BUSINESS INTELLIGENCE

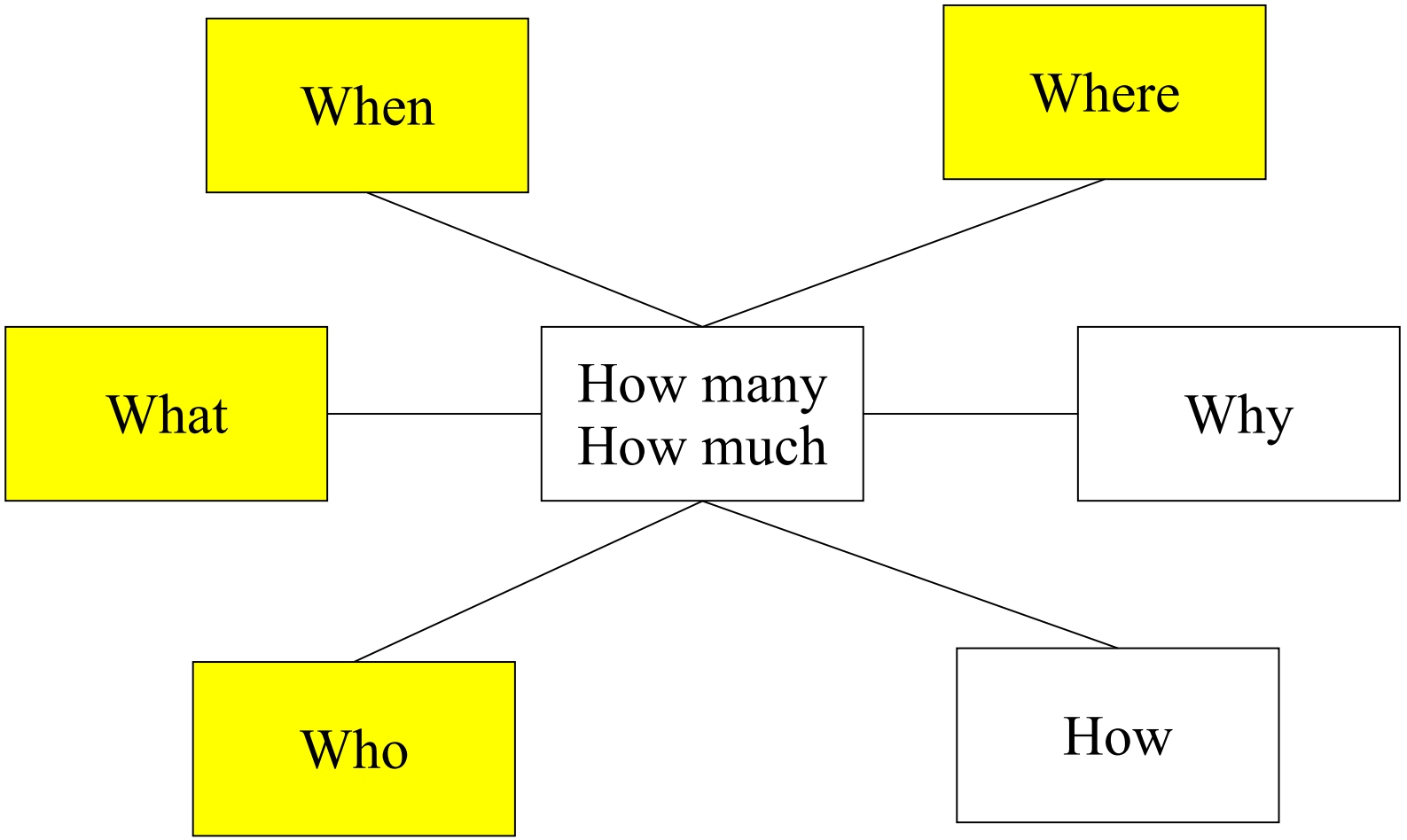
Asterio K. Tanaka
http://www.uniriotec.br/~tanaka/SAIN
tanaka@uniriotec.br

Modelagem Dimensional – Conceitos Avançados

Material baseado em originais de Maria Luiza Campos – NCE/UFRJ
Livro-Texto: The Data Warehouse Toolkit, Third Edition: The Definitive Guide to Dimensional Modeling
Ralph Kimball and Margy Ross, Wiley, 2013
Complementado com referências atuais de Ralph Kimball (http://www.kimballgroup.com/)

- Tabelas de Dimensões: campos chave e dimensões clássicas
  - When (Tempo, Data, Hora do Dia); What (Produto); Where (Loja); Why (Promoção)
- Tabelas de Fato sem Fatos (Métricas)
  - Cobertura (Promoção) e Evento
- Dimensões Degeneradas (dimensões sem tabelas)
- Extensibilidade do esquema estrela
- Modelo dimensional normalizado: Esquema Snow Flake
- Esquemas com muitas dimensões: Esquema Centopéia
- Dinâmica das Dimensões: Slowly Changing Dimension
- Dimensões com Papéis (Role Playing dimensions)
- Outros Tipos Especiais de Dimensão
  - Lixo (Junk Dimension); Dimensões muito grandes: Minidimensões; Dimensões com "outrigger"; Dimensões Multivaloradas (Bridge table)
- Tópicos Especiais sobre Fatos
  - Fatos conformados, Bus Matrix de Implementação, Tipos Clássicos de Fatos
- Agregados

Esquema Estrela de DW
5 W e 3 H



Estudo de Caso
Vendas a Varejo (Kimball 2013)

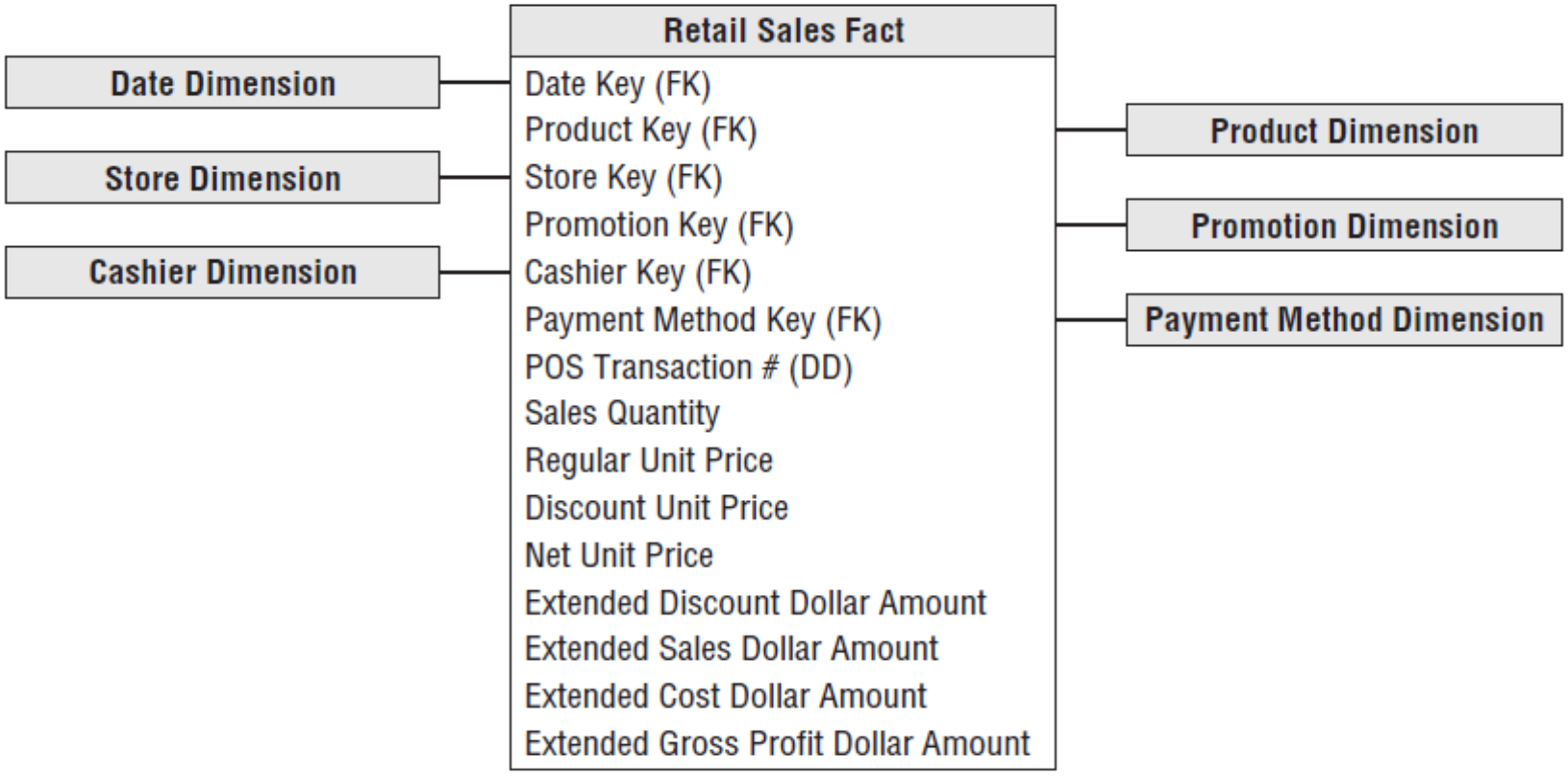


Figure 3-3: Measured facts in retail sales schema.

Campos Chaves de Tabela de Dimensões

- **Regra básica: uso de surrogates ou chaves artificiais.**
  - Ajudam a manter a estabilidade, através da neutralidade.
  - Evitam manutenção custosa de tabelas, especialmente das tabelas fatos.
  - Chaves naturais podem ter problemas de unicidade, ausência, tamanhos exagerados.
  - Chaves artificiais podem ser especificadas como inteiros de 4 bytes, alcançando até 2^{32}, isto é, mais de 2 bilhões de ocorrências (inteiros positivos), o que é mais do que necessário para qualquer tabela dimensão.
  - Chaves artificiais ficam transparentes (invisíveis) para os usuários, servindo apenas como ligação entre dimensões e fatos.
  - Campos naturais não chave poderão ser indexados, tornando as consultas amistosas.
  - Se produzidas automaticamente, deve-se ter cuidado no processo de preparação (ETL), especialmente nos reprocessamentos.
  - A única desvantagem das chaves artificiais é que não faz sentido a tabela fato ser consultada diretamente, pois os campos descritivos de filtro estarão armazenados nas dimensões.
- *Every join between dimension and fact tables in the data warehouse should be based on meaningless integer surrogate keys. You should avoid using the natural operational production codes. None of the data warehouse keys should be "smart", where you can tell something about the row just by looking at the key.*
- **Exceção para a Dimensão Data**. Atualmente se usa o formato AAAAMMDD.

Dimensão Tempo/Data (When)

- A dimensão Tempo (Data) é muito poderosa e importante em todo DW. Como tal deve ser tratada de forma diferenciada em relação às outras dimensões. Usualmente está presente em todo Data Mart, pois o DW é histórico.
- Costuma ser complexa no mundo real:
  - Dia, Mês, Trimestre, Semestre, Ano
  - Dia Acumulado no Mês, no Ano
  - Período Fiscal, Semana de Cinco Dias
  - Feriados, Fim de semana
- Qual a granularidade ideal? É claro, depende do projeto
  - Com granularidade diária, podemos organizar os dados por dias, meses, anos, por períodos fiscais (artificiais) da empresa, etc. Essa modelagem é mais flexível a mudanças nos requisitos do negócio.
- Diferente das outras dimensões, a tabela Data pode ser carregada antecipadamente, de uma só vez e não requer fonte de dados
  - Exemplo: 5 anos passados + 5 anos futuros = 10 anos = 3.650 dias (linhas na tabela)
  - Sendo o grão de tempo a data, a chave primária usada é um inteiro no formato AAAAMMDD.

Date Dimension
Date Key (PK)
Date
Full Date Description
Day of Week
Day Number in Calendar Month
Day Number in Calendar Year
Day Number in Fiscal Month
Day Number in Fiscal Year
Last Day in Month Indicator
Calendar Week Ending Date
Calendar Week Number in Year
Calendar Month Name
Calendar Month Number in Year
Calendar Year-Month (YYYY-MM)
Calendar Quarter
Calendar Year-Quarter
Calendar Year
Fiscal Week
Fiscal Week Number in Year
Fiscal Month
Fiscal Month Number in Year
Fiscal Year-Month
Fiscal Quarter
Fiscal Year-Quarter
Fiscal Half Year
Fiscal Year
Holiday Indicator
Weekday Indicator
SQL Date Stamp
...

Figure 3-4: Date dimension table.

Dimensão Data esquema

Tipo de dados SQL (Date, Time) não suportam essa riqueza de descrições, daí a necessidade de uma dimensão Data explícita.

Dimensão Data amostra

Date Key	Date	Full Date Description	Day of Week	Calendar Month	Calendar Quarter	Calendar Year	Fiscal Year-Month	Holiday Indicator	Weekday Indicator
20130101	01/01/2013	January 1, 2013	Tuesday	January	Q1	2013	F2013-01	Holiday	Weekday
20130102	01/02/2013	January 2, 2013	Wednesday	January	Q1	2013	F2013-01	Non-Holiday	Weekday
20130103	01/03/2013	January 3, 2013	Thursday	January	Q1	2013	F2013-01	Non-Holiday	Weekday
20130104	01/04/2013	January 4, 2013	Friday	January	Q1	2013	F2013-01	Non-Holiday	Weekday
20130105	01/05/2013	January 5, 2013	Saturday	January	Q1	2013	F2013-01	Non-Holiday	Weekday
20130106	01/06/2013	January 6, 2013	Sunday	January	Q1	2013	F2013-01	Non-Holiday	Weekday
20130107	01/07/2013	January 7, 2013	Monday	January	Q1	2013	F2013-01	Non-Holiday	Weekday
20130108	01/08/2013	January 8, 2013	Tuesday	January	Q1	2013	F2013-01	Non-Holiday	Weekday

Figure 3-5: Date dimension sample rows.

Vide planilha Excel fornecida pelo Kimball Group

http://www.kimballgroup.com/data-warehouse-business-intelligence-resources/books/data-warehouse-dw-toolkit/

Por que não usar Flags e Indicadores nas Dimensões?

Monthly Sales
Period: June 2013
Product Baked Well Sourdough

Holiday Indicator	Extended Sales Dollar Amount
N	1,009
Y	6,298

OR

Monthly Sales
Period: June 2013
Product Baked Well Sourdough

Holiday Indicator	Extended Sales Dollar Amount
Holiday	6,298
Non-holiday	1,009

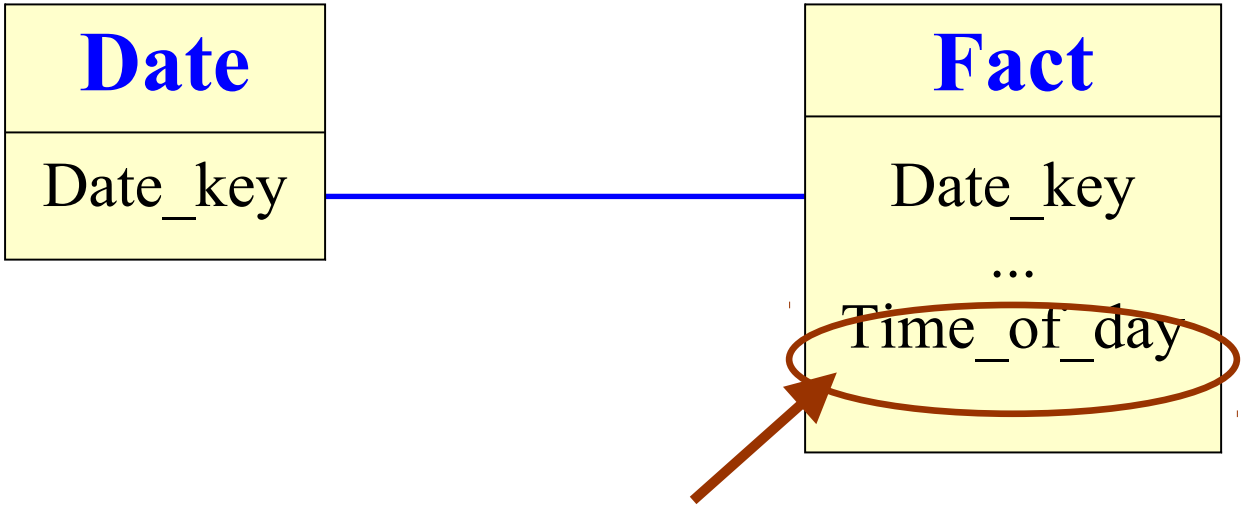
Figure 3-6: Sample reports with cryptic versus textual indicators.

Atributos de dimensão servem como rótulos em relatórios e valores em listas de filtros *pull down*.

Dimensão tempo: Horas, Minutos, Segundos

Várias soluções são possíveis, graças à extensibilidade do modelo dimensional.

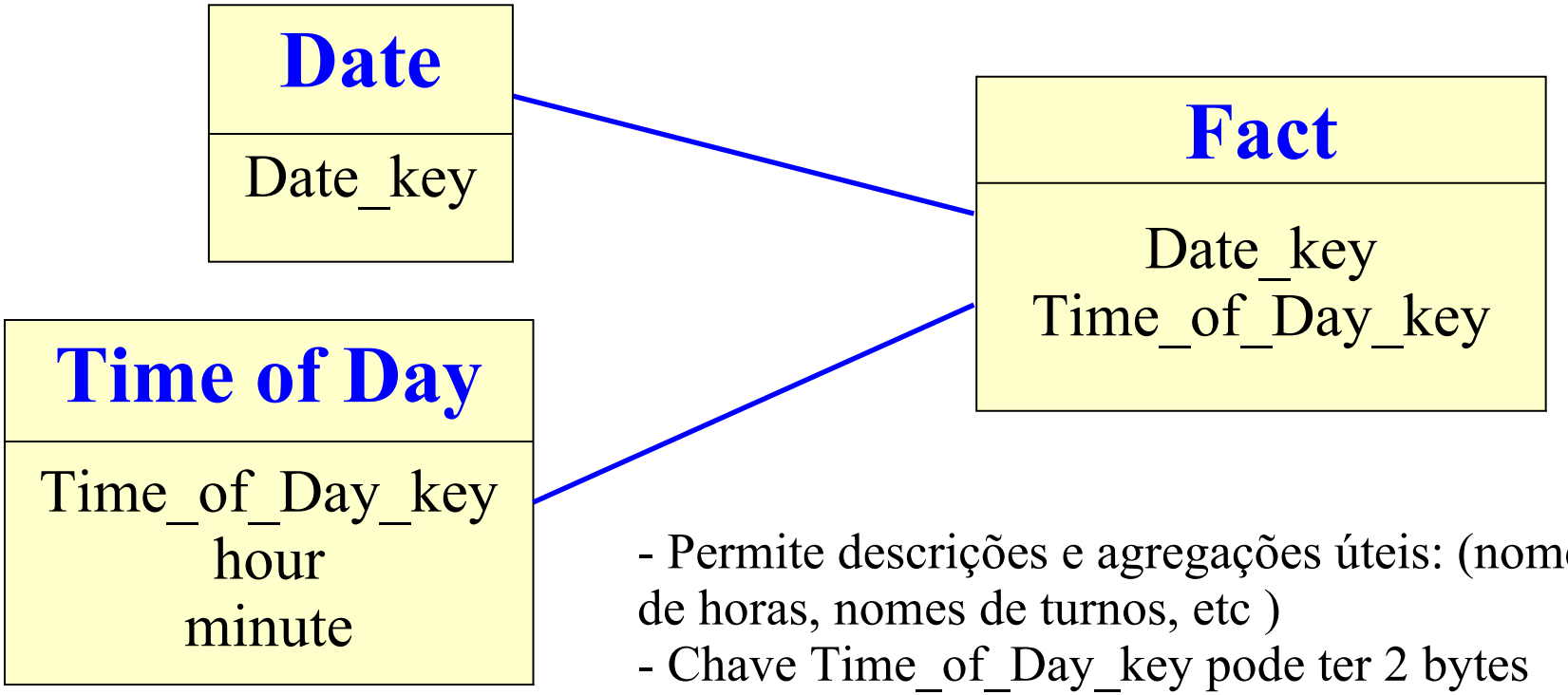
1ª Alternativa: Colocar a "hora do dia" na Tabela de Fatos



- Pode ser usado quando não há descrições adicionais sobre a hora do dia.
- Pode sobrecarregar a tabela de fatos (tipo Timestamp requer 8 bytes)
8 bytes x bilhões de linhas na tabela de fatos ...

Dimensão tempo: Horas, Minutos, Segundos

2ª Alternativa: Criar uma Dimensão Hora do Dia (recomendada por Kimball)



- Permite descrições e agregações úteis: (nomes de horas, nomes de turnos, etc)
- Chave Time_of_Day_key pode ter 2 bytes (suficiente para 24 x 60 = 1.440 minutos) ou 4 bytes (suficiente para 1.440 x 60 = 86.400 segundos).

Dimensão tempo: Horas, Minutos, Segundos

3ª Alternativa : Hora, minuto na mesma tabela de dimensão que as datas

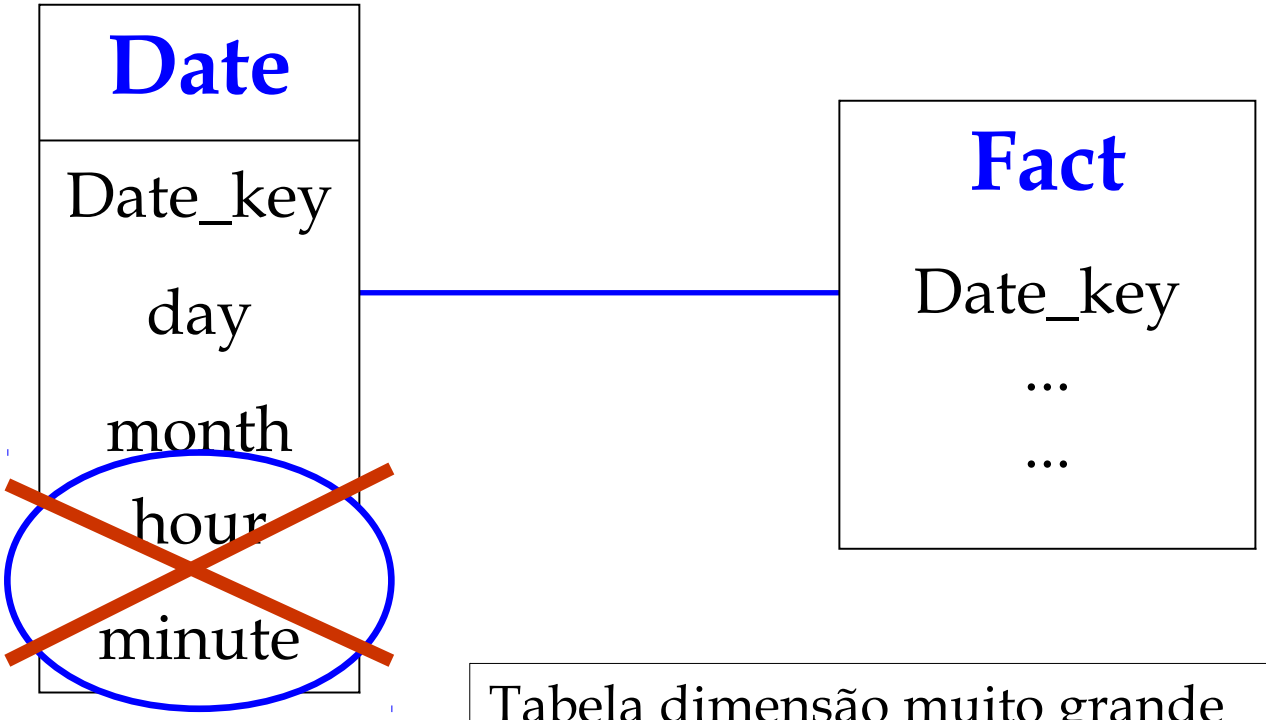


Tabela dimensão muito grande
10 anos = 3.650 x 1.440 minutos = 5.256.000 linhas (525.600 linhas cada ano adicional)

Estudo de Caso
Vendas a Varejo (Kimball 2013)

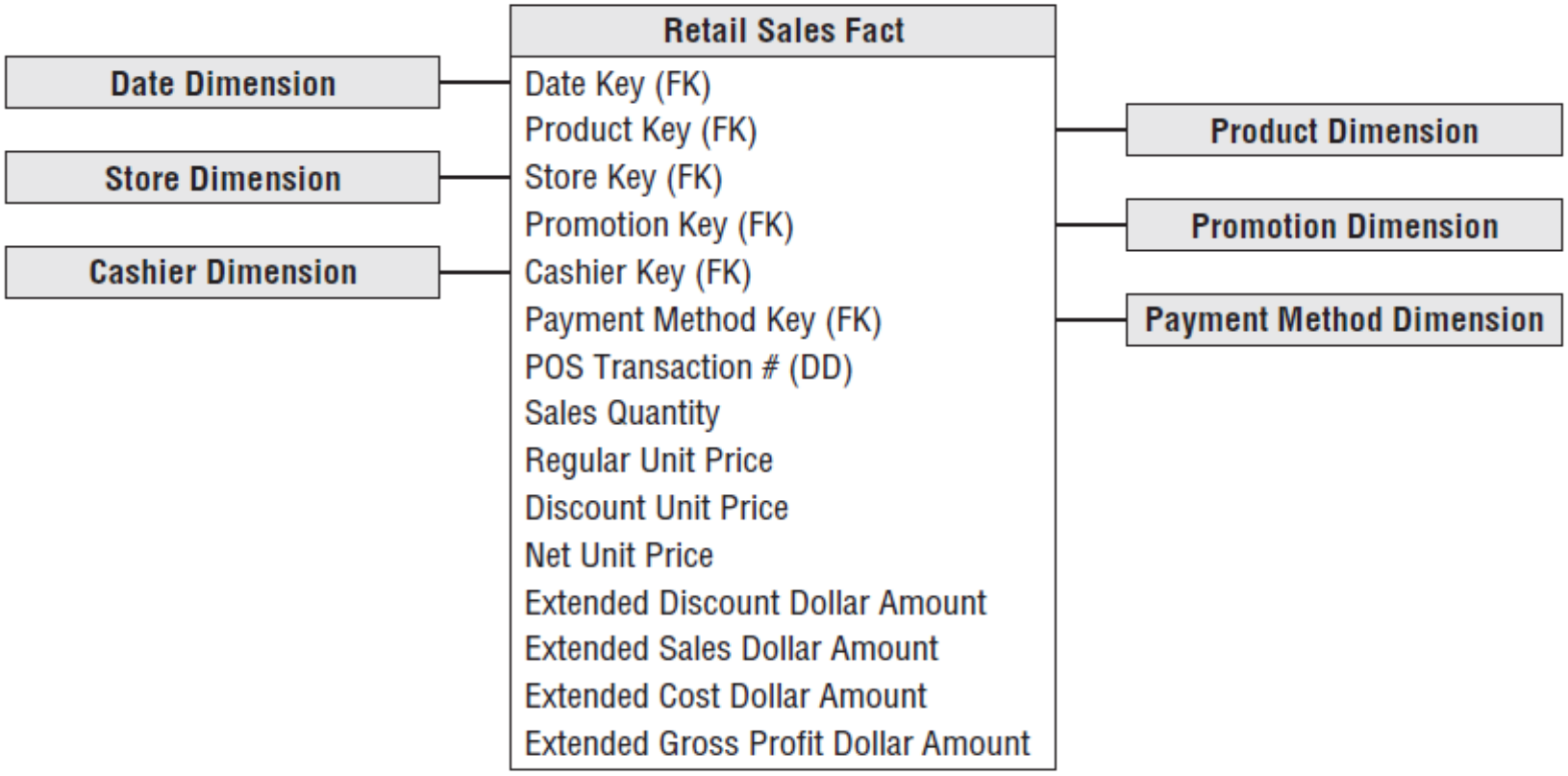


Figure 3-3: Measured facts in retail sales schema.

Product Dimension
Product Key (PK)
SKU Number (NK)
Product Description
Brand Description
Subcategory Description
Category Description
Department Number
Department Description
Package Type Description
Package Size
Fat Content
Diet Type
Weight
Weight Unit of Measure
Storage Type
Shelf Life Type
Shelf Width
Shelf Height
Shelf Depth
...

Figure 3-8: Product dimension table.

Dimensão Produto (What) - esquema

Redundância à custa da 3FN vale a pena, pois as tabelas de dimensões são pequenas em relação às tabelas de fatos.

Dimensão Produto amostra

Product Key	Product Description	Brand Description	Subcategory Description	Category Description	Department Description	Fat Content
1	Baked Well Light Sourdough Fresh Bread	Baked Well	Fresh	Bread	Bakery	Reduced Fat
2	Fluffy Sliced Whole Wheat	Fluffy	Pre-Packaged	Bread	Bakery	Regular Fat
3	Fluffy Light Sliced Whole Wheat	Fluffy	Pre-Packaged	Bread	Bakery	Reduced Fat
4	Light Mini Cinnamon Rolls	Light	Pre-Packaged	Sweeten Bread	Bakery	Non-Fat
5	Diet Lovers Vanilla 2 Gallon	Coldpack	Ice Cream	Frozen Desserts	Frozen Foods	Non-Fat
6	Light and Creamy Butter Pecan 1 Pint	Freshlike	Ice Cream	Frozen Desserts	Frozen Foods	Reduced Fat
7	Chocolate Lovers 1/2 Gallon	Frigid	Ice Cream	Frozen Desserts	Frozen Foods	Regular Fat
8	Strawberry Ice Creamy 1 Pint	Icy	Ice Cream	Frozen Desserts	Frozen Foods	Regular Fat
9	Icy Ice Cream Sandwiches	Icy	Novelties	Frozen Desserts	Frozen Foods	Regular Fat

Figure 3-7: Product dimension sample rows.

Department Name	Sales Dollar Amount
Bakery	12,331
Frozen Foods	31,776

Drill down by brand name:

Department Name	Brand Name	Sales Dollar Amount
Bakery	Baked Well	3,009
Bakery	Fluffy	3,024
Bakery	Light	6,298
Frozen Foods	Coldpack	5,321
Frozen Foods	Freshlike	10,476
Frozen Foods	Frigid	7,328
Frozen Foods	Icy	2,184
Frozen Foods	QuickFreeze	6,467

Or drill down by fat content:

Department Name	Fat Content	Sales Dollar Amount
Bakery	Nonfat	6,298
Bakery	Reduced fat	5,027
Bakery	Regular fat	1,006
Frozen Foods	Nonfat	5,321
Frozen Foods	Reduced fat	10,476
Frozen Foods	Regular fat	15,979

Figure 3-9: Drilling down on dimension attributes.

Drill Down em hierarquias de Produto

Asterio K. Tanaka

Dimensão Loja (Where)

Store Dimension
Store Key (PK)
Store Number (NK)
Store Name
Store Street Address
Store City
Store County
Store City-State
Store State
Store Zip Code
Store Manager
Store District
Store Region
Floor Plan Type
Photo Processing Type
Financial Service Type
Selling Square Footage
Total Square Footage
First Open Date
Last Remodel Date
...

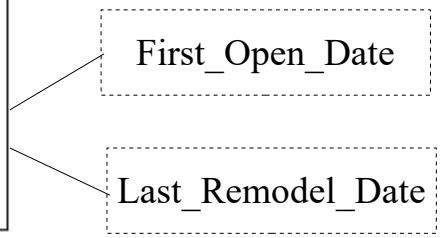
Figure 3-10: Store dimension table.

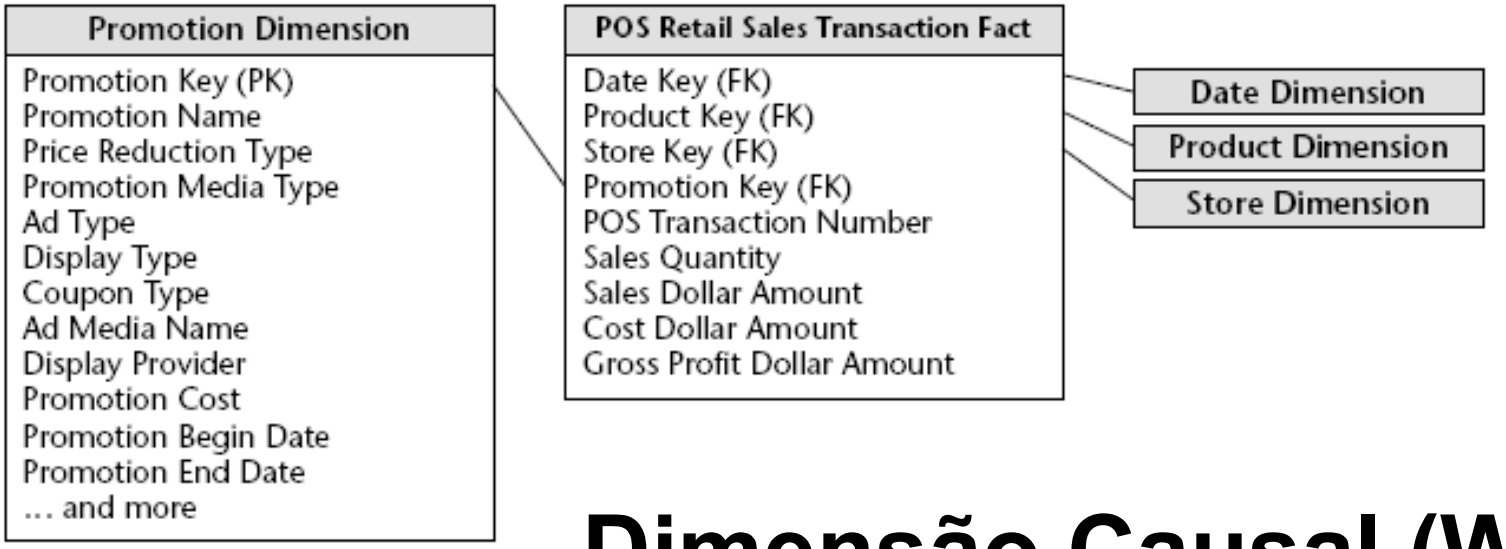
Note os atributos First Open Date e Last Remodel Date, são DATAS.

São chaves de junção com cópias da tabela de dimensão Date, declaradas como visões SQL, por exemplo

```
        CREATE VIEW First_Open_Date
(FO_day_number, FO_month, ...)
        AS SELECT day_number, month
        FROM Date
```

Esse tipo de tabela virtual para relacionar dimensões é denominado “outrigger”, com veremos adiante.





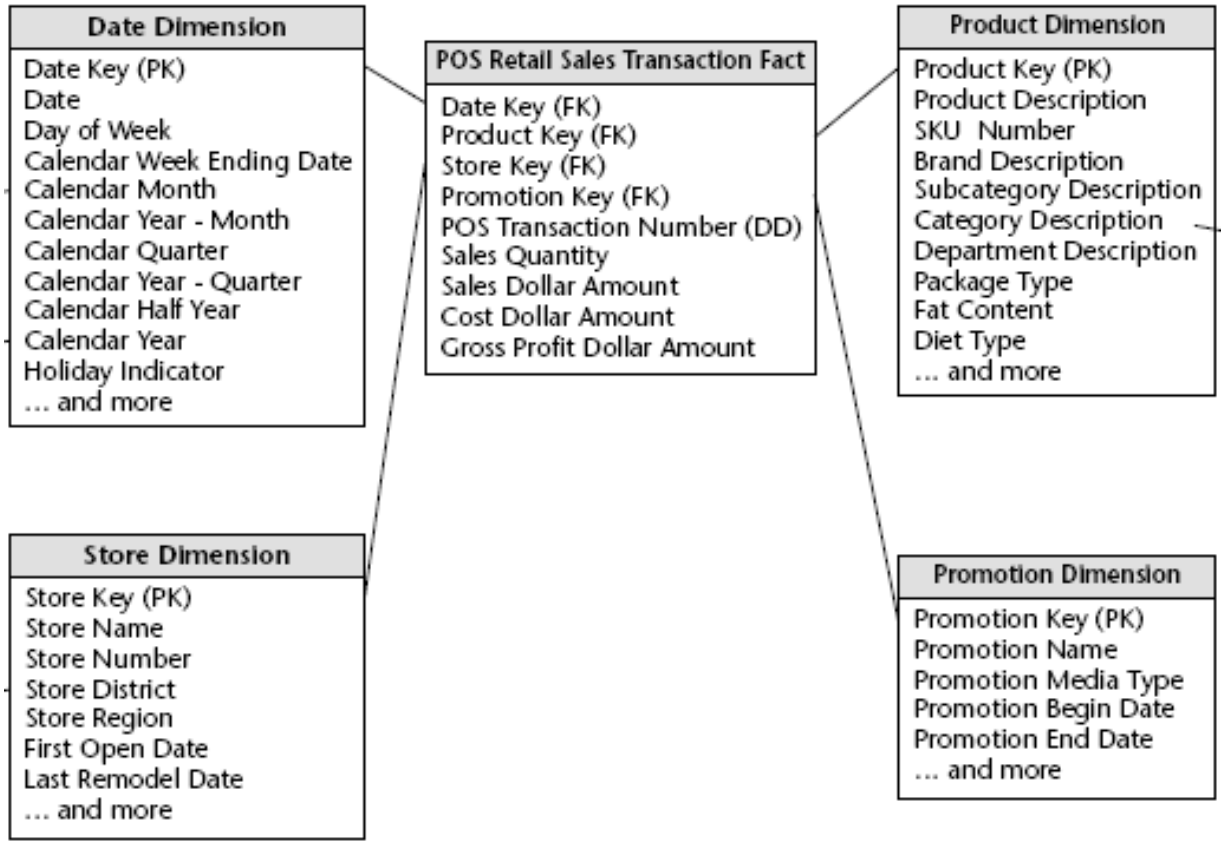
Dimensão Causal (Why) Ex: Promoção

- A dimensão Promoção do exemplo é, de fato uma COMBINAÇÃO DE DIMENSÕES causais (price reduction, ads, display, coupon) que poderiam estar em quatro tabelas separadas, com o mesmo efeito.
- No caso, estão combinadas numa única tabela de dimensão porque são altamente correlatas.
- Dimensões combinadas economizam espaço da tabela de Fatos, embora separadas pudessem ser mais bem entendidas e mais facilmente administradas.

Dimensões sem Tabelas
Dimensões Degeneradas

- Chaves de dimensão na tabela de fatos sem tabelas de dimensão correspondentes.
- Uma chave de dimensão, como o número de uma transação, número de fatura, tiquete, nota fiscal, pedido ou ordem de compra, que não tenha nenhum atributo portanto não se junta com uma tabela de dimensão.
- Esses documentos normalmente são compostos de itens, e se a granularidade da tabela de fatos for item, o número do documento estará na tabela fato apenas para permitir o agrupamento dos itens por documento.
- Dimensões degeneradas são, também, entradas para a operação de *Drill Through*, para buscar dados nos sistemas transacionais que não fizeram parte da ETL.

Dimensões sem Tabelas
Dimensões Degeneradas



POS Transaction Number é uma Dimensão Degenerada (DD)

Esquema Vendas a Varejo em Ação

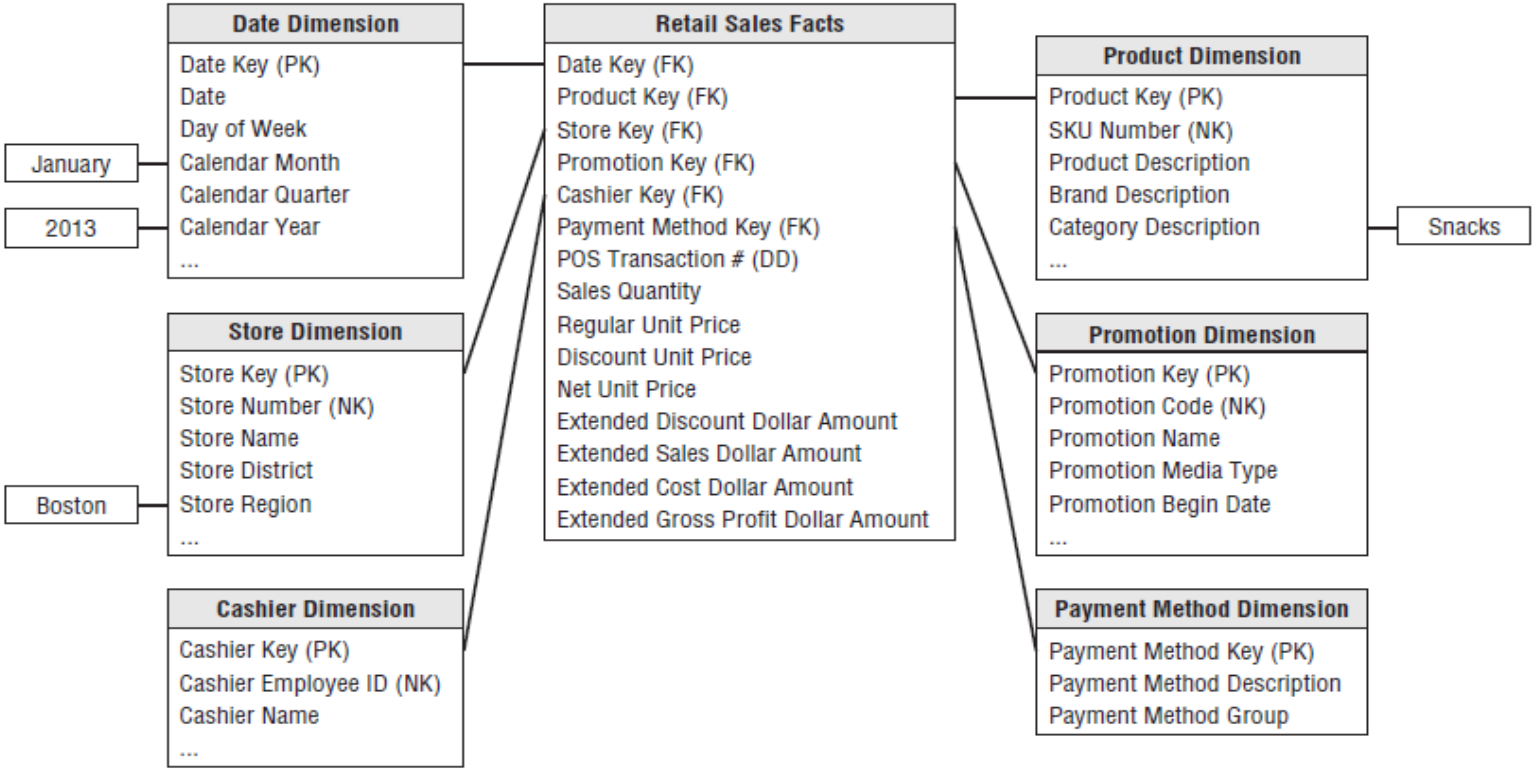


Figure 3-12: Querying the retail sales schema.

With our retail POS schema designed, let's illustrate how it would be put to use in a query environment. A business user might be interested in better understanding weekly sales dollar volume by promotion for the snacks category during January 2013 for stores in the Boston district. As illustrated in Figure 3-12, you would place query constraints on month and year in the date dimension, district in the store dimension, and category in the product dimension.

Esquema Vendas a Varejo em Ação

Calendar Week Ending Date	Promotion Name	Extended Sales Dollar Amount
January 6, 2013	No Promotion	2,647
January 13, 2013	No Promotion	4,851
January 20, 2013	Super Bowl Promotion	7,248
January 27, 2013	Super Bowl Promotion	13,798

Calendar Week Ending Date	No Promotion Extended Sales Dollar Amount	Super Bowl Promotion Extended Sales Dollar Amount
January 6, 2013	2,647	0
January 13, 2013	4,851	0
January 20, 2013	0	7,248
January 27, 2013	0	13,798

Figure 3-13: Query results and cross-tabular report.

Tabela de Fatos sem Fatos (Factless Fact Tables)

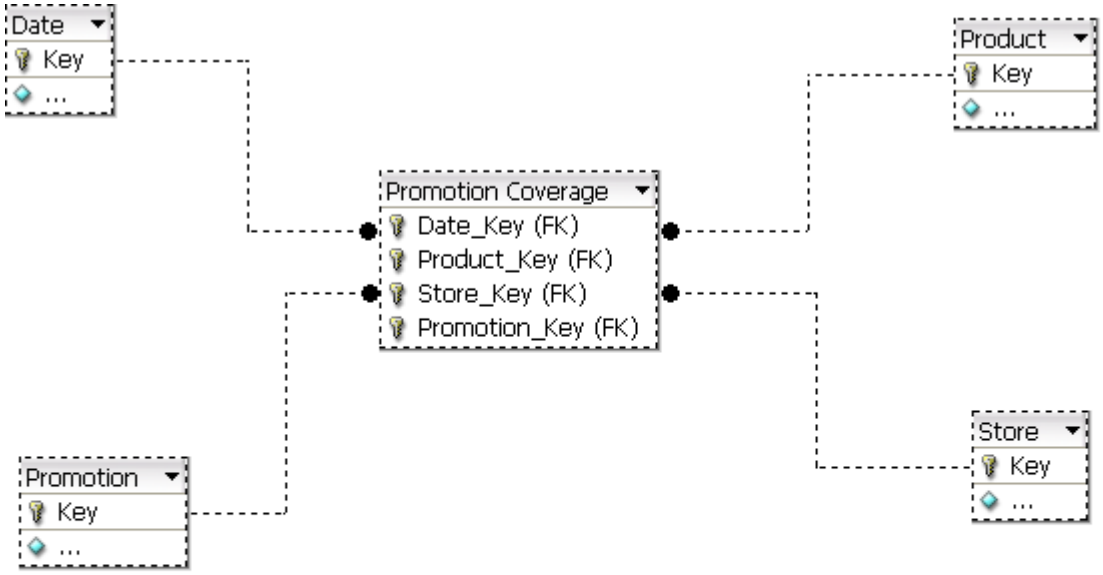
- Uma tabela de fatos que não tem fatos mas captura alguns relacionamentos muitos-para-muitos entre chaves de dimensões. Mais frequentemente usada para representar eventos ou prover informação de cobertura que não aparece em outras tabelas de fatos.
- A tabela de fatos Vendas com medidas (Retail Sales Facts) não pode responder a consultas do tipo
  - Quais produtos estavam em promoção mas não venderam?

  Por que não pode? Por que não deveria?

  A solução é criar uma Tabela de Cobertura de promoção com as mesmas dimensões da tabela de Vendas (Data, Produto, Loja, Promoção).

  Os produtos em promoção que não venderam será o conjunto diferença entre a cobertura e as vendas.

Tabela de Fatos sem Fatos
Cobertura de Promoção



Uma tabela de fatos, tipicamente sem fatos, que registra todos os produtos que estão em promoção numa determinada loja, independentemente de ser vendidos ou não.

Consulta: Quais produtos estavam em promoção mas não venderam?

```
SELECT Product_Key, ... FROM Promotion_Coverage, ... WHERE ...
MINUS
SELECT Product_Key. ... FROM POS_Retail_Sales, ... WHERE ...
```

Tabela de Fatos sem Fatos
Cobertura de Promoção

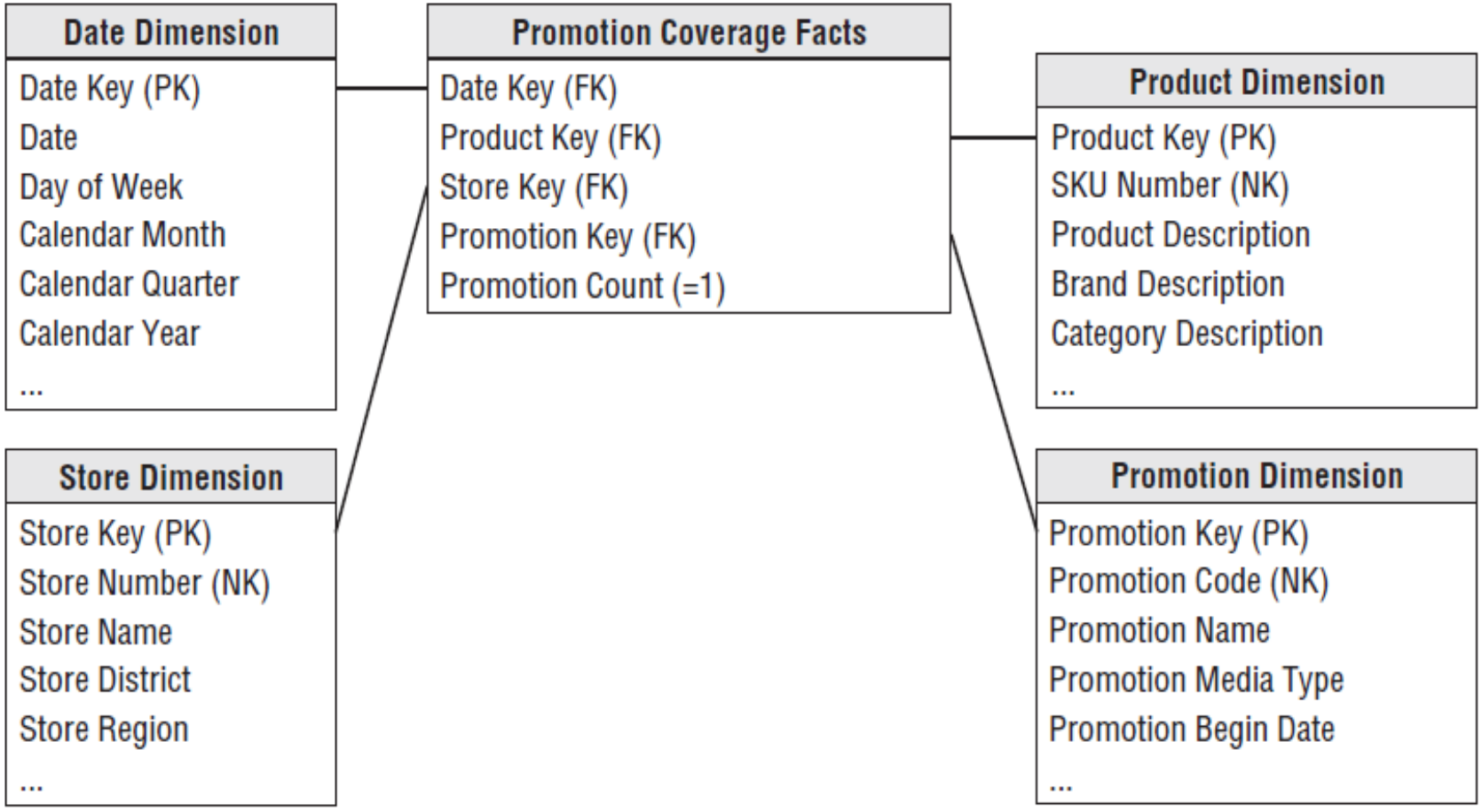
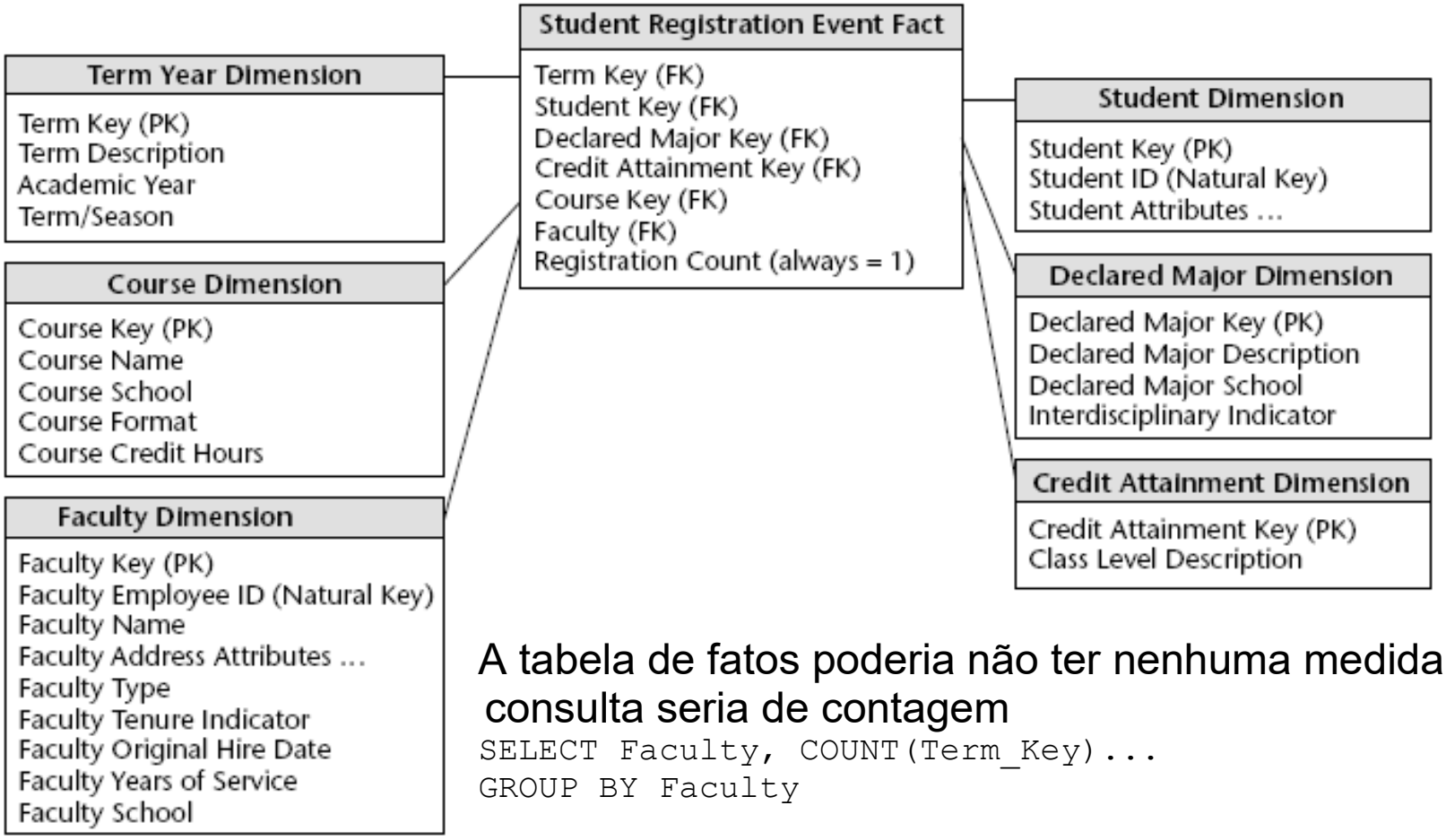


Figure 3-14: Promotion coverage factless fact table.

Tabela de Fatos sem Fatos - Eventos



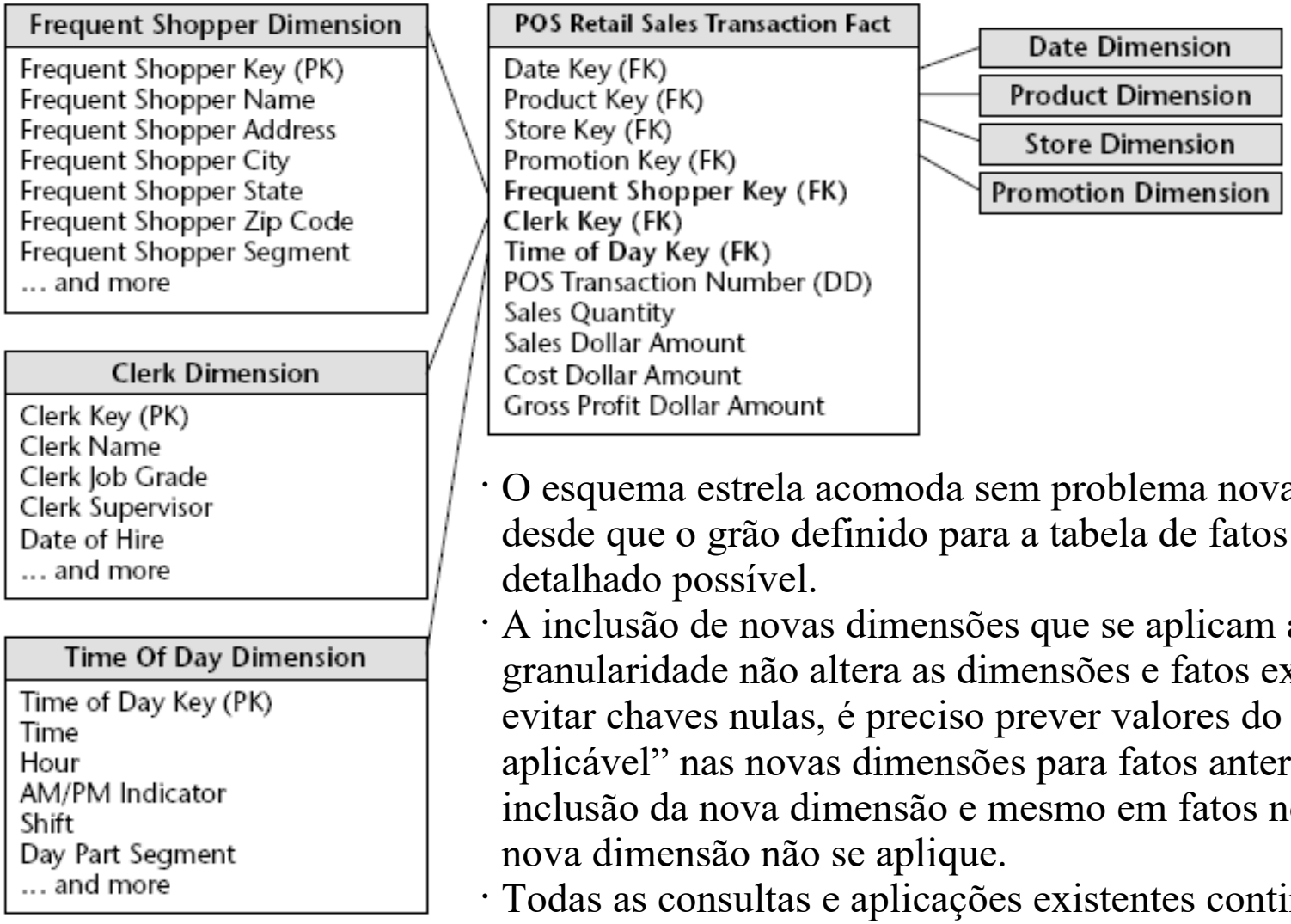
A tabela de fatos poderia não ter nenhuma medida e a consulta seria de contagem

```
SELECT Faculty, COUNT(Term_Key)...
GROUP BY Faculty
```

Ou poderia ter uma medida artificial Registration_Count apenas para tornar mais fácil a consulta

```
SELECT Faculty, SUM(Registration_Count)...
GROUP BY Faculty
```

Extensibilidade do Esquema Estrela



- O esquema estrela acomoda sem problema novas dimensões desde que o grão definido para a tabela de fatos seja o mais detalhado possível.
- A inclusão de novas dimensões que se aplicam a esse nível de granularidade não altera as dimensões e fatos existentes. Para evitar chaves nulas, é preciso prever valores do tipo "Não aplicável" nas novas dimensões para fatos anteriores à inclusão da nova dimensão e mesmo em fatos novos em que a nova dimensão não se aplique.
- Todas as consultas e aplicações existentes continuam a rodar sem nenhuma alteração.

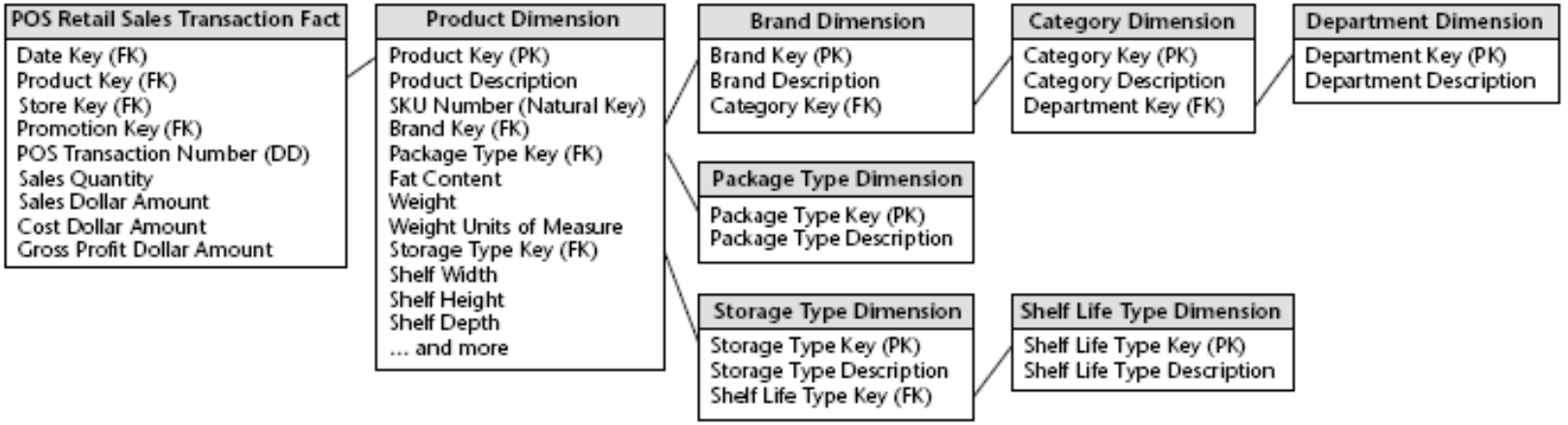
Extensibilidade do Esquema Estrela

Modificações absorvidas naturalmente pelo esquema estrela, devido a mudança nas fontes ou por decisão de modelagem, sem impacto nas aplicações existentes

- **Novos atributos de dimensões**
- **Novas dimensões**
- **Novos fatos medidos (na mesma tabela de fatos ou em nova tabela)**
- **Adição de uma fonte de dados nova envolvendo dimensões existentes assim como novas dimensões não previstas**

A extensibilidade é possível graças à simetria do esquema estrela, contanto que o grão inicial escolhido seja o mais detalhado possível pelos sistemas transacionais.

Esquema Dimensional Snow Flake



Embora aceitável, a normalização de dimensões não é recomendável por razões de desempenho e facilidade de uso

- **A quantidade de tabelas torna a apresentação do modelo mais complexa.**
- **Otimizadores do SGBD têm mais dificuldade com esquema complexo.**
- **A economia de espaço em disco é insignificante em relação ao DW completo.**
- **Snowflaking diminui a habilidade de usuários de navegar na dimensão.**
- **Snowflaking impede o uso de índices tipo Bit Map, que são usados por SGBD para indexar campos com baixa cardinalidade.**

Dimensões Outrigger

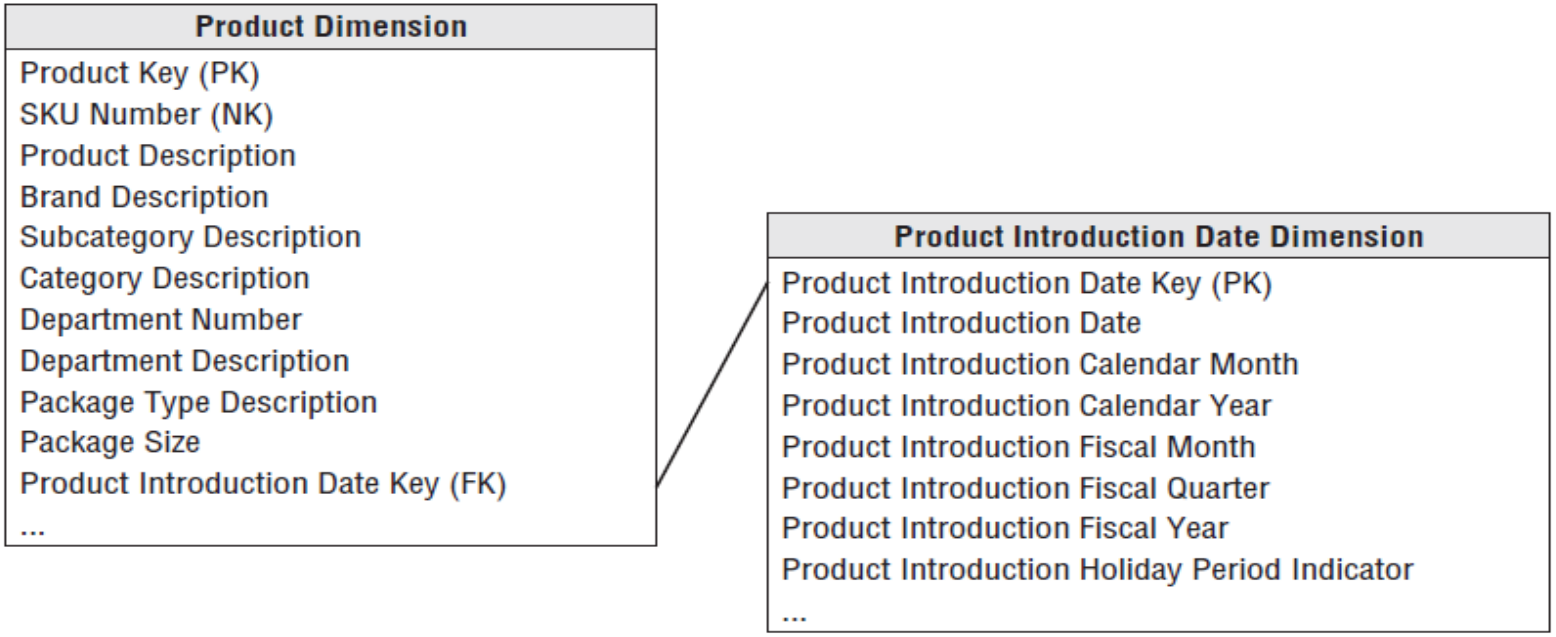
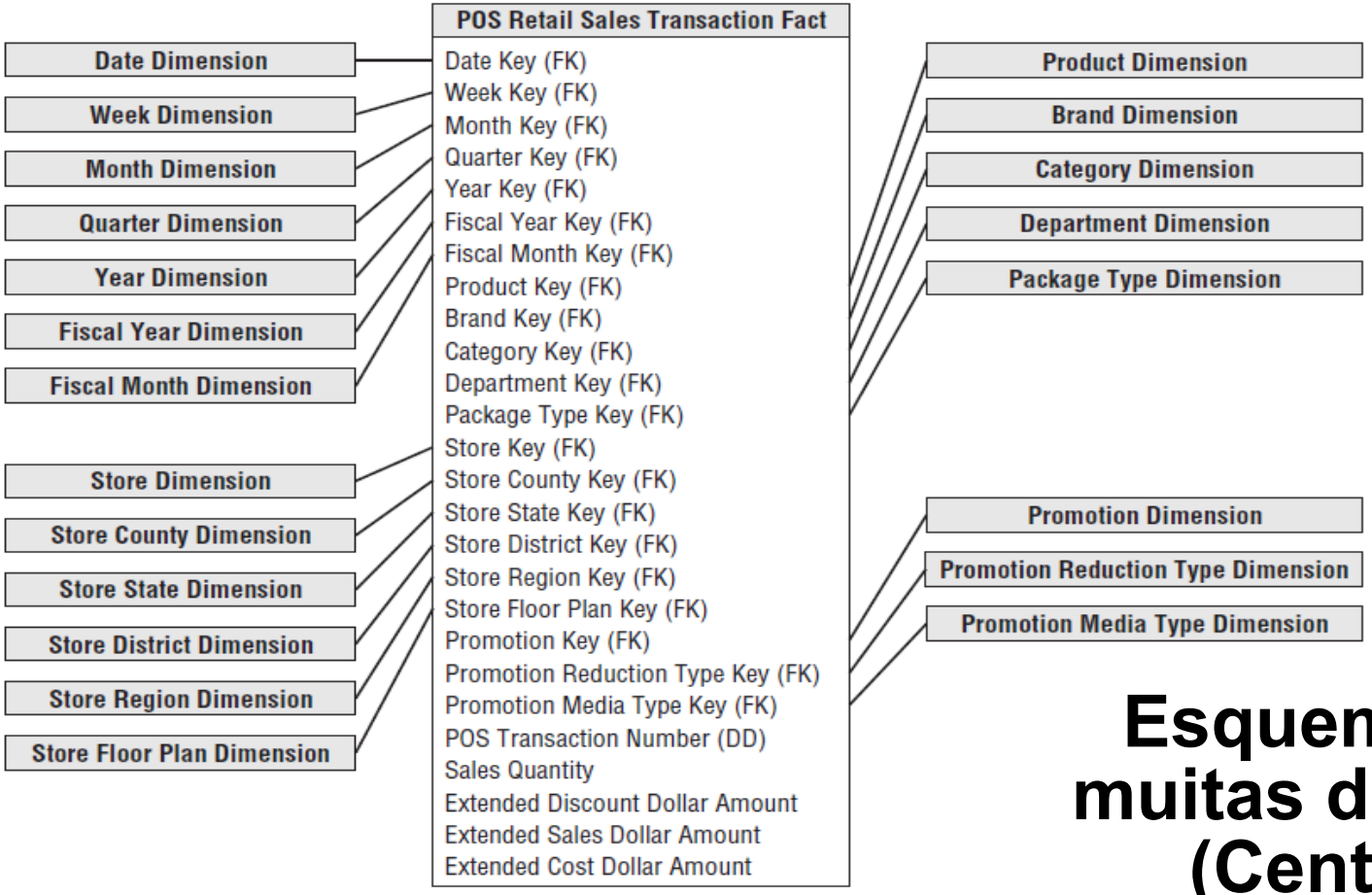


Figure 3-16: Example of a permissible outrigger.

Exemplo de situação em Retail Sales em que a normalização é aceitável.

Outrigger é uma tabela de dimensão secundária anexada a uma tabela de dimensão.

Outros exemplos ocorrem na dimensão Cliente (aplicações de CRM)



Esquemas com muitas dimensões (Centopéia)

Figure 3-17: Centipede fact table with too many normalized dimensions.

Um número de dimensões muito grande (>25) é um sinal de que muitas dimensões não são completamente independentes e deveriam ser combinadas numa única. Em geral, é um erro em modelagem dimensional representar elementos de uma hierarquia como dimensões separadas (a *menos que tenha um papel relevante em outro processo de negócio*).

Dinâmica das Dimensões

- Atualização das dimensões que mudam lentamente (Slowly Changing Dimensions)
  - Exemplos: Endereço de Cliente, Departamento de Produto.
- Várias alternativas
  - Tipo 1: Atualizar por cima do valor antigo
    - » É simples mas não preserva histórico.
  - Tipo 2: Adicionar uma nova linha com o novo valor do atributo atualizado, mantendo os demais.
    - » A nova linha particiona o histórico na tabela fato.
    - » É a técnica predominante para dimensões que mudam lentamente (slowly changing dimensions).
  - Tipo 3: Adicionar uma nova coluna, preservando o valor anterior e inserindo o novo valor na nova coluna.
    - » Permite a manutenção de duas visões simultâneas do histórico, mas dá margem a muitos valores nulos quando as mudanças são lentas.
  - Soluções híbridas, com múltiplas versões (linhas) combinadas ou não com coluna de valor anterior.
    - » Mais flexíveis e completas, porém mais complexas.

Slowly Changing Dimensions
Exemplos Tipo 1, Tipo 2, Tipo 3

Linha original

Product Key	Product Description	Department	SKU Number (Natural Key)
12345	IntelliKidz 1.0	Education	ABC922-Z

Mudança: O produto IntelliKidz 1.0 muda de departamento.

SCD Tipo 1

Product Key	Product Description	Department	SKU Number (Natural Key)
12345	IntelliKidz 1.0	Strategy	ABC922-Z

SCD Tipo 2

Product Key	Product Description	Department	SKU Number (Natural Key)
12345	IntelliKidz 1.0	Education	ABC922-Z
25984	IntelliKidz 1.0	Strategy	ABC922-Z

SCD Tipo 3

Product Key	Product Description	Department	Prior Department	SKU Number (Natural Key)
12345	IntelliKidz 1.0	Strategy	Education	ABC922-Z

SCD: Exemplo Tipo Híbrido (também chamado tipo 6 = 3+2+1)

Linha original

Product Key	Product Description	Current Department	Historical Department	SKU Number (Natural Key)
12345	IntelliKidz 1.0	Education	Education	ABC922-Z

Requisito: Preservar histórico e ao mesmo tempo suportar consultas a dados históricos de acordo com valores atuais.

Primeira mudança

Product Key	Product Description	Current Department	Historical Department	SKU Number (Natural Key)
12345	IntelliKidz 1.0	Strategy	Education	ABC922-Z
25984	IntelliKidz 1.0	Strategy	Strategy	ABC922-Z

Segunda mudança

Product Key	Product Description	Current Department	Historical Department	SKU Number (Natural Key)
12345	IntelliKidz 1.0	Critical Thinking	Education	ABC922-Z
25984	IntelliKidz 1.0	Critical Thinking	Strategy	ABC922-Z
31726	IntelliKidz 1.0	Critical Thinking	Critical Thinking	ABC922-Z

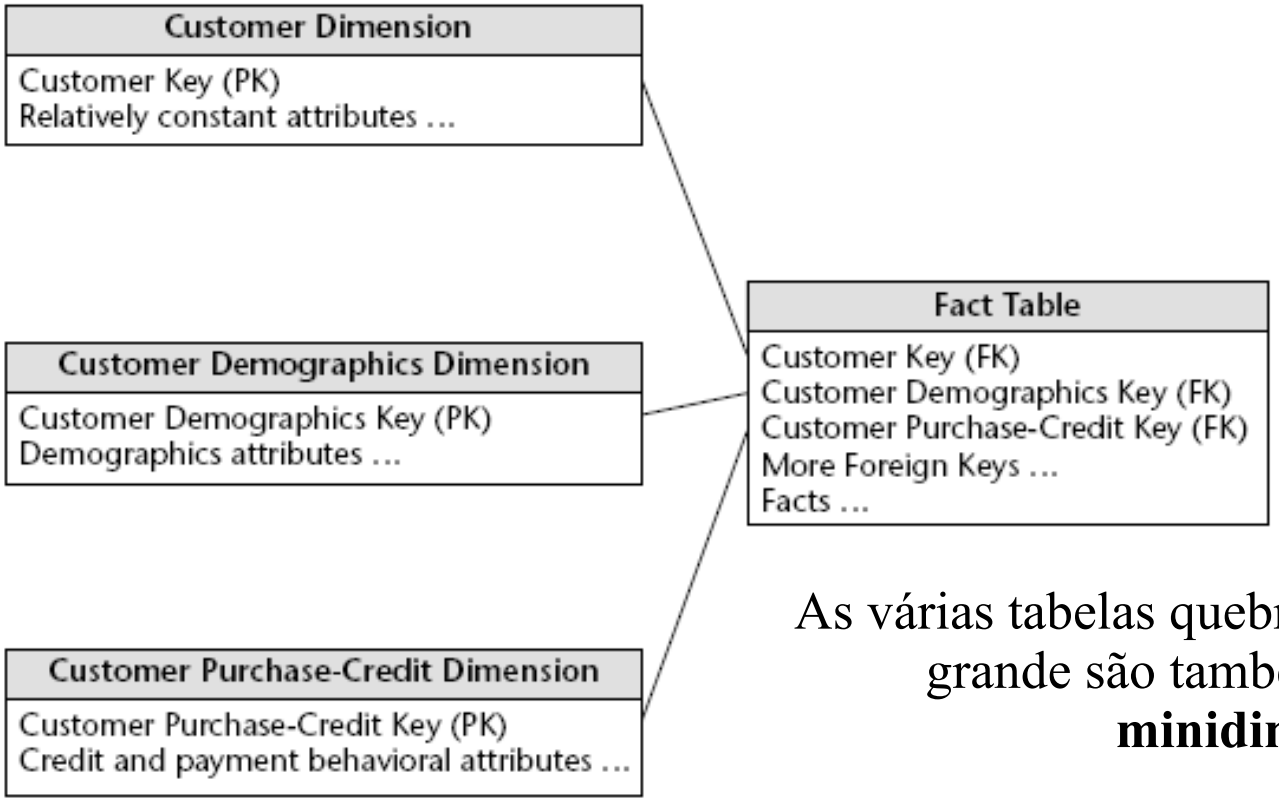
Design Tip #152 (2013)
vide

http://www.kimballgroup.com/2013/02/05/design-tip-152-slowly-changing-dimension-types-0-4-5-6-7/

SCD Type	Dimension Table Action	Impact on Fact Analysis
Type 0	No change to attribute value	Facts associated with attribute's original value
Type 1	Overwrite attribute value	Facts associated with attribute's current value
Type 2	Add new dimension row for profile with new attribute value	Facts associated with attribute value in effect when fact occurred
Type 3	Add new column to preserve attribute's current and prior values	Facts associated with both current and prior attribute alternative values
Type 4	Add mini-dimension table containing rapidly changing attributes	Facts associated with rapidly changing attributes in effect when fact occurred
Type 5	Add type 4 mini-dimension, along with overwritten type 1 mini-dimension key in base dimension	Facts associated with rapidly changing attributes in effect when fact occurred, plus current rapidly changing attribute values
Type 6	Add type 1 overwritten attributes to type 2 dimension row, and overwrite all prior dimension rows	Facts associated with attribute value in effect when fact occurred, plus current values
Type 7	Add type 2 dimension row with new attribute value, plus view limited to current rows and/or attribute values	Facts associated with attribute value in effect when fact occurred, plus current values

Dimensões com grande volume e alta volatilidade
também chamadas de Rapidly Changing Monster Dimensions (SCD tipo #4)

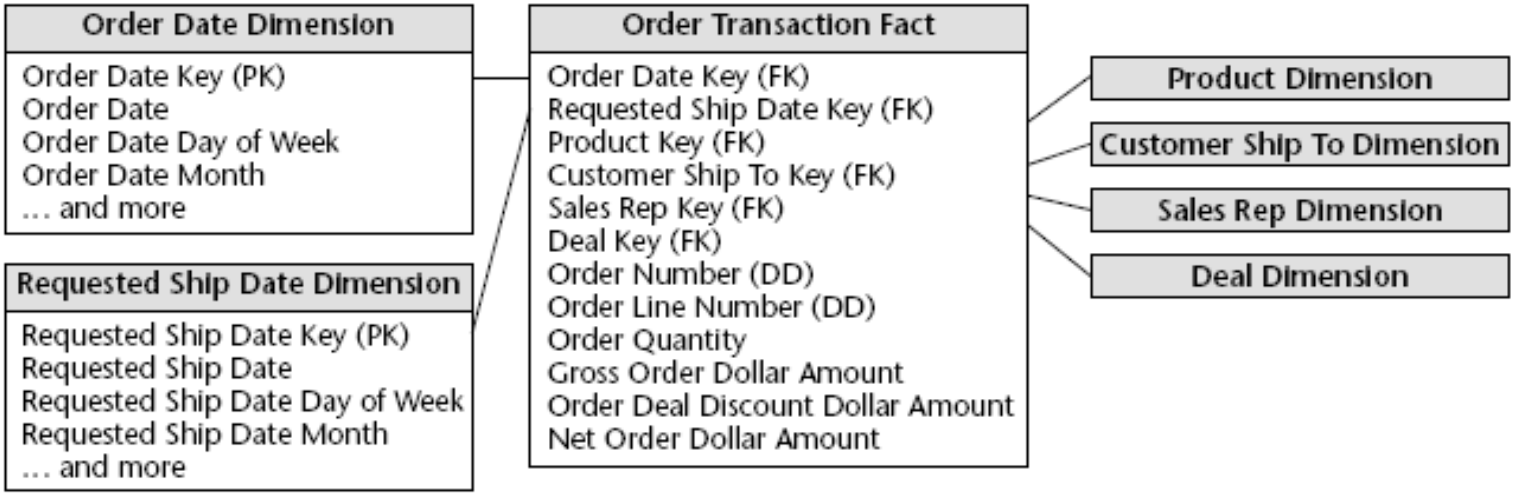
- Solução para dimensões grandes com mudanças frequentes (por exemplo, alguns atributos mudam mensalmente)
  - » Particionamento da dimensão em tabelas diferentes, separando-se dados estáticos de dados voláteis.
    - Dimensões são relacionadas entre si e ambas relacionadas com a tabela fato



As várias tabelas quebradas de uma dimensão grande são também chamadas de **minidimensões**

Dimensões com vários Papéis
Role Playing Dimensions

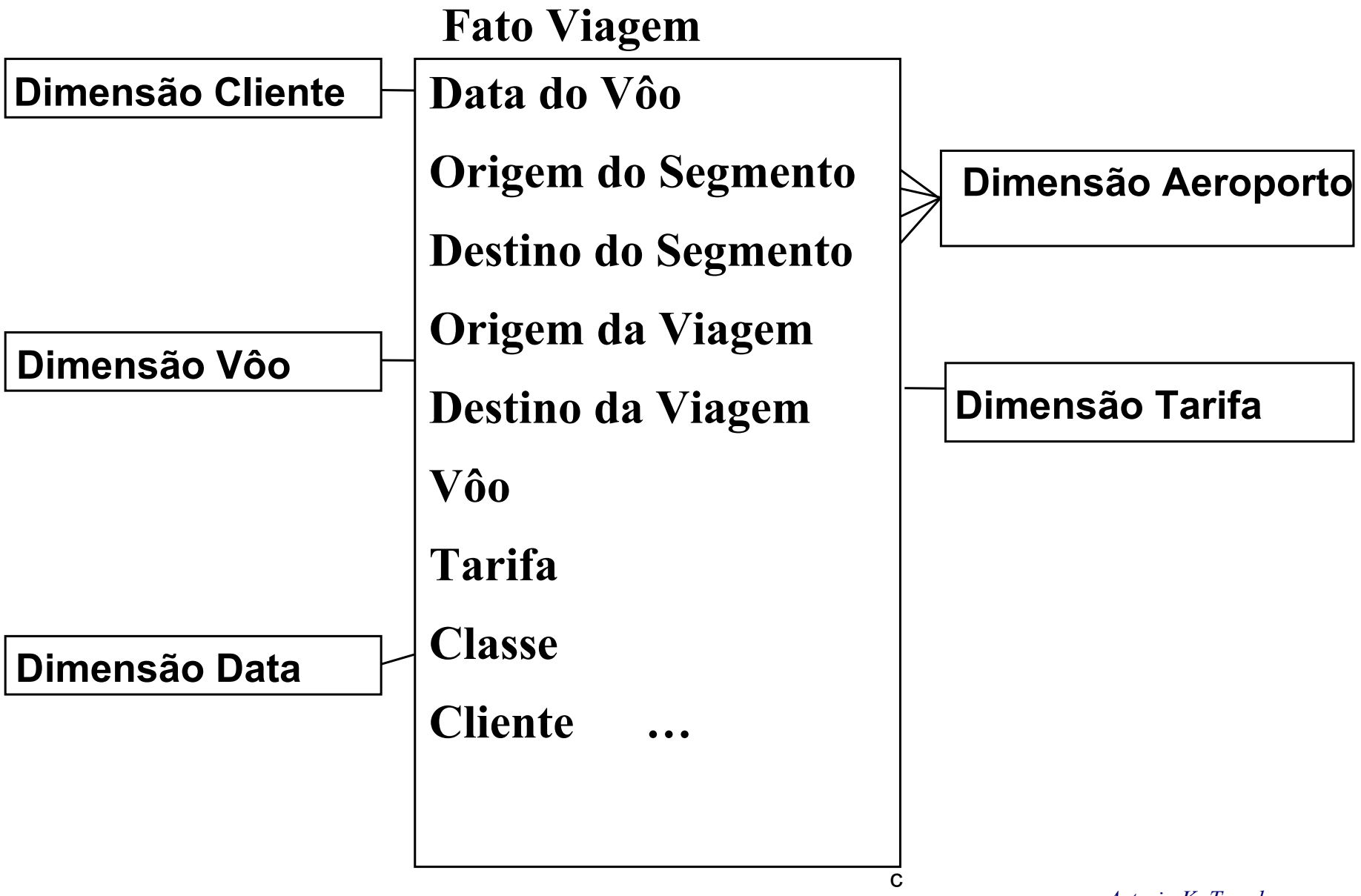
A situação onde uma mesma dimensão aparece várias vezes na mesma tabela de fatos (ex: diferentes Datas). Cada um dos papéis da dimensão é representado por uma tabela lógica separada com nomes de coluna únicos através de visões.



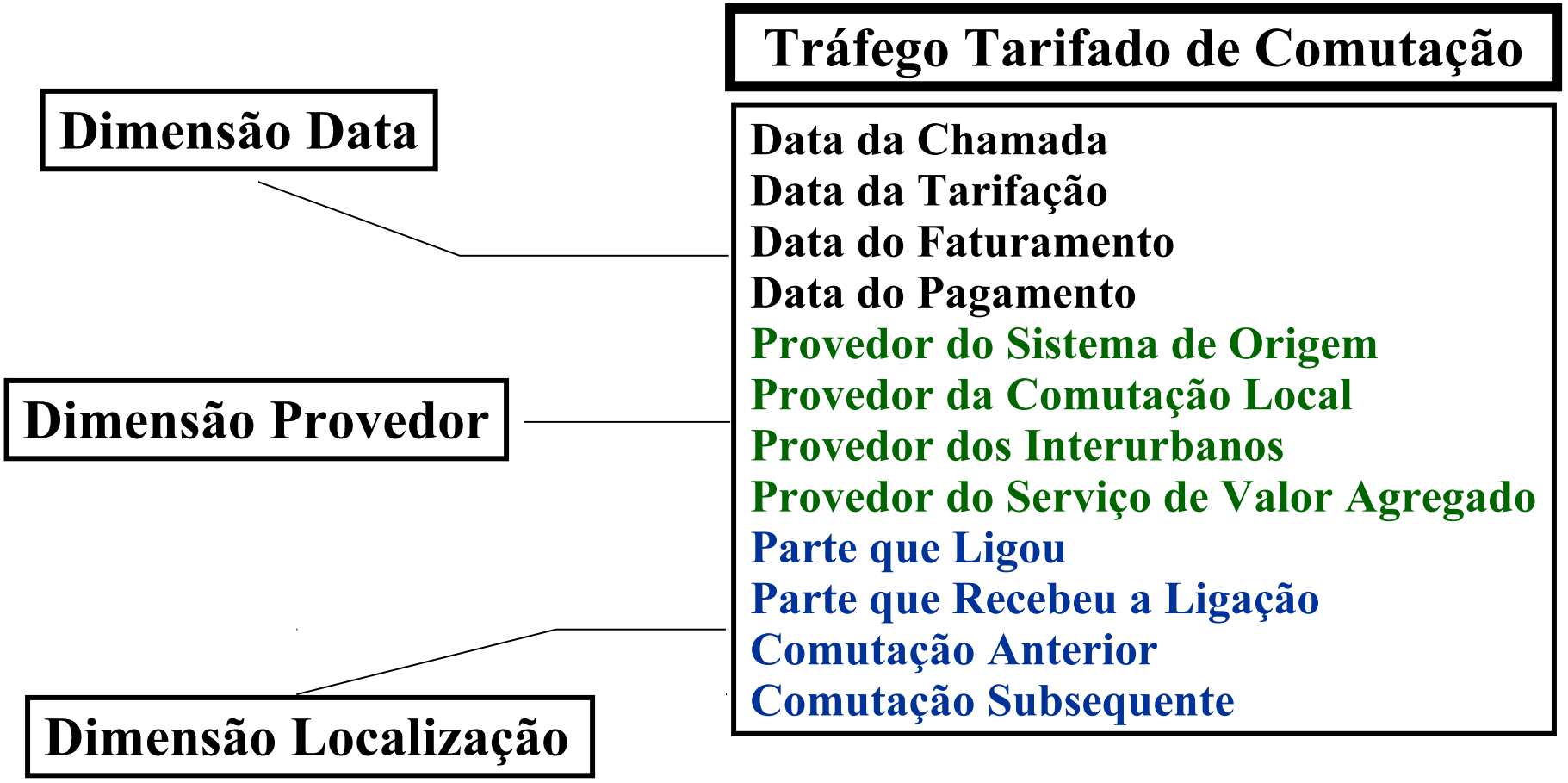

```
CREATE VIEW order_date (order_date_key, order_day_of_week,
order_month...)
AS SELECT date_key, day_of_week, month, . . . FROM Date

CREATE VIEW req_ship_date (req_ship_date_key, req_ship_day_of_week,
req_ship_month ...)
AS SELECT date_key, day_of_week, month, . . . FROM Date
```

Outros exemplos de Dimensões com papéis



Mais exemplos reais sobre dimensão com vários papéis Telecom



Outros Tipos Especiais de Dimensão

- Dimensão lixo ou sucata (junk dimension)
  - Uma dimensão abstrata com a decodificação de um grupo de flags e indicadores de baixa cardinalidade, portanto removendo os flags da tabela de fatos.
- Minidimensões (visto como SCD #4)
  - Subconjuntos de uma dimensão grande, como Cliente, que são quebrados em dimensões artificiais menores para controlar o crescimento explosivo de uma dimensão grande, com mudança rápida. Os atributos demográficos continuamente mutáveis de um cliente são frequentemente modelados como uma minidimensão separada.
- Dimensões com “Outrigger” (exemplo de Snow Flake)
  - Solução normalizada (snow flake) para conjuntos de atributos de baixa cardinalidade em dimensões grandes, como Cliente. A economia de espaço vale a pena porque a dimensão é grande, e a carga de dados é separada do restante da dimensão porque os dados provêm de fontes externas diferentes.
- Dimensões multivaloradas (tabela ponte)
  - Normalmente, uma tabela de fatos possui conexões somente para dimensões representando um valor simples, como uma data ou produto. Mas ocasionalmente, é válido conectar um registro de fato a uma dimensão representando um número aberto de valores, como o número de diagnósticos simultâneos que um paciente pode ter num momento de um mesmo tratamento. Neste caso, dizemos que a tabela de fatos tem uma dimensão multivalorada. Tipicamente manipulada por uma tabela ponte (Bridge Table) também chamada Helper Table, Tabela Associativa).

Dimensão lixo ou sucata (junk dimension)

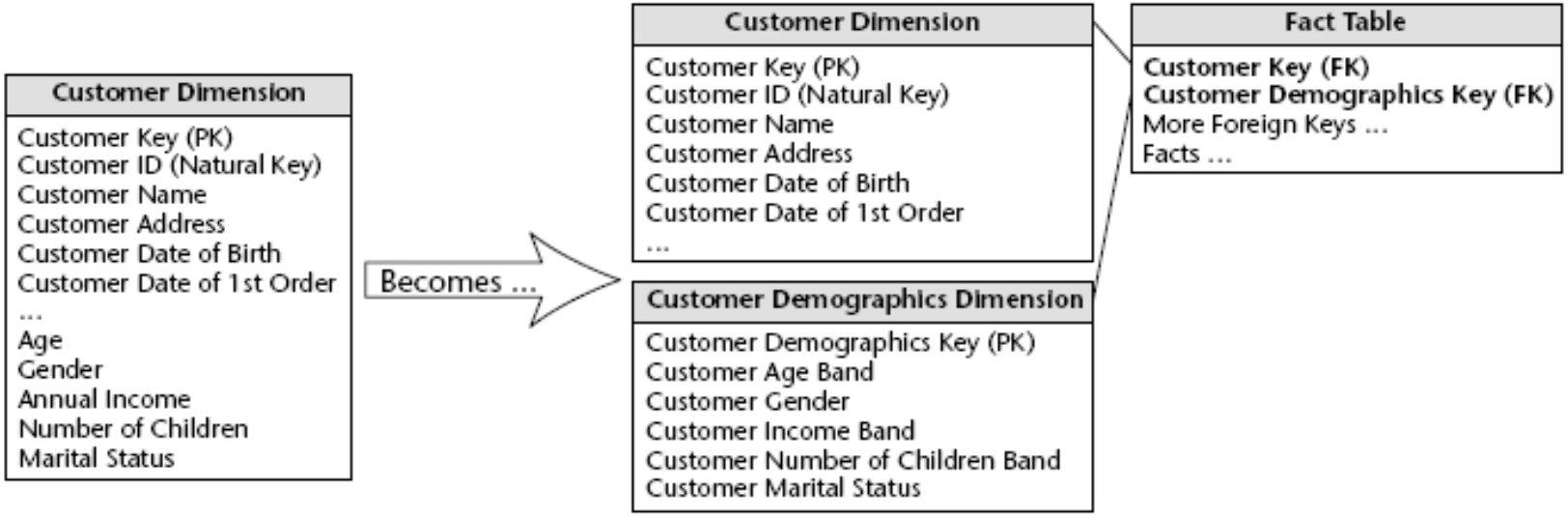
- Relacionadas com tabelas tipo código-descrição com baixa cardinalidade: Sexo, Estado Civil, Tags diversos, Textos descritivos, etc. São campos tipo miscelânea que não trazem muita correlação com os outros campos da tabela fato, mas são usados como filtro, daí serem dimensões.
- Podem ser usadas de forma combinada.
  - Exemplo: três tags binários → $2^3 = 8$ combinações possíveis
- Usado como artifício para diminuir a tabela de fatos. Exemplo:

Order Indicator Key	Payment Type Description	Payment Type Group	Order Type	Commission Credit Indicator
1	Cash	Cash	Inbound	Commissionable
2	Cash	Cash	Inbound	Non-Commissionable
3	Cash	Cash	Outbound	Commissionable
4	Cash	Cash	Outbound	Non-Commissionable
5	Visa	Credit	Inbound	Commissionable
6	Visa	Credit	Inbound	Non-Commissionable
7	Visa	Credit	Outbound	Commissionable
8	Visa	Credit	Outbound	Non-Commissionable
9	MasterCard	Credit	Inbound	Commissionable
10	MasterCard	Credit	Inbound	Non-Commissionable
11	MasterCard	Credit	Outbound	Non-Commissionable
12	MasterCard	Credit	Outbound	Commissionable

Figure 6-8: Sample rows of order indicator junk dimension.

Minidimensões

A melhor abordagem para tratar atributos em dimensões muito grandes é quebrar em uma ou mais minidimensões, cada uma contendo atributos que tenham um número limitado de valores. Exemplo: dimensão Cliente com milhões de ocorrências.



• Vide também o caso de dimensões com alta volatilidade (minidimensão com atributos que mudam rapidamente – SCD tipo #4)

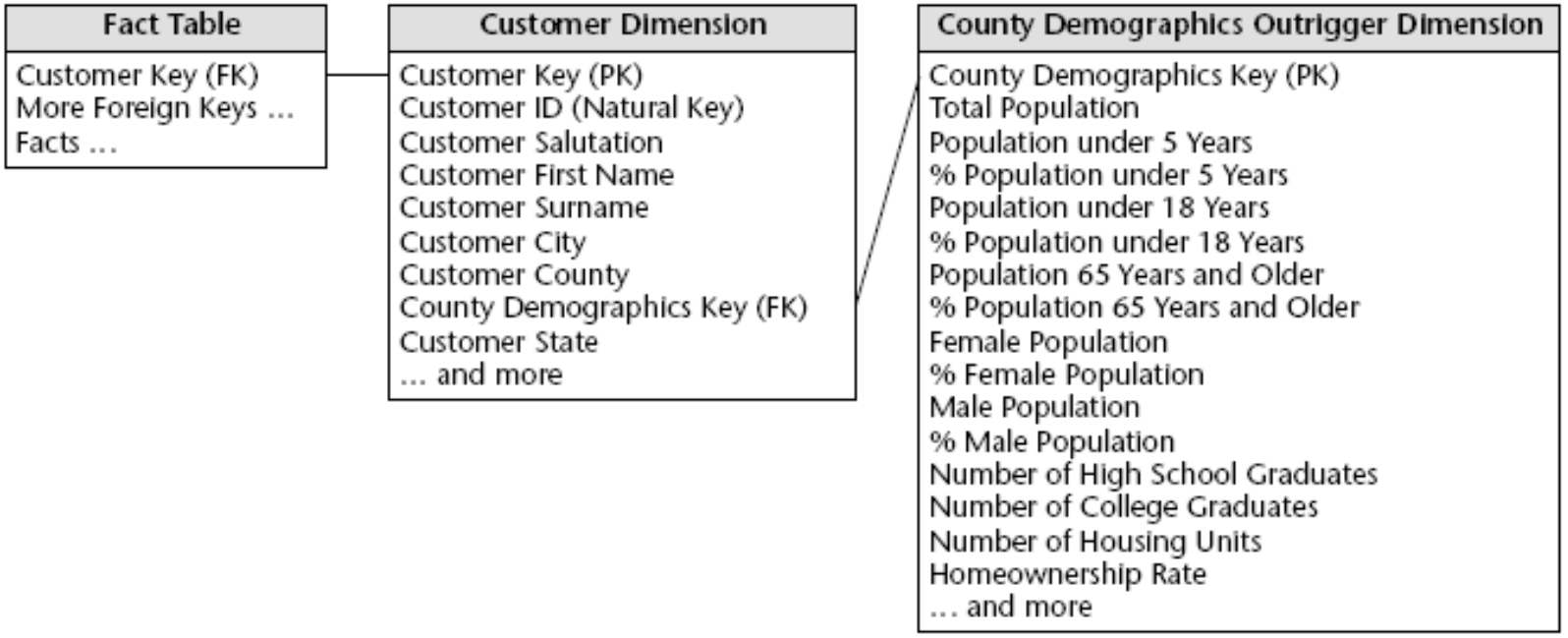
• Tamanho de cada minidimensão = Produto cartesiano da cardinalidade dos atributos da minidimensão. Exemplo acima (Age Band x Gender x Income Band X Number of Children Band x Marital Status)

10 x 2 x 10 x 5 x 5 = 5.000 linhas em Customer Demographics Dimension

Dimensões com “Outrigger”

No exemplo, o “outrigger” agrupa atributos de baixa cardinalidade, que são mantidos em tabela separada da dimensão principal (Customer) para economia de espaço, e também porque a carga dessa tabela é feita com frequência diferente e a partir de fonte externa.

Note que se a solução fosse ligar o “outrigger” diretamente à tabela de Fatos, seria uma minidimensão. Seria possível? Vantagens e desvantagens?



Dimensões Multivaloradas (Tabela Ponte)
(Bridge Table, Helper Table, Associative Table)

- Uma tabela com chave composta capturando um relacionamento muitos-para-muitos que não possa ser acomodado pela granularidade natural de uma tabela de fatos ou tabela de dimensão. Serve como uma ponte entre a tabela de fatos e a tabela de dimensão de forma a permitir dimensões multivaloradas.

- Outros exemplos de dimensões multivaloradas: titulares de conta bancária, códigos de classificações, etc

Tabela Ponte
Outro exemplo

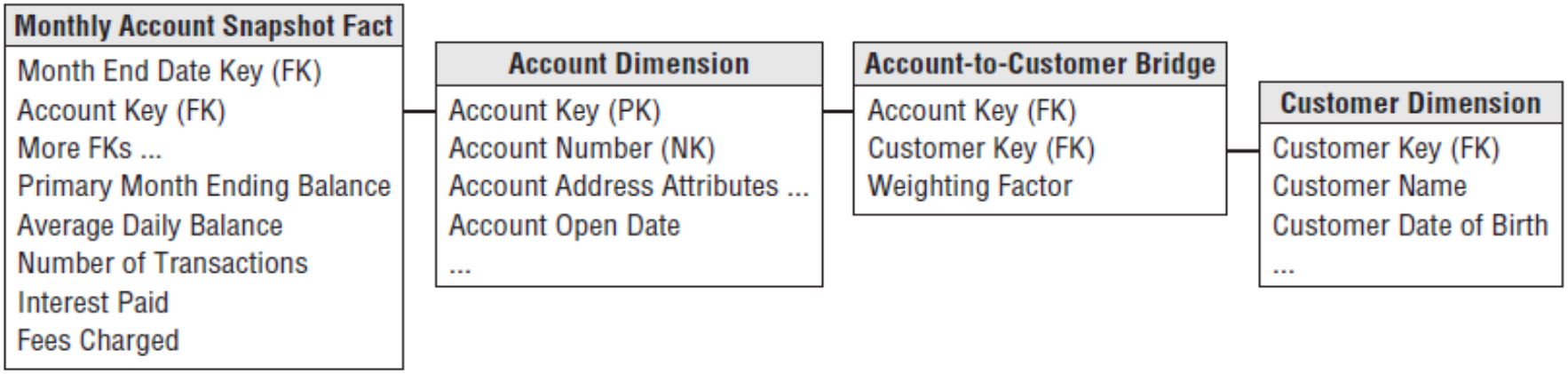


Figure 10-4: Account-to-customer bridge table with weighting factor.

- **Tabela ponte Account-to-Customer para associar múltiplos clientes com fatos de contas.**

Tópicos Especiais sobre Fatos

- **Fatos conformados**
  - **Data marts de primeiro nível , data marts consolidados.**
  - **Bus Matrix de Implementação**
- **Tipos clássicos de fatos**
  - **Transações**
  - **Instantâneos Periódicos**
  - **Instantâneos Acumulados**
- **Tabela de fatos agregados (cubos OLAP)**

Fatos Conformados

- Estabelecer dimensões conformadas para amarrar os data marts representa 95% do esforço de arquitetura de projeto. O restante do esforço consiste em estabelecer definições de fatos conformados.
- Preços, custos, lucros, medidas de qualidade, medidas de satisfação do cliente e outros KPIs são fatos que devem ser conformados. Em geral, dados de fatos não são duplicados explicitamente em múltiplos data marts. Mas isso pode ocorrer em data marts de primeiro nível (originários de um sistema fonte primário de dados) e data marts consolidados (a partir de múltiplas fontes que podem referenciar mais de um processo de negócio).
- Se os fatos forem rotulados identicamente, precisam ser definidos no mesmo contexto dimensional e com as mesmas unidades de medida de data mart para data mart.
- Algumas vezes, um fato tem uma unidade de medida natural em uma tabela de fatos e outra unidade de medida em outra tabela de fatos. Ao invés de prover um fator de conversão numa tabela de dimensão, a abordagem correta é levar o fato com as duas unidades de medida para facilitar os relatórios sem preocupação de conversão. Por exemplo, produtos medidos em caixas no depósito e em peças na loja.

Data Marts de Primeiro e Segundo Níveis

Business Process / Event	Time	Customer	Service	Rate Category	Local Svc Provider	Calling Party	Called Party	Long Dist Provider	Internal Organization	Employee	Location	Equipment Type	Supplier	Item Shipped	Account Status
Customer Billing	X	X	X	X	X			X			X				X
Service Orders	X	X	X		X			X	X	X	X	X			X
Trouble Reports	X	X	X		X	X		X	X	X	X	X	X	X	X
Yellow Page Ads	X	X		X		X			X	X	X				X
Customer Inquiries	X	X	X	X	X	X		X	X	X	X				X
Promotions & Communication	X	X	X	X	X	X		X	X	X	X	X	X	X	X
Billing Call Detail	X	X	X	X	X	X	X	X	X		X	X	X	X	X
Network Call Detail	X	X	X	X	X	X	X	X	X		X	X	X	X	X
Customer Inventory	X	X	X	X	X			X	X		X	X	X	X	X
Network Inventory	X		X						X	X	X	X	X	X	
Real Estate	X								X	X	X	X			
Labor & Payroll	X								X	X	X				
Computer Charges	X	X	X		X			X	X	X	X	X	X	X	
Purchase Orders	X								X	X	X	X	X	X	
Supplier Deliveries	X								X	X	X	X	X	X	

The Matrix Plan for the enterprise data warehouse of a large telecommunications company.

Artigo “The Matrix”, Ralph Kimball, Intelligent Enterprise, Dezembro 1999.
http://www.kimballgroup.com/1999/12/07/the-matrix/

Data Marts de Primeiro e Segundo Níveis
Fatos Consolidados

BUSINESS PROCESSES	COMMON DIMENSIONS: Date	Product	Warehouse	Store	Promotion	Customer	Employee
Issue Purchase Orders	X	X	X				
Receive Warehouse Deliveries	X	X	X				X
Warehouse Inventory	X	X	X				
Receive Store Deliveries	X	X	X	X			X
Store Inventory	X	X		X			
Retail Sales	X	X		X	X	X	X
Retail Sales Forecast	X	X		X			
Retail Promotion Tracking	X	X		X	X		
Customer Returns	X	X		X	X	X	X
Returns to Vendor	X	X		X			X
Frequent Shopper Sign-Ups	X			X		X	X

After listing the core business process rows, you might also identify more complex cross-process or consolidated rows. These consolidated rows can be extremely beneficial analytically, but they are typically much more difficult to implement given the need to combine and potentially allocate performance metrics from multiple source systems; they should be tackled after the underlying processes have been built.

Artigo “The Matrix Revisited”, Margy Ross, 2005.
http://www.kimballgroup.com/2005/12/01/the-matrix-revisited/

Bus Matrix de Implementação

Fact Table/OLAP Cube	Granularity	Facts	Date	Policyholder	Coverage	Covered Item	Employee	Policy	Claim	Claimant	3rd Party Payee
Policy Transactions											
Corporate Policy Transactions	1 row for every policy transaction	Policy Transaction Amount	Trxn Eff	X	X	X	X	X			
Auto Policy Transactions	1 row per auto policy transaction	Policy Transaction Amount	Trxn Eff	X	Auto	Auto	X	X			
Home Policy Transactions	1 row per home policy transaction	Policy Transaction Amount	Trxn Eff	X	Home	Home	X	X			
Policy Premium Snapshot											
Corporate Policy Premiums	1 row for every policy, covered item and coverage per month	Written Premium Revenue and Earned Premium Revenue Amounts	X	X	X	X	Agent	X			
Auto Policy Premiums	1 row per auto policy, covered item and coverage per month	Written Premium Revenue and Earned Premium Revenue Amounts	X	X	Auto	Auto	Agent	X			
Home Policy Premiums	1 row per home policy, covered item and coverage per month	Written Premium Revenue and Earned Premium Revenue Amounts	X	X	Home	Home	Agent	X			
Claim Events											
Claim Transactions	1 row for every claim task transaction	Claim Transaction Amount	Trxn Eff	X	X	X	X	X	X	X	X
Claim Workflow	1 row per claim	Original Reserve, Estimate, Current Reserve, Claim Paid, Salvage Collected, and Subro Collected Amounts; Loss to Open, Open to Estimate, Open to 1st Payment, Open to Subro, and Open to Closed Lags; # of Transactions	X	X	X	X	Agent	X	X	X	
Accident Involvements	1 row per loss party and affiliation on an auto claim	Accident Involvement Count	X	X	Auto	Auto		X	Auto	X	

Figure 16-7: Detailed implementation bus matrix.

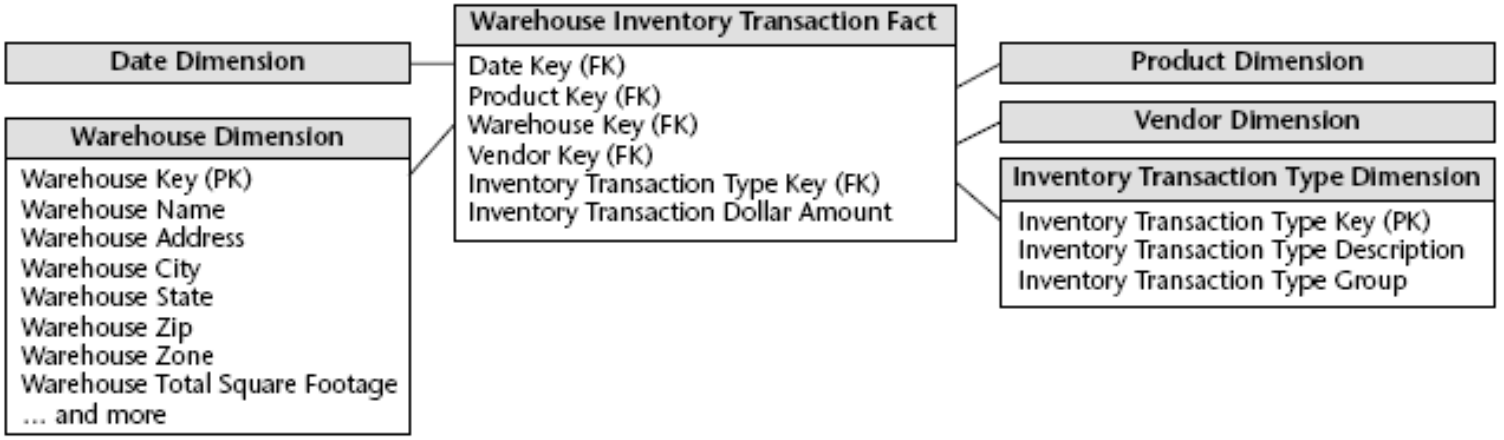
Tipos clássicos de fatos

- **Transações**
- **Instantâneos Periódicos**
- **Instantâneos Acumulados**

- **Tabela de fatos agregados (cubos OLAP)**

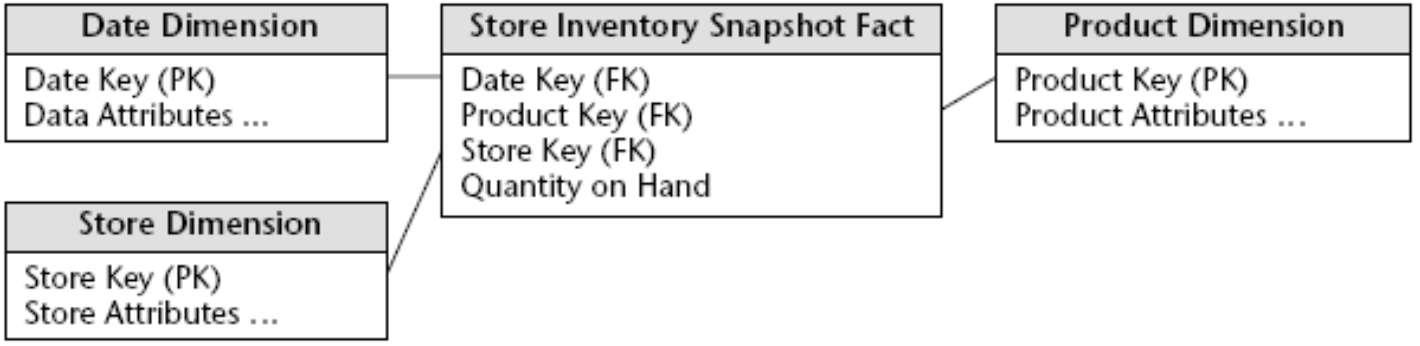
Fatos de transações

- O nível de transação individual representa a visão mais fundamental das operações do negócio. Essas tabelas de fatos representam um evento que ocorreu num ponto instantâneo do tempo.

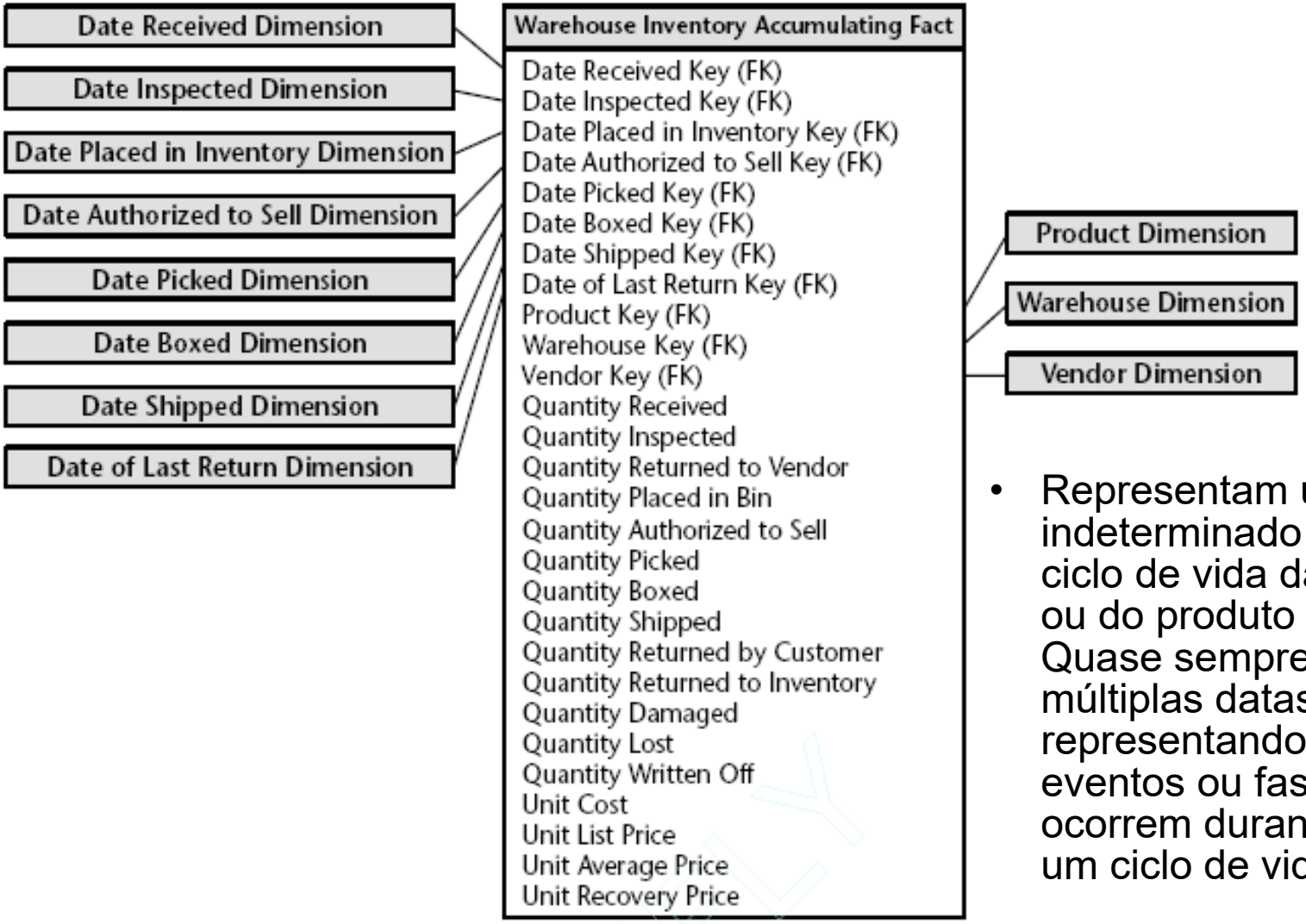


Fatos Instantâneos Periódicos

- São necessários para observar o desempenho cumulativo do negócio em intervalos de tempo regulares e previsíveis. Diferentemente do fato de transação, onde se carrega uma linha para cada ocorrência de evento, com o instantâneo periódico, tira-se uma fotografia da atividade no fim de um dia, uma semana ou um mês, e repetidamente ao fim de cada período.



Fatos instantâneos acumulados



- Representam um tempo indeterminado que cobre o ciclo de vida da transação ou do produto ou pessoa. Quase sempre possuem múltiplas datas, representando os múltiplos eventos ou fases que ocorrem durante o curso de um ciclo de vida.

Tipos clássicos de fatos

Tabela de Comparação dos Tipos de Fatos (2ª Edição, 2002)

CHARACTERISTIC	TRANSACTION GRAIN	PERIODIC SNAPSHOT GRAIN	ACCUMULATING SNAPSHOT GRAIN
Time period represented	Point in time	Regular, predictable intervals	Indeterminate time span, typically short-lived
Grain	One row per transaction event	One row per period	One row per life
Fact table loads	Insert	Insert	Insert and update
Fact row updates	Not revisited	Not revisited	Revisited whenever activity
Date dimension	Transaction date	End-of-period date	Multiple dates for standard milestones
Facts	Transaction activity	Performance for predefined time interval	Performance over finite lifetime

Tipos clássicos de fatos

Tabela de Comparação dos Tipos de Fatos (3ª Edição, 2013)

	Transaction	Periodic Snapshot	Accumulating Snapshot
Periodicity	Discrete transaction point in time	Recurring snapshots at regular, predictable intervals	Indeterminate time span for evolving pipeline/workflow
Grain	1 row per transaction or transaction line	1 row per snapshot period plus other dimensions	1 row per pipeline occurrence
Date dimension(s)	Transaction date	Snapshot date	Multiple dates for pipeline's key milestones
Facts	Transaction performance	Cumulative performance for time interval	Performance for pipeline occurrence
Fact table sparsity	Sparse or dense, depending on activity	Predictably dense	Sparse or dense, depending on pipeline occurrence
Fact table updates	No updates, unless error correction	No updates, unless error correction	Updated whenever pipeline activity occurs

Figure 4-7: Fact table type comparisons.

Tipos clássicos de fatos

Explicitando tipos de fatos na Matriz do DW
Exemplo: Recursos Humanos

	Fact Type	Date	Position	Employee	Organization	Benefit
Hiring Processes						
Employee Position Snapshot	Periodic	X	X	Empl Mgr	X	
Employee Requisition Pipeline	Accumulating	X	X	Empl Mgr	X	
Employee Hiring	Transaction	X	X	Empl Mgr	X	
Employee "On Board" Pipeline	Accumulating	X	X	Empl Mgr	X	
Benefits Processes						
Employee Benefits Eligibility	Periodic	X		X	X	X
Employee Benefits Application	Accumulating	X		X	X	X
Employee Benefit Participation	Periodic	X		X	X	X
Employee Management Processes						
Employee Headcount Snapshot	Periodic	X		X	X	X
Employee Compensation	Transaction	X		X	X	X
Employee Benefit Accruals	Transaction	X		X	X	X
Employee Performance Review Pipeline	Accumulating	X		Empl Mgr	X	X
Employee Performance Review	Transaction	X		Empl Mgr	X	X
Employee Prof Dev Completed Courses	Transaction	X		X	X	
Employee Disciplinary Action Pipeline	Accumulating	X		Empl Mgr	X	
Employee Separations	Transaction	X		Empl Mgr	X	

Figure 9-4: Bus matrix rows for HR processes.

Hierarquias Recursivas

Também conhecidas como auto-relacionamentos

Lidando com Hierarquias Recursivas (1)

One approach is to include the manager's employee key as another foreign key in the fact table, as shown in Figure 9-5. This manager employee key joins to a role-playing employee dimension where every attribute name refers to "manager" to differentiate the manager's profile from the employee's. This approach associates the employee and their manager whenever a row is inserted into a fact table. BI analyses can easily filter and group by either employee or manager attributes with virtually identical query performance because both dimensions provide symmetrical access to the fact table. The downside of this approach is these dual foreign keys must be embedded in every fact table to support managerial reporting.

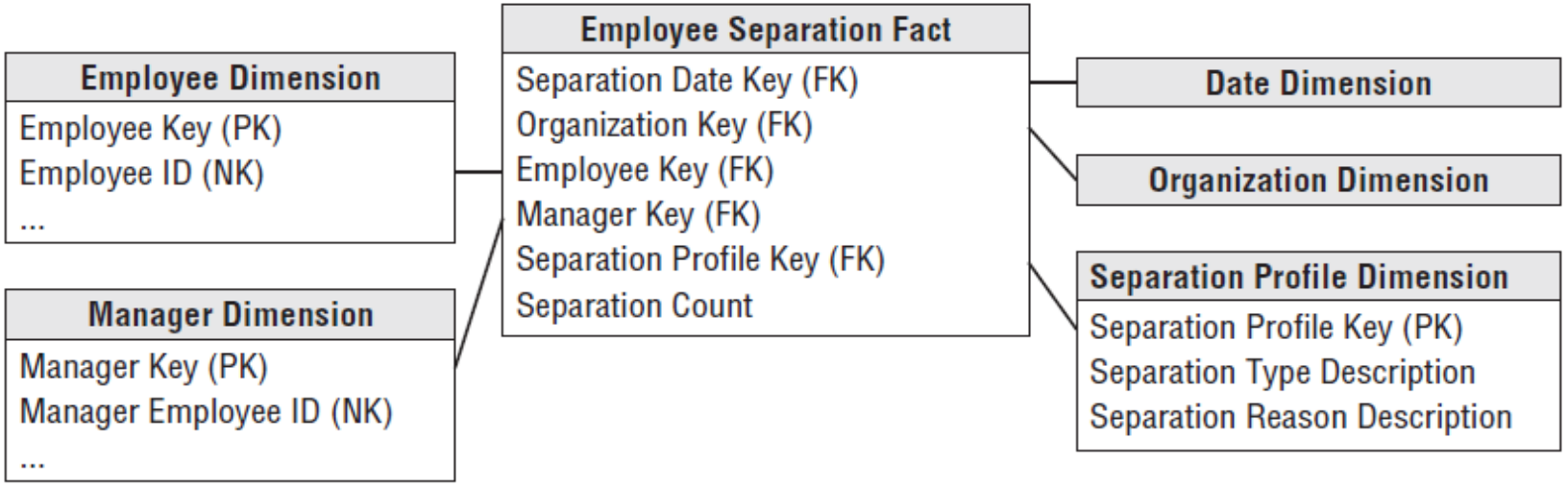


Figure 9-5: Dual role-playing employee and manager dimensions.

Lidando com Hierarquias Recursivas (2)

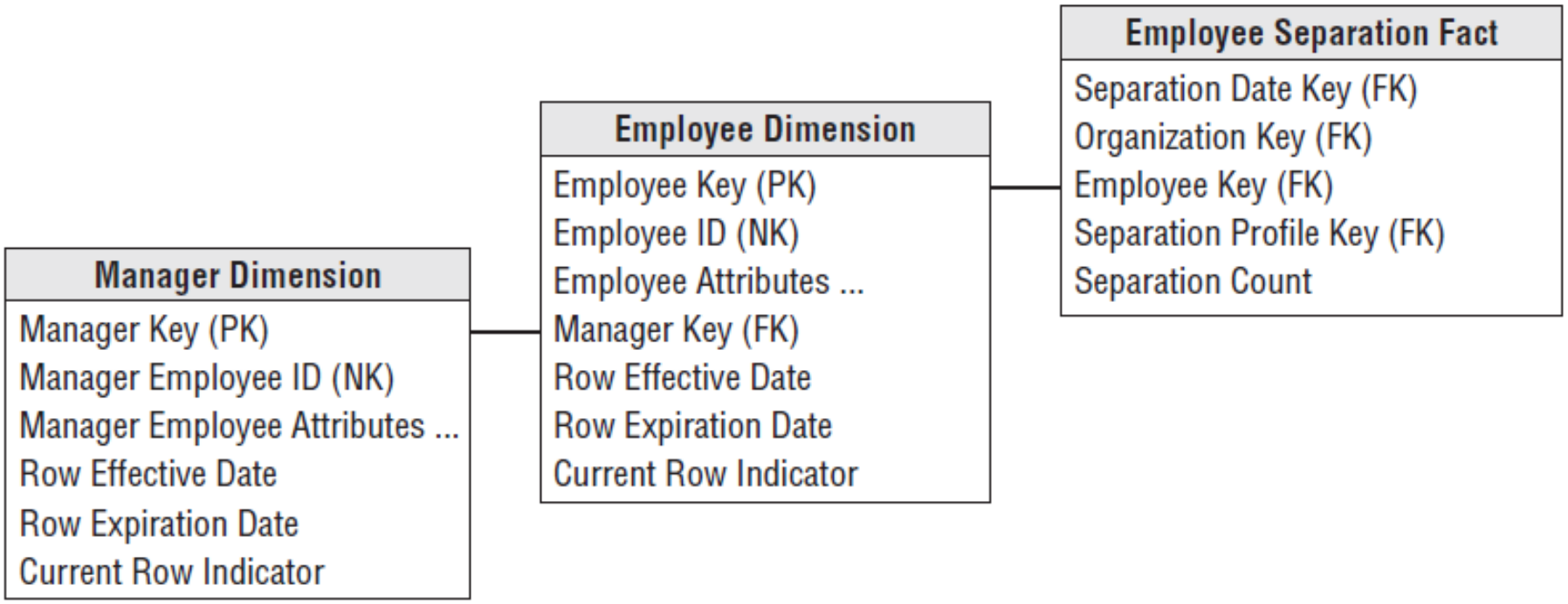


Figure 9-6: Manager role-playing dimension as an outrigger.

If the manager's foreign key in the employee dimension is designated as a type 2 attribute, then new employee rows would be generated with each manager change. However, we encourage you to think carefully about the underlying ETL business rules.

Chaves Surrogate em Tabelas de Fatos

- Chaves surrogate são usadas em tabelas de dimensões para servir de ligação (chaves estrangeiras) em tabelas de fatos
- Chaves surrogate em tabelas de fato, não associadas com nenhuma dimensão, podem ser úteis quando atribuídas sequencialmente durante o processo de ETL, para serem usadas, em situações especiais:
  - Como a chave primária simples de uma tabela de fatos;
  - Para servir como um identificador imediato de uma linha da tabela de fatos sem navegar múltiplas dimensões para ETL;
  - Para permitir um processo ininterrupto ao desfazer ou reassumir uma carga;
  - Para permitir que operações de atualização (UPDATE) em tabelas de fatos sejam decompostas em operações menos arriscadas de INSERT seguido de DELETE.

Tabelas de Fatos Agregados
Cubos OLAP

- São resultantes de simples “rollup” de dados de tabelas de fatos atômicos, com o objetivo de acelerar o desempenho de consultas.
- Devem estar disponíveis juntamente com as tabelas de fatos atômicos para que as ferramentas de BI possam escolher o nível apropriado de agregação em tempo de consulta. Este processo é conhecido como “aggregate navigation”.
- Um conjunto de agregados bem projetado deve se comportar como índices e visões em bancos de dados, que aceleram o desempenho das consultas mas não são percebidos diretamente pelos usuários.
- Cubos OLAP agregados com medidas sumarizadas também são construídos como agregados relacionais, mas os cubos OLAP são acessados diretamente pelos usuários de negócios.

Tabelas de Fatos Agregados

- **Materializar (armazenar) ou não?**
  - **Dilema similar à criação de índices e materialização de visões (views) em bancos de dados**
- **Critérios para definição de agregados**
  - **Passam pela análise dos principais tipos de informação necessárias e pela dificuldade de se obtê-las diretamente das tabelas granulares.**
  - **Exemplo:**

TDLoja (chave-loja, nome-loja, endereco-loja, cidade, estado, ***regiao***)

TDProduto (chave-produto, descricao, marca, ***categoria***, tipo-embalagem, departamento)

TDData (chave-dia, data-completa, dia, ***mês***, ***ano***, período-fiscal, estação)

TFVendas (chave-loja, chave-produto, chave-dia, valor-vendido-real, custo-real, lucro, qtd-vendida)

Hierarquias de dimensões

REGIÃO → LOJA

CATEGORIA → PRODUTO

ANO → MÊS → DATA

Agregados: O que materializar?

- Combinações possíveis
  - Ternárias: LOJA X PRODUTO X DATA
    → 2 x 2 x 3 = 12 combinações
  - Binárias:
    » LOJA X PRODUTO + LOJA X DATA + PRODUTO X DATA
    →2 x 2 + 2 x 3 + 2 x 3 = 16 combinações
  - Unárias:
    » LOJA + PRODUTO + DATA
    →2 + 2 + 3 = 7 combinações
  - Total = 35 combinações de agregações

- Quais deveriam ser materializadas e armazenadas?

- Importante conhecer a distribuição de valores agregados por dimensão
  - Ex: LOJA

```
SELECT nome-loja, COUNT(*)
FROM TFVendas, TDLoja
WHERE TFVendas.chave-loja = TDLoja.chave-loja
GROUP BY nome-loja
```

Agregados

- **Cuidados na definição dos agregados**
  - Valores aditivos
    - » Nem todas as métricas armazenadas nas tabelas granulares são aditivas em todas as dimensões (fatos semi-aditivos ou não aditivos). Isto significa que os atributos das tabelas fatos de agregados poderão ser diferentes das tabelas fatos granulares.
  - Precisão
    - » Deve-se definir criteriosamente a precisão dos valores aditivos de agregados, que deverão ser maiores do que os usados nos respectivos valores das tabelas granulares (para evitar overflow na adição)
    - » Fatos e dimensões agregados devem estar em tabelas fisicamente diferentes das tabelas granulares, mesmo que o número de tabelas cresça muito. Ferramentas de análise (OLAP, por exemplo) possuem mecanismo de navegação de agregados que escondem a complexidade da estrutura.

Agregados

- **Exemplos**
  - Agregação por loja, para todos os produtos, todos os dias.
  - Agregação por loja, por mês, para todos os produtos.
  - Agregação por regiao de venda, por mês, por categoria.

Agregados

- **Exemplos**

  – Agregação por loja, para todos os produtos, todos os dias.

```
INSERT INTO AG-LOJA AS
SELECT nome-loja, sum(valor-vendido-real), sum(custo-real)
FROM TDLoja, TFVendas
WHERE TDLoja.chave-loja=TFVendas.chave-loja
GROUP BY nome-loja
```

  – Agregação por loja, por mês, para todos os produtos.

```
INSERT INTO AG-LOJA-MÊS AS
SELECT nome-loja, mês, sum(valor-vendido-real), sum(custo-real)
FROM TDLoja, TFVendas, TDDia
WHERE TDLoja.chave-loja=TFVendas.chave-loja AND
      TFVendas.chave-dia=TDDia .chave-dia
GROUP BY nome-loja, mês
```

Agregados

- **Exemplos**
  - Agregação por regiao de venda, por mês, por categoria.

```
INSERT INTO AG-REG-CAT-MES AS
SELECT regiao, mês, categoria, sum(valor-vendido-real), sum(custo-real)
FROM TDLoja, TFVendas.TDProduto, TDDia
WHERE TDLoja.chave-loja=TFVendas.chave-loja AND
      TFVendas.chave-dia=TDDia .chave-dia AND
      TFVendas.chave-produto=TDProduto.chave-produto
GROUP BY regiao, mês, categoria
```

- **Cuidados operacionais**
  - Modelos separados (agregados e granulares) para evitar contenções mútuas no momento de carga ou atualização.
  - Carga total versus Atualização incremental: Tempo de processamento versus Complexidade de programas
  - Carga/atualização pode requerer processamento paralelo, para otimização
- **Utilização de agregados**
  - Navegador de agregados: camada de interface entre a ferramenta OLAP e o servidor de DW. O navegador realiza transparentemente a conversão de comandos SQL granulares nos equivalentes que trabalham informações agregadas.

Dez Erros Comuns a Evitar em Modelagem Dimensional Toolkit 3ª Edição, pag 397-401

- **Erro 10:** Colocar atributos de texto usados para restrições e agrupamento numa tabela de fatos.
- **Erro 9:** Limitar atributos descritivos verbosos em dimensões para economizar espaço.
- **Erro 8:** Separar hierarquias e níveis de hierarquia em dimensões múltiplas.
- **Erro 7:** Ignorar a necessidade de cuidar de mudanças em atributos de dimensões.
- **Erro 6:** Resolver todos os problemas de desempenho de consultas adicionando mais hardware.

Dez Erros Comuns a Evitar em Modelagem Dimensional Toolkit 3ª Edição, pag 397-401

- **Erro 5:** Usar chaves operacionais ou “inteligentes” para junções de tabelas de dimensão com tabela de fatos.
- **Erro 4:** Negligenciar a declaração de grão e depois a consistência com o grão da tabela de fatos.
- **Erro 3:** Projetar o modelo dimensional baseado em um relatório específico.
- **Erro 2:** Esperar que usuários consultem dados de nível atômico mais baixo num formato normalizado.
- **Erro 1:** Falhar em conformar fatos e dimensões através de diferentes data marts.

As 10 Regras Essenciais para a Modelagem de Dados Dimensional

http://www.kimballgroup.com/2009/05/the-10-essential-rules-of-dimensional-modeling/

- Rule #1: Load detailed atomic data into dimensional structures.
- Rule #2: Structure dimensional models around business processes.
- Rule #3: Ensure that every fact table has an associated date dimension table.
- Rule #4: Ensure that all facts in a single fact table are at the same grain or level of detail.
- Rule #5: Resolve many-to-many relationships in fact tables.
- Rule #6: Resolve many-to-one relationships in dimension tables.
- Rule #7: Store report labels and filter domain values in dimension tables.
- Rule #8: Make certain that dimension tables use a surrogate key.
- Rule #9: Create conformed dimensions to integrate data across the enterprise.
- Rule #10: Continuously balance requirements and realities to deliver a DW/BI solution that's accepted by business users and that supports their decision-making.

Vide tradução para Português neste link da empresa Ambiente Livre