

www.sqlbi.com



Microsoft Partner

Gold Business Intelligence
Gold Data Platform

**SSAS
MAESTRO**
by Microsoft



Why is data modeling important?

Introduction to data modeling

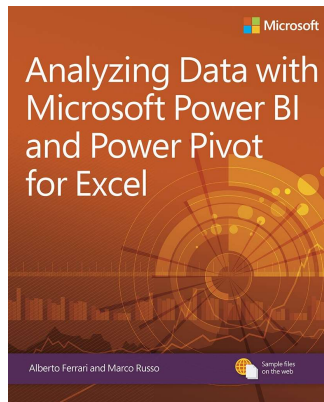
Marco Russo, SQLBI

Mail: marco@sqlbi.com

Twitter: @marcorus



We write
Books



We teach
Courses



We provide
Consulting



Remote
Consulting



Power BI/SSAS
Optimization



BI Architectural
Review



On-Site
Consulting



Custom Training
& Mentoring

We are recognized
BI Experts

Microsoft Partner

Gold Business Intelligence
Gold Data Platform

**Microsoft
CERTIFIED**
Technology Specialist

**Microsoft
CERTIFIED**
IT Professional

**Microsoft
CERTIFIED**
Trainer

MVP Microsoft
Most Valuable
Professional

**SSAS
MAESTRO**



www.sqlbi.com

Agenda

- Granularity, business entities
- Facts and dimensions
- Header/detail tables
- Multiple fact tables

Slides and demo



<https://sql.bi/402792>

Working with a single table



- In Excel, you work with a single table
- As simple as it is, it is already a data model
- It comes with several limitations
 - Number of rows: less than 1 Million
 - Speed and memory usage are not optimal
 - Can only perform basic calculations
- The limit on size becomes a limit on the data model

Granularity

- Granularity is the level of detail of your table
- The more the columns, the higher the granularity
- Higher granularity
 - More detailed information
 - More powerful model
 - Increase in the number of rows
- Lower granularity
 - Faster and smaller model
 - Less analytical power

Granularity and table size

- Increasing granularity increase the size of the model
 - More columns → More rows
- You quickly hit the limit of 1M rows of Excel...

Category	Sales
Bikes	10,000
Helmets	5,000

Category	Subcategory	Sales
Bikes	Cross bikes	8,000
Bikes	Mountain bikes	2,000
Helmets	Colorful helmets	2,000
Helmets	Lightweight helmets	3,000

Scattered information



- Higher granularity is not always the best choice
- Too high is as bad as too low
- Example: yearly income repeated on every row

ProductCategoryName	
<input type="checkbox"/> Audio	
<input type="checkbox"/> Cameras and camcorders	
<input type="checkbox"/> Cell phones	
<input type="checkbox"/> Computers	
<input type="checkbox"/> Games and Toys	
<input checked="" type="checkbox"/> Home Appliances	
<input type="checkbox"/> Music, Movies and Audio Books	
<input type="checkbox"/> TV and Video	

BrandName	AverageYearlyIncome
Wide World Importers	\$9,765,456.65
Proseware	\$9,586,214.41
Northwind Traders	\$2,230,398.67
Litware	\$9,170,201.49
Fabrikam	\$9,461,956.24
Contoso	\$8,307,093.90
Adventure Works	\$9,614,894.80
Total	\$8,957,859.39

Leveraging the data model

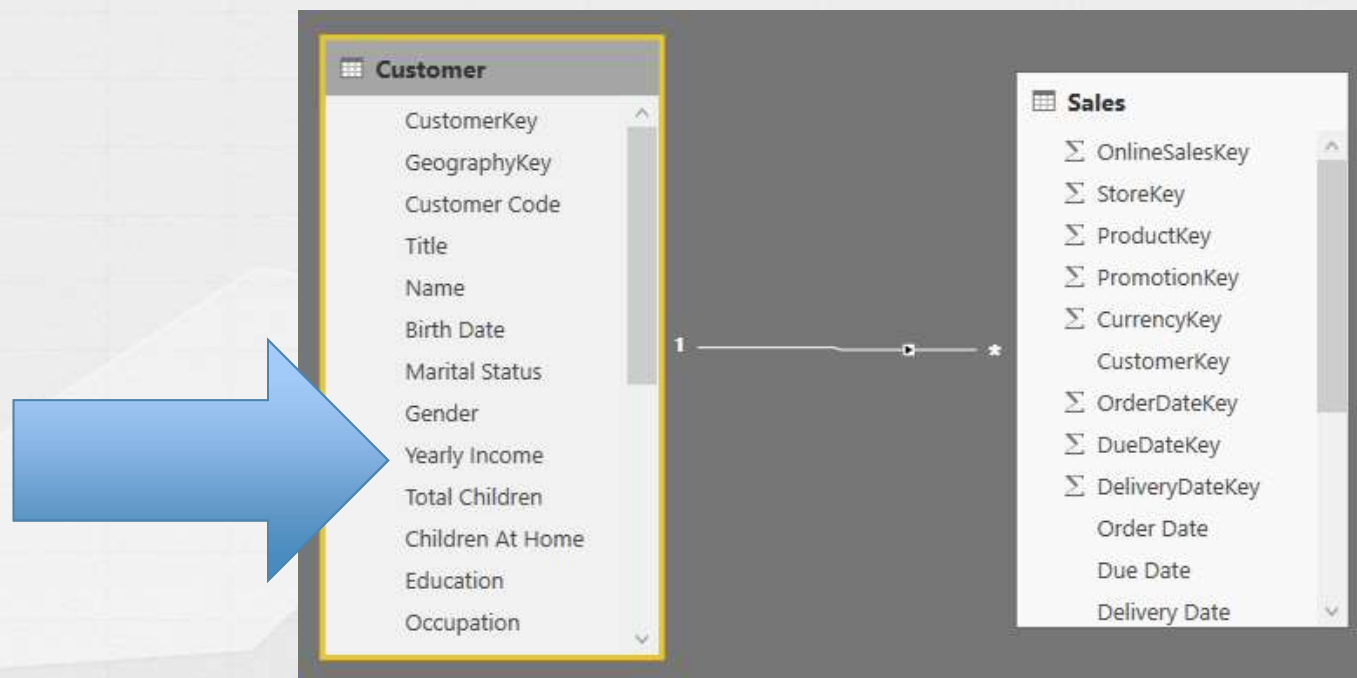


- Using the data model, you can load multiple tables
- Load Customers and Sales as separate tables
- Two tables need to be linked through a relationship
- `Sales[CustomerKey] = Customer[CustomerKey]`

If YearlyIncome is a customer related information, then you need a separate Customer table to store it.

Customer is a business entity

- Being a business entity, it deserves a table by itself



Business Entities

- Each business has different entities
 - Products, Customers, Resellers
 - Patients, Medications, Doctors
 - Claims, Customers
 - Teams, Workers, Buildings, Projects
 - Software, Features, Bugs, Customers
- Each business entity has unique characteristics

1 Entity = 1 Table

The difference between facts and dimensions is the core of star schemas and BI modeling

Facts and dimensions



Separation between facts and dimensions

- Different entities need different ways of handling
- Fact: *something that happened*
 - The sale of a product to a customer
 - A cash withdrawal on an ATM machine
 - The signature of an order
 - The prescription of a medical treatment
- Dimension: *something that describes a fact*
 - Attribute of a fact
 - The name of the customer, or of the patient
 - The date when the fact happened
 - The currency of the cash withdrawal

Some important notes about facts

- The fact table is typically the largest one
 - You can have billions of sales
 - But it is unlikely you have billions of customers
- Facts are associated with numbers
 - Amount withdrawn, quantity of the product sold
- Not all numbers belong to facts
 - Price of a product: belongs to the dimension
 - Quantity of products: belongs to the fact table
- Measures are typically aggregation of numbers from the fact table
 - `SUM (Sales[Quantity])`

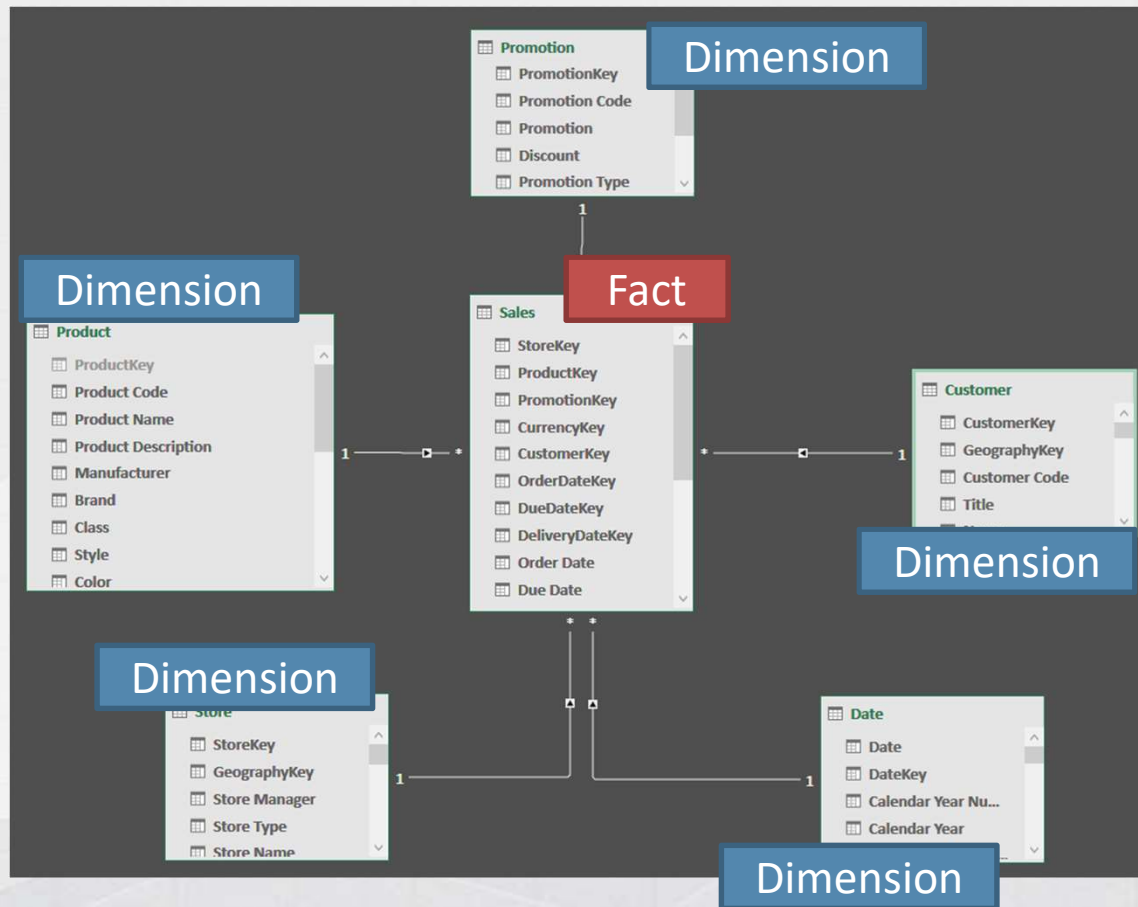
Some important notes about dimensions

- Dimensions are typically small tables
 - 1 million rows, for a dimension, is already a large number
- Dimensions contain a lot of strings
 - Customer name, address, gender, education
 - Product color, size, brand, manufacturer
- You typically slice by dimensions
- When you compute something, you count them
 - Number of customers
 - Number of products
- Dimensions are not related each other

What makes a dimension?

- One business entity = one table
- Attributes of an entity in the same table
- Customer is a business entity
 - Attributes: city, country, region, education, gender, age
- Usually Country is not an entity
 - It is an attribute of other dimensions
 - Country of customer, country of store
- Exception: demographic data
 - Measure: population (fact table)
 - Dimension: country (which is an entity in this model)

Introducing star schemas



Why choosing a different model?

- If the model is not the right one
 - DAX code tend to be very complex
 - Formulas are hard to think at
 - Complexity turns into performance issues
- With the correct model
 - DAX code is simple, as it should be
 - Performance are great
- Building the right model requires experience

Is your model a different one?

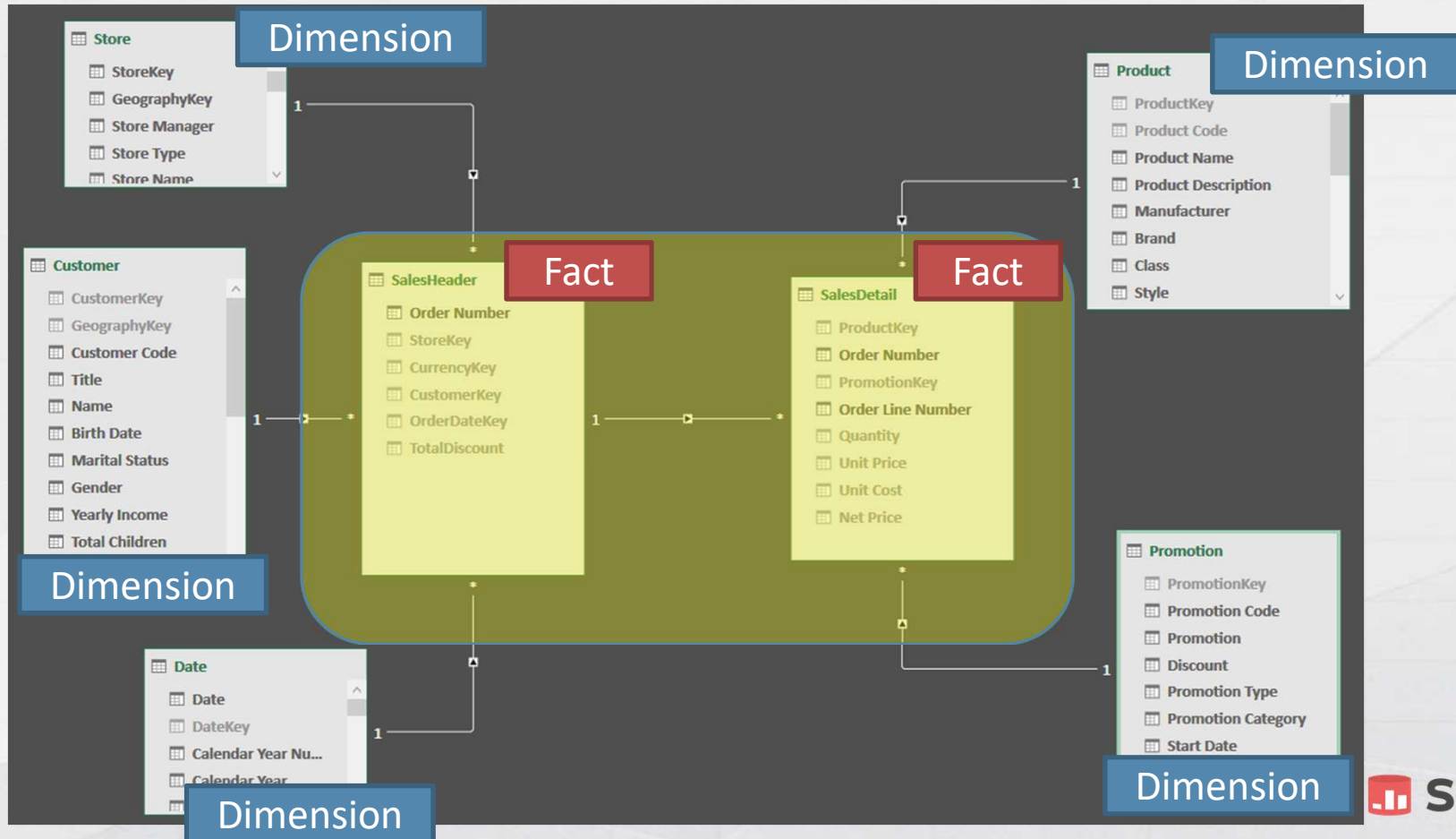
- At the beginning, you always feel your model is different than the standard ones
- 99.9% of the times, this is not the case
- Do not deviate from standard modeling, unless you really know what you are doing
- Business Intelligence was born in **1958**
- In **60 years**, we analyzed nearly any existing model
- And we found star schemas to be the best option

Let us see a first deviation from star schemas

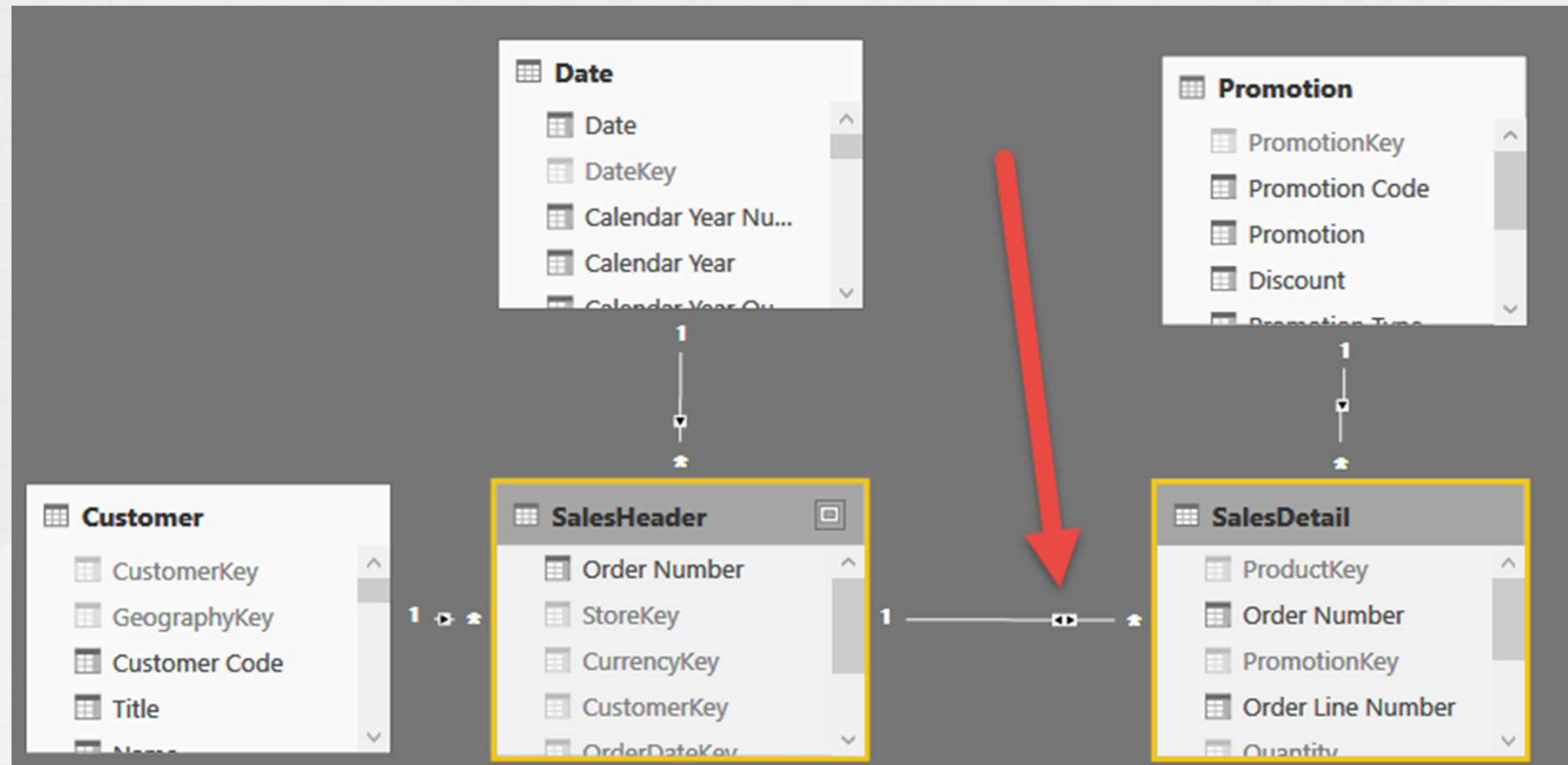
Header / detail tables



Sales headers and sales details



Bidirectional filtering is not the way to go



Numbers are still incorrect



- What is the column total showing?
- It is using the many-to-many pattern, very hard to read

Brand ▲	CY 2007	CY 2008	CY 2009	Total
A. Datum	\$23,396.50	\$6,625.26	\$6,818.09	\$36,839.84
Adventure Works	\$31,014.00	\$11,358.74	\$10,899.31	\$53,272.05
Contoso	\$51,975.46	\$24,946.38	\$28,580.97	\$105,502.81
Fabrikam	\$14,426.09	\$22,265.30	\$17,478.13	\$54,169.52
Litware	\$13,876.12	\$17,539.56	\$14,796.98	\$46,212.66
Northwind Traders	\$21,013.75	\$3,754.67	\$5,361.97	\$30,130.39
Proseware	\$12,812.25	\$9,740.69	\$15,318.93	\$37,871.87
Southridge Video	\$23,171.83	\$4,380.71	\$4,788.64	\$32,341.19
Tailspin Toys	\$1,029.51	\$707.81	\$1,775.78	\$3,513.10
The Phone Company	\$4,027.90	\$7,067.17	\$6,330.28	\$17,425.35
Wide World Importers	\$7,125.39	\$19,366.80	\$14,494.74	\$40,986.93
Total	\$158,506.66	\$120,458.29	\$116,667.34	\$395,632.29

SUM is 458,265.70

Computing the discount as a percentage

At the header level, the discount can be computed as a percentage, instead of a fixed value.

This way, we allocate the discount proportionally.

Calculated columns work fine for this.

`SalesHeader[DiscountPct] =`

```
DIVIDE (
    SalesHeader[TotalDiscount],
    SUMX (
        RELATEDTABLE ( SalesDetail ),
        SalesDetail[Unit Price] * SalesDetail[Quantity]
    )
)
```

Order Number	StoreKey	CurrencyKey	CustomerKey	OrderDateKey	TotalDiscount	DiscountPct
20080604724008	307	1	13009	20080604	€ 0.54	10.00%
200805105CS561	307	1	19098	20080510	€ 13.99	10.00%
20070605820430	307	1	9431	20070605	€ 181.80	10.00%
20070510215734	307	1	4735	20070510	€ 32.90	10.00%
200801156CS531	307	1	19092	20080115	€ 311.99	15.00%
200704013CS473	307	1	19082	20070401	€ 698.60	20.00%
20071115726159	307	1	15160	20071115	€ 1.33	15.00%
200905028CS712	307	1	19122	20090502	€ 97.80	10.00%
20070422714011	307	1	3012	20070422	€ 5.59	20.00%
200902218CS699	307	1	19116	20090221	€ 335.98	20.00%
20070213824162	307	1	13163	20070213	€ 307.36	20.00%
200902076CS697	307	1	19115	20090207	€ 12.20	20.00%
20071227722905	307	1	11906	20071227	€ 13.91	15.00%
20080414822856	307	1	11857	20080414	€ 363.78	20.00%

Computing over multiple star schemas

Multiple fact tables



Denormalized fact tables



Purchases	Sales
Quantity	Quantity
Unit cost	Unit Price
ProductName	ProductName
ColorName	ColorName
Manufacturer	Manufacturer
Date	Date
BrandName	BrandName
ProductSubcategoryName	ProductSubcategoryName
ProductCategoryName	ProductCategoryName
Purchase Amount	Sales Amount

BrandName	Purchase Amount	Sales Amount
A. Datum	2,533,963.42	30,202,685.54
Adventure Works	6,048,167.59	30,202,685.54
Contoso	12,314,395.68	30,202,685.54
Fabrikam	10,003,071.13	30,202,685.54
Litware	6,377,548.93	30,202,685.54
Northwind Traders	1,713,836.80	30,202,685.54
Proseware	5,305,305.29	30,202,685.54
Southridge Video	2,199,989.35	30,202,685.54
Tailspin Toys	646,571.47	30,202,685.54
The Phone Company	3,045,608.33	30,202,685.54
Wide World Importers	4,151,139.81	30,202,685.54
Total	54,339,597.80	30,202,685.54

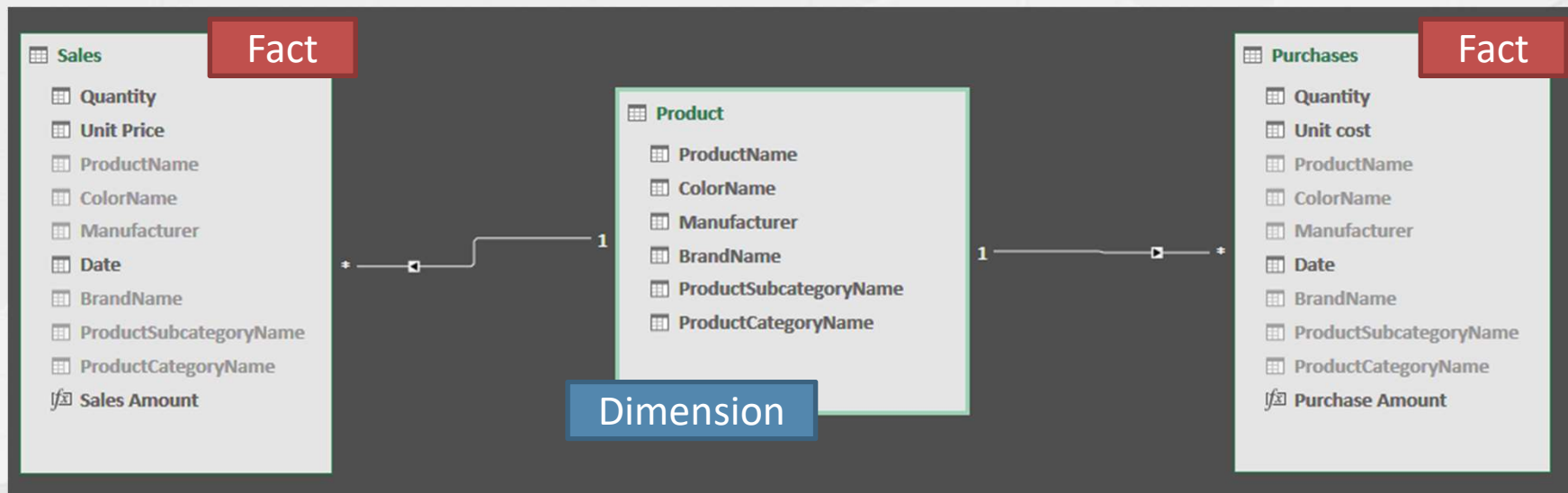
Using multiple fact tables

- Very common scenario
 - Sales and purchases
 - Orders and shipments
 - Sales and weather information
- What we cover in this section
 - Build the correct set of dimensions
 - Use one fact table to filter the other one(s)

Building a star schema



- A proper star schema is nearly always the best choice
- But how do we build the Product table?



Options to build the new dimension

- Use an SQL view, if feasible
- Use M code in Power Query
 - Available in Excel and Power BI
- Use DAX code and build a calculated table
 - Available in Power BI and SSAS 2016
- You need a key for the new dimension
 - Easy in SQL
 - Harder in M or DAX, if primary key not already available

Example of DAX code for the dimension

Here you can see an example of the new dimension built with DAX.
In this case, we will use the product name for the key, because nothing better is available.

Products =

```
DISTINCT (
    UNION (
        ALL (
            Sales[ProductName],
            Sales[ColorName],
            Sales[Manufacturer],
            Sales[BrandName],
            Sales[ProductCategoryName],
            Sales[ProductSubcategoryName]
        ),
        ALL (
            Purchases[ProductName],
            Purchases[ColorName],
            Purchases[Manufacturer],
            Purchases[BrandName],
            Purchases[ProductCategoryName],
            Purchases[ProductSubcategoryName]
        )
    )
)
```

ProductName	ColorName	Manufacturer	BrandName
Contoso Bright Light battery E20 blue	Blue	Contoso, Ltd	Contoso
Contoso Reserve Pen -Tablet Pen E200 Blue	Blue	Contoso, Ltd	Contoso
Contoso Multimedia Speakers M25 Blue	Blue	Contoso, Ltd	Contoso
Contoso USB Wave Multi-media Keyboard E280 Blue	Blue	Contoso, Ltd	Contoso
Contoso Wireless Notebook Optical Mouse M35 Blue	Blue	Contoso, Ltd	Contoso
Contoso USB Optical Mouse E200 Blue	Blue	Contoso, Ltd	Contoso
Contoso Smart Battery M901 Blue	Blue	Contoso, Ltd	Contoso
Contoso Digital Camera Accessory kit M200 Blue	Blue	Contoso, Ltd	Contoso
Contoso Wireless Laser Mouse M55 Silver	Silver	Contoso, Ltd	Contoso
Contoso Optical Wheel OEM PS/2 Mouse E60 Silver	Silver	Contoso, Ltd	Contoso
Contoso Bluetooth Notebook Mouse E70 Silver	Silver	Contoso, Ltd	Contoso
Contoso Wireless Laser Mouse E50 Silver	Silver	Contoso, Ltd	Contoso
Contoso Wireless Notebook Optical Mouse M35 Silver	Silver	Contoso, Ltd	Contoso
Contoso Laptop Keyboard X105 Silver	Silver	Contoso, Ltd	Contoso
Contoso Bluetooth Notebook Mouse X305 Silver	Silver	Contoso, Ltd	Contoso
Contoso Multimedia Speakers M25 Brown	Brown	Contoso, Ltd	Contoso
Contoso Bright Light battery E20 Pink	Pink	Contoso, Ltd	Contoso

Conclusions

- Data modeling is a required skill for Power BI
- Besides, it is incredibly fun!
- A good data model leads to simple DAX
- Simple, fast, easy to maintain
- Most of the times, the source data model is not your best choice
- Don't be a data consumer, become a data modeler

Thank you!



Check our articles, whitepapers and courses on
www.sqlbi.com

Slides and demo



<https://sql.bi/402792>