

# CRIANDO UM MODELO DE DADOS

# CRIANDO UM MODELO DE DADOS



Nesta seção, abordaremos **tópicos básicos de modelagem de dados**, como normalização, tabelas de fatos e dimensões, chaves primárias e estrangeiras, cardinalidade de relacionamento e fluxo de filtro

## TEMAS QUE ABRIREMOS:

Modelagem de Dados 101

Normalização

Fatos e Dimensões

Chaves primárias e estrangeiras

Cardinalidade

Fluxo de filtro

Esquemas Comuns

Hierarquias

## OBJETIVOS PARA ESTA SEÇÃO:

- Compreender os princípios básicos da modelagem de dados, incluindo normalização, tabelas de fatos e dimensões e esquemas comuns
- Criar relacionamentos de tabelas usando chaves primárias e estrangeiras e discutir diferentes tipos de cardinalidade de relacionamento
- Configurar filtros de relatório e rastrear o contexto do filtro à medida que ele flui entre tabelas relacionadas no modelo
- Explore opções de modelagem de dados, como hierarquias, categorias de dados e campos ocultos



# O QUE É UM MODELO DE DADOS?

The screenshot shows three separate data models in a Power BI interface:

- Product Lookup:** Contains fields like Model Name, Product Color, Product Cost, Product Description, Product Key, and Product Name.
- Sales Data:** Contains fields like Customer Key, Index, Order Date, Order Line Item, Order Number, Order Quantity, Product Key, Stock Date, Territory Key, and so on.
- Returns Data:** Contains fields like Product Key, Return Date, Return Quantity, and Territory Key.

Este **NÃO** É um modelo de dados

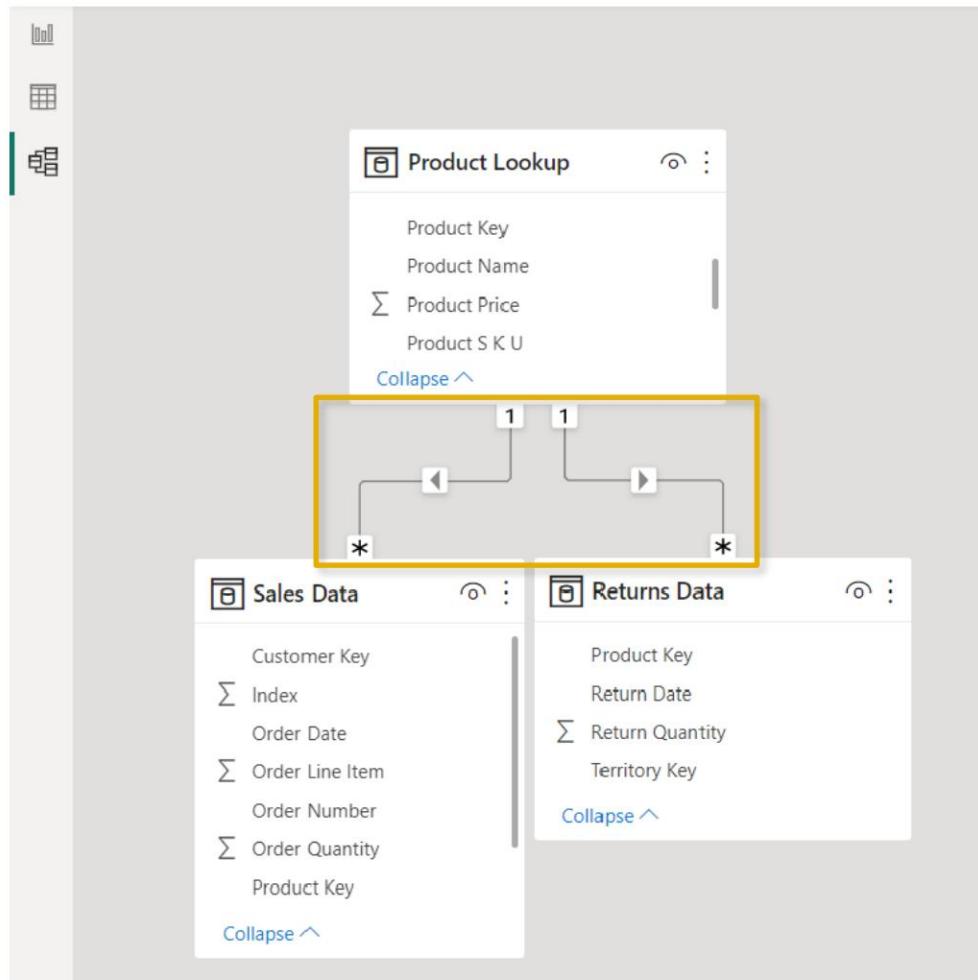


- Esta é uma coleção de tabelas independentes, que não compartilham conexões ou relacionamentos
- Se você tentou visualizar **Pedidos e Devoluções por produto**, isso é o que você obteria

ProductName	OrderQuantity	ReturnQuantity
All-Purpose Bike Stand	84,174	1,828
AWC Logo Cap	84,174	1,828
Bike Wash - Dissolver	84,174	1,828
Cable Lock	84,174	1,828
Chain	84,174	1,828
Classic Vest, L	84,174	1,828
Classic Vest, M	84,174	1,828
Classic Vest, S	84,174	1,828
Fender Set - Mountain	84,174	1,828
<b>Total</b>	<b>84,174</b>	<b>1,828</b>



# O QUE É UM MODELO DE DADOS?



Este É um modelo de dados!



- As tabelas são conectadas por meio de relacionamentos, com base em um campo comum (chave do produto)
- Agora os dados **de vendas e devoluções** podem ser filtrados usando campos da tabela **de pesquisa de produtos !**

ProductName	OrderQuantity	ReturnQuantity
All-Purpose Bike Stand	234	8
AWC Logo Cap	4,151	46
Bike Wash - Dissolver	1,706	25
Classic Vest, L	182	4
Classic Vest, M	182	7
Classic Vest, S	157	8
Fender Set - Mountain	3,960	54
Half-Finger Gloves, L	840	18
Half-Finger Gloves, M	918	16
<b>Total</b>	<b>84,174</b>	<b>1,828</b>



# NORMALIZAÇÃO DE BANCO DE DADOS

**Normalização** é o processo de organização de tabelas e colunas em um banco de dados relacional para reduzir a redundância e preservar a integridade dos dados. É comumente usado para:

- **Elimine dados redundantes** para diminuir o tamanho das tabelas e melhorar a velocidade e a eficiência do processamento
- **Minimizar erros e anomalias provenientes** de modificações de dados (inserção, atualização ou exclusão de registros)
- **Simplifique as consultas** e estruture o banco de dados para análises significativas



Em um banco de dados normalizado, cada tabela deve servir a um propósito **distinto e específico**  
(ou seja, *informações sobre produtos, registros de transações, atributos de clientes, detalhes de lojas, etc.*)

date	product_id	quantity	product_brand	product_name	product_sku	product_weight
1/1/1997	869	5	Nationeel	Nationeel Grape Fruit Roll	52382137179	17
1/7/1997	869	2	Nationeel	Nationeel Grape Fruit Roll	52382137179	17
1/3/1997	1	4	Washington	Washington Berry Juice	90748583674	8.39
1/1/1997	1472	3	Fort West	Fort West Fudge Cookies	37276054024	8.28
1/6/1997	1472	2	Fort West	Fort West Fudge Cookies	37276054024	8.28
1/5/1997	2	4	Washington	Washington Mango Drink	96516502499	7.42
1/1/1997	76	4	Red Spade	Red Spade Sliced Chicken	62054644227	18.1
1/1/1997	76	2	Red Spade	Red Spade Sliced Chicken	62054644227	18.1
1/5/1997	3	2	Washington	Washington Strawberry Drink	58427771925	13.1
1/7/1997	3	2	Washington	Washington Strawberry Drink	58427771925	13.1
1/1/1997	320	3	Excellent	Excellent Cranberry Juice	36570182442	16.4

Os modelos que não são normalizados contêm **dados redundantes e duplicados**. Nesse caso, todos os campos específicos do produto poderiam ser armazenados em uma tabela separada contendo um registro exclusivo para cada **ID de produto**

Isto pode não parecer crítico agora, mas pequenas ineficiências podem tornar-se grandes problemas em grande escala!



# TABELAS DE FATOS E DIMENSÃO

Os modelos de dados geralmente contêm dois tipos de tabelas: tabelas **de fatos** (“dados”) e tabelas **de dimensões** (“pesquisa”):

- As **tabelas de fatos** contêm **valores numéricos** ou métricas usadas para resumo (*vendas, pedidos, transações, visualizações de página etc.*)
- As **tabelas de dimensões** contêm **atributos descritivos** usados para filtragem ou agrupamento (*produtos, clientes, datas, lojas, etc.*)

date	product_id	quantity
1/1/1997	869	5
1/1/1997	1472	3
1/1/1997	76	4
1/1/1997	320	3
1/1/1997	4	4
1/1/1997	952	4
1/1/1997	1222	4
1/1/1997	517	4
1/1/1997	1359	4
1/1/1997	357	4
1/1/1997	1426	5
1/1/1997	190	4
1/1/1997	367	4
1/1/1997	250	5
1/1/1997	600	4
1/1/1997	702	5

Esta tabela **de fatos** contém valores **de quantidade**, juntamente com campos **de data** e **product\_id**

date	day_of_month	month	year	weekday	week_of_year	week_ending	month_name	quarter
1/1/1997	1	1	1997	Wednesday	1	1/5/1997	January	Q1
1/2/1997	2	1	1997	Thursday	1	1/5/1997	January	Q1
1/3/1997	3	1	1997	Friday	1	1/5/1997	January	Q1
1/4/1997	4	1	1997	Saturday	1	1/5/1997	January	Q1
1/5/1997	5	1	1997	Sunday	2	1/5/1997	January	Q1
1/6/1997	6	1	1997	Monday	2	1/12/1997	January	Q1

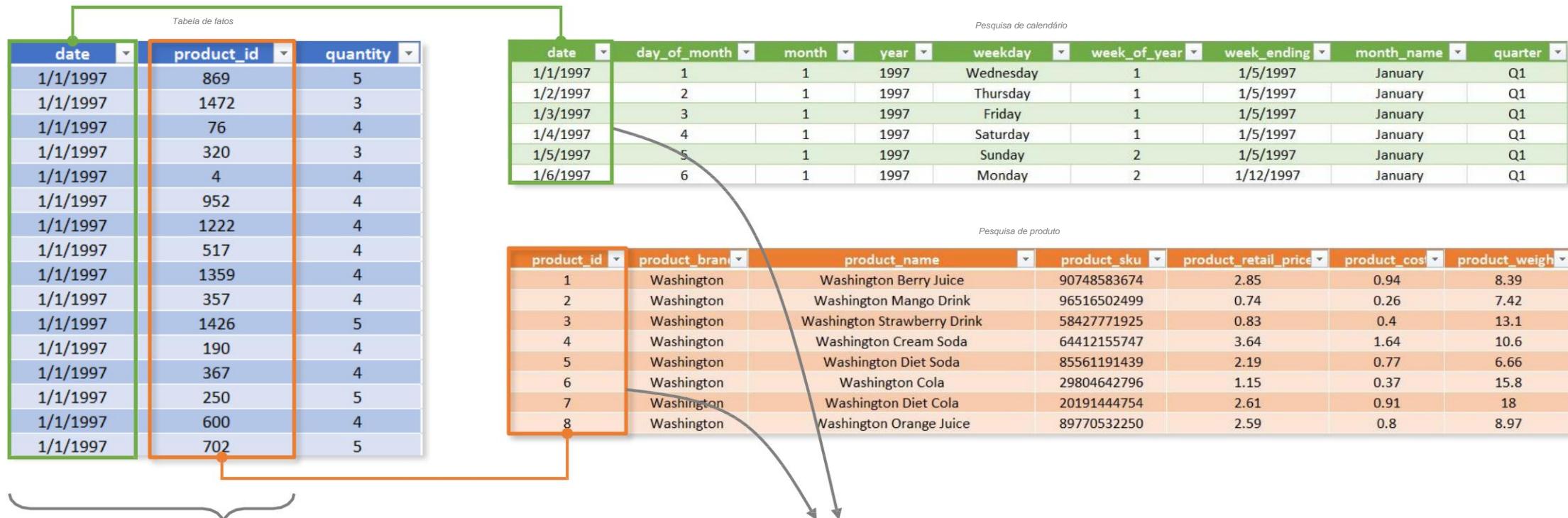
Esta tabela **de pesquisa de calendário** contém atributos sobre cada **data** (mês, ano, trimestre, etc.)

product_id	product_brand	product_name	product_sku	product_retail_price	product_cost	product_weight
1	Washington	Washington Berry Juice	90748583674	2.85	0.94	8.39
2	Washington	Washington Mango Drink	96516502499	0.74	0.26	7.42
3	Washington	Washington Strawberry Drink	58427771925	0.83	0.4	13.1
4	Washington	Washington Cream Soda	64412155747	3.64	1.64	10.6
5	Washington	Washington Diet Soda	85561191439	2.19	0.77	6.66
6	Washington	Washington Cola	29804642796	1.15	0.37	15.8
7	Washington	Washington Diet Cola	20191444754	2.61	0.91	18
8	Washington	Washington Orange Juice	89770532250	2.59	0.8	8.97

Esta tabela **de pesquisa de produto** contém atributos sobre cada **product\_id** (marca, SKU, preço, etc.)



# CHAVES PRIMÁRIAS E ESTRANGEIRAS



Estas são chaves **estrangeiras** (FK)

*Eles contêm várias instâncias de cada valor e estão relacionados às **chaves primárias** nas tabelas de dimensão*

Estas são **chaves primárias** (PK)

*Eles identificam exclusivamente cada linha da tabela e se relacionam com **chaves estrangeiras** em tabelas de fatos*



# RELACIONAMENTOS VS. TABELAS MESCLADAS



*Não posso simplesmente mesclar consultas ou usar funções de pesquisa para reunir tudo em uma única tabela?*

- Homem anônimo confuso

Campos da tabela de fatos originais					Atributos da tabela de pesquisa de calendário					Atributos da tabela de pesquisa de produtos		
date	product_id	quantity	day_of_month	month	year	weekday	month_name	quarter	product_brand	product_name	product_sku	product_weight
1/1/1997	869	5	1	1	1997	Wednesday	January	Q1	Nationaleel	Nationaleel Grape Fruit Roll	52382137179	17
1/7/1997	869	2	7	1	1997	Tuesday	January	Q1	Nationaleel	Nationaleel Grape Fruit Roll	52382137179	17
1/3/1997	1	4	3	1	1997	Friday	January	Q1	Washington	Washington Berry Juice	90748583674	8.39
1/1/1997	1472	3	1	1	1997	Wednesday	January	Q1	Fort West	Fort West Fudge Cookies	37276054024	8.28
1/6/1997	1472	2	6	1	1997	Monday	January	Q1	Fort West	Fort West Fudge Cookies	37276054024	8.28
1/5/1997	2	4	5	1	1997	Sunday	January	Q1	Washington	Washington Mango Drink	96516502499	7.42
1/1/1997	76	4	1	1	1997	Wednesday	January	Q1	Red Spade	Red Spade Sliced Chicken	62054644227	18.1
1/1/1997	76	2	1	1	1997	Wednesday	January	Q1	Red Spade	Red Spade Sliced Chicken	62054644227	18.1
1/5/1997	3	2	5	1	1997	Sunday	January	Q1	Washington	Washington Strawberry Drink	58427771925	13.1
1/7/1997	3	2	7	1	1997	Tuesday	January	Q1	Washington	Washington Strawberry Drink	58427771925	13.1
1/1/1997	320	3	1	1	1997	Wednesday	January	Q1	Excellent	Excellent Cranberry Juice	36570182442	16.4

Você pode, mas é extremamente ineficiente!

- A mesclagem de tabelas cria **redundância** e muitas vezes requer **significativamente mais memória e capacidade de processamento** para análise em comparação com um modelo relacional com diversas tabelas pequenas



# A VISÃO DO MODELO

**Faixa de menu (Ajuda doméstica)**

**Tela modelo**

**Guias de layout do modelo**

**Painel de propriedades**  
(*nome, sinônimo, formato, etc.*)

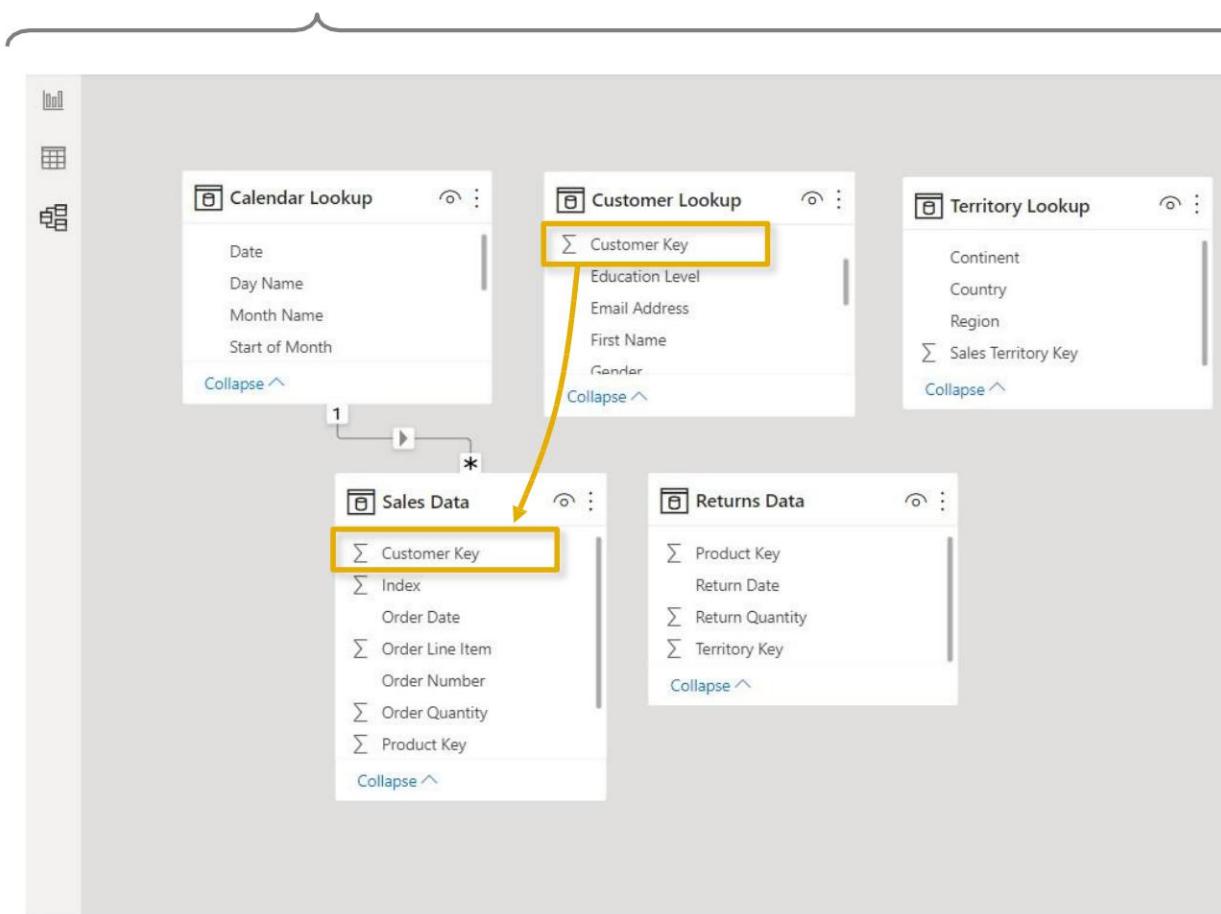
**Lista de dados/campos**

**Ver opções**  
(*Zoom, redefinir layout, ajustar à página*)

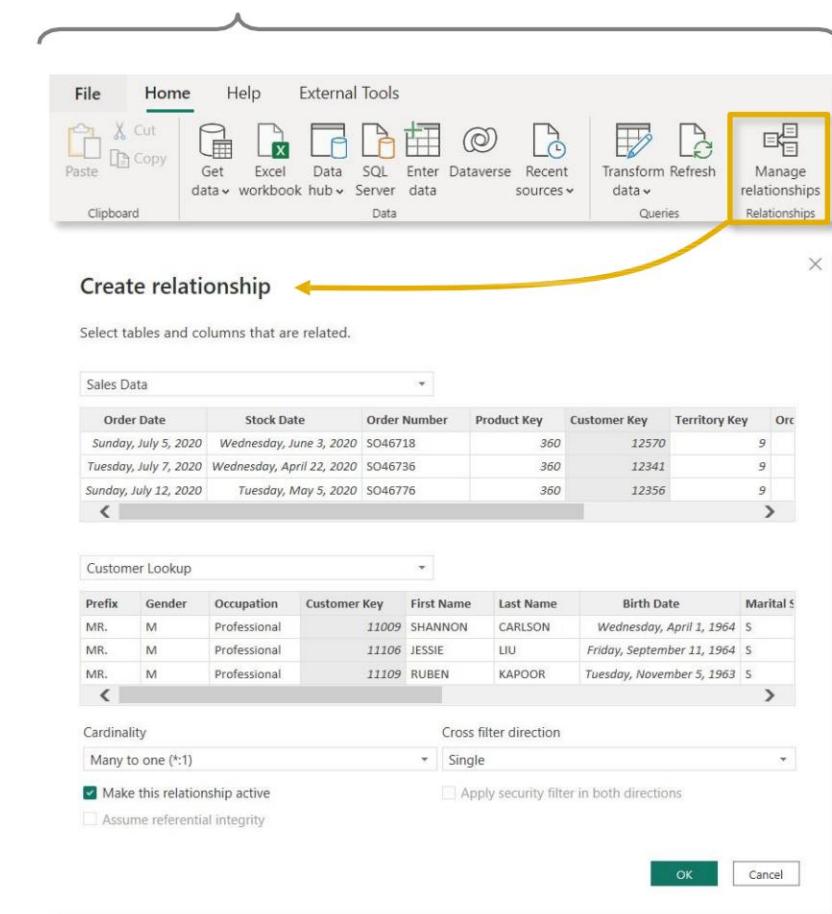


# CRIANDO RELACIONAMENTOS DE MESA

**OPÇÃO 1:** Clique e arraste para conectar chaves primárias e estrangeiras na visualização do **modelo**



**OPÇÃO 2:** Adicionar ou detectar relacionamentos usando a caixa de diálogo **Gerenciar Relacionamentos**





# GERENCIANDO E EDITANDO RELACIONAMENTOS

**Manage relationships**

The screenshot shows the Power BI ribbon with the 'Relationships' icon highlighted. A large yellow arrow points from this icon down to the 'Manage relationships' dialog box. The dialog box displays a list of existing relationships between tables and columns, with several checkboxes checked. At the bottom, there are buttons for 'New...', 'Autodetect...', 'Edit...', and 'Delete'. The 'Edit...' button is also highlighted with a yellow box.

Inicie a caixa de diálogo **Gerenciar Relacionamentos** ou clique duas vezes em um relacionamento para modificá-lo

## Edit relationship

Select tables and columns that are related.

Sales Data

Order Date	Stock Date	Order Number	Product Key	Customer Key	Territory Key	Orc
Sunday, July 5, 2020	Wednesday, June 3, 2020	SO46718	360	12570	9	
Tuesday, July 7, 2020	Wednesday, April 22, 2020	SO46736	360	12341	9	
Sunday, July 12, 2020	Tuesday, May 5, 2020	SO46776	360	12356	9	

Customer Lookup

Prefix	Gender	Occupation	Customer Key	First Name	Last Name	Birth Date	Marital S
MR.	M	Professional	11009	SHANNON	CARLSON	Wednesday, April 1, 1964	S
MR.	M	Professional	11106	JESSIE	LIU	Friday, September 11, 1964	S
MR.	M	Professional	11109	RUBEN	KAPOOR	Tuesday, November 5, 1963	S

Cardinality: Many to one (\*:1)      Cross filter direction: Single

Make this relationship active       Apply security filter in both directions

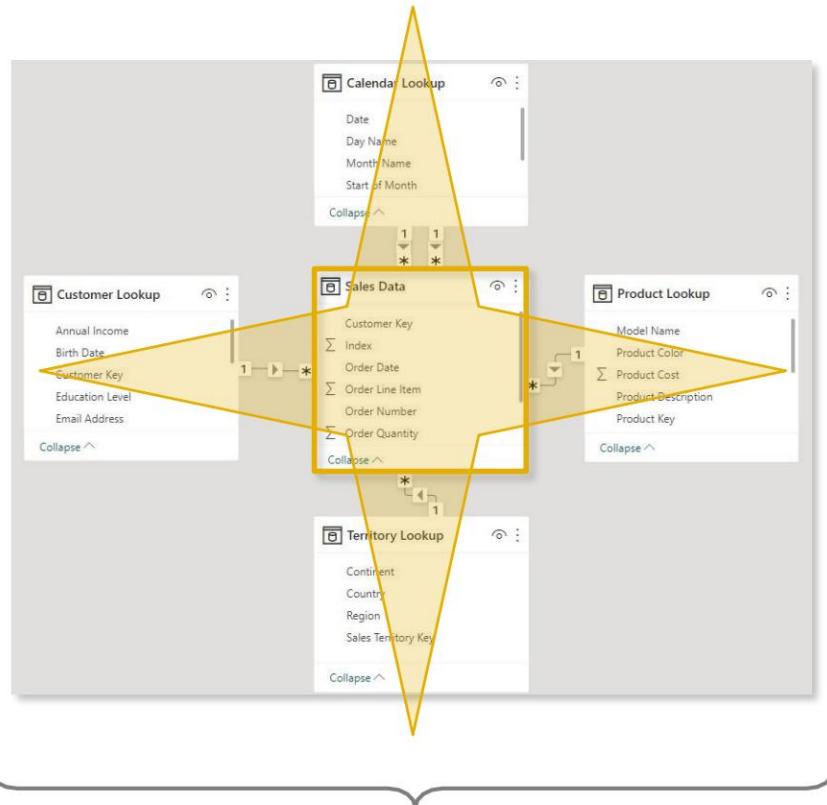
Assume referential integrity

OK      Cancel

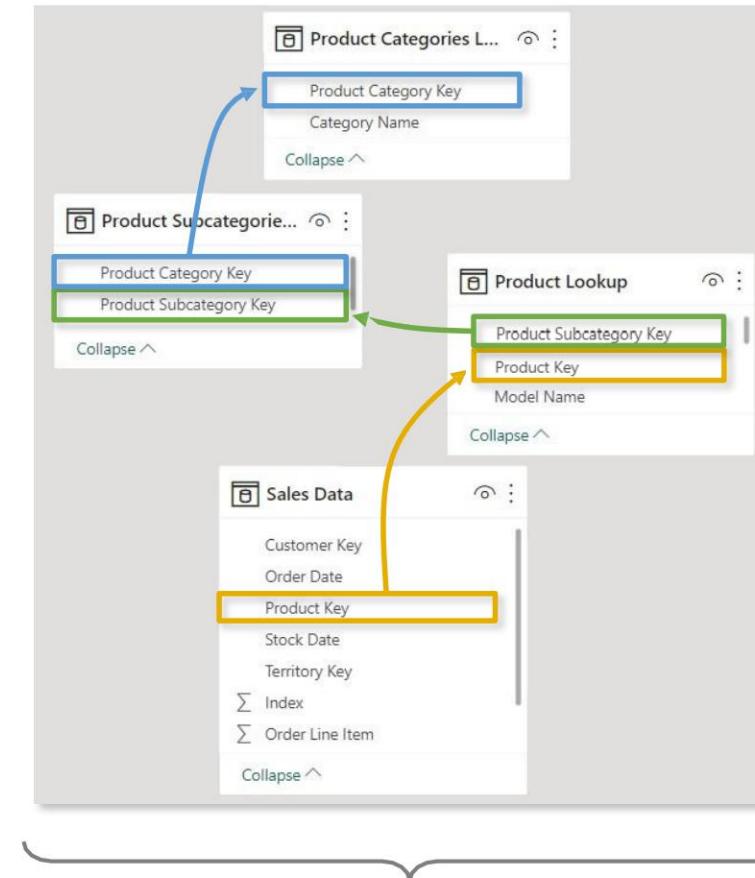
As ferramentas de edição permitem **ativar ou desativar** relacionamentos e gerenciar **a cardinalidade e a direção do filtro** – mais sobre isso em breve!



# ESQUEMAS DE ESTRELA E FLOCO DE NEVE



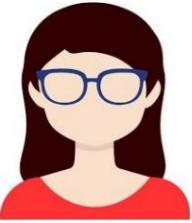
Um **esquema em estrela** é o tipo mais simples e comum de modelo de dados, caracterizado por uma única tabela de fatos cercada por tabelas de dimensões relacionadas.



Um **esquema floco de neve** é uma extensão de uma estrela e inclui relacionamentos entre tabelas de dimensões e tabelas de subdimensões relacionadas.



# TAREFA: RELACIONAMENTOS DE MESA



**NOVA MENSAGEM**

De: **Dana Modelle (Analista)**

Assunto: **Preciso de um favor...**

Ei,

Ethan compartilhou o modelo de dados em que você está trabalhando e podemos ter um problema...

Ontem à noite deixei meu laptop aberto e meu gato Dennis, de alguma forma, colocou as patas em nosso modelo. Agora todos os relacionamentos acabaram!

Você poderia reconstruir o modelo, incluindo todas as três tabelas de produtos? Eu devo-te uma!

-Dana

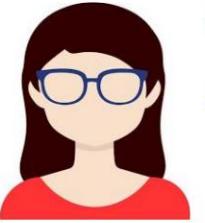
[Reply](#) [Forward](#)

## *Objetivos principais*

1. Exclua todos os relacionamentos de tabela existentes
2. Crie um esquema em estrela criando relacionamentos entre as tabelas Vendas, Calendário, Cliente, Produto e Territórios
3. Conecte todas as três tabelas de produtos (Produto, Subcategoria, Categoria) em um esquema de floco de neve
4. Use o visual da matriz para confirmar que você pode filtrar valores de quantidade do pedido usando campos de cada tabela de dimensão



# SOLUÇÃO: RELACIONAMENTOS DE TABELA

  NOVA MENSAGEM

De: **Dana Modelle (Analista)**

Assunto: Preciso de um favor...

Ei,

Ethan compartilhou o modelo de dados em que você está trabalhando e podemos ter um problema...

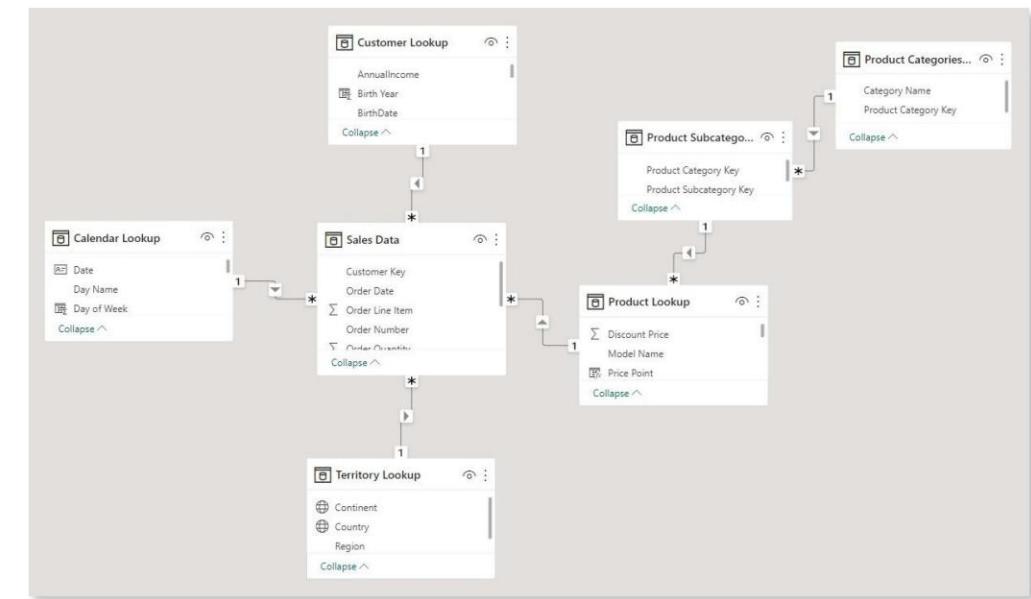
Ontem à noite deixei meu laptop aberto e meu gato Dennis, de alguma forma, colocou as patas em nosso modelo. Agora todos os relacionamentos acabaram!

Você poderia reconstruir o modelo, incluindo todas as três tabelas de produtos? Eu devo-te uma!

-Dana

[Reply](#) [Forward](#)

## Visualização da solução





# DICA PRO: RELACIONAMENTOS ATIVOS E INATIVOS

**Edit relationship**

Select tables and columns that are related.

Order Date	Stock Date	Order Number	Product Key	Customer Key	Territory Key	Order Line Item	C
7/5/2020	6/3/2020	SO46718	360	12570	9	1	1
7/7/2020	4/22/2020	SO46736	360	12341	9	1	1
7/12/2020	5/5/2020	SO46776	360	12356	9	1	1

Date	Day Name	Start of Week	Start of Month	Month Name	Start of Year	Year
1/1/2020	Wednesday	12/29/2019	1/1/2020	January	1/1/2020	2020
1/2/2020	Thursday	12/29/2019	1/1/2020	January	1/1/2020	2020
1/3/2020	Friday	12/29/2019	1/1/2020	January	1/1/2020	2020

Cardinality: Many to one (\*:1)

Make this relationship active

Apply security filter in both directions

Assume referential integrity

OK Cancel

**Properties**

Relationship

Table: Sales Data Column: Order Date

Cardinality: Many to one (\*:1)

Table: Calendar Lookup Column: Date

Make this relationship active

Yes

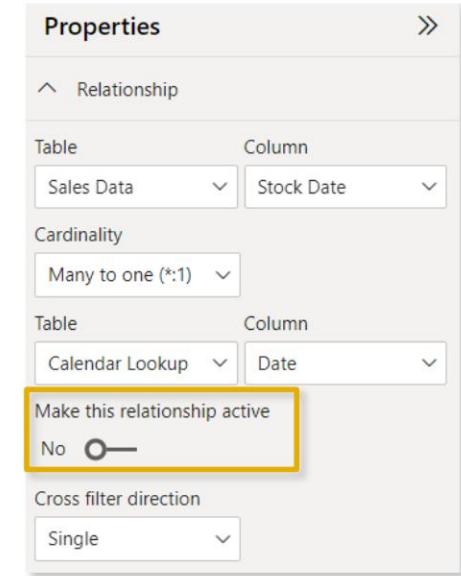
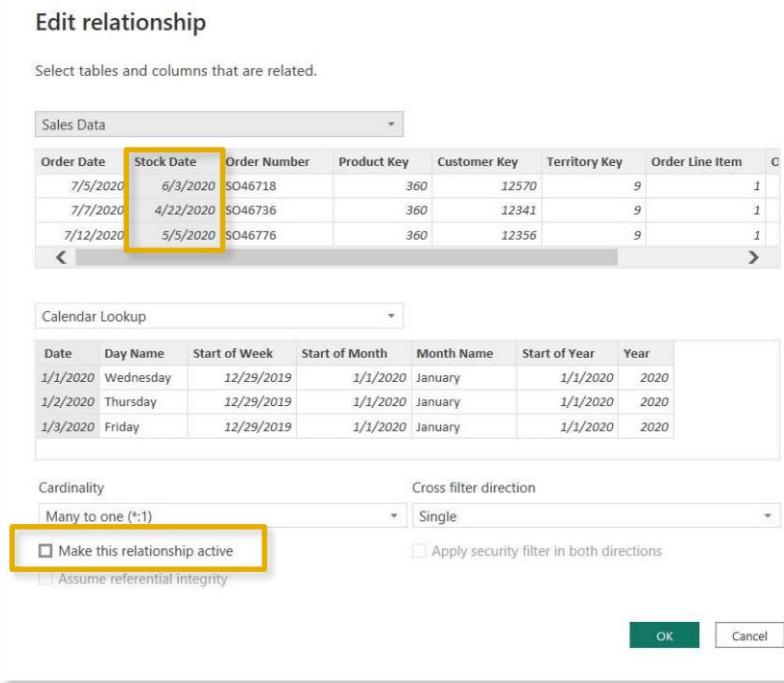
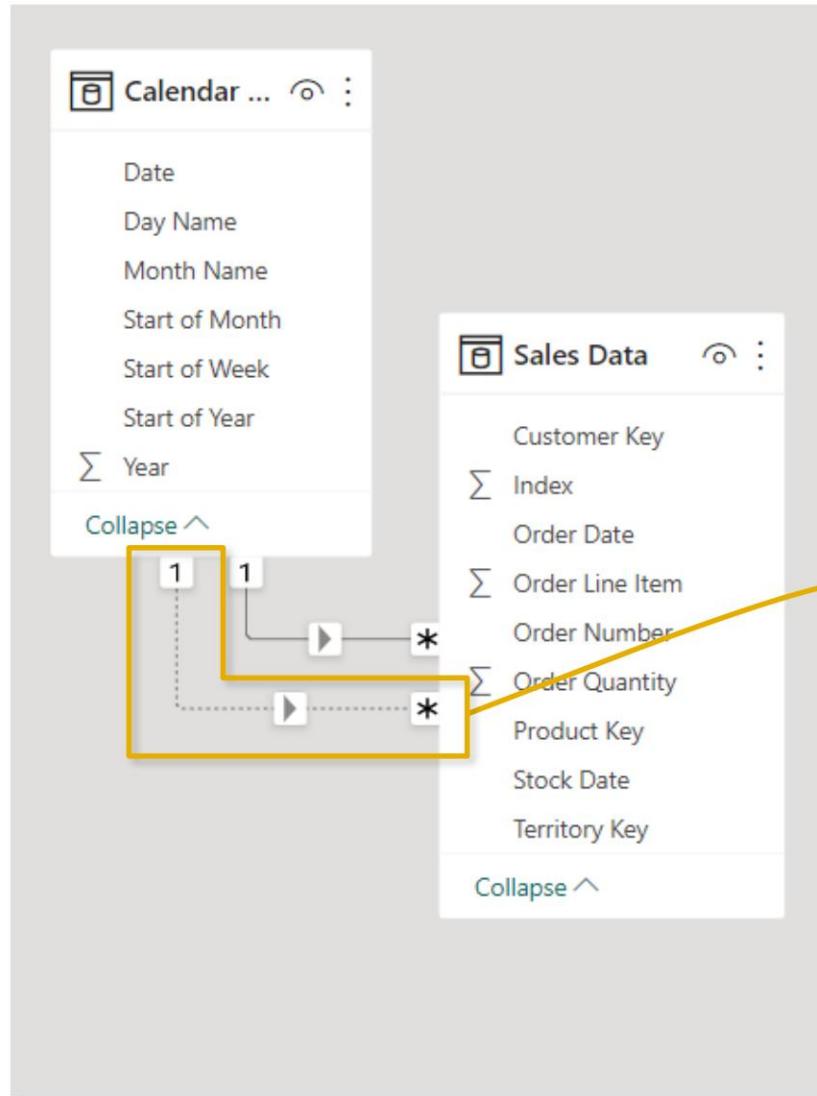
Cross filter direction: Single

A tabela **Dados de vendas** contém dois campos de data (**Data do pedido** e **Data do estoque**), mas só pode haver **um relacionamento ativo** com a chave Data na tabela Calendário

Você pode definir relacionamentos como ativos ou inativos na caixa de diálogo **Editar Relacionamentos** ou nas **Propriedades** (você deve desativar um antes de ativar outro)



# DICA PRO: RELACIONAMENTOS ATIVOS E INATIVOS

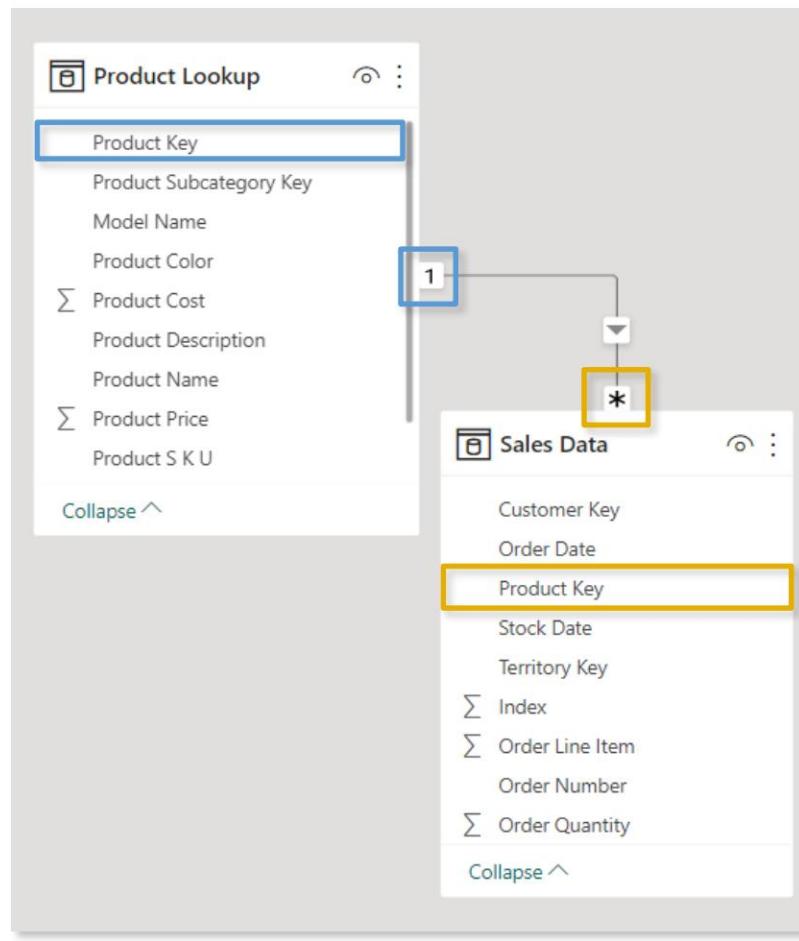


A tabela **Dados de vendas** contém dois campos de data (**Data do pedido e Data do estoque**), mas só pode haver **um relacionamento ativo** com a chave Data na tabela Calendário

Você pode definir relacionamentos como ativos ou inativos na caixa de diálogo **Editar Relacionamentos** ou nas **Propriedades** (você deve desativar um antes de ativar outro)



# CARDINALIDADE DE RELACIONAMENTO



**Cardinalidade** refere-se à singularidade dos valores em uma coluna

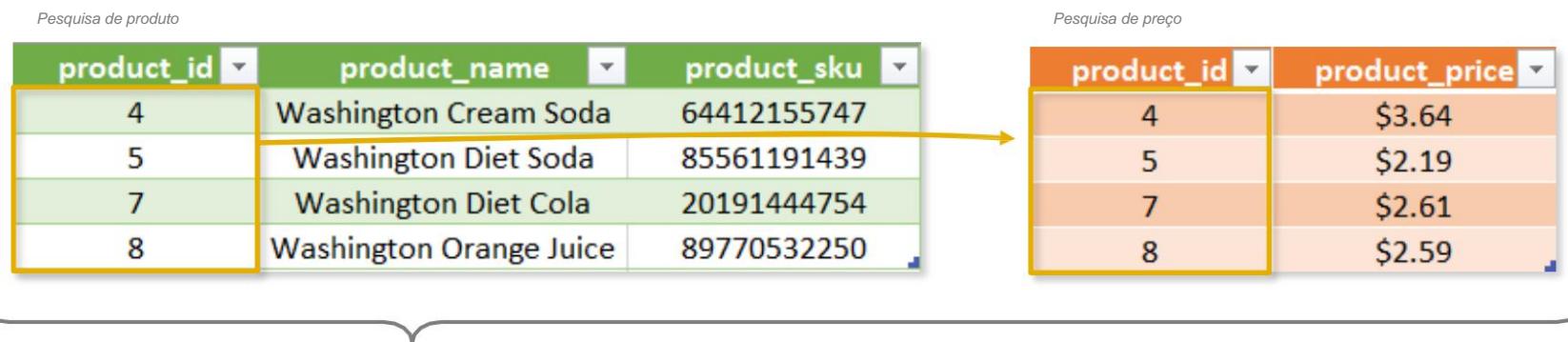
- Idealmente, todos os relacionamentos no modelo de dados devem seguir um **Cardinalidade um-para-muitos** : uma instância de cada chave primária e **muitas** instâncias de cada chave estrangeira

Neste exemplo, há apenas **UMA instância de cada Chave de Produto** na tabela **Produto** (indicada por “1”), pois cada linha contém **atributos de um único produto** (nome, SKU, descrição, preço, etc.)

Existem **MUITAS instâncias de cada chave de produto** na tabela **Vendas** (indicadas por um asterisco \*), pois há **várias vendas para cada produto**



# EXEMPLO: CARDINALIDADE UM PARA UM



- Conectar as duas tabelas acima usando **product\_id** cria um **relacionamento um para um**, já que cada ID de produto aparece apenas uma vez em cada tabela
- Este não é necessariamente um relacionamento “ruim”, mas você pode simplificar o modelo mesclando as tabelas em uma tabela de dimensão única e válida

product_id	product_name	product_sku	product_price
4	Washington Cream Soda	64412155747	\$3.64
5	Washington Diet Soda	85561191439	\$2.19
7	Washington Diet Cola	20191444754	\$2.61
8	Washington Orange Juice	89770532250	\$2.59

*NOTA:* isso ainda respeita as regras de normalização, pois todas as linhas são únicas e capturam atributos específicos do produto



# EXEMPLO: CARDINALIDADE DE MUITOS PARA MUITOS

Pesquisa de produto

product_id	product_name	product_sku
4	Washington Cream Soda	64412155747
4	Washington Diet Cream Soda	81727382373
5	Washington Diet Soda	85561191439
7	Washington Diet Cola	20191444754
8	Washington Orange Juice	89770532250

Vendas

date	product_id	transactions
1/1/2017	4	12
1/2/2017	4	9
1/3/2017	4	11
1/1/2017	5	16
1/2/2017	5	19
1/1/2017	7	11

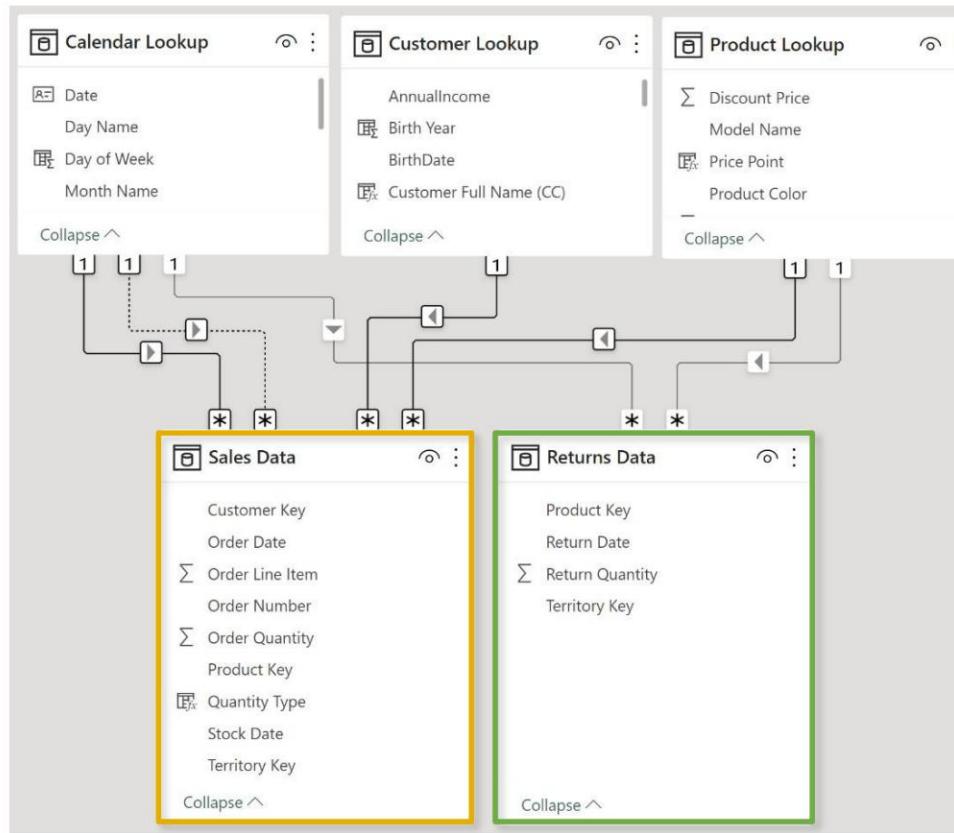


This relationship has cardinality Many-Many. This should only be used if it is expected that neither column (product\_id and product\_id) contains unique values, and that the significantly different behavior of Many-many relationships is understood. [Learn more](#)

- Se tentarmos conectar as tabelas acima usando **product\_id**, obteremos um **relacionamento muitos-para-muitos** aviso, pois existem várias instâncias de product\_id em ambas as tabelas
- Mesmo se forçarmos esta relação, como saberíamos qual produto foi realmente vendido em cada data – **Cream Soda** ou **Diet Cream Soda**?



# CONECTANDO MÚLTIPAS TABELAS DE FATOS



Este modelo contém duas tabelas de fatos: **Dados de Vendas** e **Dados de retorno**

- Como não há relacionamento de chave primária/estrangeira, não podemos conectá-los diretamente entre si
- Mas *podemos* conectar cada tabela de fatos a pesquisas relacionadas, o que nos permite filtrar dados de vendas e devoluções **usando campos de qualquer tabela de pesquisa compartilhada**
- Podemos visualizar pedidos e devoluções por produto, pois ambas as tabelas estão relacionadas à Pesquisa de Produto, mas não podemos visualizar devoluções por cliente, pois não existe relacionamento

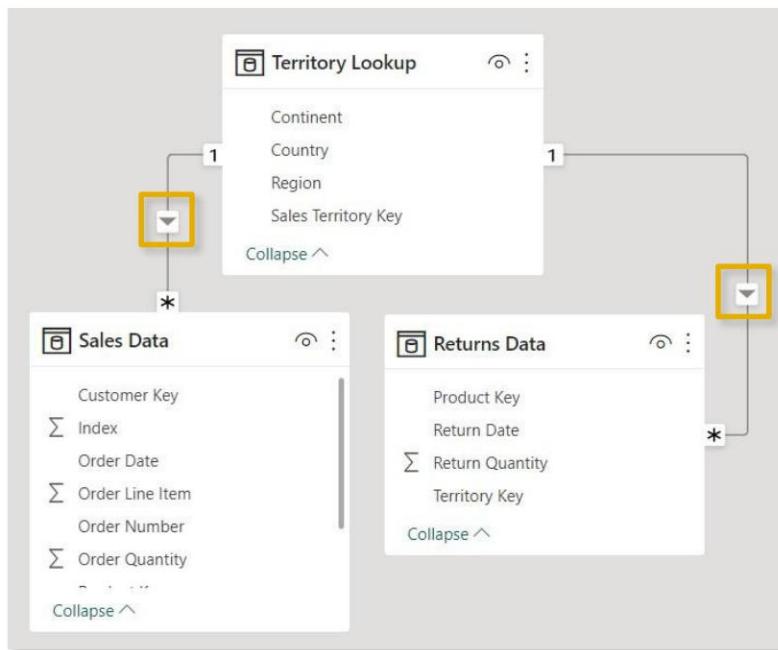


## Ei, isso é importante!

De modo geral, as tabelas de fatos devem **se conectar por meio de tabelas de dimensões compartilhadas**, e não diretamente entre si.



# CONTEXTO E FLUXO DO FILTRO



Aqui temos duas tabelas de dados ( Dados de Vendas e **Dados de Devoluções**), conectadas ao **Territory Lookup**

As setas mostram a **direção do filtro** e apontam de um **(1)** lado do relacionamento para muitos **(\*)** lados

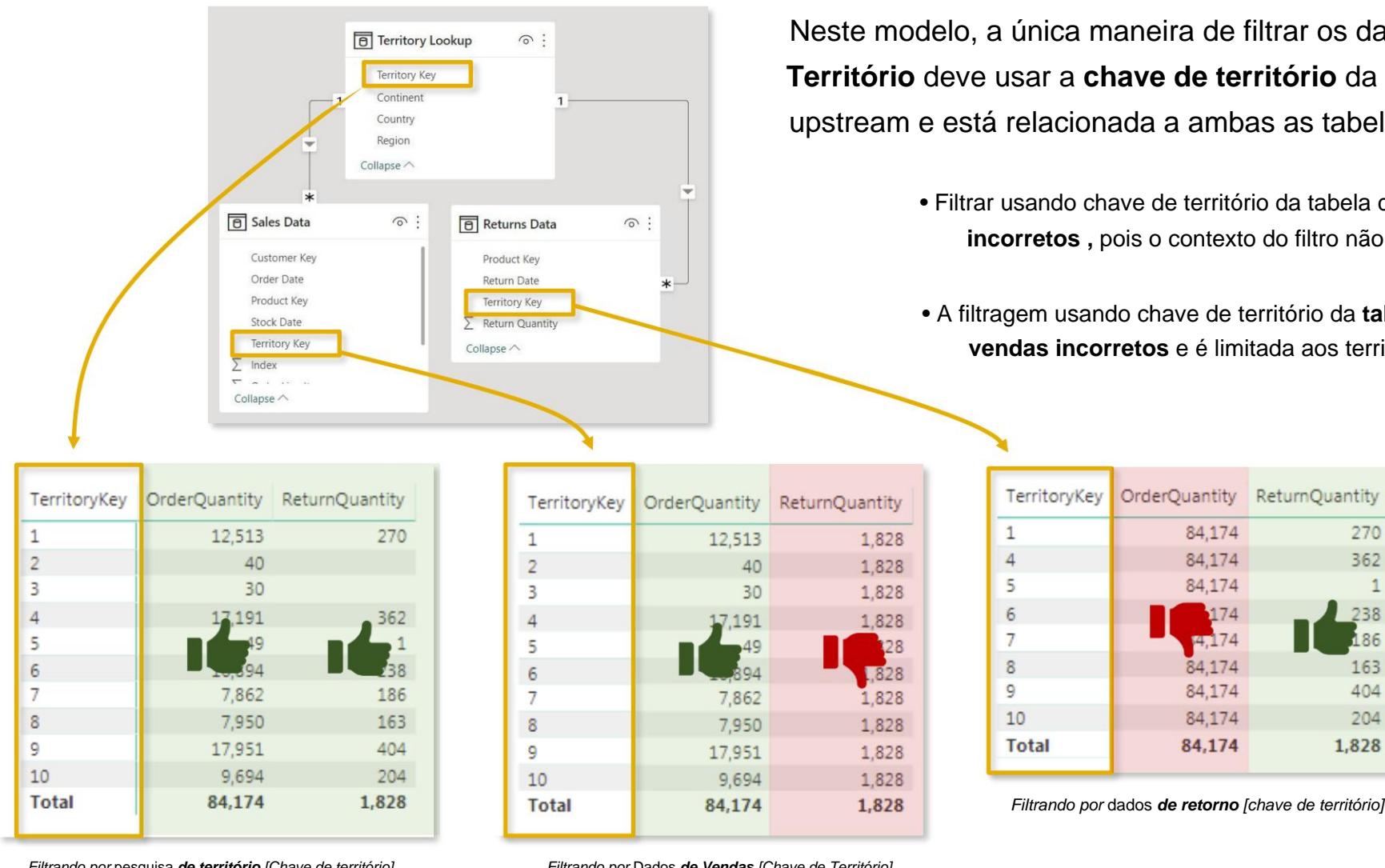
- Quando você filtra uma tabela, esse **contexto de filtro** é passado para qualquer tabelas “downstream” relacionadas, seguindo a direção da seta
- O contexto do filtro NÃO PODE fluir “upstream”



**DICA PRO:** organize tabelas de pesquisa acima das tabelas de fatos em seu modelo como um lembrete visual de que **os filtros sempre fluem downstream**



# EXEMPLO: FLUXO DO FILTRO



Neste modelo, a única maneira de filtrar os dados **de Vendas e Devoluções** por **Território** deve usar a **chave de território** da tabela de pesquisa, que é upstream e está relacionada a ambas as tabelas de fatos

- Filtrar usando chave de território da tabela comercial gera **valores de retorno incorretos**, pois o contexto do filtro não pode fluir para nenhuma outra tabela
- A filtragem usando chave de território da **tabela de devoluções** gera **valores de vendas incorretos** e é limitada aos territórios existentes na tabela de devoluções



# FILTROS BIDIRECIONAIS

**Edit relationship**

Select tables and columns that are related.

Sales Data

Order Date	Stock Date	Order Number	Product Key	Customer Key	Territory Key	Or
Sunday, July 5, 2020	Wednesday, June 3, 2020	SO46718	360	12570	9	
Tuesday, July 7, 2020	Wednesday, April 22, 2020	SO46736	360	12341	9	
Sunday, July 12, 2020	Tuesday, May 5, 2020	SO46776	360	12356	9	

Territory Lookup

Region	Country	Continent	Sales Territory Key
Northwest	United States	North America	1
Northeast	United States	North America	2
Central	United States	North America	3

Cardinality

Many to one (\*:1)

Make this relationship active

Assume referential integrity

Cross filter direction: Both

OK Cancel

**Relationship Diagram:** Shows three tables: Sales Data, Territory Lookup, and Returns Data. Sales Data has a many-to-one relationship with Territory Lookup (Sales Territory Key) and a many-to-many relationship with Returns Data (Territory Key). Territory Lookup has a one-to-many relationship with Sales Data and a one-to-many relationship with Returns Data. Returns Data has a many-to-many relationship with Sales Data. A yellow arrow points from the Sales Data table to the Returns Data table, indicating the flow of filters between them.

**Properties Panel:**

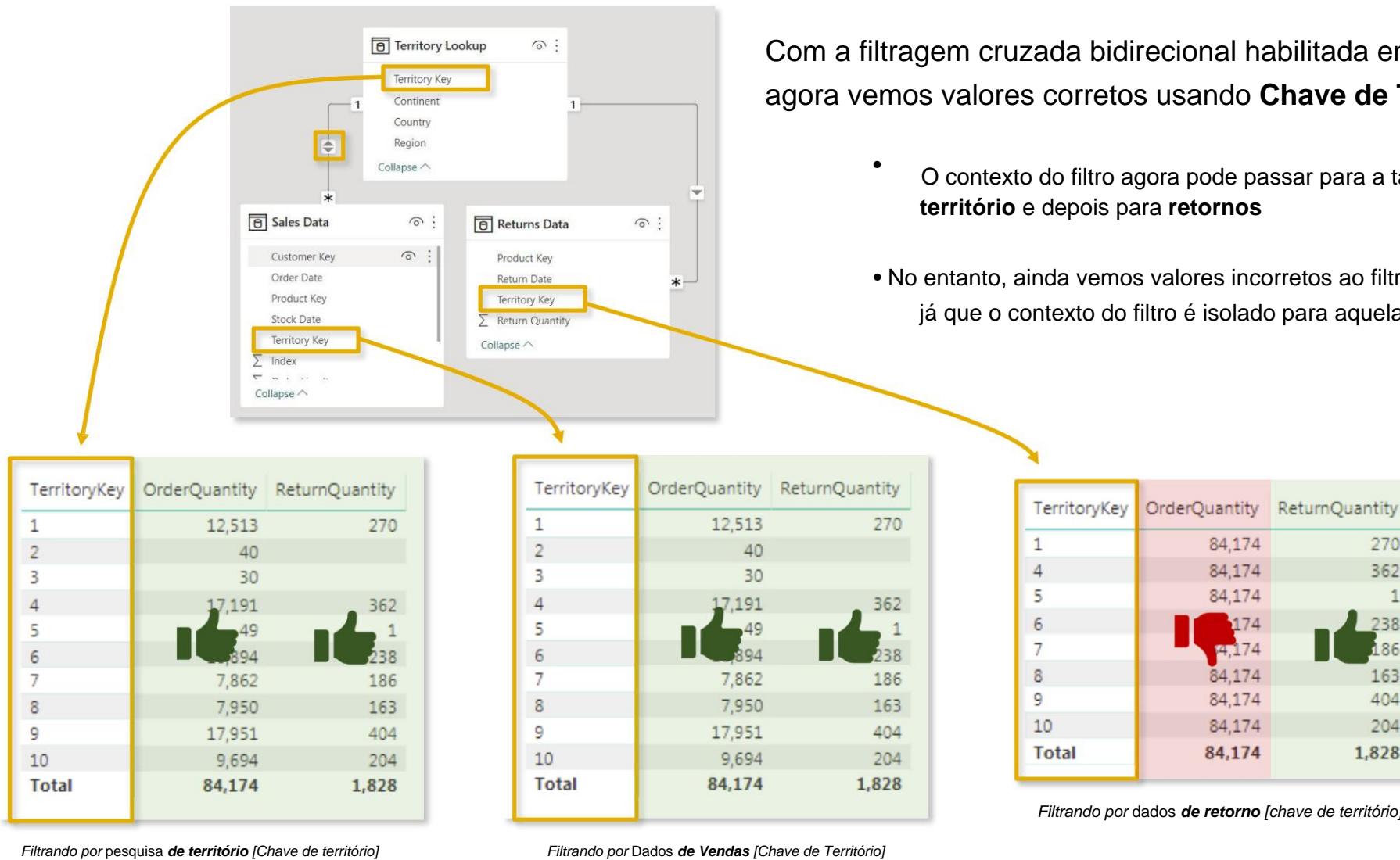
- Relationship:** Table: Sales Data, Column: Territory Key; Cardinality: Many to one (\*:1); Table: Territory Lookup, Column: Sales Territory Key.
- Cross filter direction:** Both (highlighted with a yellow box).
- Apply security filter in both directions:** Yes (radio button selected).

Atualizar a **direção do filtro cruzado** de **Único** para **Ambos** permite que o contexto do filtro flua em qualquer direção

- Neste exemplo, os filtros aplicados à Salestable podem passar para a tabela **Territory Lookup** e depois para **Returns**



# EXEMPLO: FILTROS BIDIRECIONAIS



Com a filtragem cruzada bidirecional habilitada entre **Vendas** e **Território**, agora vemos valores corretos usando **Chave de Território** de *qualquer* tabela

- O contexto do filtro agora pode passar para a tabela **de pesquisa de território** e depois para **retornos**
- No entanto, ainda vemos valores incorretos ao filtrar usando **Territory Key** da **Returnstable**, já que o contexto do filtro é isolado para aquela única tabela



# EXEMPLO: FILTROS BIDIRECIONAIS



Neste caso, habilitamos a filtragem cruzada bidirecional entre o **Tabelas de retornos e territórios**

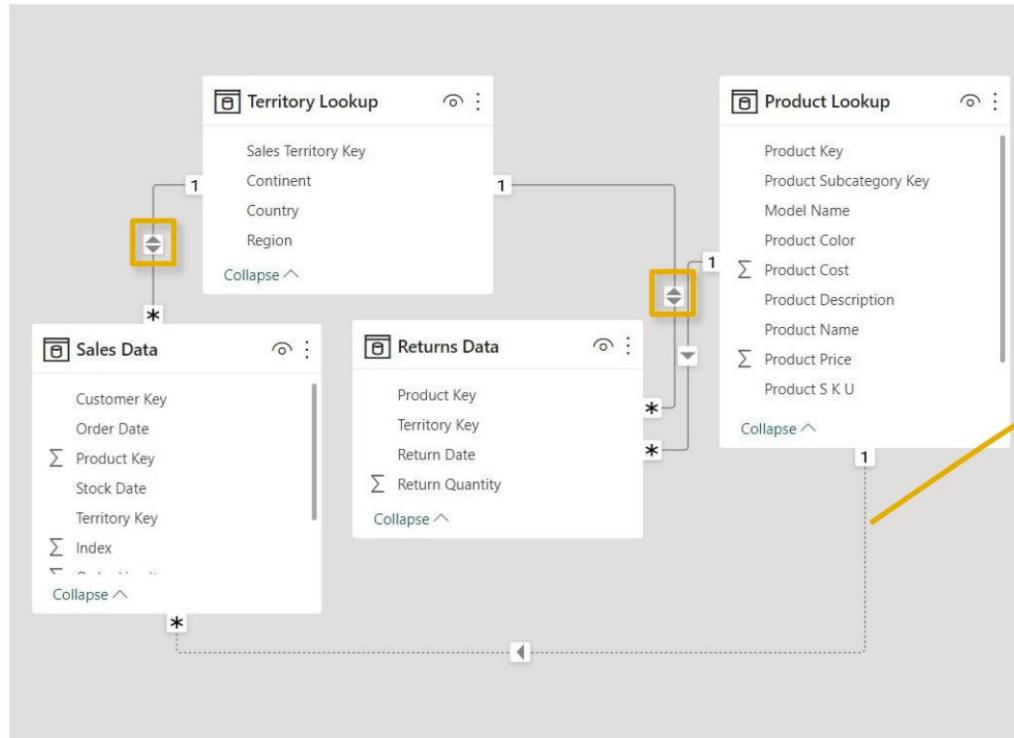
- Como esperado, agora vemos valores incorretos ao filtrar usando chave de território da tabela **Vendas**, já que o contexto do filtro está isolado nessa tabela única
- Embora os valores *pareçam* corretos ao filtrar usando chave de território da **Returnstable**, estamos **faltando dados de vendas** de quaisquer territórios que não apareceram na tabela de devoluções (especificamente Territórios **2 e 3**)

Os territórios 2 e 3 não existem na tabela **Returns**, portanto não são incluídos no contexto de filtro que passa para **Pesquisa e vendas** de território

Filtrando por dados de retorno [chave de território]



# AMBIGUIDADE



Use filtros bidirecionais com cuidado e **somente quando necessário**

- O uso de vários filtros bidirecionais pode causar **ambiguidade** ao introduzir vários caminhos de filtro entre tabelas

! You can't create a direct active relationship between Sales\_Data and Product\_Lookup because that would introduce ambiguity between tables Product\_Lookup and Territory\_Lookup. To make this relationship active, deactivate or delete one of the relationships between Product\_Lookup and Territory\_Lookup first.

Neste exemplo, o contexto de filtro da tabela **Produto** pode ser transmitido para **Retorna** e até **Pesquisa de Território**, que seria filtrada com base no Chaves de território passadas do Returnstable

Também com um relacionamento ativo entre **Produto** e **Vendas**, o contexto do filtro poderia passar pela **Tabela de Vendas ou Devolução** para chegar ao **Território Tabela de pesquisa**, que pode gerar contexto de filtro conflitante

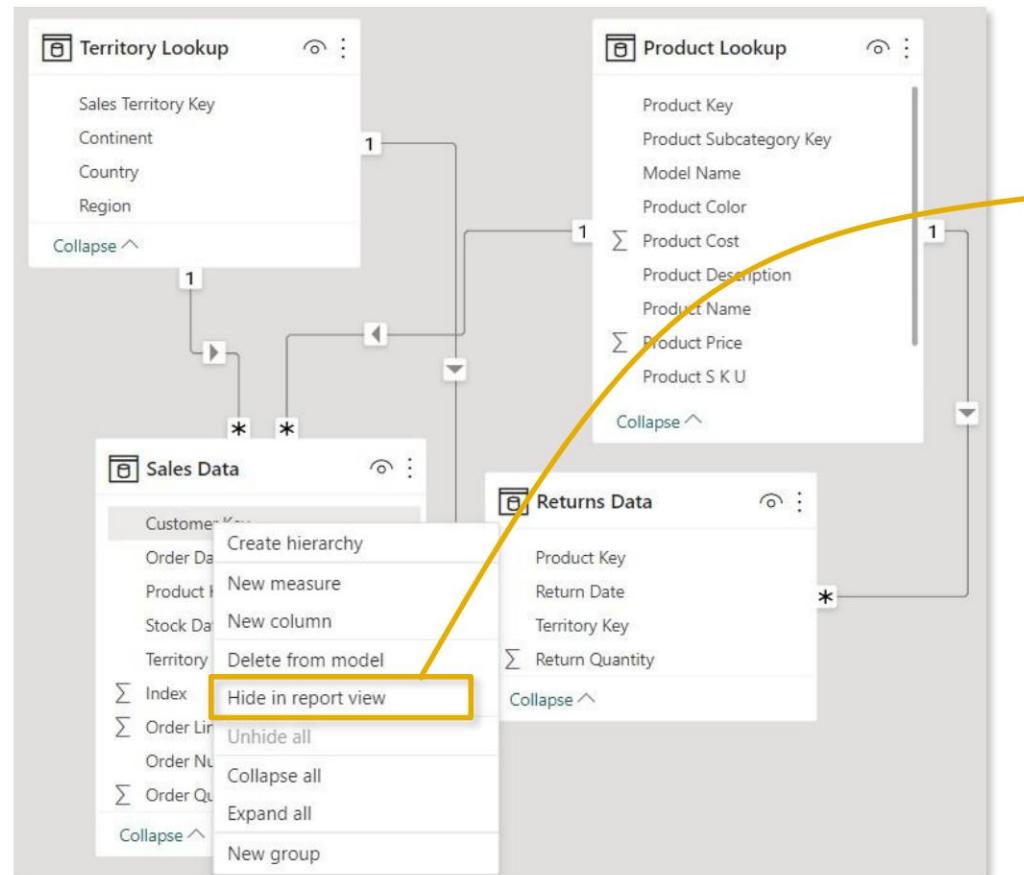
**DICA PRO:** Projete seus modelos com **filtros unidirecionais** e **cardinalidade 1:muitos**, a menos

que relacionamentos mais complexos sejam absolutamente necessários





# ESCONDENDO CAMPOS



**Ocultar na Visualização de Relatório** torna os campos inacessíveis a partir do Guia Relatório, mas ainda disponível nas visualizações **Dados e Modelo**

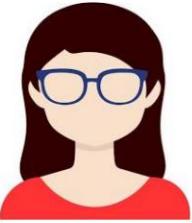
- Isso pode ser controlado clicando com o botão direito em um campo na visualização Dados ou Modelo ou selecionando “**Está oculto**” no painel Propriedades
- Isso é comumente usado para evitar que os usuários filtrem usando campos inválidos, reduzir a confusão ou ocultar métricas irrelevantes da visualização



**DICA PRO:** Oculte as **chaves estrangeiras** nas tabelas de fatos para forçar os usuários a filtrar usando **chaves primárias** nas tabelas de dimensão



# ATRIBUIÇÃO: FLUXO DO FILTRO

 NOVA MENSAGEM

De: **Dana Modelle** (Analista)

Assunto: **Larry ficou desonesto!**

Olá, temos outro problema.

Larry, do departamento de vendas, acabou de me enviar esta captura de tela. Acho que ele deve ter baixado nosso modelo de Power BI e mexido em alguns relacionamentos, porque SEI que tivemos vendas do produto 338.

Você pode ajudar a diagnosticar o que está acontecendo e evitar que ele faça isso novamente?

-Dana

PS Kevin diz oi 

[Reply](#)   [Forward](#)

## Objetivos principais

1. Replique a matriz de Larry abaixo para diagnosticar o que ele deve ter feito com o modelo\*

Product Key	Sum of Order Quantity	Sum of Return Quantity
322	55	2
324	72	3
326	65	3
328	75	4
330	51	6
332	64	2
334	63	2
336	50	1
340	56	1
342	72	1
346	24	2

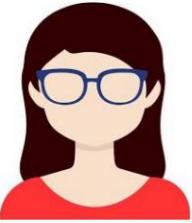
- Qual produto é o nº 338?
- Por que a matriz de Larry não mostrava nenhum pedido?

2. Oculte quaisquer chaves estrangeiras restantes para evitar que outros usuários cometam o mesmo erro

\*Dica: pode ser necessário alterar temporariamente um relacionamento para bidirecional



# SOLUÇÃO: FLUXO DO FILTRO

  NOVA MENSAGEM

De: **Dana Modelle** (Analista)

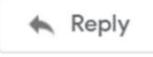
Assunto: **Larry ficou desonesto!**

Olá, temos outro problema.

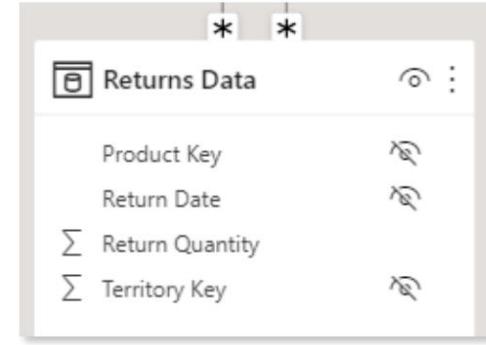
Larry, do departamento de vendas, acabou de me enviar esta captura de tela. Acho que ele deve ter baixado nosso modelo de Power BI e mexido em alguns relacionamentos, porque SEI que tivemos vendas do produto 338. Você pode ajudar a diagnosticar o que está acontecendo e evitar que ele faça isso novamente?

-Dana

PS Kevin diz oi 

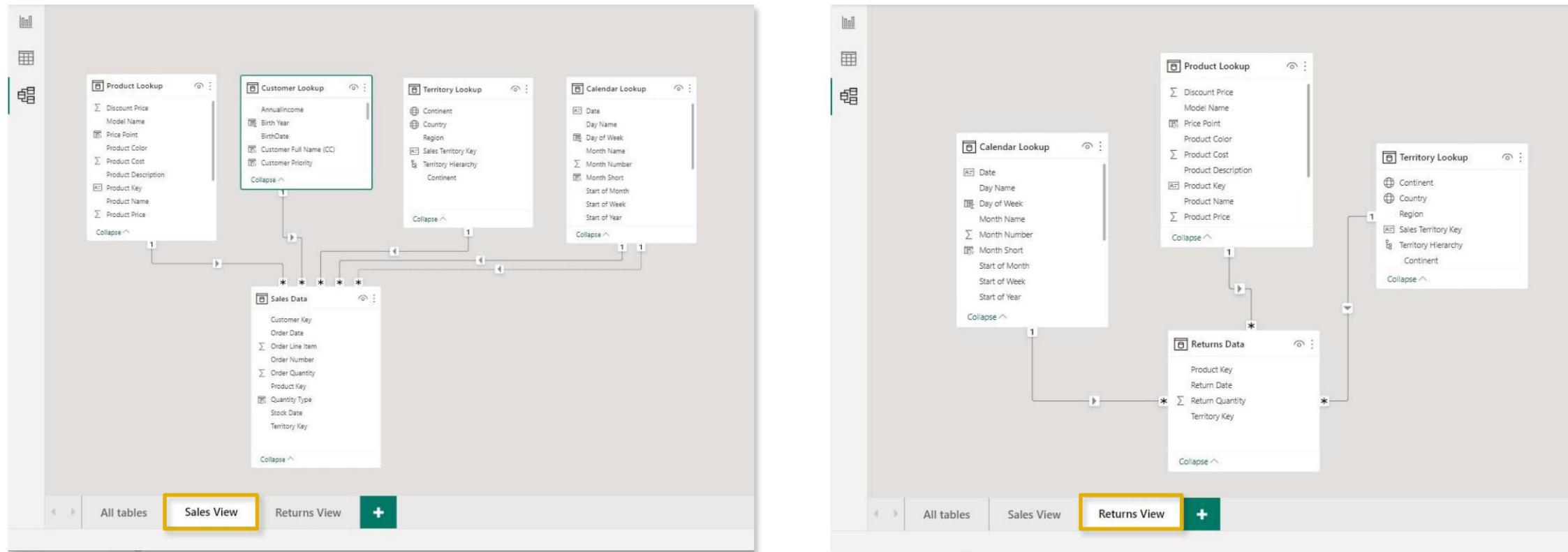
 Reply     Forward

## Visualização da solução

1. Larry deve ter mudado o relacionamento entre **Returns Data** e **Product Lookup** para **bidirecional** e filtrou sua matriz usando `product_id` da tabela **Returns**
  - Bicicleta de estrada (Road-650 Preto, 44)
  - O produto 338 não existe nas Devoluções tabela, então foi excluído quando esse filtro contexto passado para a tabela Vendas
2. 



# DICA PRO: LAYOUTS DE MODELO



**Os layouts de modelo** permitem criar visualizações personalizadas para mostrar partes específicas de modelos grandes e complexos

- Aqui criamos uma **Visualização de Vendas** exibindo apenas tabelas relacionadas a vendas e uma **Visualização de Devoluções** exibindo apenas tabelas relacionadas para retornos (*Nota: isso não cria tabelas duplicadas*)



# FORMATOS E CATEGORIAS DE DADOS

File Home Help External Tools **Table tools** **Column tools**

Name: Country Data type: Text

Structure

Region	Country	Continent	Sales Territory Key
Northwest	United States	North America	1
Northeast	United States	North America	2
Central	United States	North America	3
Southwest	United States	North America	4
Southeast	United States	North America	5
Canada	Canada	North America	6
France	France	Europe	7
Germany	Germany	Europe	8
Australia	Australia	Pacific	9
United Kingdom	United Kingdom	Europe	10

\$% Format Text **Formatting**

Σ Summarization Don't summarize

Data category: **Country**

Uncategorized  
Address  
Place  
City  
County  
State or Province  
Postal code  
**Country**  
Continent  
Latitude  
Longitude  
Web URL  
Image URL  
Barcode

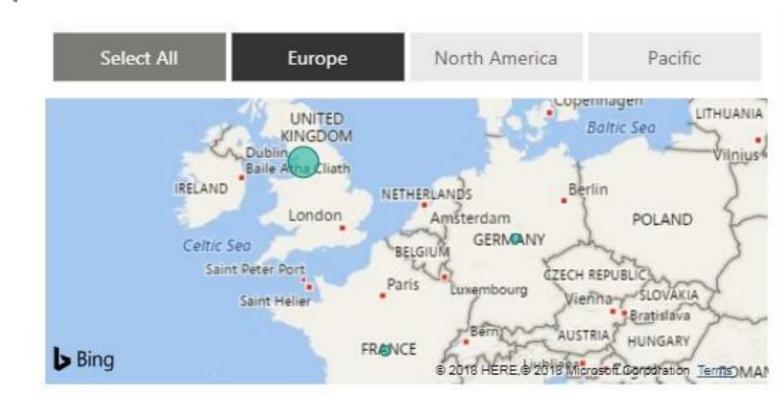
Sort by column Sort

Personalize **os formatos de dados** no menu Ferramentas de coluna na seção

Visualização de dados ou painel Propriedades na visualização **Modelo**

Atribuir **categorias de dados para** campos geoespaciais, URLs ou códigos de barras

- Isso é comumente usado para ajudar o Power BI a mapear campos baseados em localização, como endereços, países, cidades, coordenadas, CEPs, etc.





# HIERARQUIAS

**Hierarquias** são grupos de colunas que refletem vários níveis de granularidade

- Por exemplo, uma **hierarquia Geográfica** pode incluir campos **País, Estado e Cidade**
- As hierarquias são tratadas como um **único item** em tabelas e relatórios, permitindo que os usuários façam “detalhamento” e “detalhamento” em cada nível

The image consists of three side-by-side screenshots from the Microsoft Power BI desktop application:

- Screenshot 1 (Left):** Shows a context menu for the "Product Price" column. The "Create hierarchy" option is highlighted with a yellow box and has a yellow arrow pointing to it from the text below.
- Screenshot 2 (Middle):** Shows the "Territory Hierarchy" node expanded under the "Territory Lookup" section. The "Continent" node is also highlighted with a yellow box and has a yellow arrow pointing to it from the text below.
- Screenshot 3 (Right):** Shows a context menu for the "Country" column. The "Add to hierarchy" option is highlighted with a yellow box and has a yellow arrow pointing to it from the text below.

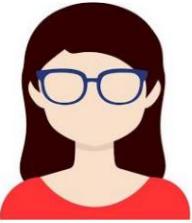
No painel **Dados**, clique com o botão direito em um campo e selecione **Criar hierarquia**

Esta hierarquia contém “Continente” e é denominada “**Hierarquia de Território**”

Clique com o botão direito em outro campo (como “País”) e selecione **Adicionar à hierarquia** (ou arraste-o!)



# ATRIBUIÇÃO: HIERARQUIAS



**NOVA MENSAGEM**

De: **Dana Modelle** (Analista)

Assunto: Adicionando uma hierarquia de datas

Bom dia!

Esperando que você possa ajudar com uma solicitação rápida.

Como faremos muitas análises de séries temporais, Ethan nos pediu para adicionar uma hierarquia de datas ao modelo para que os usuários pudessem visualizar rapidamente tendências em qualquer nível de granularidade (ano, mês, dia, etc.).

Por favor, adicione isso antes da nossa ligação da tarde. Obrigado!

-Dana

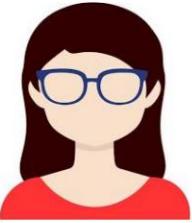
**Reply** **Forward**

## *Objetivos principais*

1. Crie uma nova hierarquia com base no campo **Início do Ano** e nomeie-a como “**Hierarquia de Data**”
2. Clique com o botão direito ou arraste para adicionar campos até que sua hierarquia contenha o seguinte (nesta ordem):
  - **Início do ano**
  - **Início do mês**
  - **Início da semana**
  - **Data**
3. Adicione sua nova hierarquia ao visual de matriz (em linhas) e pratique o detalhamento entre cada nível de granularidade



# SOLUÇÃO: HIERARQUIAS

  NOVA MENSAGEM

De: **Dana Modelle** (Analista)

Assunto: Adicionando uma hierarquia de datas

Bom dia!

Esperando que você possa ajudar com uma solicitação rápida.

Como faremos muitas análises de séries temporais, Ethan nos pediu para adicionar uma hierarquia de datas ao modelo para que os usuários pudessem visualizar rapidamente tendências em qualquer nível de granularidade (ano, mês, dia, etc.).

Por favor, adicione isso antes da nossa ligação da tarde. Obrigado!

-Dana

[Reply](#) [Forward](#)

---

## Visualização da solução

---

Data >

Search

> Measure Table

✓ Calendar Lookup

     Average Revenue per Customer

     Date

✓ Date Hierarchy

    Start of Year

    Start of Month

    Start of Week

     Date



## MELHORES PRÁTICAS DO MODELO DE DADOS



### Concentre-se na construção de um modelo normalizado desde o início

- Aproveite os *relacionamentos* e certifique-se de que cada mesa atenda a um propósito claro e distinto



### Organize tabelas de dimensões acima das tabelas de dados em seu modelo

- Isso serve como um lembrete visual de que os filtros sempre fluem “a jusante”



### Evite relacionamentos complexos, a menos que seja absolutamente necessário • Procure usar

*relacionamentos de tabela 1 para muitos e filtros unidirecionais sempre que possível*



### Ocultar campos da visualização do relatório para evitar contexto de filtro inválido

- Isso força os usuários do relatório a filtrar usando chaves primárias de tabelas de dimensão