

Database
Relationship
Data Modeling
Entity
Bridge Table
Cardinality

Advanced Data Modeling with Power BI

#PowerBler



Marc Lelijveld

Data & Analytics consultant
Macaw Netherlands



Marc.Lelijveld@outlook.com



[@MarcLelijveld](https://twitter.com/MarcLelijveld)



linkedin.com/in/MarcLelijveld



Data-Marc.com

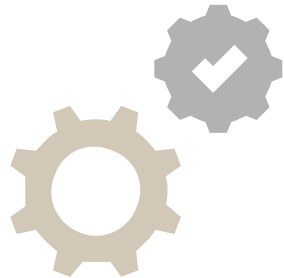


Set the stage

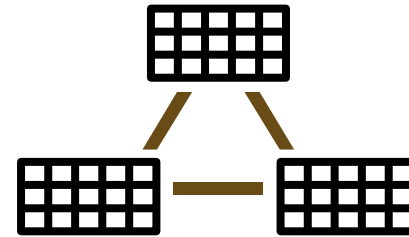
Gather



Clean



Model

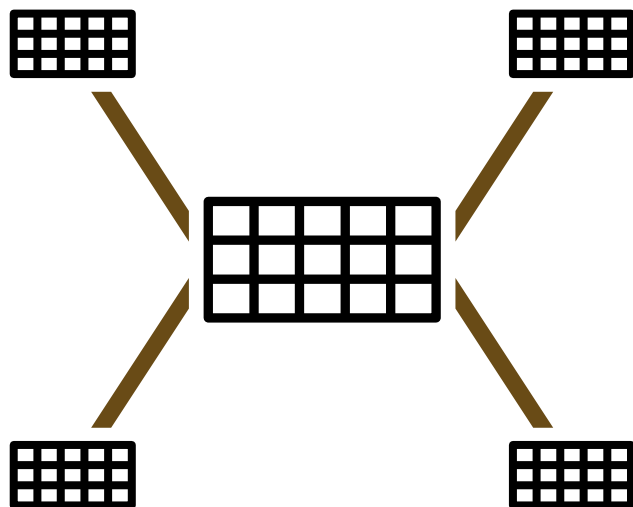


Visualize

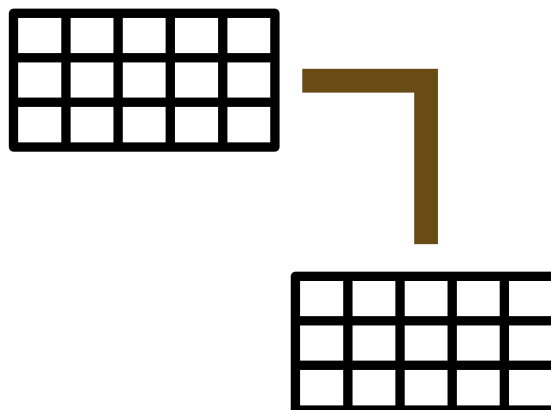


Your basic understanding

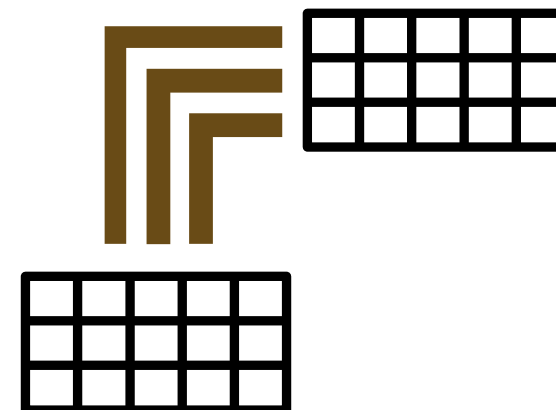
Star Schema



Relationship types



Role Playing Dimensions



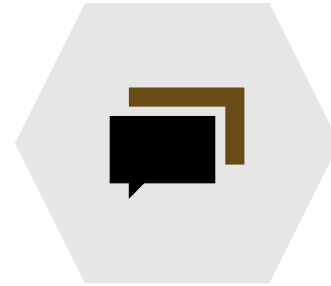
Learning objectives



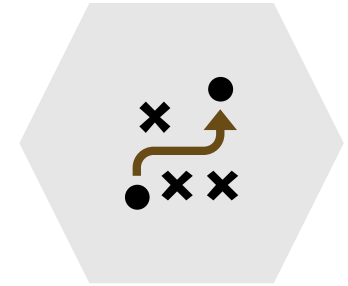
Explain data modeling
best practices in
general



Leverage one and bi-
directional relationships



Successfully implement
aggregations



Work with real live
scenarios containing
multiple fact tables

Today's agenda

- Relationships revisited
- Relationship direction
- Working with multiple fact tables
- Implementing Aggregations
- Wrap-up





Relationships revisited

Relationship types

1 to 1

Every **individual record** in dataset A is mapped to one **individual record** in dataset B

Dataset A
Customer information

Customer ID	Country
AW000111024	United States
AW00019377	Germany

Dataset B
Customer information

Customer ID	Date of Birth
AW000111024	9 april 1990
AW00019377	9 april 1983

1 to many

One record of dataset A is mapped to **multiple records** in dataset B

Dataset A
Customer information

Customer ID	Country
AW000111024	United States
AW00019377	Germany

Dataset B
Sales Information

Customer ID	Product	Order Quantity
AW00011024	CA-1098	2
AW00019377	BC-M005	1
AW00019377	CA-1098	1
AW00019377	FE-6654	1
AW00019377	HL-U509-B	1
AW00019377	TI-M602	1
AW00019377	TT-M928	1
AW00019377	WB-H098	1

Many to many

Multiple records of dataset A are mapped to **multiple records** in dataset B

Dataset A
Customer information

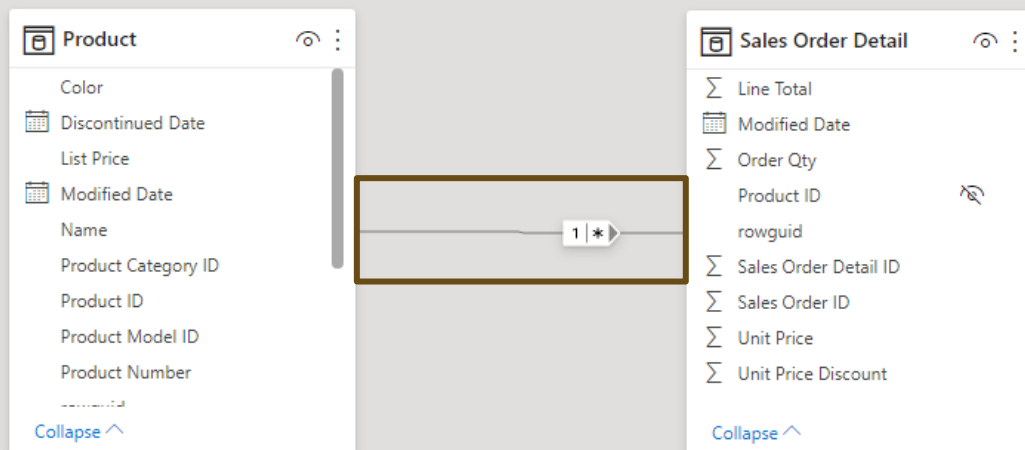
Customer ID	Store ID	Store Type
AW00011024	ON-1	Online
AW00011024	ST-1	Regional City Store
AW00011024	ST-2	Regional City Store
AW00019377	ON-1	Online
AW00019377	ST-1	Regional City Store
AW00019377	ST-2	Regional City Store

Dataset B
Sales Information

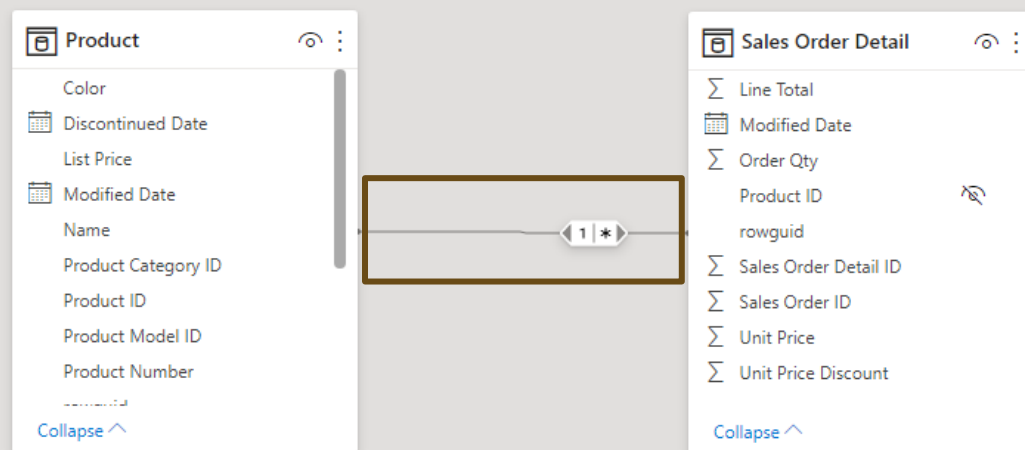
Customer ID	Product	Order Quantity
AW00011024	CA-1098	2
AW00011024	TT-M928	5
AW00019377	BC-M005	1
AW00019377	CA-1098	1
AW00019377	FE-6654	1
AW00019377	HL-U509-B	1
AW00019377	TI-M602	1
AW00019377	TT-M928	1
AW00019377	WB-H098	1

Relationship direction

Singular



Bi-directional



Relationship direction

Bi-directional relationships can result in surprising results, especially when working with multiple fact tables

- ➔ It can have performance impacts and 'overfilter'
- ➔ Can lead to ambiguity

Try to avoid bi-directional relationships

You can influence the direction of a relationship for the context of a calculation by CROSSFILTER in DAX:

- ➔ `CROSSFILTER(column1, column2, direction)`



Relationships direction

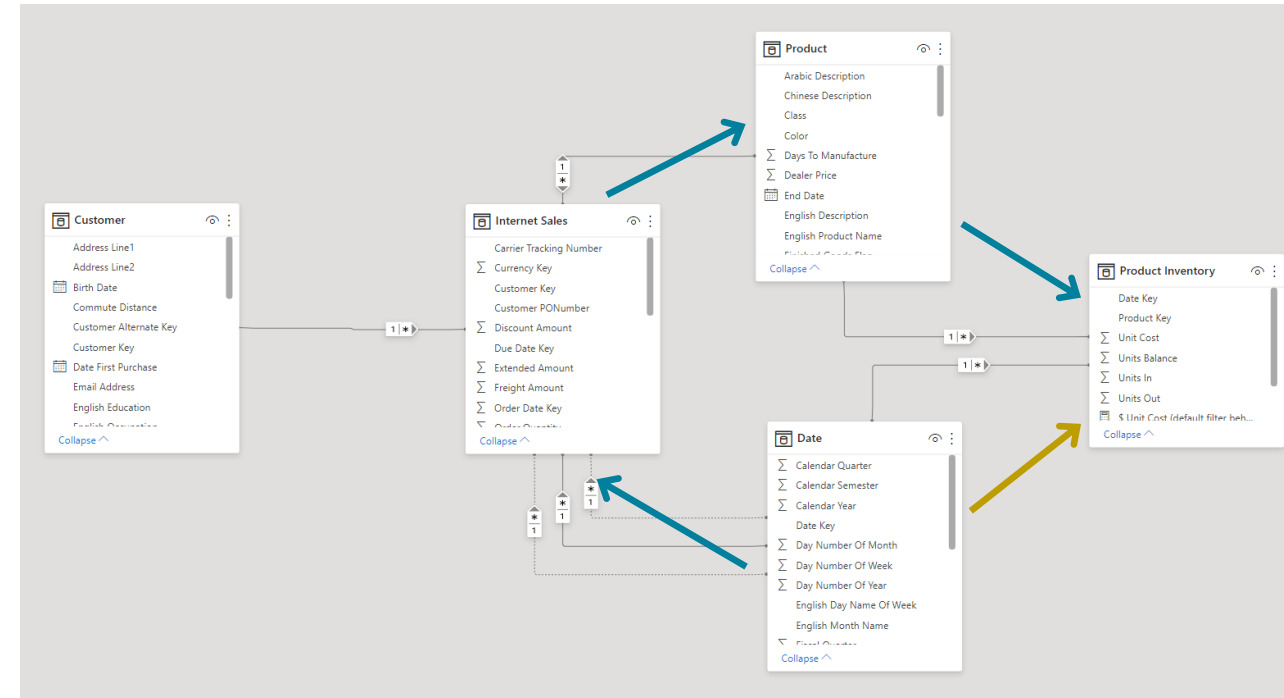
Ambiguous data models

Two filter paths to the same table

→ Leads to unexpected results

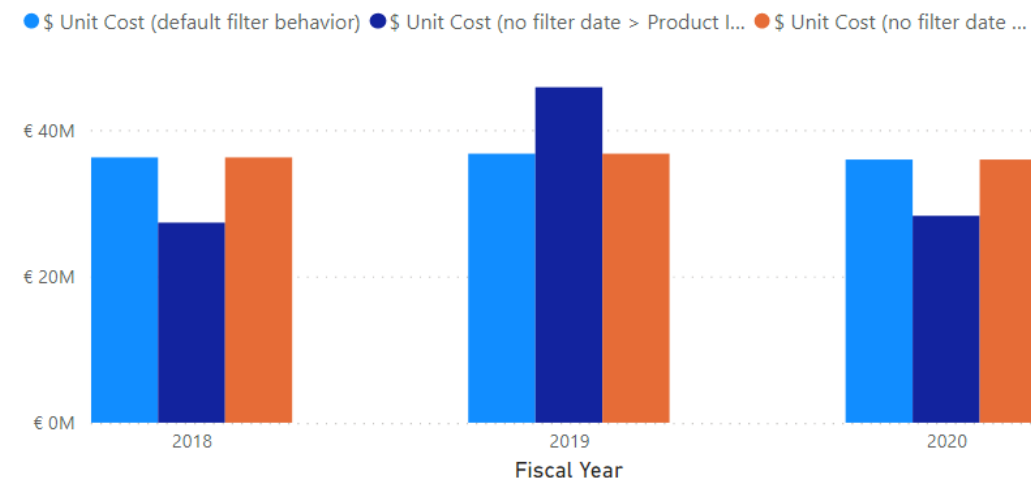
Can happen with bi-directional relationships

→ Avoid bi-directional relationships as much as possible



Ambiguous data models

- Can lead to unpredictable results
- Each of these measures calculates the same, but removes one of the relationships
- By using CROSSFILTER you can change relationships in a measure context

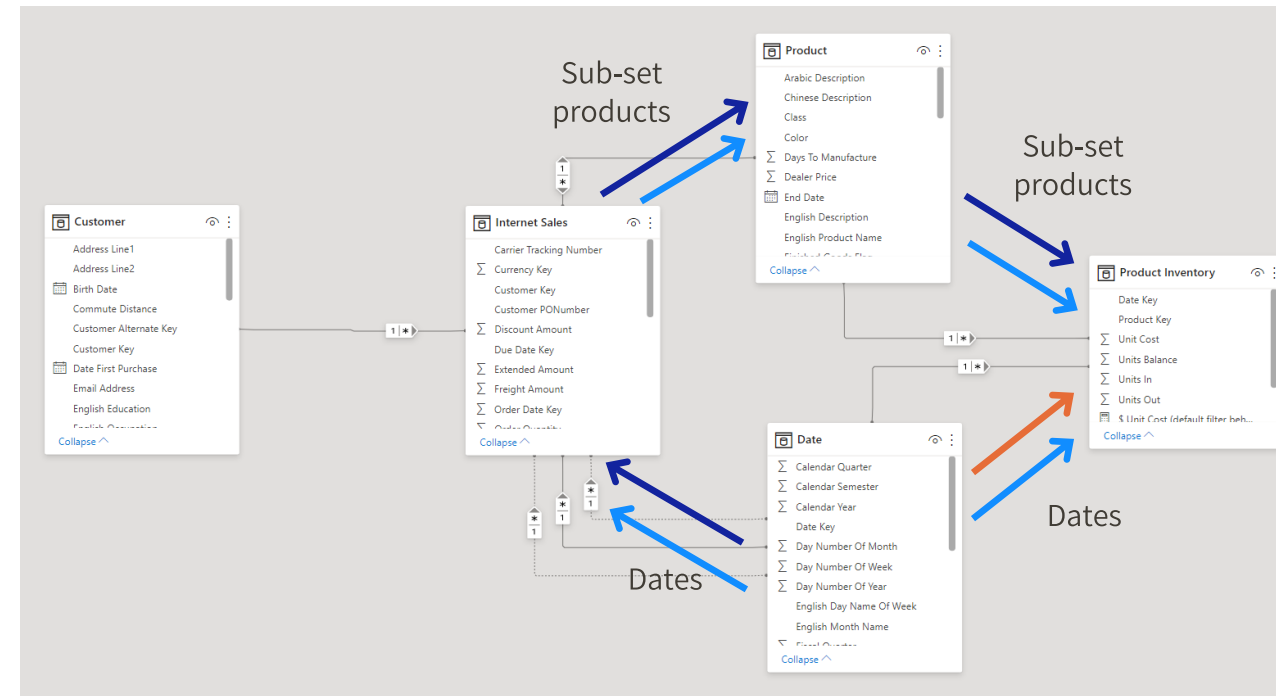
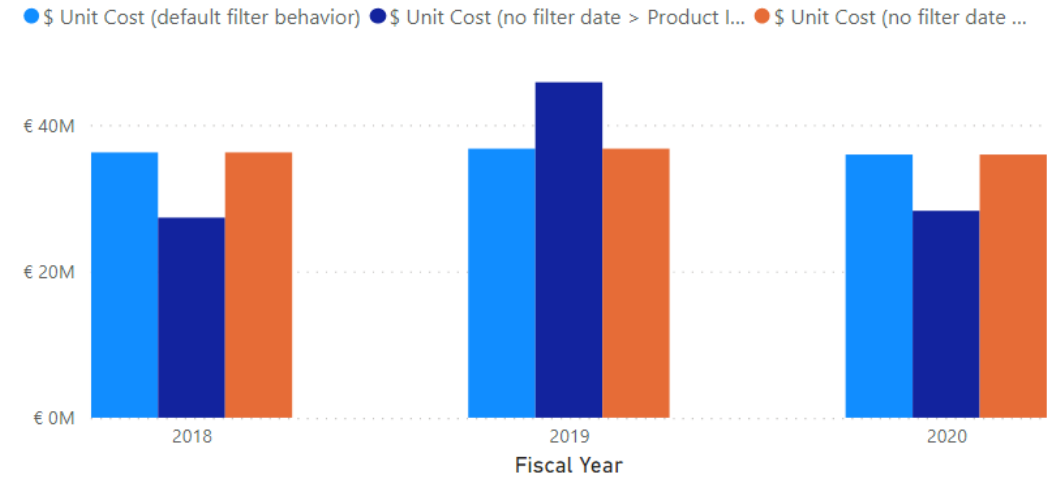


Fiscal Year	\$ Unit Cost (default filter behavior)	\$ Unit Cost (no filter date > Product Inventory)	\$ Unit Cost (no filter date > Internet Sales)
2018	€ 36.244.707,62	€ 27.337.603,34	€ 36.244.707,62
2019	€ 36.758.094,11	€ 45.834.231,75	€ 36.758.094,11
2020	€ 35.954.533,19	€ 28.267.611,68	€ 35.954.533,19
Total	€ 108.957.334,92	€ 108.957.334,92	€ 108.957.334,92

Ambiguous data models

Unexpected filter behavior

- Result will be filtered by a subset of products as part of the Internet Sales (dark blue filter path)
- Result will be filtered by a subset of dates (orange filter path)



Demo – relationships direction

Demo time!

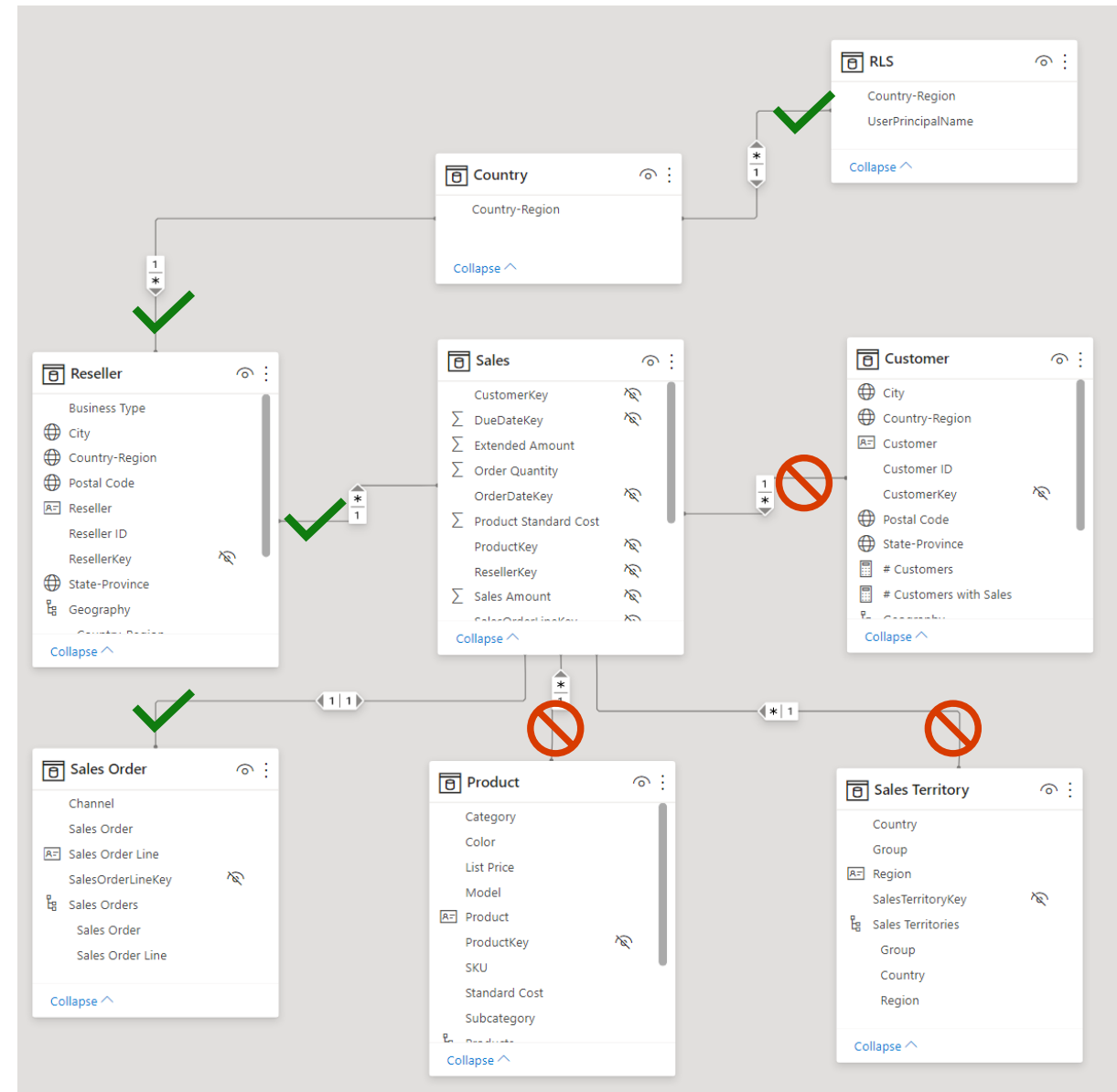
Because life is boring without risks



Relationships & Row level security

Does not filter dimensional tables

- Except over 1:1 bi-directional relationships
- Possible when adding a RLS filter to each table individually
- Requires a specific setup regarding relationships



Relationships & Row level security

Uses single directional filters by default

- you can set 'Apply the Filter Direction when using Row Level security' on a bi-directional relationship to force RLS to use bi-directional filtering
- Useful for dynamic row level security

Edit relationship

Select tables and columns that are related.

RLS

Country-Region	UserPrincipalName
Australia	user 1
Canada	user 2
Germany	user 3

Country

Country-Region
[Not Applicable]
Australia
Canada

Cardinality

Many to one (*:1)

Cross filter direction

Both

☒ Make this relationship active

☐ Assume referential integrity

☒ Apply security filter in both directions

OK

Cancel

Demo – Row Level Security

Demo time!

Because life is boring without risks





Handle multiple fact tables

Multiple fact tables

Multiple fact tables are common in real-life scenarios. Common scenarios:

Fact tables that you can append into one because they share common dimensions

Facts on different levels of granularity, different topics or that do not share common dimensions

Not sure what scenario you have? Use a mapping table.

	Currency	Customer	Due date	Employee	Order date	Product	Promotion	Reseller	Sales territory	Ship date
Internet sales	X	X	X		X	X	X		X	X
Reseller sales	X		X	X	X	X	X	X	X	X

Multiple fact tables

	Currency	Customer	Due date	Employee	Order date	Product	Promotion	Reseller	Sales territory	Ship date
Internet sales	X	X	X		X	X	X		X	X
Reseller sales	X		X	X	X	X	X	X	X	X

If you decide to append these, you will get (blanks) when you use one of the unmatched columns



Implementing Aggregations

Aggregations

Benefits

- ➔ Report visualizations are faster
- ➔ Balanced architecture by combining Direct Query and Import storage modes

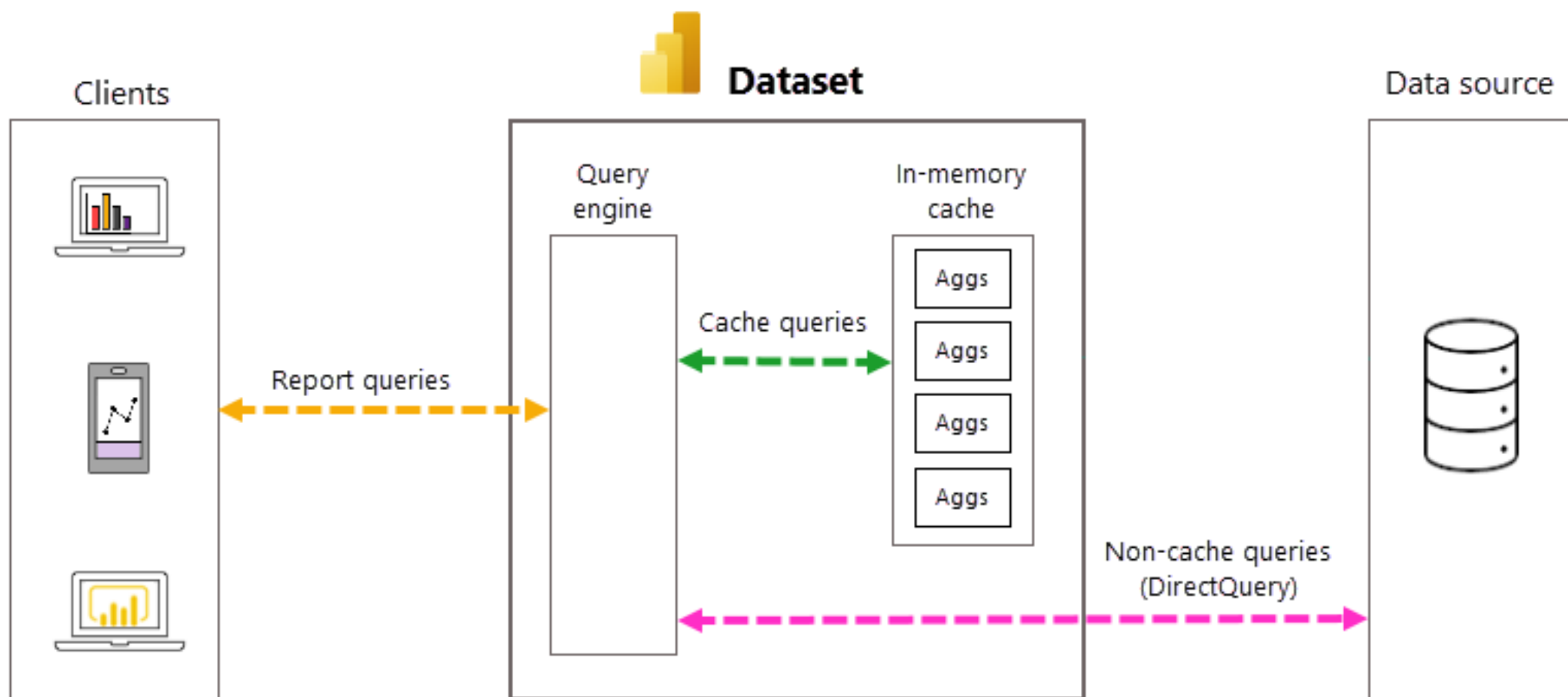
Store data at a higher level of granularity than the original table

The following aggregations are available:

count, groupby, max, min, sum, and count table rows

- ➔ Aggregated data is stored in-memory (imported), details are accessed through DirectQuery
- ➔ You can create the aggregated table in the Data Transformations (Power Query) or in your source (preferred)

Aggregations



Aggregations & storage modes

Aggregation, that hit based on relationships, require *regular* relationships.

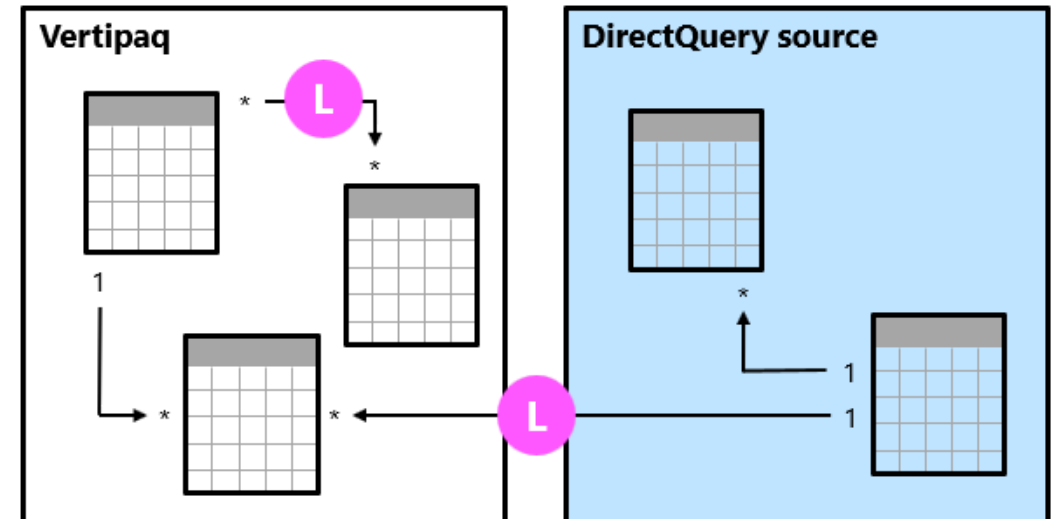
Regular relationships include the following storage mode combinations, where both tables are from a single source group:

Table on the many side	Table on the 1 side
Dual	Dual
Import	Import or Dual
DirectQuery	DirectQuery or Dual

Relationships & storage modes

A model relationship is *limited* when there's no guaranteed "one" side. It can be the case for three reasons:

- The relationship uses a Many-to-many cardinality type (even if one or both columns contain unique values)
- The storage mode combination is Import and DirectQuery
- The relationship is cross source group



Impact of limited relationships

Cross source group relationships have performance implications.

Limited optimization: joins are done on-demand for DirectQuery.

No blank rows: table joins are achieved by using INNER JOIN.

→ Blank rows are not added for referential integrity violations

Additional restrictions:

→ RELATED DAX function cannot be used to retrieve the 'one' side of the relationship

→ Enforcing RLS requires you to check the following checkbox 

Cardinality	Cross filter direction
Many to many (*:*)	Both
<input checked="" type="checkbox"/> Make this relationship active	<input type="checkbox"/> Apply security filter in both directions
<input type="checkbox"/> Assume referential integrity	

Demo – Aggregations

Demo time!

Because life is boring without risks



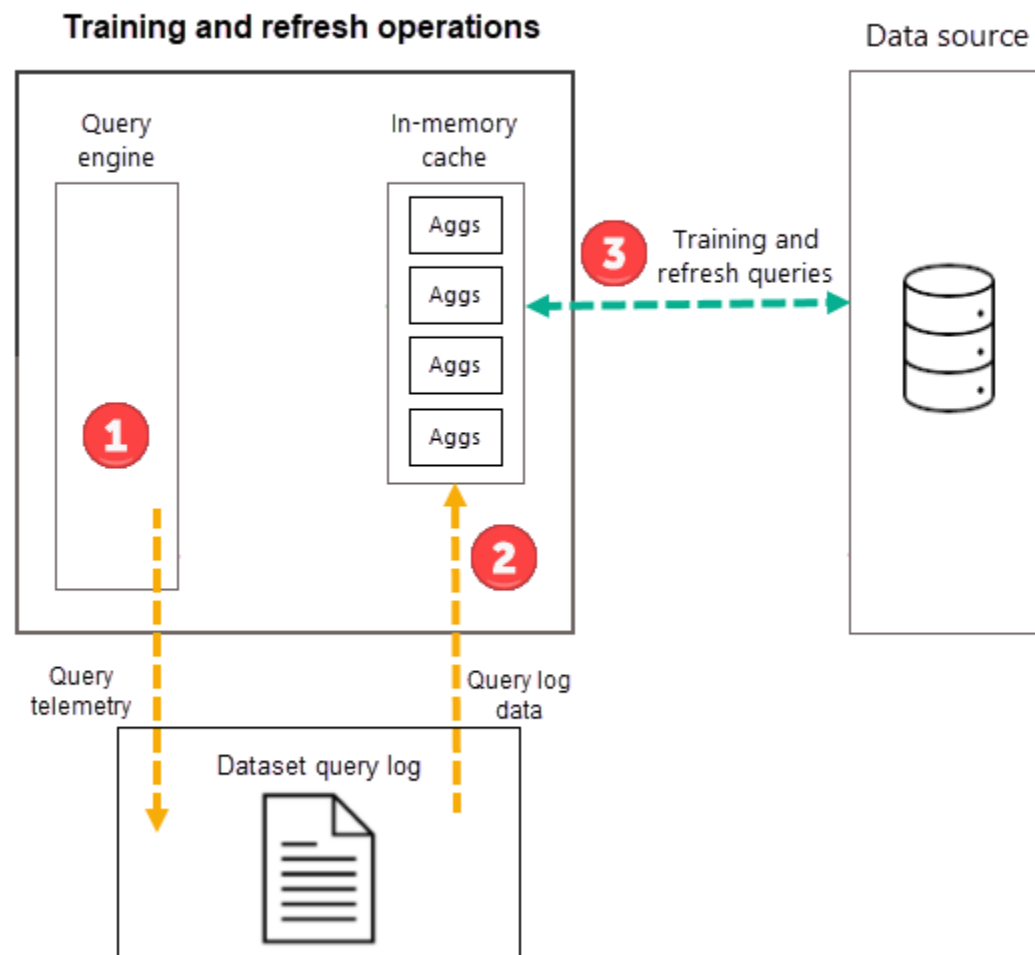
Automatic Aggregations – Preview!

Power BI Premium per User, Premium Capacity and Embedded datasets

Automatic aggregations based on Query logs (7 days)

Supported sources during preview:

- Azure SQL Database
- Azure Synapse Dedicated SQL pool
- Google BigQuery
- Snowflake



Wrap up

LET'S
RECAP...

- Use a **star schema** or snowflake data model to get the best out of Power BI.
- Be careful leveraging **bi-directional** relationships
- Avoid **ambiguous** data models
- Consider leveraging **aggregations** to help analysis of big data
- Be prepared to deal with **multiple fact tables**

Resources

- **Learning path: Model data in Power BI:**
<https://aka.ms/DataModelingLearningPath>
- **Learning path: Use DAX in Power BI Desktop:**
<https://aka.ms/DAXLearningPath>
- **Aggregations in Power BI Desktop:**
<https://aka.ms/PowerBIDesktopAggregations>
- **DAX function: CROSSFILTER:**
<https://aka.ms/DAXCrossfilter>
- **Model relationships in Power BI Desktop:**
<https://aka.ms/ModelRelationships>
- **Whitepaper: Bidirectional cross-filtering for Power BI Desktop:**
<https://aka.ms/BidirectionalCrossFilteringWhitepaper>
- **AdventureWorks 2020 demo dataset:**
<https://aka.ms/AW2020Dataset>

LET'S
RECAP...

Want to review all this content?

LET'S
RECAP...

Webinar Series: Mastering Data Modeling with Power BI

Episode 1 - Data Modeling 101: Increasing the Impact of Power BI

by Jeroen ter Heerdt and Marc Lelijveld

[Watch now](#)

Webinar Series: Mastering Data Modeling with Power BI Episode 2 - Learn Advanced Data Modeling with Power BI

by Jeroen ter Heerdt and Marc Lelijveld

[Watch now](#)

Webinar Series: Mastering Data Modeling with Power BI

Episode 3 - Data Modeling for Experts with Power BI

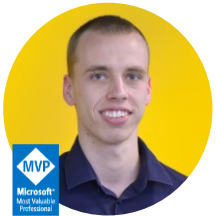
by Jeroen ter Heerdt and Marc Lelijveld

[Register to watch on September 8th, 2021 10AM – 11AM Pacific Time](#)

Find them all here: <https://docs.microsoft.com/en-us/power-bi/fundamentals/webinars>
or: aka.ms/DataModelingWebinarEpisode{1/2/3}



Thanks for attending!



Marc Lelijveld

Data & Analytics consultant
Macaw Netherlands

✉ Marc.Lelijveld@outlook.com

🐦 [@MarcLelijveld](https://twitter.com/MarcLelijveld)

in linkedin.com/in/MarcLelijveld

🌐 Data-Marc.com

