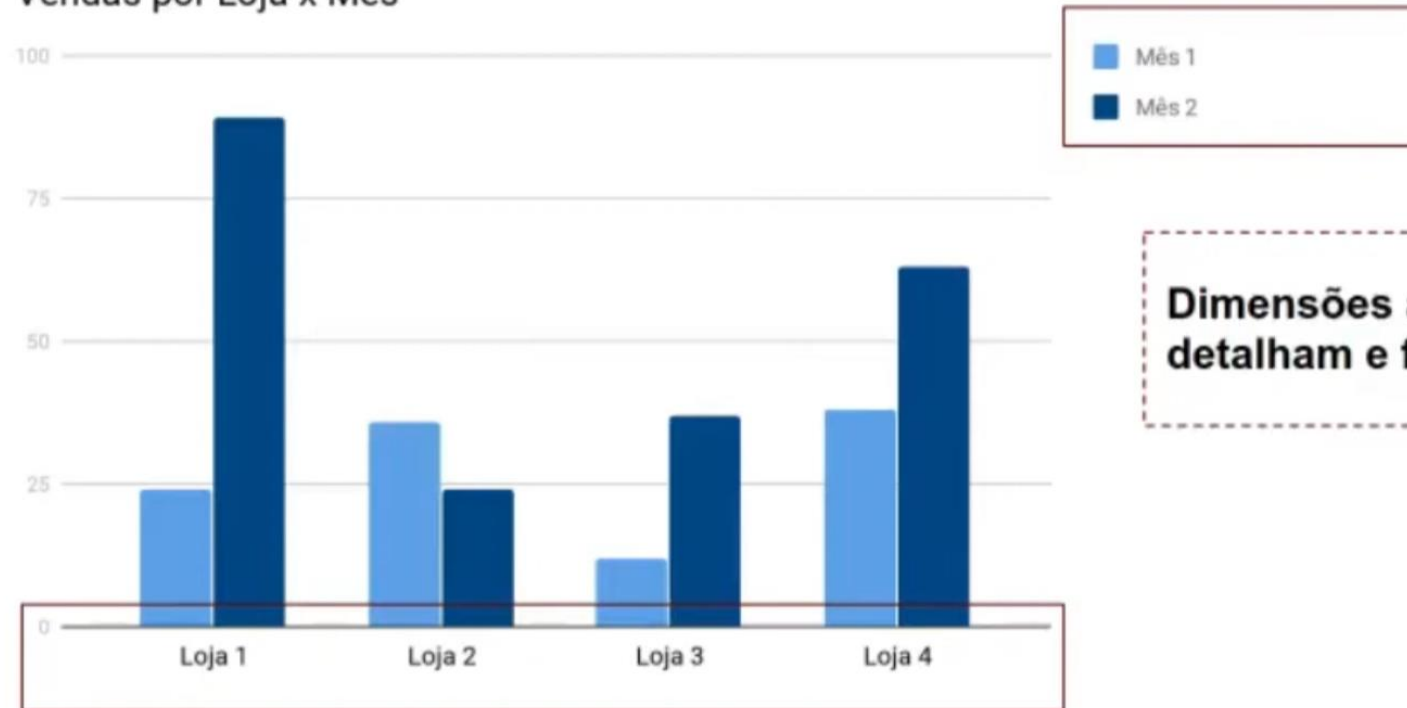
Dimensões

- Fatos são eventos
- Dimensões são atributos que qualificam esses eventos
- Exemplo:
 - Cidade de um cliente
 - Endereço
 - Data de um registro de evento
 - Local
 - Marca
 - Fornecedor

Dimensões

- Podem ser agrupados em um relatório/análise/BI

Vendas por Loja x Mês



**Dimensões agrupam,
detalham e filtram fatos**

Dimensões

- Dados de dimensão **são utilizados para filtrar e agrupar os fatos, permitindo observar os fatos a partir de diferentes perspectivas.**
- **Seus valores não podem ter duplicidades** (exemplo: cada linha representa um produto específico, um cliente específico...)

Dimensões

- **Alguns exemplos:**

- **Produto** (nome do produto, detalhe do produto, linha do produto, unidade de venda do produto...)
- **Cliente** (nome cliente, cpf, cnpj, endereço, cidade, estado...)

Dimensões

- Uma dimensão é uma coleção de atributos textuais que são altamente correlacionados entre si.

Exemplo: Clientes

ID da dimensão	ID Cliente	Nome	Cidade	UF
1	20	João da Silva	Rio de Janeiro	RJ
2	25	Carla Santos	Curitiba	PR
3	30	Roberto Almeida	São Paulo	SP

Dimensão não possui chaves estrangeiras!

Dimensões – Que acontece quando as dimensões se alteram?

Quando podem se alterar na tabela fonte

- Updates
- Mutáveis

Dimensões – Que acontece quando as dimensões se alteram?

- **Slowly Changing Dimensions** (Dimensões que Mudam Lentamente, em português)
- Retrata as dimensões que sofrem atualizações em seus campos e os classifica pelo tipo de mudança existente em cada uma delas.
- Vários tipos de **SCD** podem ser identificados no DW, variando de acordo com as características de atualizações das dimensões

Dimensões – Que acontece quando as dimensões se alteram?

- **Slowly Changing Dimensions** (Dimensões que Mudam Lentamente, em português)
- As alternativas mais comuns de SCD são:
 - SCD Tipo 0
 - SCD Tipo 1,
 - SCD Tipo 2,
 - SCD Tipo 3 e o
 - SCD Híbrido.

Exemplo

ID Cliente	Nome	Cidade
20	João da Silva	Rio de Janeiro

Tabela de Clientes do ERP

ID Cliente	Nome	Cidade
20	João da Silva	Curitiba

Tabela de Clientes do ERP

Como acompanhar?

SCD tipo 0

- Esse tipo geralmente ignora.

Dimensão **não se altera.**

ID Cliente	Nome	Cidade
20	João da Silva	Curitiba

Tabela de Clientes do ERP

SK Cliente	ID Cliente	Nome	Cidade (SCD Tipo 0)
1	20	João da Silva	Rio de Janeiro

Dimensão e fonte
ficam com estado
distinto

dim_clientes

SCD tipo 0

- Esse tipo geralmente ignora.
- Uso: talvez estado civil

Dimensão **não se altera**.

ID Cliente	Nome	Cidade
20	João da Silva	Curitiba

Tabela de Clientes do ERP

SK Cliente	ID Cliente	Nome	Cidade (SCD Tipo 0)
1	20	João da Silva	Rio de Janeiro

dim_clientes

Dimensão e fonte
ficam com estado
distinto

SCD tipo 1

- Esse tipo geralmente é chamado de método “ **Sobrescrever** ”.
- Nesse método, quaisquer alterações nos dados de dimensão simplesmente substituem o estado anterior dos dados com a mesma chave.

SCD tipo 1

Dimensão **se altera** mas perdemos o histórico

ID Cliente	Nome	Cidade
20	João da Silva	Curitiba

Tabela de Clientes do ERP

SK Cliente	ID Cliente	Nome	Cidade (SCD Tipo 1)
1	20	João da Silva	Curitiba

Dimensão e fonte
ficam com o mesmo
estado

dim_clientes

SCD tipo 1

- Embora muito simples de implementar, este método sofre de uma grande desvantagem.
- Devido ao mecanismo de substituição, você não apenas perde o histórico anterior da dimensão, mas também o estado da tabela de fatos à qual ela é anexada.

SCD tipo 1

- Observe como o antigo endereço residencial é simplesmente **substituído** pelo endereço novo, o histórico do endereço anterior é perdido.
- As repercussões da perda do histórico podem ser graves nos casos em que uma agregação da tabela de fatos é afetada pela alteração na dimensão.
- Nesse caso, sem o histórico, torna-se extremamente difícil rastrear o motivo pelo qual o valor de agregação foi afetado.

SCD tipo 1

- Devido às deficiências citadas acima, o SCD Tipo 1 raramente é usado em plataformas de dados modernas.
- Portanto, precisamos de métodos melhores, que nos permitam realizar alterações nas dimensões, preservando as referências **anteriores** para usos ativos.
- No geral, basta usar o SCD Tipo 1 se seus cálculos *não se* importarem com o estado anterior dos dados ou com as repercussões que isso causa no futuro.

SCD Tipo 2

- Também conhecido como método “ **Adicionar um novo registro** ”.
- Nesse método, o registro de alteração é adicionado como um novo registro à tabela de dimensões e marcado como “ ***Atual*** ” ou “ ***Ativo*** ”.
- Além disso, a versão anterior do registro é marcada como “ ***Expired*** ” ou “ ***Inactive*** ”.

SCD Tipo 2

- As várias versões (atual e histórica) de um registro são ligadas usando uma chave **substituta** .
- No nível da tabela, o SCD Tipo 2 é implementado adicionando colunas de carimbo de data/hora ***StartDate*** e ***EndDate*** para cada linha na tabela de dimensões.
- Além disso, uma coluna de ***status*** é adicionada para marcar se o registro é ***atual*** ou ***expirado*** status.

SCD Tipo 2

ID Cliente	Nome	Cidade	Modificado
20	João da Silva	São Paulo	8/06/2016

Tabela de Clientes do ERP

SK Cliente	Efetivo	Expirado	Atual	ID Cliente	Nome	Cidade (SCD Tipo 2)
1	10/04/2015	8/06/2016	0	20	João da Silva	Rio de Janeiro
2	8/06/2016	NULL	1	20	João da Silva	São Paulo

SCD Tipo 2

ID Cliente
20

Tabela de Clientes d

Por esse motivo usamos uma chave primaria diferente ao dado de origem

SK Cliente	Efetivo	Expirado	Atual	ID Cliente	Nome	Cidade (SCD Tipo 2)
1	10/04/2015	8/06/2016	0	20	João da Silva	Rio de Janeiro
2	8/06/2016	NULL	1	20	João da Silva	São Paulo

SCD Tipo 2

- Usando a abordagem SCD Tipo 2, você pode rastrear cronologicamente o histórico de alterações ao longo do tempo e manter as referências a tabelas de fatos de maneira cronológica.
- MAS implementação é um pouco complicada em comparação com o SCD Tipo 1.

SCD Tipo 2

- O aplicativo que mantém a tabela de dimensões precisa ser codificado de forma que a adição do novo registro com a versão atual e a expiração da versão anterior sejam realizadas em uma única transação.
- Além disso, toda consulta que vai contra a tabela de dimensão precisa filtrar status= **Current** .
- **Existe uma alternativa mais simples**, ainda exploramos outro método que, de certa forma, **é simplesmente uma extensão do método SCD Tipo 1**.

SCD Tipo 3

- Também conhecido como método “ **Adicionar um novo campo** ”.
- Para cada alteração, a versão anterior e a versão atual são armazenadas como duas colunas distintas na mesma linha da tabela de dimensões.
- O SCD Tipo 3 é relativamente mais fácil de implementar em comparação com o SCD Tipo 2, o histórico inclui apenas as versões atual e anterior.

SCD Tipo 3

- O fato de que apenas uma quantidade limitada de histórico está disponível torna o caso de uso do SCD Tipo 3 um pouco limitado. Mas a facilidade de implementação o torna um tanto desejável.

SCD Tipo 3

ID	name	address	previous_address	Phone
1	Julie Rich	Ap #255-3031 Dui Avenue, Billings, USA		1-528-884-4331
2	Eden Byrd	Ap #476-7527 Aenean Road, Harrisburg, USA		1-375-910-8385
3	Magee Cash	P.O. Box 333, 5820 Vel Avenue, New Haven, USA		1-853-137-7417

Change of Address

ID	name	address	previous_address	Phone
1	Julie Rich	Ap #255-3031 Dui Avenue, Billings, USA		1-528-884-4331
2	Eden Byrd	Ap #476-7527 Aenean Road, Harrisburg, USA		1-375-910-8385
3	Magee Cash	687 Magnis Rd., New Haven, USA	P.O. Box 333, 5820 Vel Avenue, New Haven, USA	1-853-137-7417

Tipos de chaves

Chave natural ou de negócio

- Chave primária da fonte de dados

ID	Nome	Cidade
20	João da Silva	Rio de Janeiro
25	Carla Santos	Curitiba
30	Roberto Almeida	São Paulo

Fonte

Chave natural ou de negócio

- Chave primária da fonte de dados

Não deve ser usada como chave da dimensão!

ID	Nome	Cidade
20	João da Silva	Rio de Janeiro
25	Carla Santos	Curitiba
30	Roberto Almeida	São Paulo

ID da dimensão	ID Cliente	Nome	Cidade
1	20	João da Silva	Rio de Janeiro
2	25	Carla Santos	Curitiba
3	30	Roberto Almeida	São Paulo

Fonte

Dimensão

Chave Surrougate

- Gera uma combinação única de registros (Problema - duplicação)
 - Chave Composta

ID da dimensão	ID Cliente	Nome	Cidade
20_joao_da_silva_rio_de_janeiro	20	João da Silva	Rio de Janeiro
25_carla_santos_curitiba	25	Carla Santos	Curitiba
30_roberto_almeida_sao_paulo	30	Roberto Almeida	São Paulo

Chave Surrougate

- Gera uma combinação única de registros
 - Chave Auto-incremental

ID da dimensão	ID Cliente	Nome	Cidade
01	20	João da Silva	Rio de Janeiro
02	25	Carla Santos	Curitiba
03	30	Roberto Almeida	São Paulo

* Pode ter um limite de opções em tabelas muito grandes

Chave Surrougate

- Gera uma combinação única de registros
 - Chave “Hasheada”

ID da dimensão	ID Cliente	Nome	Cidade
XyZjb1231nASD	20	João da Silva	Rio de Janeiro
48Juhybbcanhu!asd	25	Carla Santos	Curitiba
Has37hbasdaaa22Z	30	Roberto Almeida	São Paulo

Abordagens de desenvolvimento de *data warehouses*

- **O modelo de Inmon:** abordagem EDW (de cima para baixo)
- **O modelo de Kimball:** abordagem por *data mart* (de baixo para cima)

Abordagens de desenvolvimento de *data warehouses*

- O modelo de Inmon: abordagem EDW (de cima para baixo)
- O modelo de Kimball: abordagem por *data mart* (de baixo para cima)



Qual modelo é o melhor?

Abordagens de desenvolvimento de *data warehouses*

- O modelo de Inmon: abordagem EDW (de cima para baixo)
- O modelo de Kimball: abordagem por *data mart* (de baixo para cima)



Qual modelo é o melhor?

O que atender a necessidade do seu negócio.



Detalhando um pouco as abordagens...

- **Top Down (Inmon)**

Faz um todo, que irá atender a todos os setores

- **Vantagens:** centralização, visão globalização
- **Desvantagem:** implementação lenta e risco alto



Detalhando um pouco as abordagens...

- **Botton Up (Kimball)**

Faz por setores e depois sobe para um todo

- **Vantagens:** implementação e retorno rápido
- **Desvantagem:** integrações entre áreas, descentralização



Detalhando um pouco as abordagens...

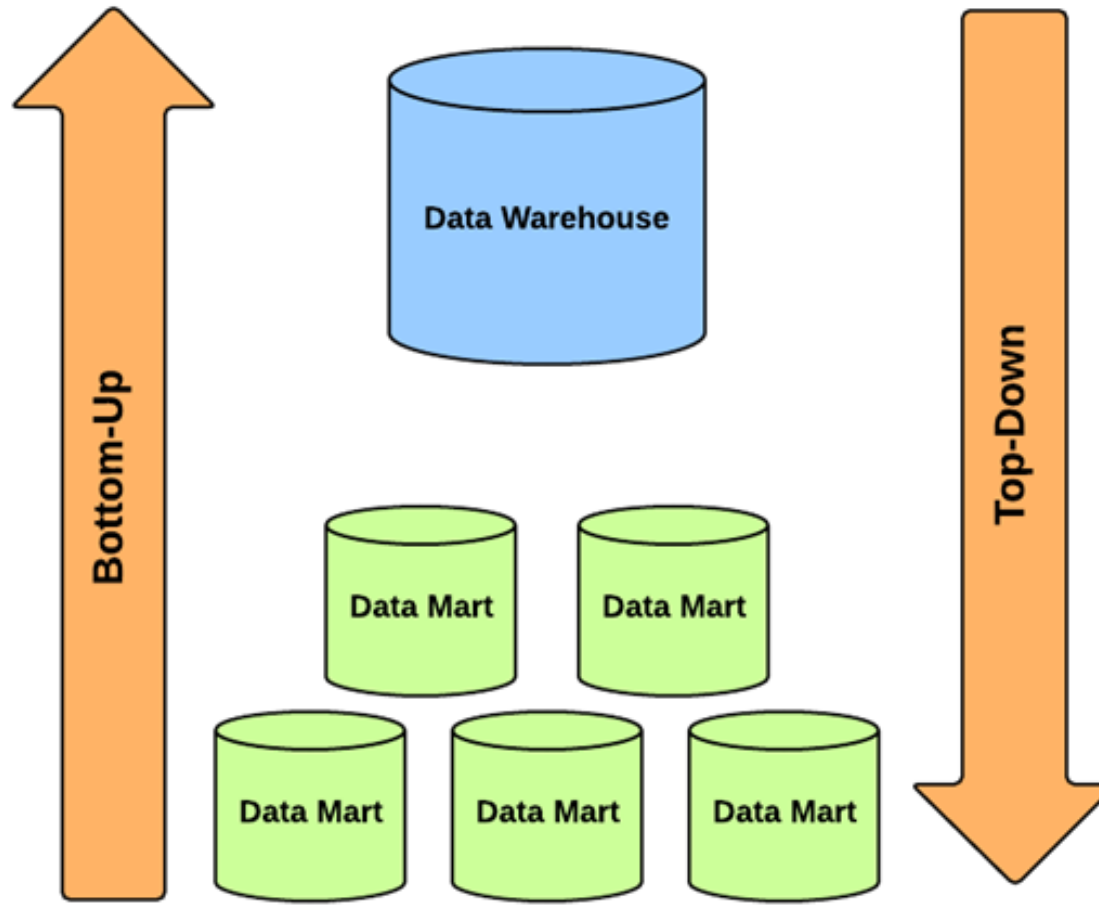
- **Combinada**

Planejamento *Top Down* e Desenvolvimento *Bottom Up*

- Planejamento como um todo (e não apenas setorial) e a implementação (desenvolvimento) feita setorialmente e não como um todo.



Detalhando um pouco as abordagens...



Detalhando um pouco as abordagens...

- **Combinada**

Planejamento *Top Down* e Desenvolvimento *Bottom Up*

- Planejamento como um todo (e não apenas setorial) e a implementação (desenvolvimento) feita setorialmente e não como um todo.



Modelagem Dimensional

- Um Data Warehouse é montado utilizando Modelagem Dimensional (ou **Multidimensional**), ou seja, um sistema baseado em recuperação de dados que suporta **alto volume** de acesso via consultas.

Multidimensionalidade

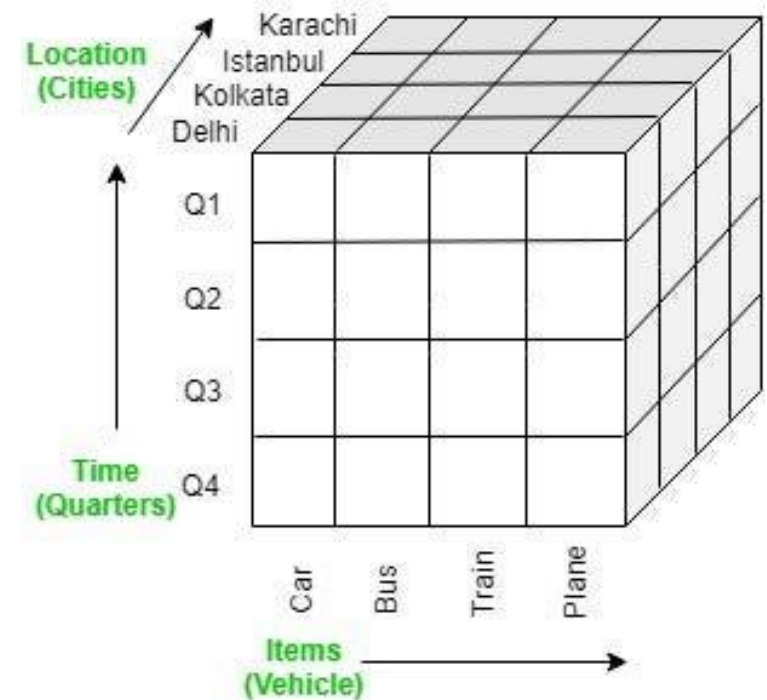
- A capacidade de organizar, apresentar e analisar dados (**fatos** — **vendas** por exemplo) por várias **dimensões**, como:
 - por regiões,
 - por produtos,
 - por vendedores e
 - por tempo (quatro dimensões, no exemplo)

Apresentação multidimensional:

- **Dimensões:** produtos, vendas, mercado, segmentos, unidades de negócios, localizações geográficas, canais de distribuição, países ou indústrias
- **Tempo:** diário, semanal, mensal, bimestral, trimestral, semestral ou anual

Operações OLAP em Cubos

- Operações nas dimensões, claro, sempre baseado em um Fato...
- Exemplo de algumas das operações, nas dimensões: **Location**, **Time** e **Items**...



Operações OLAP em Cubos

- **Segmentar ou fatiar (*Slice*)**

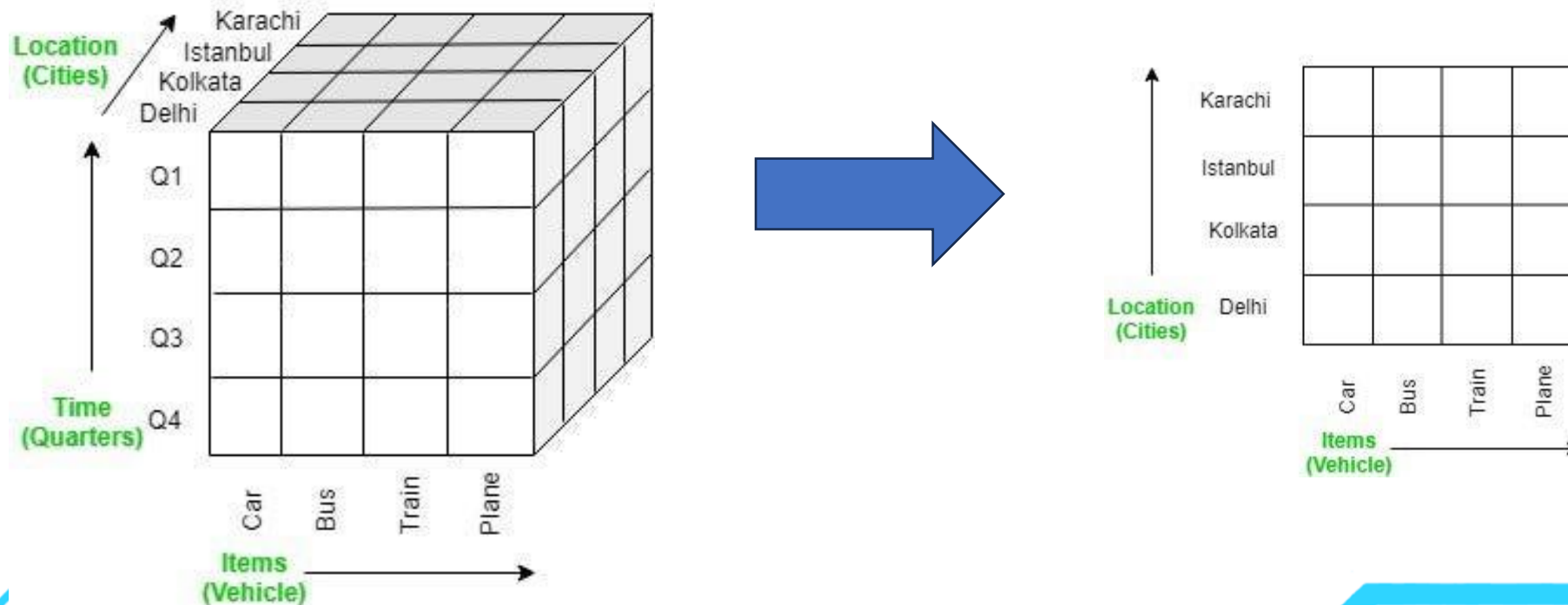
Ele seleciona uma única dimensão do cubo OLAP que resulta na criação de um novo subcubo (“corte para um valor fixo”).

Com base no cubo de estudo fornecido, o *Slice* é executado na dimensão Time = “Q1” (Trimestre).

Operações OLAP em Cubos

- Segmentar ou fatiar (*Slice*)

Com base no cubo de estudo fornecido, o *Slice* é executado na dimensão Time = “Q1” (Trimestre).



Operações OLAP em Cubos

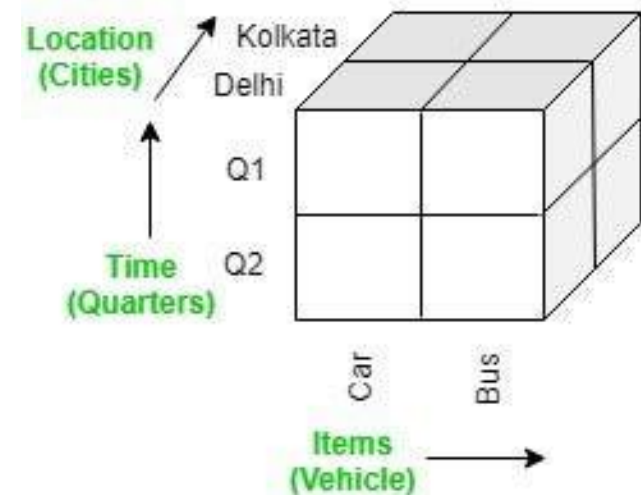
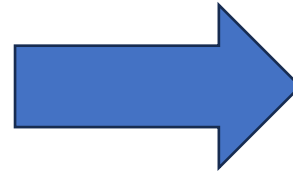
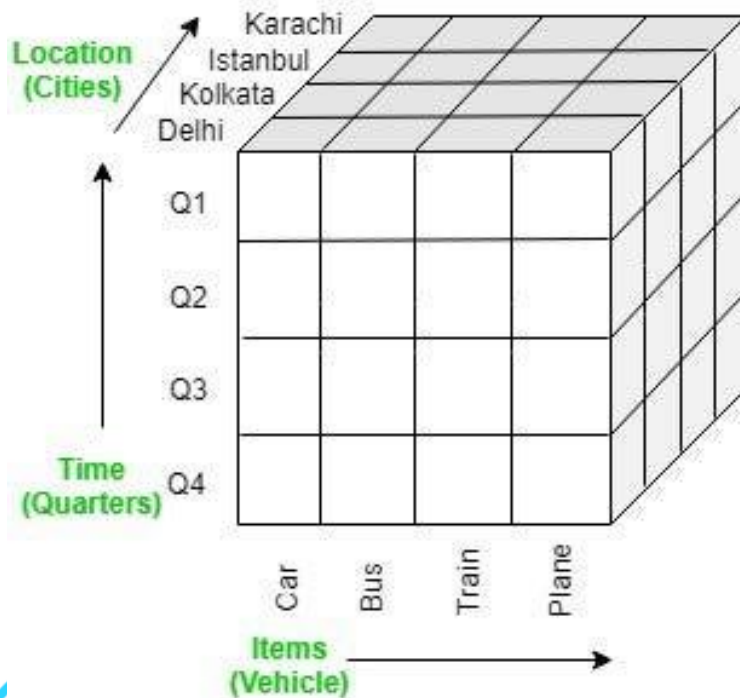
- **Dividir ou picar (dice)**

Ele retorna um subcubo do cubo OLAP selecionando duas ou mais dimensões (“seleção de faixas de valores”).

Operações OLAP em Cubos

- **Dividir ou picar (dice)**

Ele retorna um subcubo do cubo OLAP selecionando duas ou mais dimensões (“seleção de faixas de valores”).



Operações OLAP em Cubos

- Aumentar/diminuir o foco (*drill-down / roll-up*)

O usuário navega através de níveis de dados, indo desde os mais resumidos (menos foco — *roll-up*) até os mais detalhados (mais foco *drill-down*)

Operações OLAP em Cubos

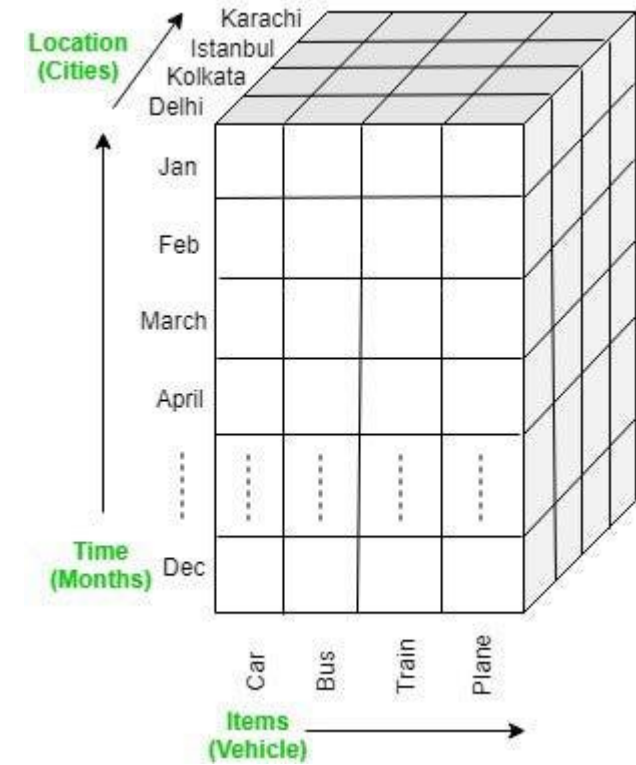
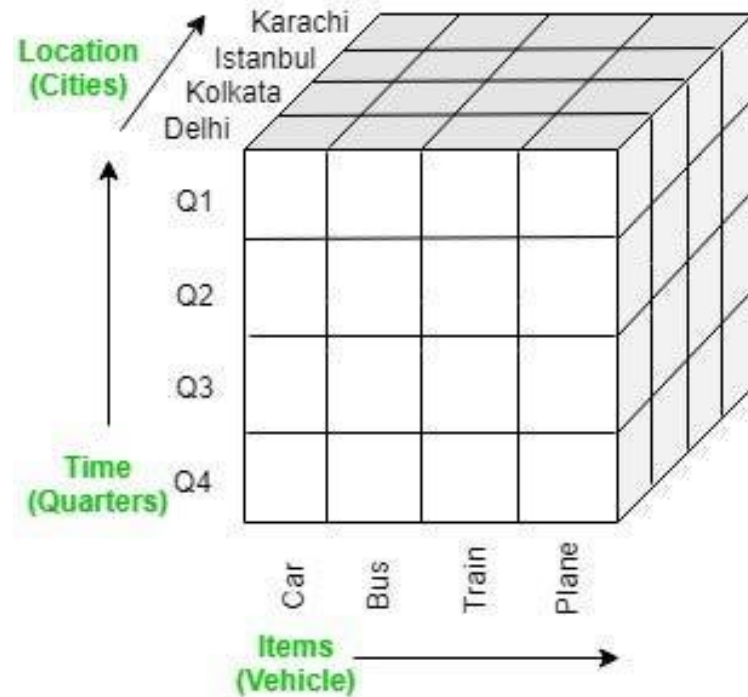
- Aumentar/diminuir o foco (*drill-down / roll-up*)

Na operação de ***drill down***, os dados menos detalhados são convertidos em dados altamente detalhados.

Pode ser feito por:

- Descendo na hierarquia do conceito
- Adicionando uma nova dimensão

Operações OLAP em Cubos



Operações OLAP em Cubos

- Já na ***roll up***, é justamente o contrário do anterior.

Ele executa a agregação no cubo OLAP.

Pode ser feito por:

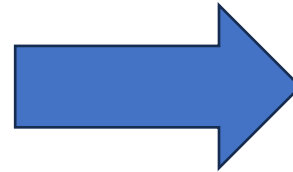
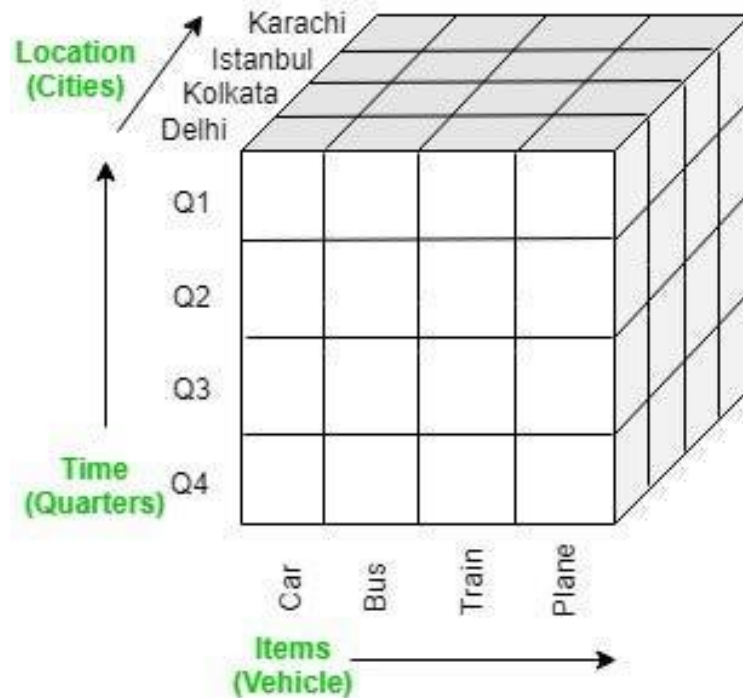
- Subindo na hierarquia do conceito
- Reduzindo as dimensões

Operações OLAP em Cubos

- Fazer o pivô (pivot)
- Usado para modificar a orientação dimensional de um relatório ou de uma página de exibição de consulta.
- Também é conhecida como **operação de *rotação***, pois gira a visão atual para obter uma nova visão da representação.
- No subcubo obtido após a operação de ***slice***, a execução da operação de pivô fornece uma nova visão do mesmo.

Operações OLAP em Cubos

- Fazer o pivô (pivot)



A 2D table diagram representing the result of a pivot operation. The dimensions are:

- Items (Vehicle)**: Car, Bus, Train, Plane
- Location (Cities)**: Delhi, Kolkata, Istanbul, Karachi

Car				
Bus				
Train				
Plane				
	Delhi	Kolkata	Istanbul	Karachi

Representação de dados (Tipos de Modelos)

- Esquema em estrela (*Star Schema*)
- Esquema em floco de neve (*Snow flake Schema*)
- Esquema Galaxy (*Galaxy Schema*)

Modelo Star Schema

- É um modelo dimensional comumente implantado em banco de dados relacionais.
- Caracteriza-se por tabelas fatos associadas à dimensões por meio de chaves estrangeiras.

Modelo Star Schema

- A sua estruturação otimiza consultas em grandes volumes de dados.
- Um ponto importante a considerar é que ***star schemas* são modelos não normalizados de dados**, que adicionam certa redundância (dimensões) em prol de ganhos de performance no momento das consultas.

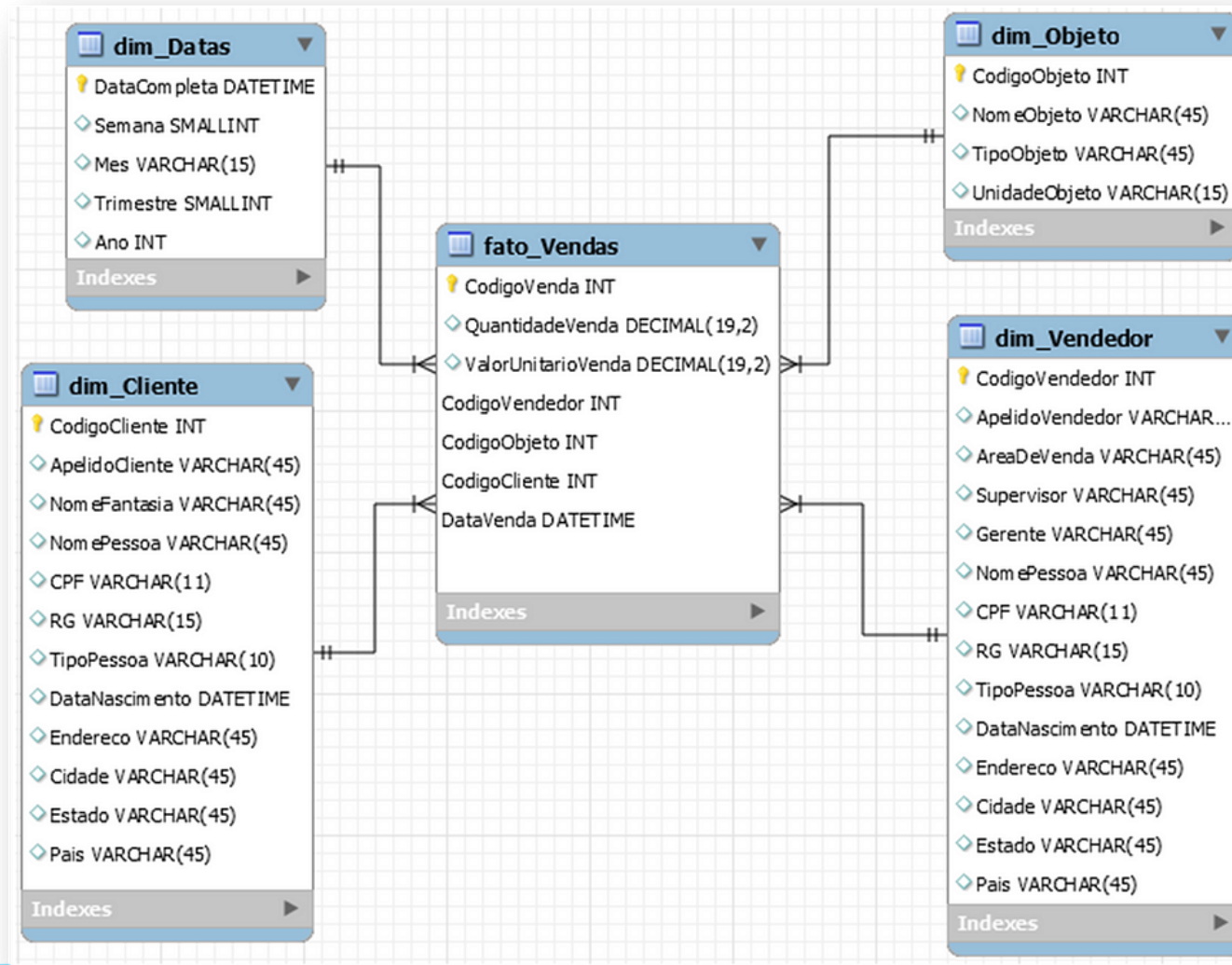
Modelo Star Schema

- O nome “**estrela**” se dá devido à disposição em que se encontram as tabelas, sendo a tabela fato centralizada relacionando-se com diversas outras tabelas de dimensão.

Modelo Star Schema

- Veja um exemplo da estrutura do Star Schema. Nesse modelo os **dados** são “**desnormalizados**” para evitar *joins* entre tabelas, **diminuindo o tempo de consultas**, no entanto devido a repetição de dados, **utiliza mais espaço em disco**.

Modelo Star Schema



Modelo Star Schema

- Vantagem desse modelo é a eficiência na extração de dados, o que é um grande diferencial em se tratando de um data warehouse

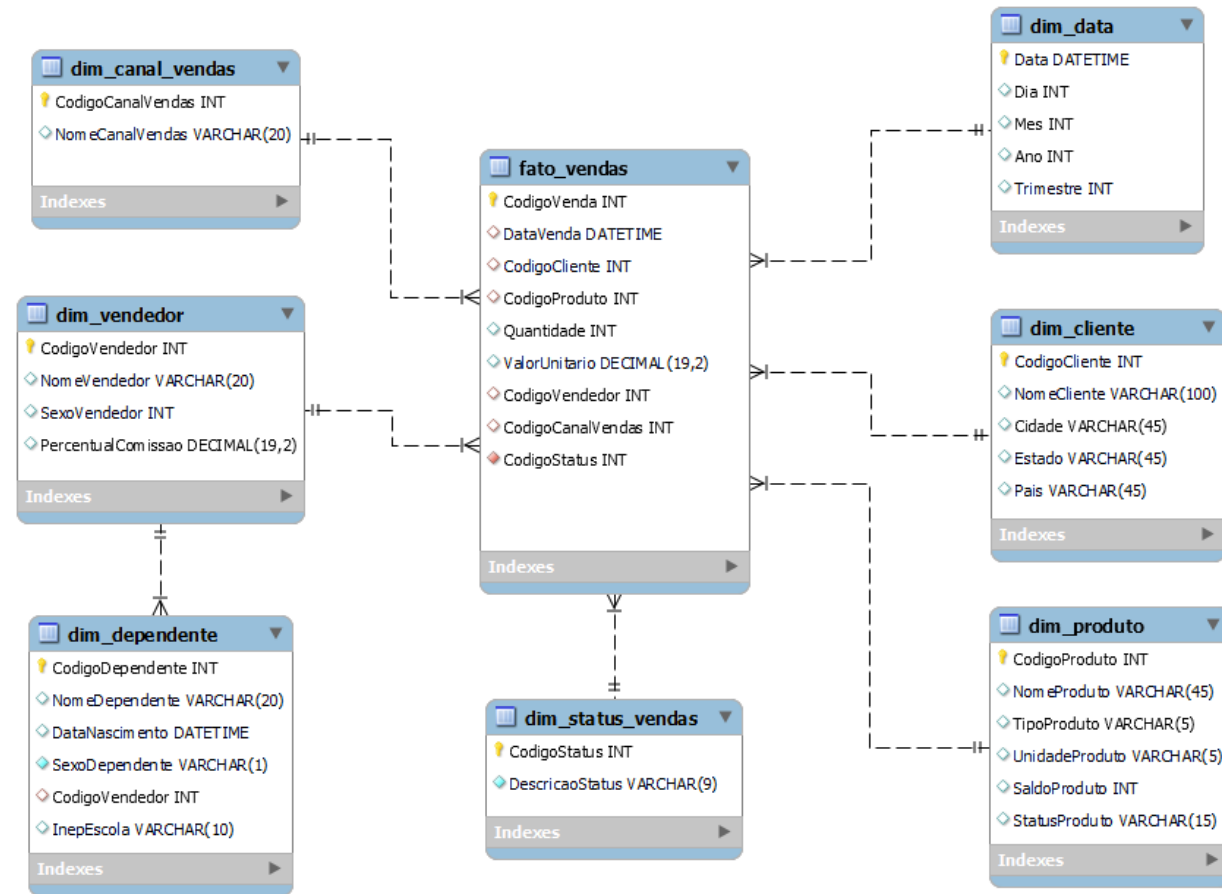
Modelo Snow Flake

- É uma variação do modelo *star schema* em que **criam-se hierarquias de dimensões (normalizadas) sempre que necessário.**
- Um dos **benefícios** desta **estratégica de modelagem** é **otimizar o armazenamento, visto que as redundancias de dados são minimizados** (ou mesmo removidas).

Modelo Snow Flake

- Todavia, acrescenta-se complexidade nas consultas pela maior quantidade de tabelas e relacionamentos envolvidos.
- Há também o custo computacional associado com uma maior quantidade de junções entre tabelas, que podem impactar negativamente no tempo de resposta às consultas do usuário.

Modelo Snow Flake



Modelo Galaxy

- Também conhecido como esquema de Constelação de Fatos.
- Contém mais de **1 tabela de fatos**. As dimensões que são compartilhadas são chamadas de dimensões **conformadas**.

Técnicas de Modelagem Multidimensional

- Segue uma sequência de imagens de como transformar uma Modelagem Relacional — OLTP (base de dados relacional — operacional) para uma **Modelagem Multidimensional (OLAP)**.

Técnicas de Modelagem Multidimensional

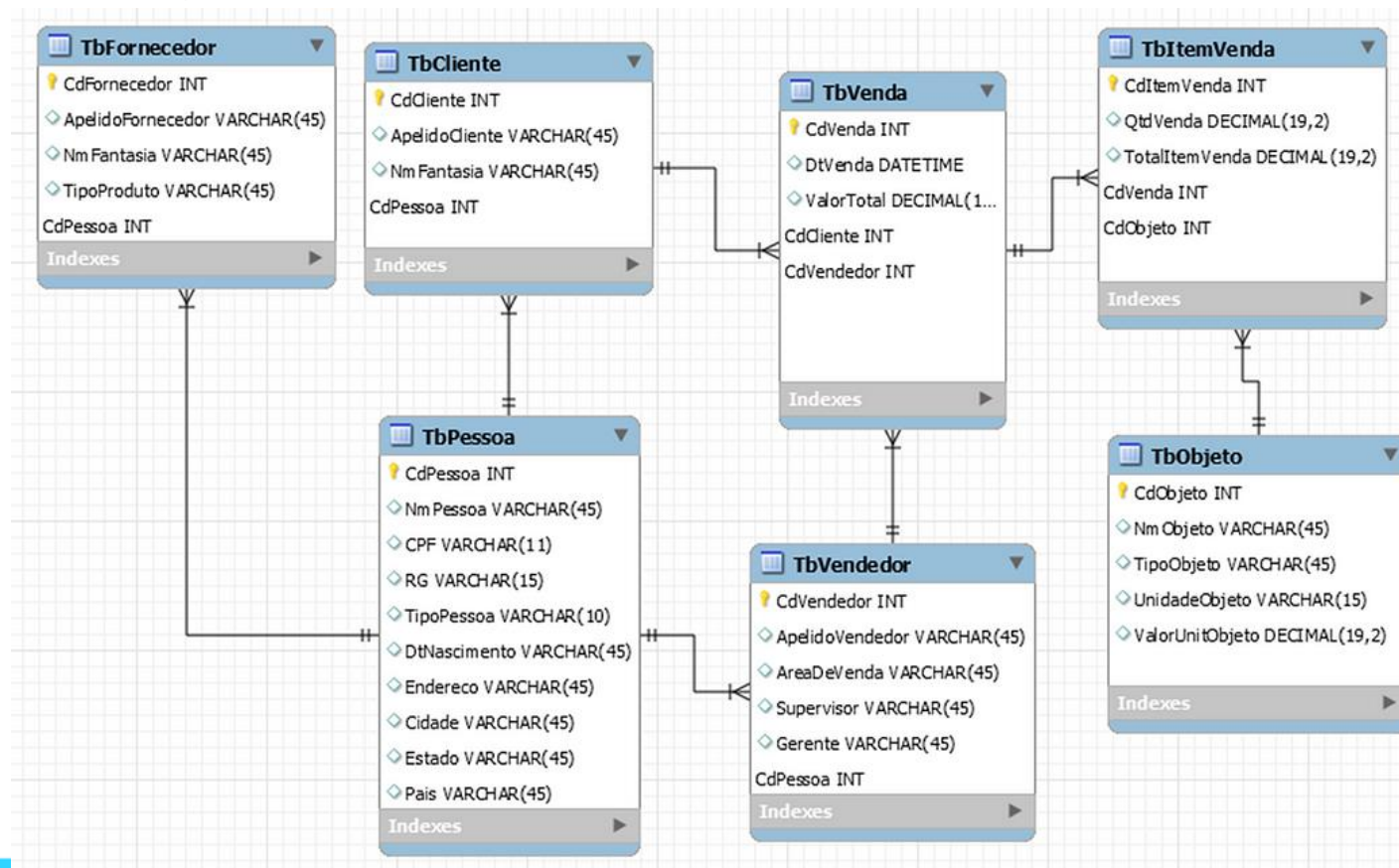
- Do relacional para o Multidimensional:

Modelo Relacional (OLTP)

Modelo que é utilizado nas bases de dados operacionais (ERP, CRM...). No nosso exemplo, temos uma modelagem relacional de uma base de dados de vendas.

Técnicas de Modelagem Multidimensional

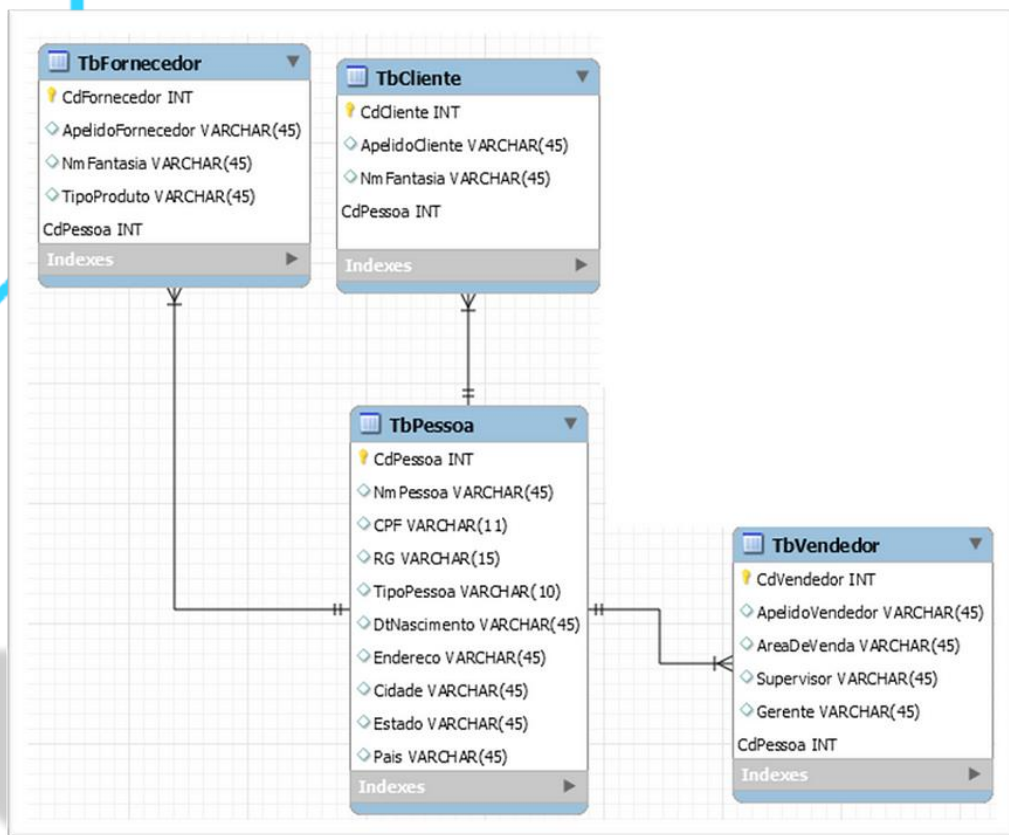
- Do relacional para o Multidimensional:



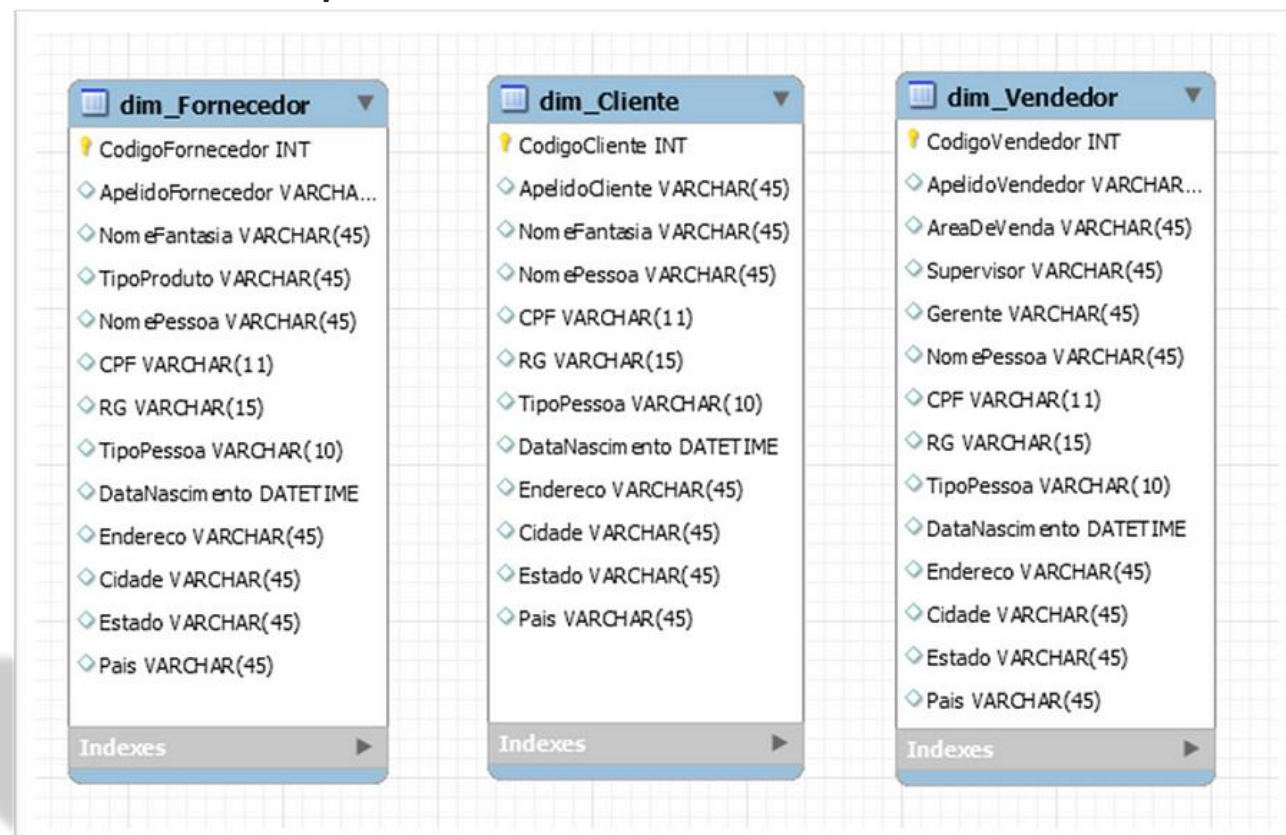
Técnicas de Modelagem Multidimensional

- Exemplo 1 — Dimensões envolvendo a tabela Pessoa

Do Relacional ...

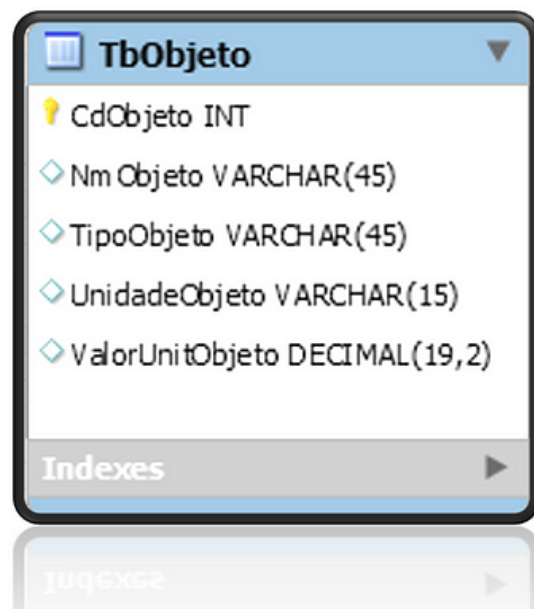


para o Multidimensional

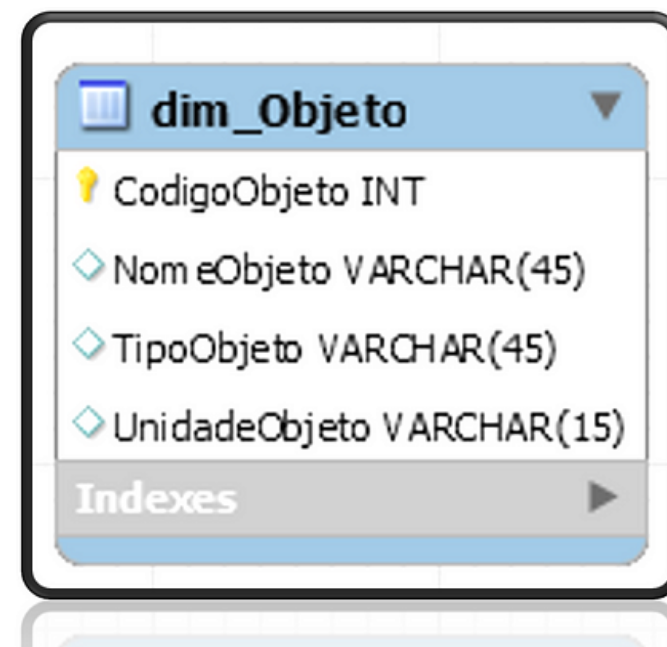


Técnicas de Modelagem Multidimensional

- Exemplo 2 — dimensão Objeto
- Do Relacional para



- o Multidimensional

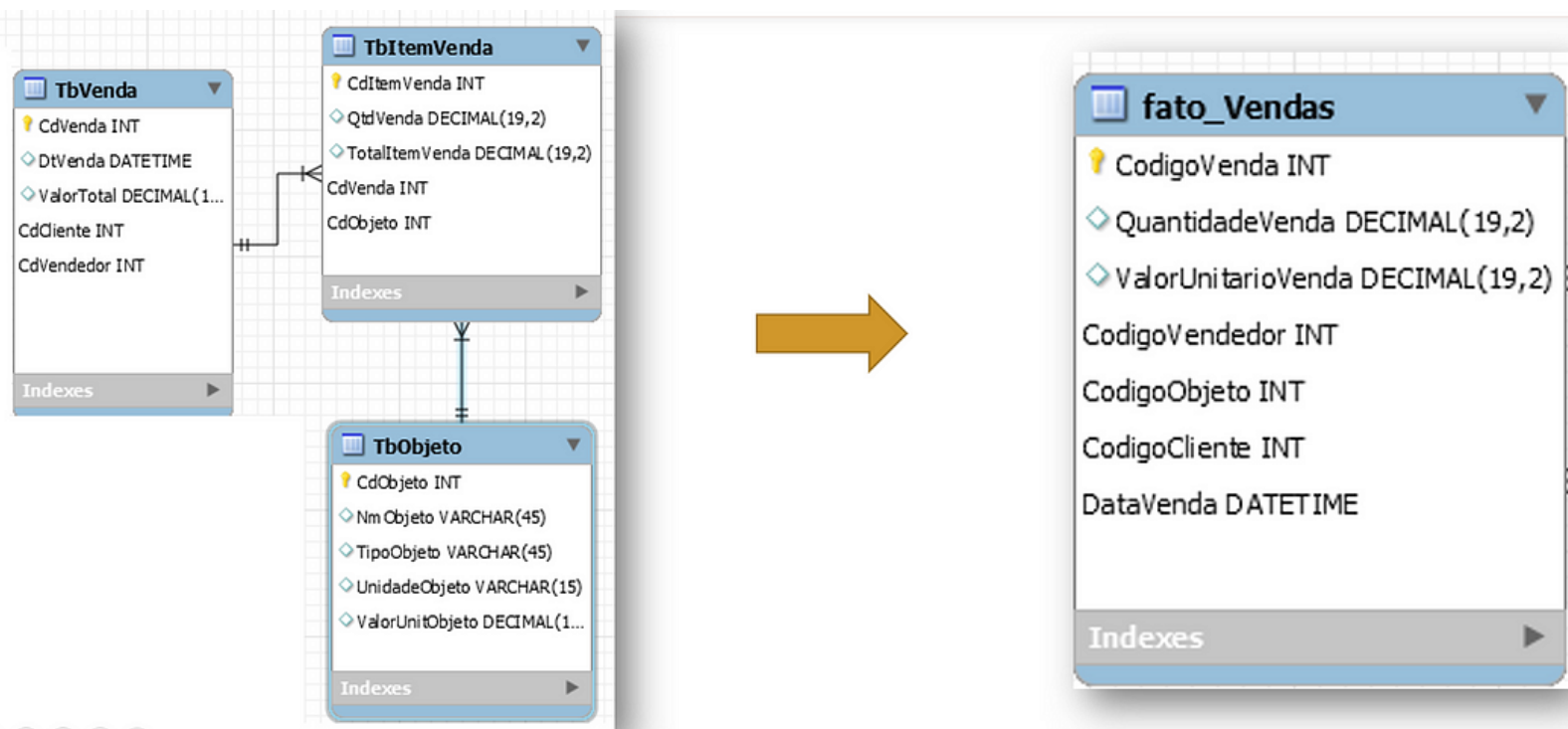


Técnicas de Modelagem Multidimensional

- Exemplo 3 — fato Vendas

Do Relacional para

o Multidimensional



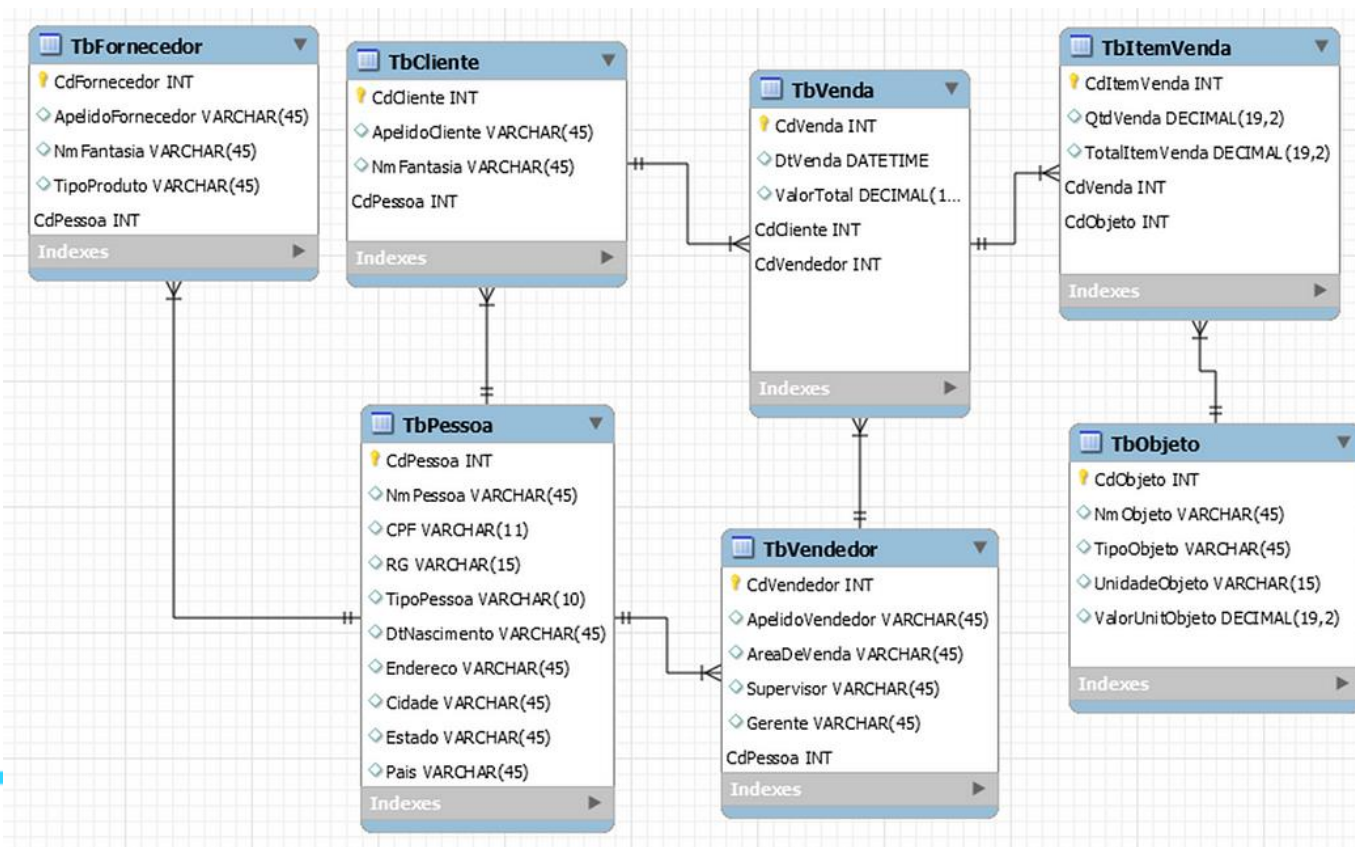
Técnicas de Modelagem Multidimensional

- **Exemplo 4 — completo**

Aqui é o modelo completo, transformando uma Modelagem Relacional (OLTP) para uma Modelagem Multidimensional

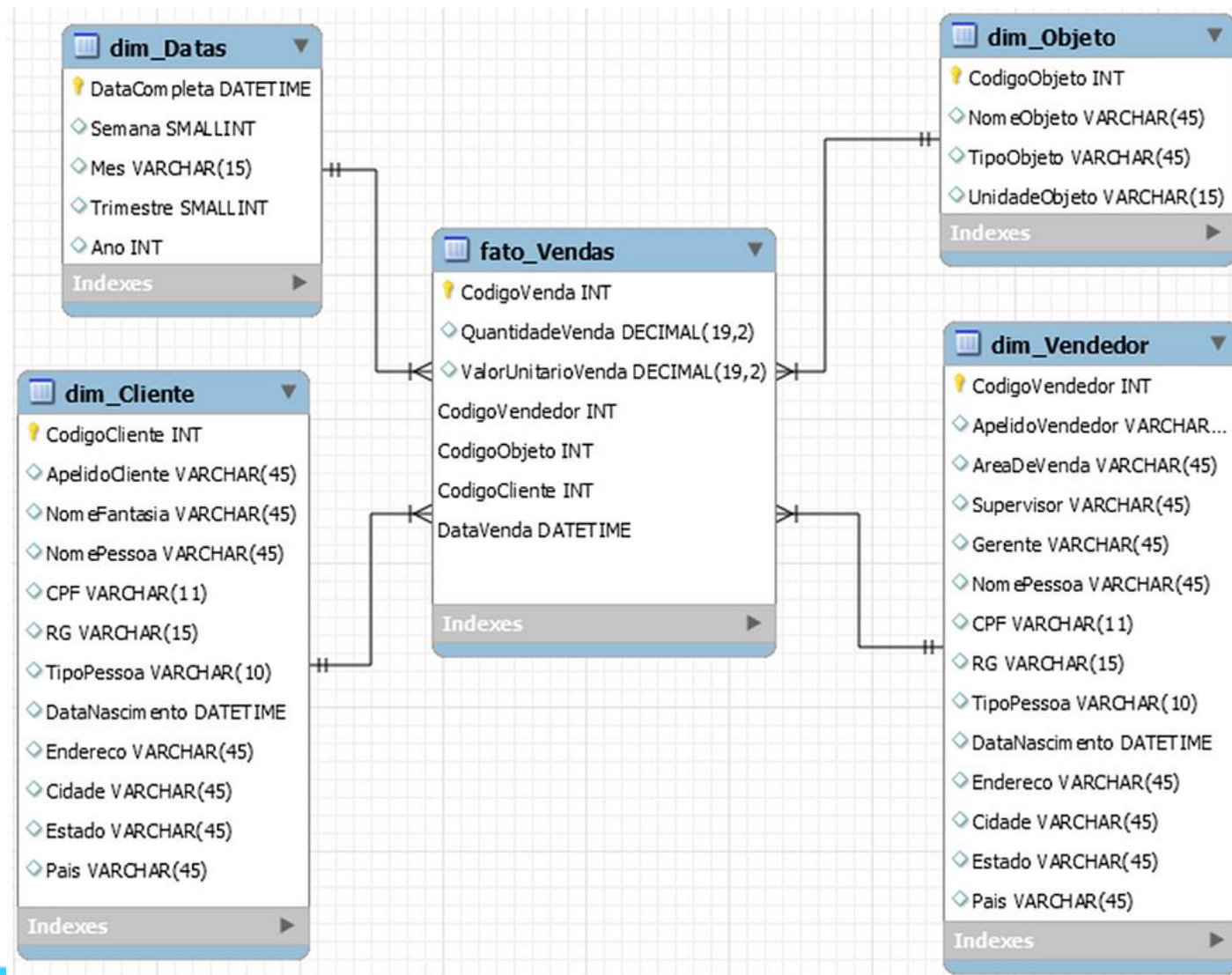
Técnicas de Modelagem Multidimensional

- Exemplo 4 — completo
Do Relacional



Técnicas de Modelagem Multidimensional

- Exemplo 4 — completo para o Multidimensional



Factless Fact Tables

- As tabelas de fato sem fato são encontradas na modelagem de data warehouse esta tabela não possui nenhuma medida ela contém apenas chaves estrangeiras para tabelas dimensionais, sendo suficiente para responder a questões relevantes.

Factless Fact Tables

- As tabelas de fatos sem fatos também podem ser usadas para analisar o que não aconteceu, essas consultas sempre têm duas partes:
 - uma tabela de cobertura sem fatos que contém todas as possibilidades de eventos que podem acontecer e
 - uma tabela de atividades que contém os eventos que ocorreram.
- Quando a atividade é subtraída da cobertura, o resultado é o conjunto de eventos que não ocorreram.

Sei que não é tudo maravilha... quais os problemas mais comuns no dia a dia?

- **Falta**

- Documentação
- Não tem Diagramas de Entidade e Relacionamento
- Não tem Dicionário de dados — [Gerar dicionário \(SQL Server\)](#)
- Na maior parte das vezes, temos apenas as tabelas do banco de dados
- “Data Warehouses” pegando dados direto de produção
- “Data Warehouses” sem utilizar modelagem multidimensional

Problemas na implementação de data warehouses

- Estabelecer expectativas que você **não pode cumprir**
- Carregar o *data warehouse* com informações **só** porque estão **disponíveis**
- Acreditar que **projetar** bases de dados com *data warehouses* é o mesmo que projetar bases de dados transacionais
- Escolher um gestor de *data warehouse* voltado para a tecnologia, em vez de **voltado aos usuários**

Visualização de dados

- É a apresentação de informações em formato imagético ou gráfico.
- Ela permite que tomadores de decisão vejam os resultados de análises visualmente, para que possam compreender conceitos difíceis ou identificar novos padrões.

Storytelling

- **Contar histórias** é a descrição detalhada de conceitos, opiniões e experiências pessoais por meio de narrativas que provocam respostas **emocionais** e **insights**.
- Trata-se de aproveitar as histórias para envolver seu público ou esclarecer qualquer coisa. É uma das maneiras mais fáceis de levar o público junto para tornar uma conversa animada e envolvente



Banco de Dados Aplicado

Cristiano Santos

cristiano.santos@amf.edu.br

