



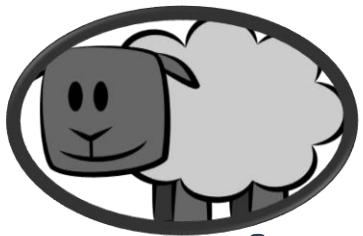
# **Unlock the Power of Power BI with Calculation Groups**

Jason Romans

# Jason Romans

Senior BI Engineer

Builder of Models



**The Dax Shepherd**

thedaxshepherd@gmail.com

<https://thedaxshepherd.net/>



*Live in Nashville, TN*



*Started as SQL Server DBA*

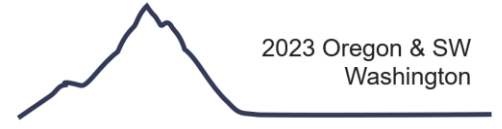


*Transitioned to the Microsoft BI Stack*



*Work on everything from SQL Server Integration Services, SQL Server Database, Analysis Services, and Power BI*

# Thank You Sponsors!



Our sponsors make this event possible. Please thank and support them.



# Agenda

---

1. Why Calculation Groups
2. How to Create
3. Calculation Items
4. Multiple Calculation Groups
5. Conclusion

# Agenda

---

## **1. Why Calculation Groups**

2. How to Create

3. Calculation Items

4. Multiple Calculation Groups

5. Conclusion

It starts with one measure...

Sales Amount

Now, need to create a version for different time periods

Sales Amount MTD – Month to Date

Sales Amount QTD – Quarter to Date

Sales Amount YTD – Year to Date

Sales Amount SPLY – Same Period Last Year

Can you do the  
same for this  
measure?

---

Total Cost

- Total Cost QTD
- Total Cost MTD
- Total Cost YTD
- Total Cost SPLY



And Now These...

- Margin
- Margin %
- Volume
- EBITDA



It starts to get a  
little tangled

---



# Difficult to Manage Variations

---

For each measure

--you may have at least 5  
variations

# Not easy to switch between on visual

---

If you have YTD measures

- want to see Same Period Last Year

**We need a better way**

---

# The Pattern is the Same

---



- Sales Amount
- Total Cost
- Margin

YTD

YTD

YTD

# It is the measure that changes



Sales Amount	YTD
Total Cost	YTD
Margin	YTD

# Placeholder for the Measure



Sales Amount	YTD
Total Cost	YTD
Margin	YTD

Measure\_Variable??




# Rest of the Code is the Same



Sales Amount	YTD
Total Cost	YTD
Margin	YTD

`CALCULATE (`  
`Measure_Variable,`  
`DATESYTD ('Date'[Date]))`





# Only Need to Write Pattern One Time

YTD =

```
CALCULATE (
    Measure_Variable,
    DATESYTD ( 'Date'[Date] )
)
```

- 
- This is the business use case  
Calculation Groups are designed to solve
  - Just need a way to handle the  
Measure\_Variable

# Introducing SELECTEDMEASURE()

---



YTD =

```
CALCULATE (
    SELECTEDMEASURE(),
    DATESYTD ( 'Date'[Date] )
)
```

# Measure Selected

---



YTD = [Sales Amount]

CALCULATE (

SELECTEDMEASURE(),

DATESYTD ( 'Date' [Date] )

)

# Replaces Measure in DAX Code

---



YTD =

```
CALCULATE (
    [Sales Amount],
    DATESYTD ( 'Date'[Date] )
)
```

- 
- We have a way to handle different measures
  - What are some of the benefits we expect to achieve

# Benefits of Calculation Groups

---

- Fewer measures to maintain
- Reuse already established Measures
- Use on multiple measures
- Control formatting

---

But there is more – you can use  
Calculation Groups in a slicer for the  
user to choose between



# What Does this Look Like

---

# Time Intelligence – 1 Measure

## Sales Amount

Year	Current	QTD	YTD	SPLY	Time Intelligence
2018	4,984,304.80	1,237,622.62	4,984,304.80	3,033,305.02	<div>Current</div> <div>MTD</div> <div>QTD</div> <div>YTD</div> <div>SPLY</div>
Jan	636,983.88	636,983.88	636,983.88		
Feb	788,062.88	1,425,046.76	1,425,046.76		
Mar	269,320.40	1,694,367.17	1,694,367.17		
Apr	27,644.10	27,644.10	1,722,011.26		
May	510,486.78	538,130.88	2,232,498.04	168,392.56	
Jun	413,325.23	951,456.11	2,645,823.27	263,600.69	
Jul	348,977.82	348,977.82	2,994,801.09	204,281.19	
Aug	392,499.67	741,477.49	3,387,300.76	312,793.50	
Sep	359,381.42	1,100,858.91	3,746,682.18	334,423.50	
Oct	394,691.65	394,691.65	4,141,373.83	402,067.05	
Nov	329,846.47	724,538.12	4,471,220.30	438,804.70	
Dec	513,084.50	1,237,622.62	4,984,304.80	908,941.83	
<b>Total</b>	<b>4,984,304.80</b>	<b>1,237,622.62</b>	<b>4,984,304.80</b>	<b>3,033,305.02</b>	

# Time Intelligence – 2 Measures

## Sales Amount & Total Cost

Time Intelligence YTD			SPLY	
Year	Sales Amount	Total Cost	Sales Amount	Total Cost
☐ 2018	4,984,304.80	2,193,394.99	3,033,305.02	1,335,411.56
Jan	636,983.88	278,700.85		
Feb	1,425,046.76	618,998.60		
Mar	1,694,367.17	737,421.60		
Apr	1,722,011.26	748,655.63		
May	2,232,498.04	969,453.25	168,392.56	69,007.25
Jun	2,645,823.27	1,154,001.07	263,600.69	116,482.16
Jul	2,994,801.09	1,304,716.63	204,281.19	89,144.35
Aug	3,387,300.76	1,477,938.90	312,793.50	143,531.77
Sep	3,746,682.18	1,639,576.74	334,423.50	147,223.18
Oct	4,141,373.83	1,813,779.76	402,067.05	182,879.12
Nov	4,471,220.30	1,962,431.50	438,804.70	197,499.12
Dec	4,984,304.80	2,193,394.99	908,941.83	389,644.61
Total	4,984,304.80	2,193,394.99	3,033,305.02	1,335,411.56

Time Intelligence

☐ Current

☐ MTD

☐ QTD

■ YTD

■ SPLY

Time Intelligence

☐ Current

☐ MTD

☐ QTD

☒ YTD

☒ SPLY

# 2 Measures with Time Intelligence and Statistics (MAX, MIN, AVG)

Both MAX Time Intelligence Year	No Time Intelligence		YTD	
	Current		YTD	
	Sales Amount	Total Cost	Sales Amount	Total Cost
☐ 2018	<b>788,062.88</b>	<b>340,297.75</b>	<b>788,062.88</b>	<b>340,297.75</b>
Jan	636,983.88	278,700.85	636,983.88	278,700.85
Feb	788,062.88	340,297.75	788,062.88	340,297.75
Mar	269,320.40	118,423.00	788,062.88	340,297.75
Apr	27,644.10	11,234.03	788,062.88	340,297.75
May	510,486.78	220,797.62	788,062.88	340,297.75
Jun	413,325.23	184,547.82	788,062.88	340,297.75
Jul	348,977.82	150,715.56	788,062.88	340,297.75
Aug	392,499.67	173,222.27	788,062.88	340,297.75
Sep	359,381.42	161,637.84	788,062.88	340,297.75
Oct	394,691.65	174,203.02	788,062.88	340,297.75
Nov	329,846.47	148,651.74	788,062.88	340,297.75
Dec	513,084.50	230,963.49	788,062.88	340,297.75
<b>Total</b>	<b>788,062.88</b>	<b>340,297.75</b>	<b>788,062.88</b>	<b>340,297.75</b>

# Are Calculation Groups New

---

- Since about 2019
- Needed to use Tabular Editor
- Power BI Desktop October 2023 Release (Preview)
  - Will increase the adoption rate

# Agenda

---

1. Why Calculation Groups
- 2. How to Create**
3. Calculation Items
4. Multiple Calculation Groups
5. Conclusion

# Two Ways

---

## Tabular Editor

- Up till the October 2023 release - only way

## Power BI Desktop (Preview)

- October 2023 release or newer

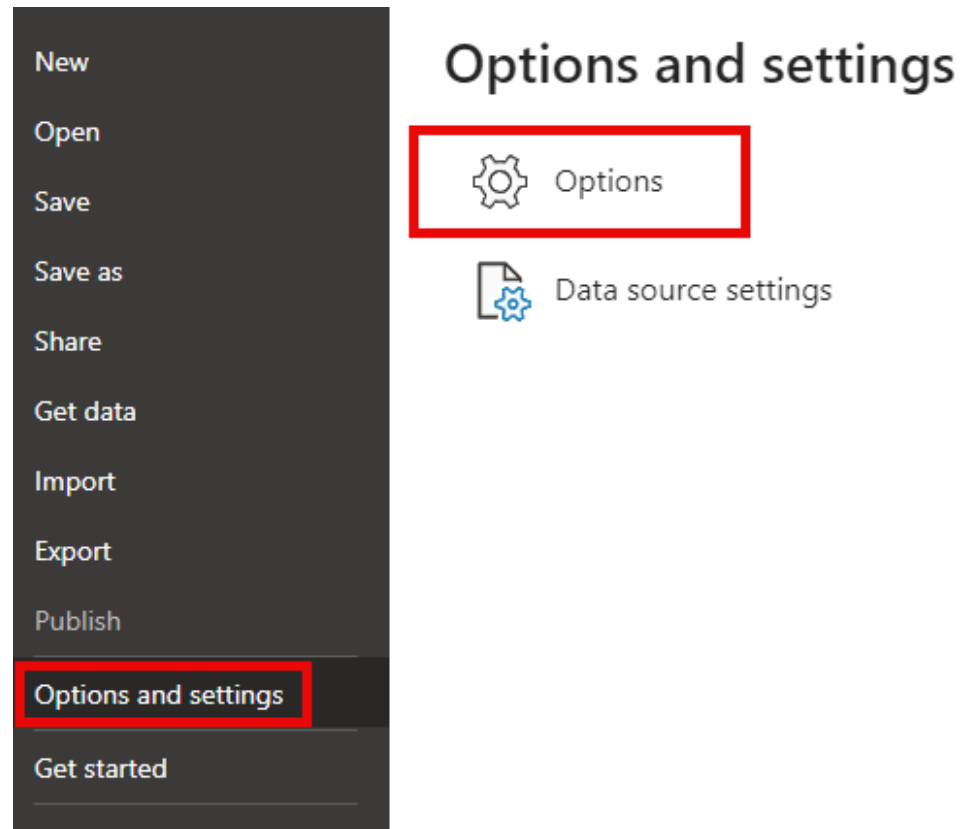
# Creating in Power BI Desktop

---

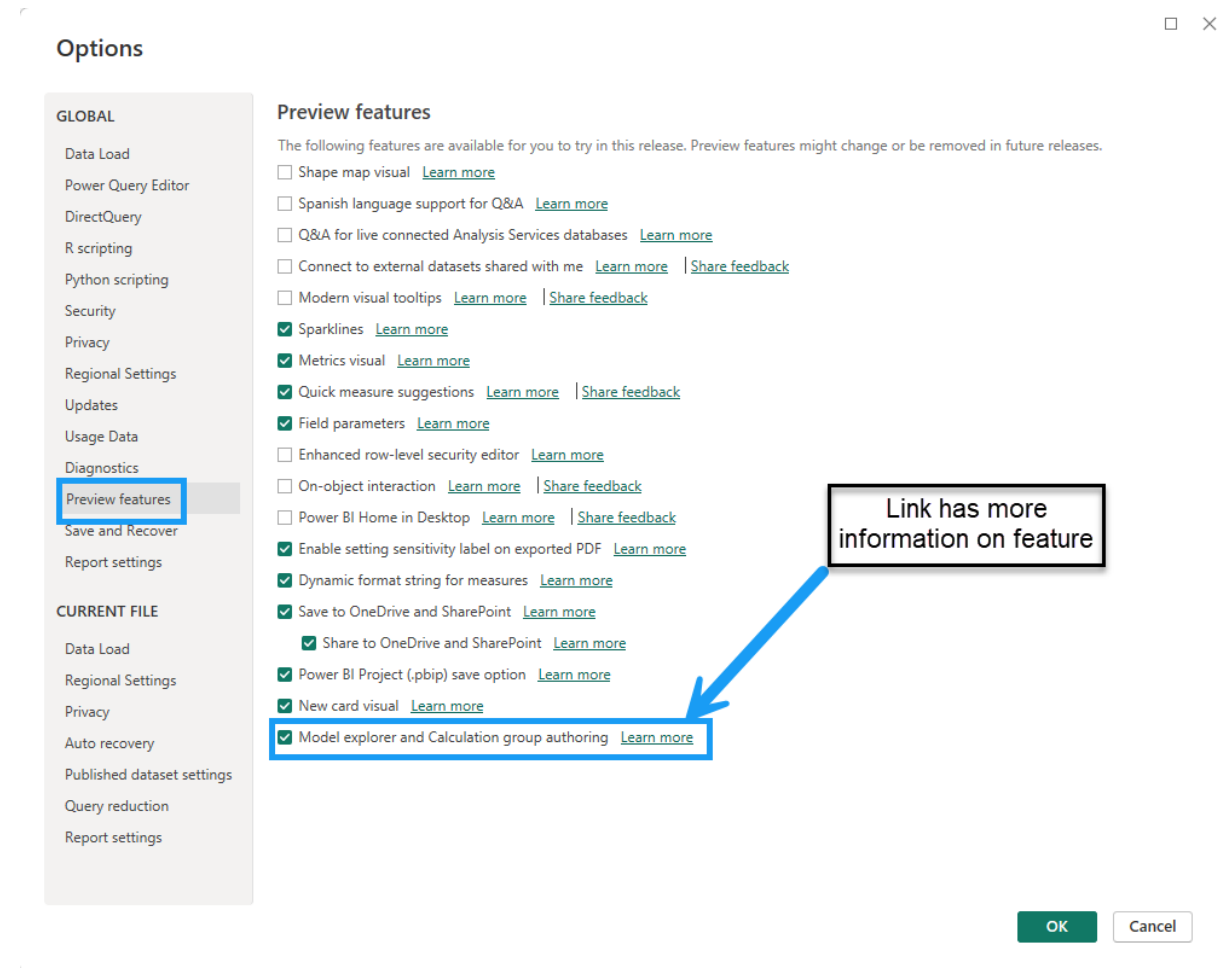


# Power BI Desktop Enable Preview - Options

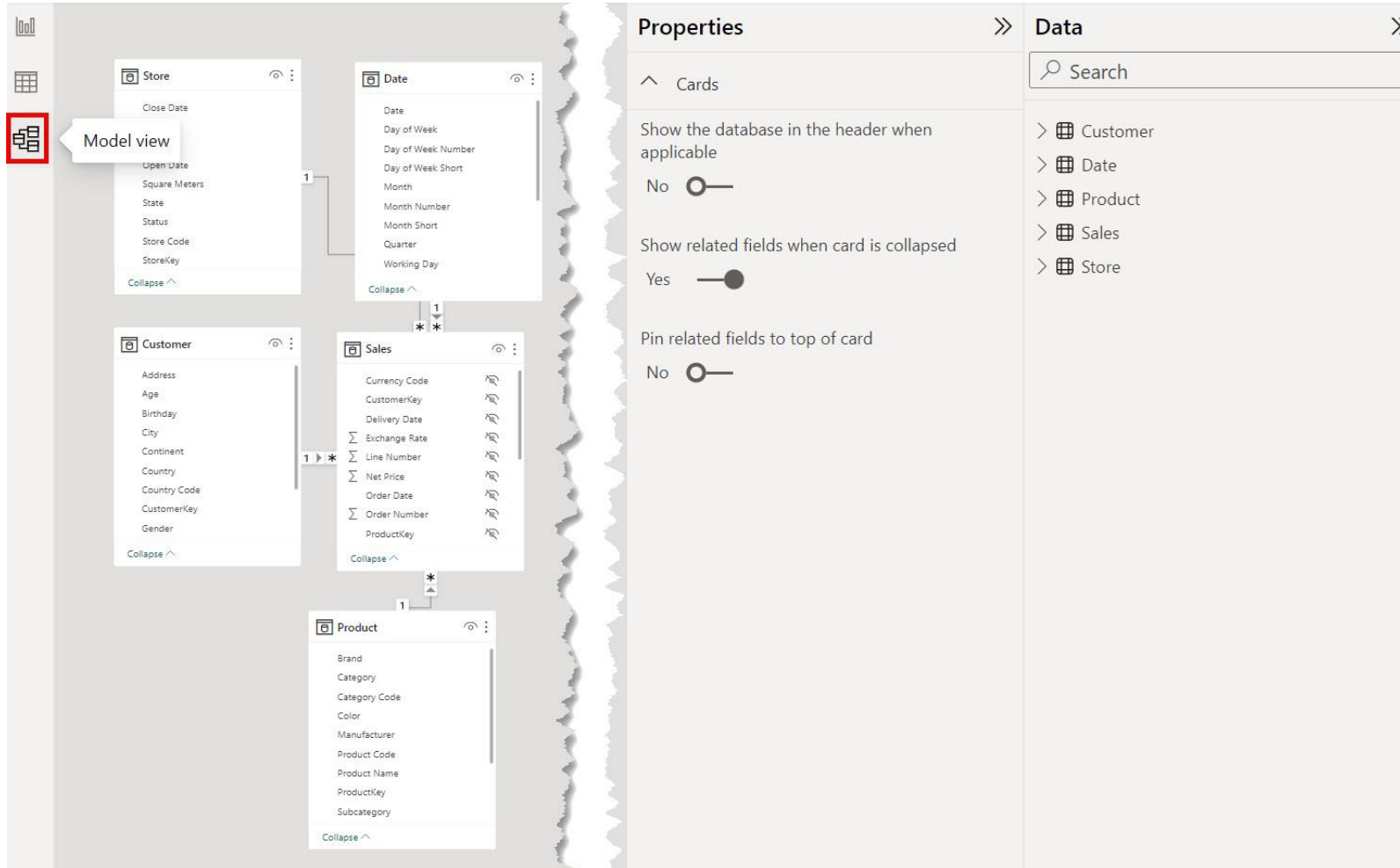
---



# Power BI Enable Preview



# Power BI – Model Explorer Not Enabled



# Power BI – Model View - Tables

The screenshot displays the Power BI Model View interface. On the left, a list of tables is shown: Store, Date, Customer, Sales, and Product. Each table is represented by a card with its fields listed below. The Store card includes fields like Close Date, Open Date, Square Meters, State, Status, Store Code, and StoreKey. The Date card includes Date, Day of Week, Day of Week Number, Day of Week Short, Month, Month Number, Month Short, Quarter, and Working Day. The Customer card includes Address, Age, Birthday, City, Continent, Country, Country Code, CustomerKey, and Gender. The Sales card includes Currency Code, CustomerKey, Delivery Date, Exchange Rate, Line Number, Net Price, Order Date, Order Number, and ProductKey. The Product card includes Brand, Category, Category Code, Color, Manufacturer, Product Code, Product Name, ProductKey, and Subcategory. Relationships are indicated by lines connecting the tables: Store to Date (1 to 1), Store to Sales (1 to \*), Date to Sales (\* to \*), Customer to Sales (1 to \*), and Product to Sales (\* to 1). A red box highlights the 'Model view' icon in the top left corner. On the right, the 'Properties' pane shows settings for 'Cards', including 'Show the database in the header when applicable' (set to No), 'Show related fields when card is collapsed' (set to Yes), and 'Pin related fields to top of card' (set to No). The 'Data' pane on the far right shows a list of tables: Sales, Customer, Date, Product, and Store, with a search bar above them.

**Model view**

**Properties**

**Cards**

Show the database in the header when applicable

No ☐

Show related fields when card is collapsed

Yes ☒

Pin related fields to top of card

No ☐

**Data**

**Tables** **Model**

Search

- > Sales
- > Customer
- > Date
- > Product
- > Store

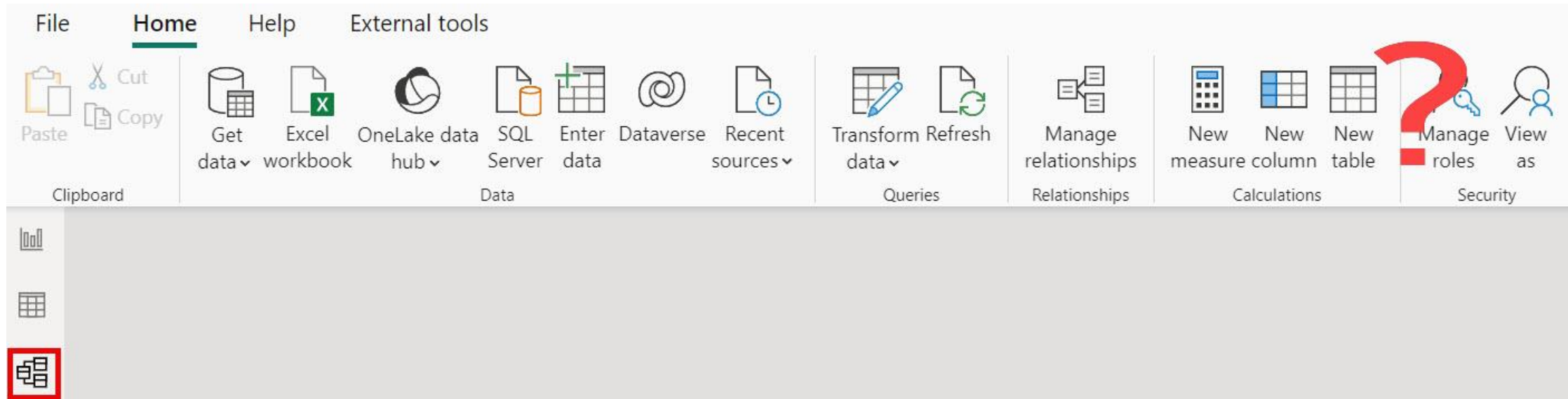
# Power BI – Model View – Model Explorer

The screenshot displays the Power BI interface in the Model View. On the left, a navigation pane shows icons for Reports, Tables, and the Model View (highlighted with a red box). The main area shows a diagram of the data model with four tables: Store, Date, Customer, and Sales. The Store table is connected to the Date table (1 to 1 relationship). The Date table is connected to the Sales table (1 to many relationship). The Customer table is connected to the Sales table (1 to many relationship). The Product table is also connected to the Sales table (1 to many relationship). Each table has a list of columns and a 'Collapse' button.

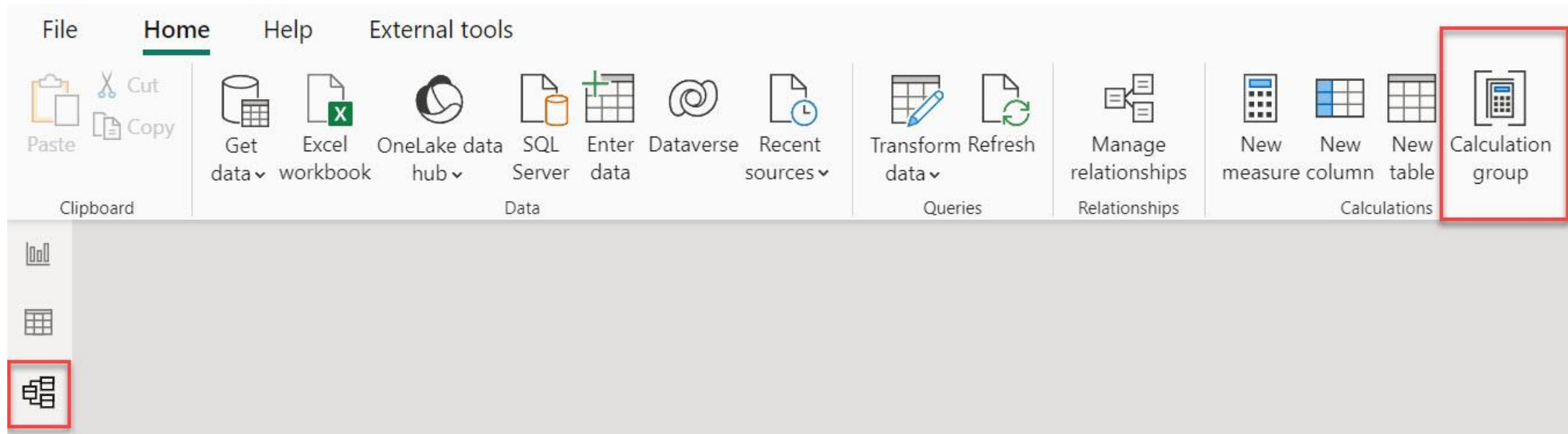
On the right, the Properties pane is visible, showing the 'General' tab. The Name is 'Semantic model' and the Description is 'Enter a description'. The Server is 'localhost:57762' and the Compatibility Level is '1567'. The Cultures section shows 'en-US'. The Discourage implicit measures toggle is set to 'No'.

Next to the Properties pane is the Data pane, which shows the 'Model' tab selected. It displays a search bar and a list of items under the 'Semantic model' category: Calculation groups (0), Cultures (1), Measures (5), Perspectives (0), Relationships (4), Roles (0), and Tables (5).

# Model Explorer Off – No Calculation Group

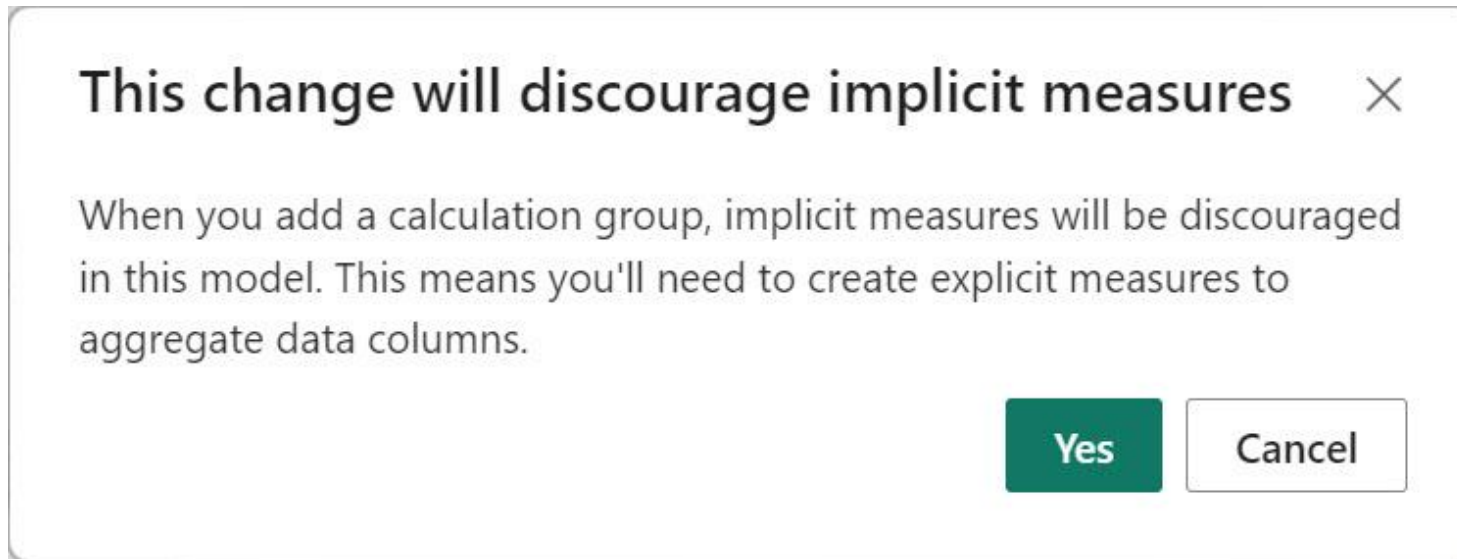


# Click Calculation Group



# Discourage Implicit Measures

---









A little bit to unpack here



# What is an Implicit Measure?

---

▼  Sales

- ☐  Margin
- ☐  Margin %
- ☐  $\Sigma$  Quantity
- ☐  Sales Amount
- ☐  Total Cost
- ☐  Total Quantity
- ☐  $\Sigma$  Unit Cost

# Implicit Measure Example

Measure with formatting

Year	Count of Quantity	Count of Quantity	Sum of Unit Cost	Total Quantity
2017	10	2781	\$414,250.11	8,599
May	10	127	\$17,740.371	406
Jun	10	254	\$36,632.162	790
Jul	10	215	\$29,543.376	659
Aug	10	314	\$46,432.916	1,001
Sep	10	330	\$47,392.8505	1,016
Oct	10	350	\$54,882.921	1,116
Nov	10	393	\$60,455.6185	1,179
Dec	10	798	\$121,169.895	2,432
2018	10	5756	\$714,666.855	17,922
Jan	10	616	\$94,177.396	1,904
Feb	10	708	\$106,474.482	2,314
Mar	10	259	\$36,448.617	818
Apr	9	50	\$4,657.09	138
May	10	590	\$66,370.44	1,896
Jun	10	525	\$61,193.05	1,628
Jul	10	422	\$47,152.8	1,107

Distinct Non-Distinct

Sum

Visualizations

Build visual

Rows

Year

Month Short

Columns

Add data fields here

Values

Count of Quantity

Count of Quantity

Sum of Unit Cost

Total Quantity

# Implicit Versus Explicit Measures

---

- Implicit – Drag the value column to visual and it automatically aggregates it
  - It is creating the calculation for you
- Explicit
  - Control formatting
  - Only expose measures to developers and users
  - Can change the logic of the measure – Like a SQL View

# “Discourage” Implicit Measures

---

This change will discourage implicit measures ×

When you add a calculation group, implicit measures will be discouraged in this model. This means you'll need to create explicit measures to aggregate data columns.

Yes

Cancel

# Discourage Implicit Measures

---

This change will ~~discourage~~ implicit measures ×

When you add a calculation group, implicit measures will be discouraged in this model. This means you'll need to create explicit measures to aggregate data columns.

Yes

Cancel

# Discourage Implicit Measures

---

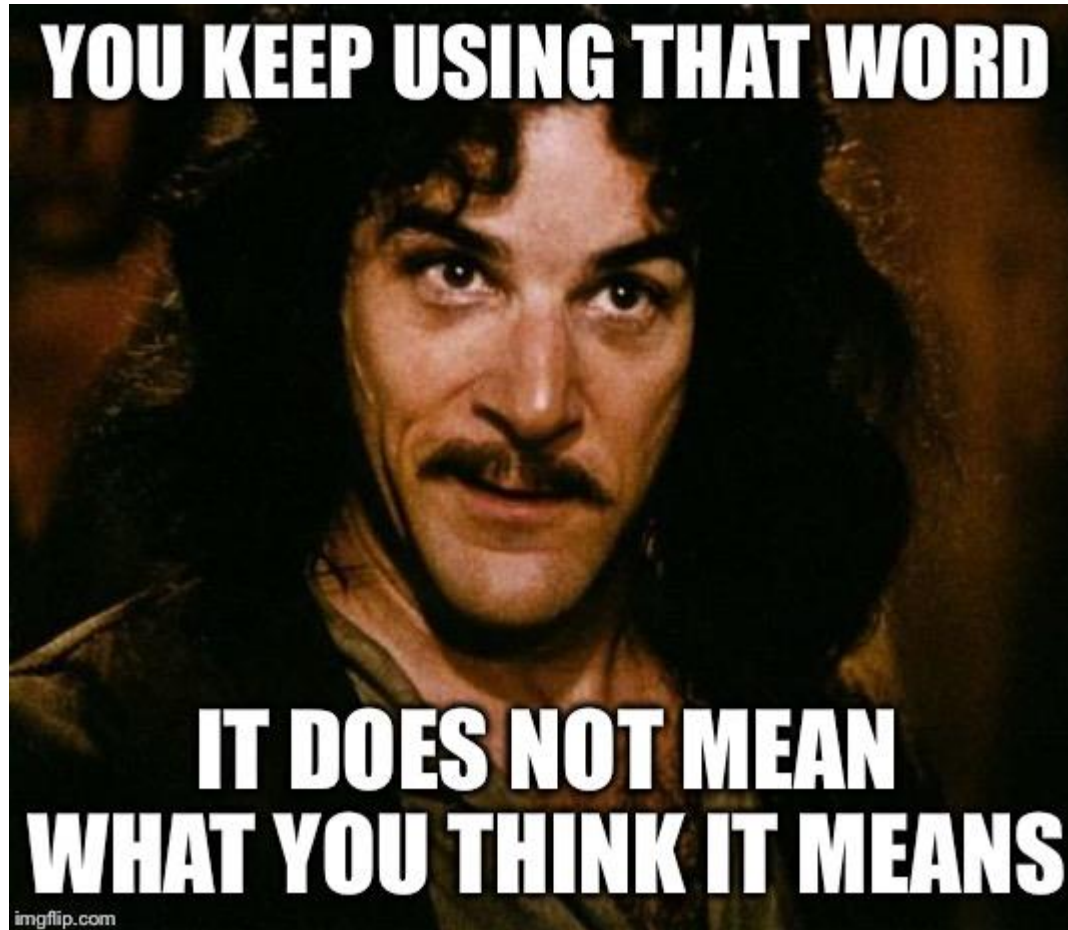
This change will **Prohibit** implicit measures ×

When you add a calculation group, implicit measures will be discouraged in this model. This means you'll need to create explicit measures to aggregate data columns.

Yes

Cancel

# Discourage



# Discourage Implicit Measures Behavior

Year					
\$0.95 \$1.425 \$1.99 \$2.94 \$3.35					
2018	8.24	7.70	5.13	29.40	229.41
Jan					
Feb					
Mar		7.70			
Apr					
May	2.85		5.13		42.21

Visualizations

Build visual

Filters

Rows

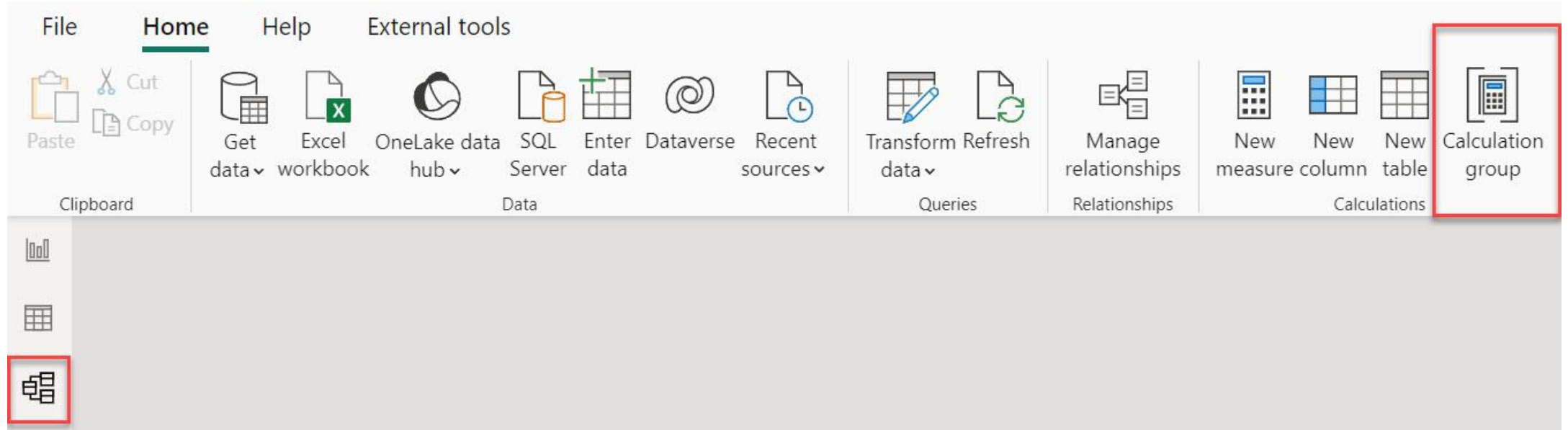
Columns

Values

Cannot put the fields on Values. Must put on Columns, Rows, etc.



# Return to Creating Calculation Group



# We accept this change

---

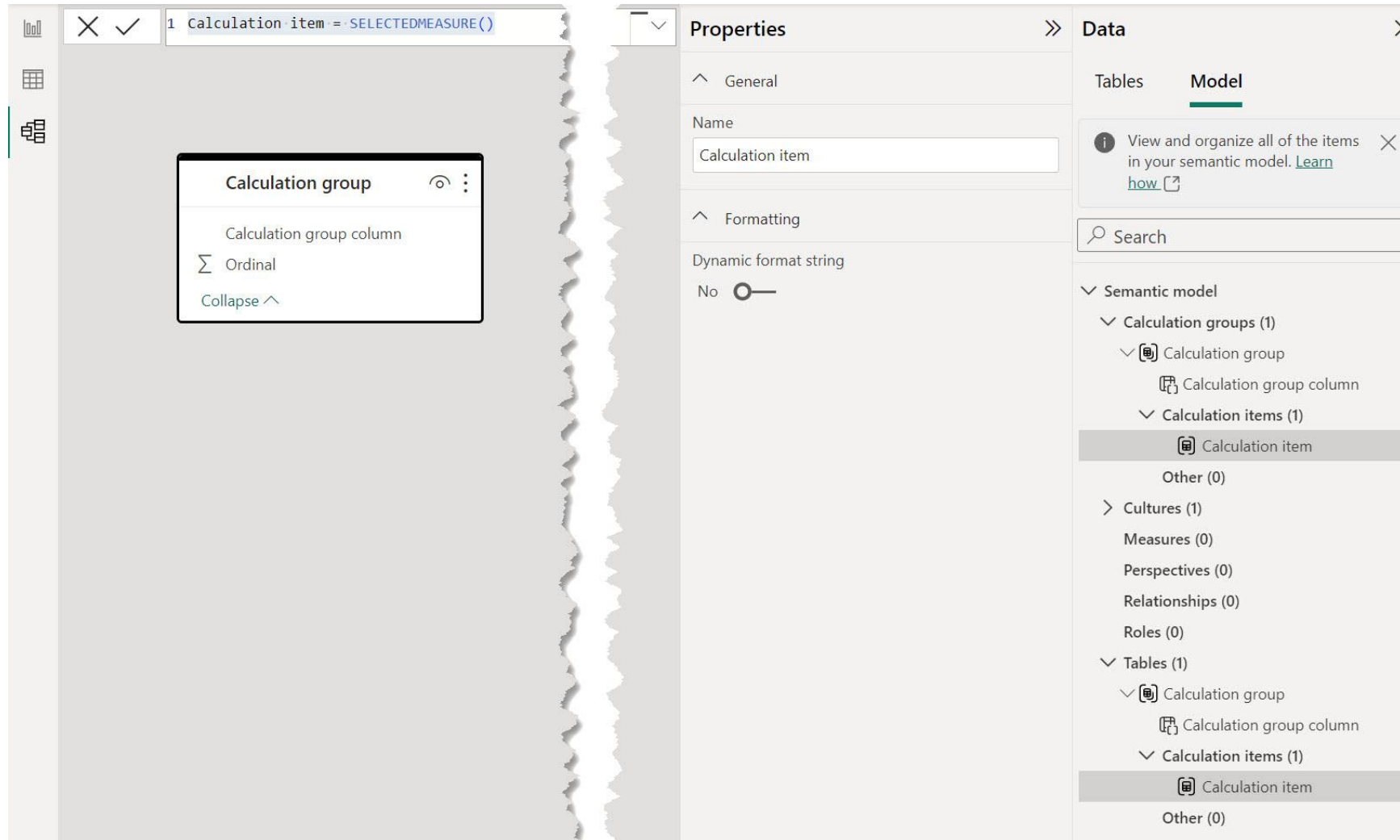
## This change will discourage implicit measures ×

When you add a calculation group, implicit measures will be discouraged in this model. This means you'll need to create explicit measures to aggregate data columns.

Yes

Cancel

# New Calculation Group View



# Calculation Group is Just a table

The image shows a screenshot of the Power BI Desktop interface, illustrating that a Calculation Group is essentially a table. The main view on the left shows a 'Calculation group' with a 'Calculation group column' and an 'Ordinal' value. A large red arrow points from the word 'Table' to the 'Calculation group' box. The right side of the image shows the 'Properties' and 'Data' panes. The 'Properties' pane shows the 'Name' of the Calculation group as 'Calculation item'. The 'Data' pane shows the 'Semantic model' structure, where the 'Calculation group' is listed under 'Tables (1)' and 'Calculation items (1)'. The 'Calculation item' is highlighted in both panes, indicating its role as a table within the model.

1 Calculation item = `SELECTEDMEASURE()`

**Calculation group**

- Calculation group column
- $\Sigma$  Ordinal
- Collapse ^

**Table**

**Properties**

- General
  - Name: Calculation item
- Formatting
  - Dynamic format string: No

**Data**

Tables **Model**

View and organize all of the items in your semantic model. [Learn how](#)

Search

**Semantic model**

- Calculation groups (1)
  - Calculation group
    - Calculation group column
    - Calculation items (1)
      - Calculation item
- Other (0)
- Cultures (1)
- Measures (0)
- Perspectives (0)
- Relationships (0)
- Roles (0)
- Tables (1)
  - Calculation group
    - Calculation group column
    - Calculation items (1)
      - Calculation item
- Other (0)

# Table has a column for Calculation Items

---

Calculation Item is just a value in the column

---

# Formula for new Calculation Item

The screenshot displays the Power BI Desktop interface. On the left, a 'Calculation group' context menu is open, showing options: 'Calculation group column', 'Σ Ordinal', and 'Collapse ^'. The formula bar at the top shows the formula: `1 Calculation item = SELECTEDMEASURE()`. A large red arrow points from the text 'Formula for First Item' to the formula bar. On the right, the 'Properties' pane is visible, showing the 'Name' field set to 'Calculation item'. The 'Data' pane on the far right shows the 'Model' view, with a tree structure listing 'Calculation groups (1)', 'Calculation items (1)', and 'Other (0)'. The 'Calculation items (1)' node is expanded, showing a single 'Calculation item'.

**Formula for First Item**

Calculation item = SELECTEDMEASURE()

Properties

General

Name

Calculation item

Formatting

Dynamic format string

No

Data

Tables

Model

View and organize all of the items in your semantic model. [Learn how](#)

Search

Semantic model

Calculation groups (1)

Calculation group

Calculation group column

Calculation items (1)

Calculation item

Other (0)

Cultures (1)

Measures (0)

Perspectives (0)

Relationships (0)

Roles (0)

Tables (1)

Calculation group

Calculation group column

Calculation items (1)

Calculation item

Other (0)





# Model Properties

Properties

>>

^ General

Name  
Semantic model

Description

Server  
localhost:58371

Compatibility Level  
1567

Cultures  
• en-US

Discourage implicit measures  
No ☐

Data

>>

Tables

Model

i

View and organize all of the items in your semantic model. [Learn how](#) [↗](#)

×

✓ Semantic model

Calculation groups (0)  
    > Cultures (1)  
    > Measures (2)  
        Perspectives (0)  
    > Relationships (4)  
        Roles (0)  
    > Tables (5)

# Discourage Implicit Measures Setting

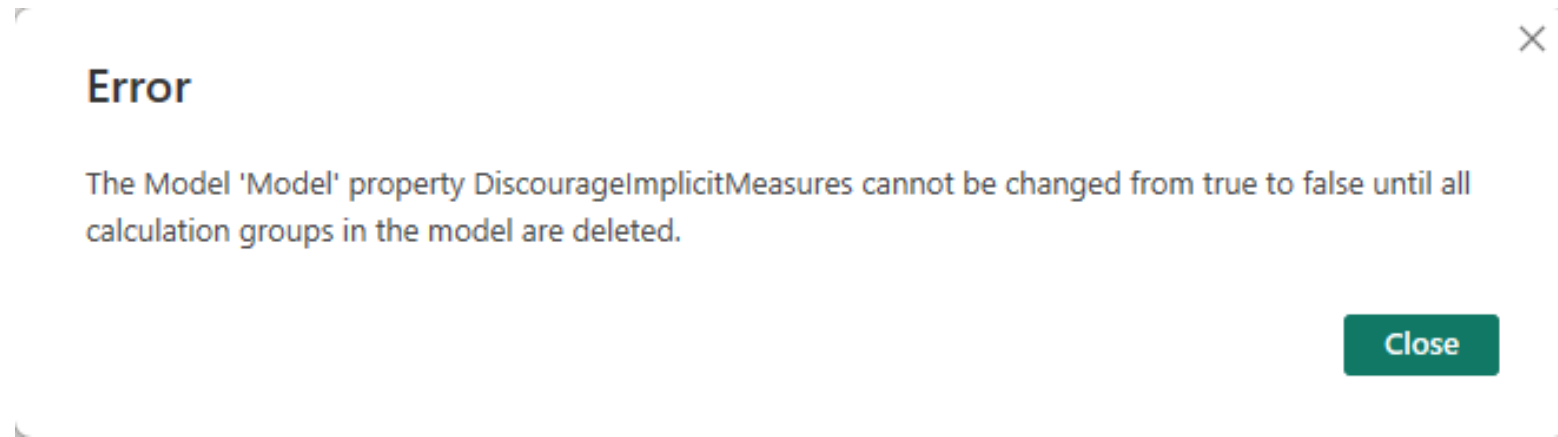
The screenshot displays the SQL Server Enterprise Manager interface. On the left, the 'Properties' pane is open to the 'General' tab for a 'Semantic model'. The 'Name' is 'Semantic model' and the 'Description' field is empty. The 'Server' is 'localhost:63754' and the 'Compatibility Level' is '1567'. Under 'Cultures', 'en-US' is listed. At the bottom, the 'Discourage implicit measures' setting is shown with a toggle switch currently set to 'Yes'. This section is highlighted with a red rectangle. On the right, the 'Data' pane is open to the 'Model' tab, showing a tree view of the semantic model's contents: Calculation groups (1), Cultures (1), Measures (5), Perspectives (0), Relationships (4), Roles (0), and Tables (6).

Properties	Data
<b>General</b>	<b>Model</b>
Name Semantic model	View and organize all of the items in your semantic model. <a href="#">Learn how</a>
Description Enter a description	Search
Server localhost:63754	▼ Semantic model
Compatibility Level 1567	> Calculation groups (1)
Cultures • en-US	> Cultures (1)
Discourage implicit measures Yes	> Measures (5)
	Perspectives (0)
	> Relationships (4)
	Roles (0)
	> Tables (6)

We can flip this switch back to No if we delete all Calculation Groups

# Discourage Implicit Measures Error

---



# Can reenale if all Calculation Groups Deleted

---

The screenshot displays the SSDT interface with two main panes: Properties and Data. The Properties pane on the left shows the 'General' tab for a semantic model. The 'Discourage implicit measures' property is set to 'No' and is highlighted with a red rectangular box. The Data pane on the right shows the 'Model' tab, which lists the semantic model's components: Calculation groups (0), Cultures (1), Measures (2), Perspectives (0), Relationships (4), Roles (0), and Tables (5). The 'Tables' section is expanded, showing a list of tables: Customer, Date, Product, Sales, and Store.

**Properties** >>

General

Name  
Semantic model

Description  
Enter a description

Server  
localhost:61535

Compatibility Level  
1567

Cultures  
• en-US

Discourage implicit measures  
No ☐

**Data** >>

Tables **Model**

View and organize all of the items in your semantic model. [Learn how](#)

Search

Semantic model

- Calculation groups (0)
- > Cultures (1)
- > Measures (2)
- Perspectives (0)
- > Relationships (4)
- Roles (0)
- Tables (5)
  - > Customer
  - > Date
  - > Product
  - > Sales
  - > Store

# Can reenable if all Calculation Groups Deleted

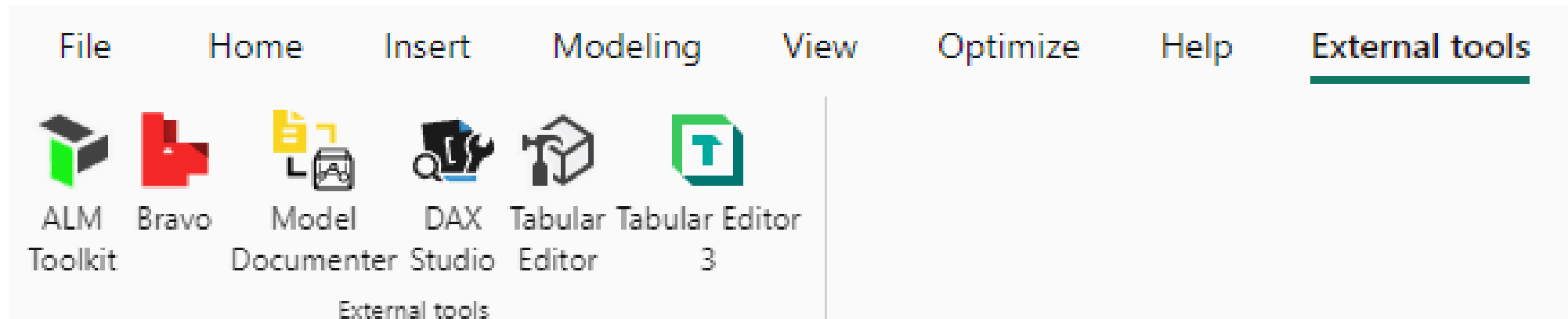
---

# Creating in Tabular Editor

---

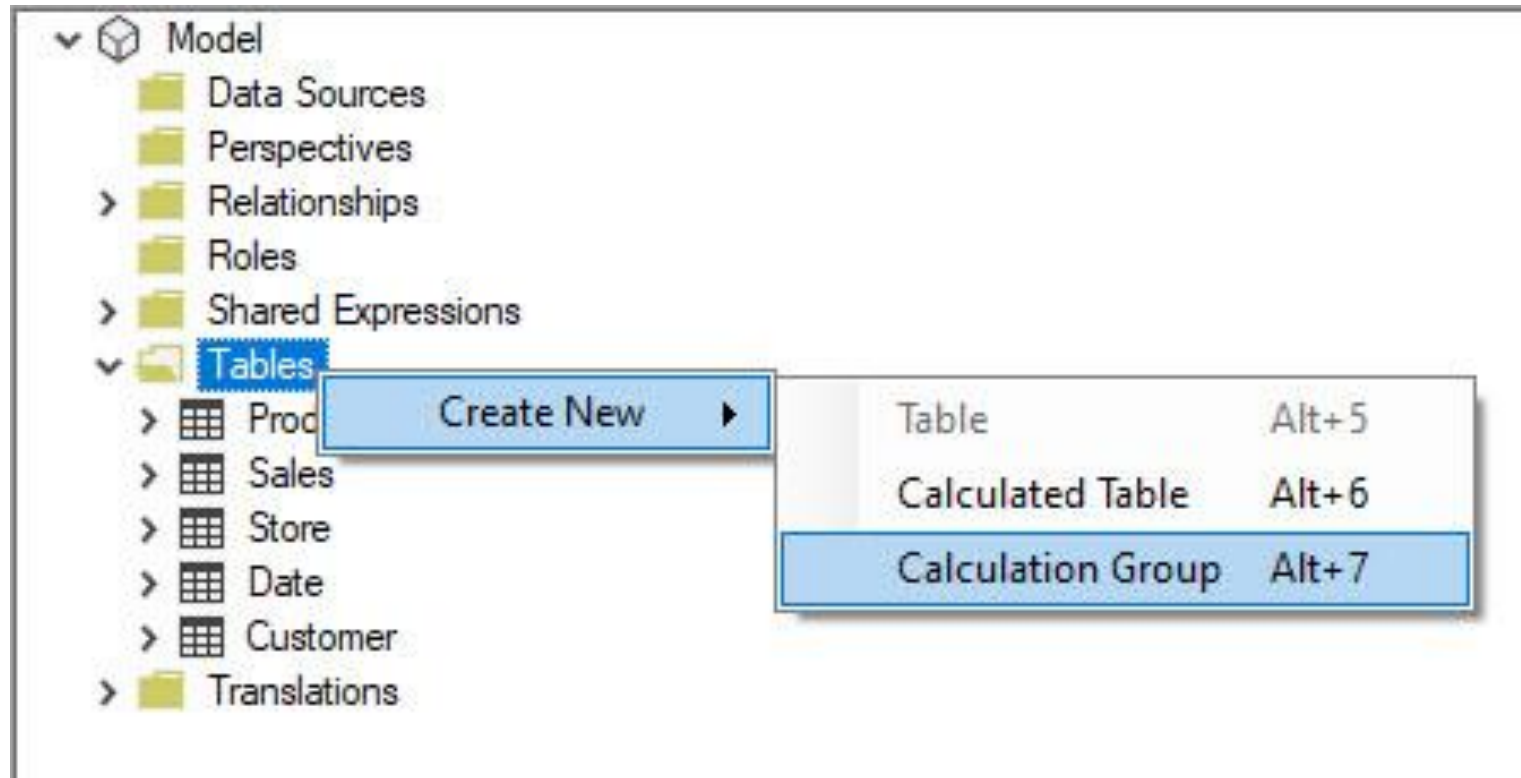
# Full Install – Will appear in External Tools Tab

---



# New Calculation Group – Under Tables

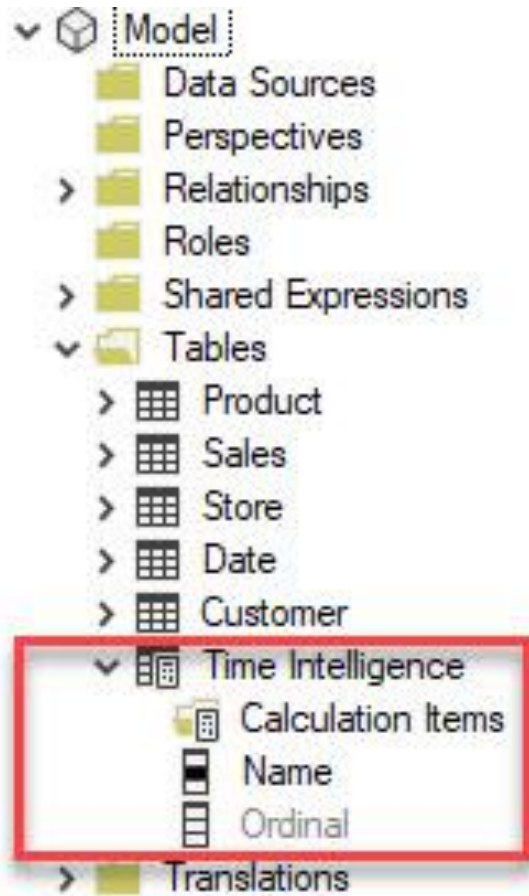
---





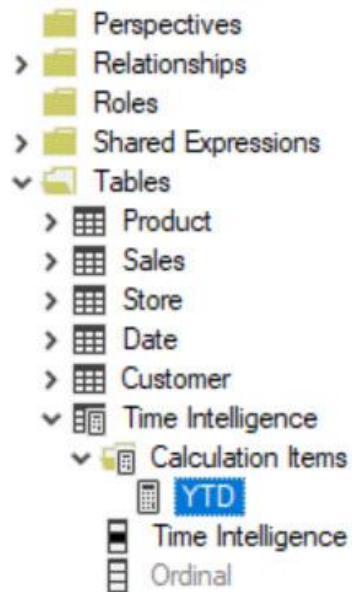
# New Calculation Group

---



# Calculation Item

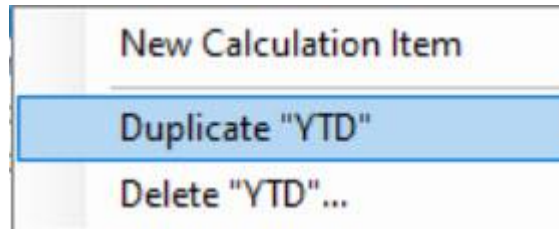
---



```
1 CALCULATE(  
2     SELECTEDMEASURE(),  
3     DATESYTD('Date'[Date])  
4 )
```

# Tabular Editor 2 Tricks - Duplicate

---



# TE 2: Duplicate – Make Changes

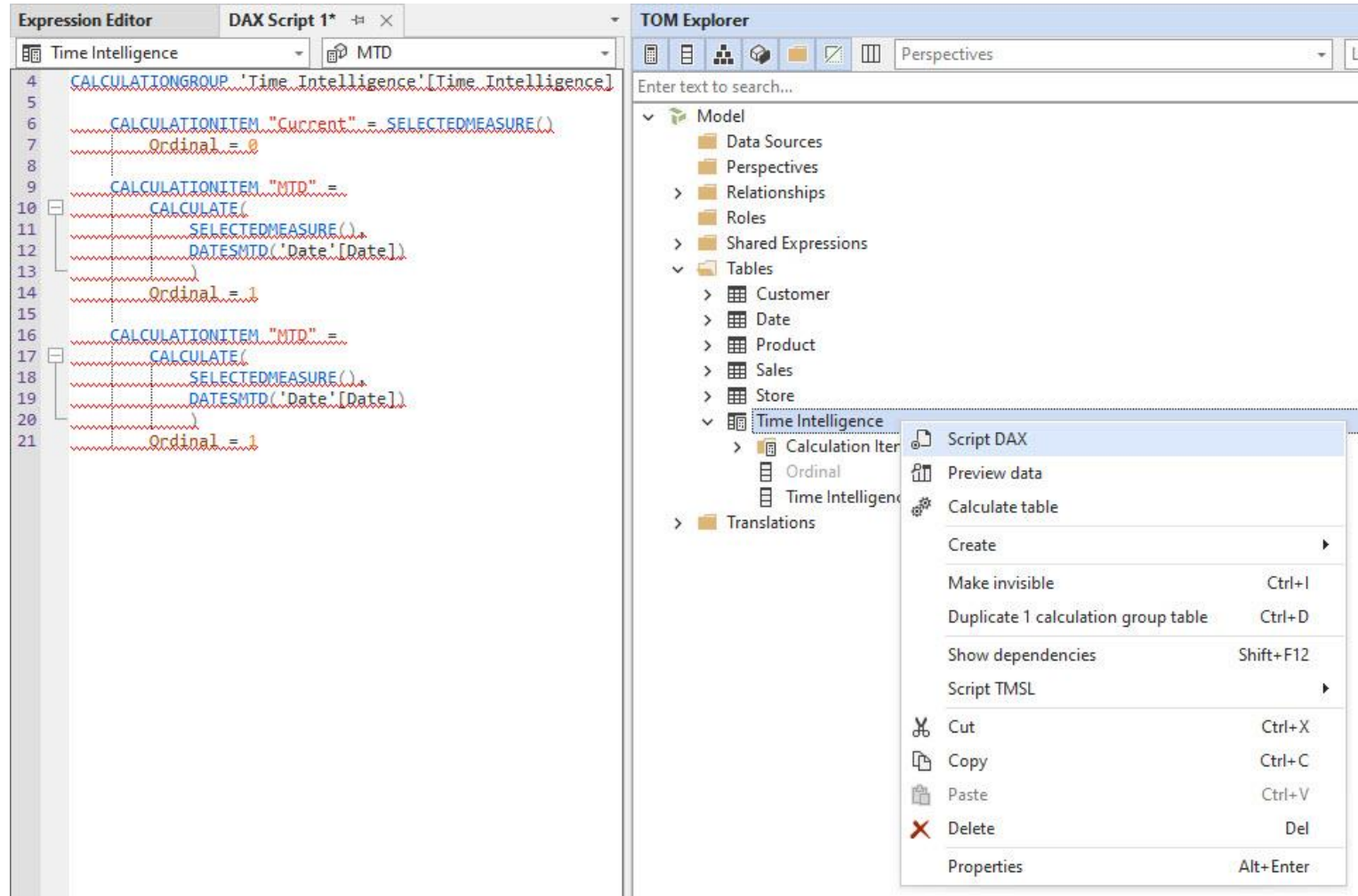
---

The screenshot displays the SQL Server Data Tools (SSDT) interface. On the left, the 'Model' tree is expanded, showing a hierarchy of 'Data Sources', 'Perspectives', 'Relationships', 'Roles', 'Shared Expressions', and 'Tables'. Under 'Tables', there is a list of tables: 'Product', 'Sales', 'Store', 'Date', 'Customer', and 'Time Intelligence'. Under 'Time Intelligence', there is a list of 'Calculation Items': 'YTD' and 'YTD copy'. The 'YTD copy' item is highlighted with a red rectangle. On the right, the 'Property: Expression' pane shows the DAX formula for the selected item, which is:

```
1 CALCULATE(  
2     SELECTEDMEASURE(),  
3     DATESYTD('Date'[Date])  
4 )
```


The formula is displayed in a blue font. The 'DATESYTD' function is highlighted with a red rectangle. The formula is numbered 1 through 4 on the left side of the pane.

# Tabular Editor 3 – Script DAX



# Power BI - Refresh is Required

---

 One or more calculation groups need to be manually refreshed.

Refresh now



# Agenda

---

1. Why Calculation Groups
2. How to Create
- 3. Calculation Items**
4. Multiple Calculation Groups
5. Conclusion

# Ordinal

---



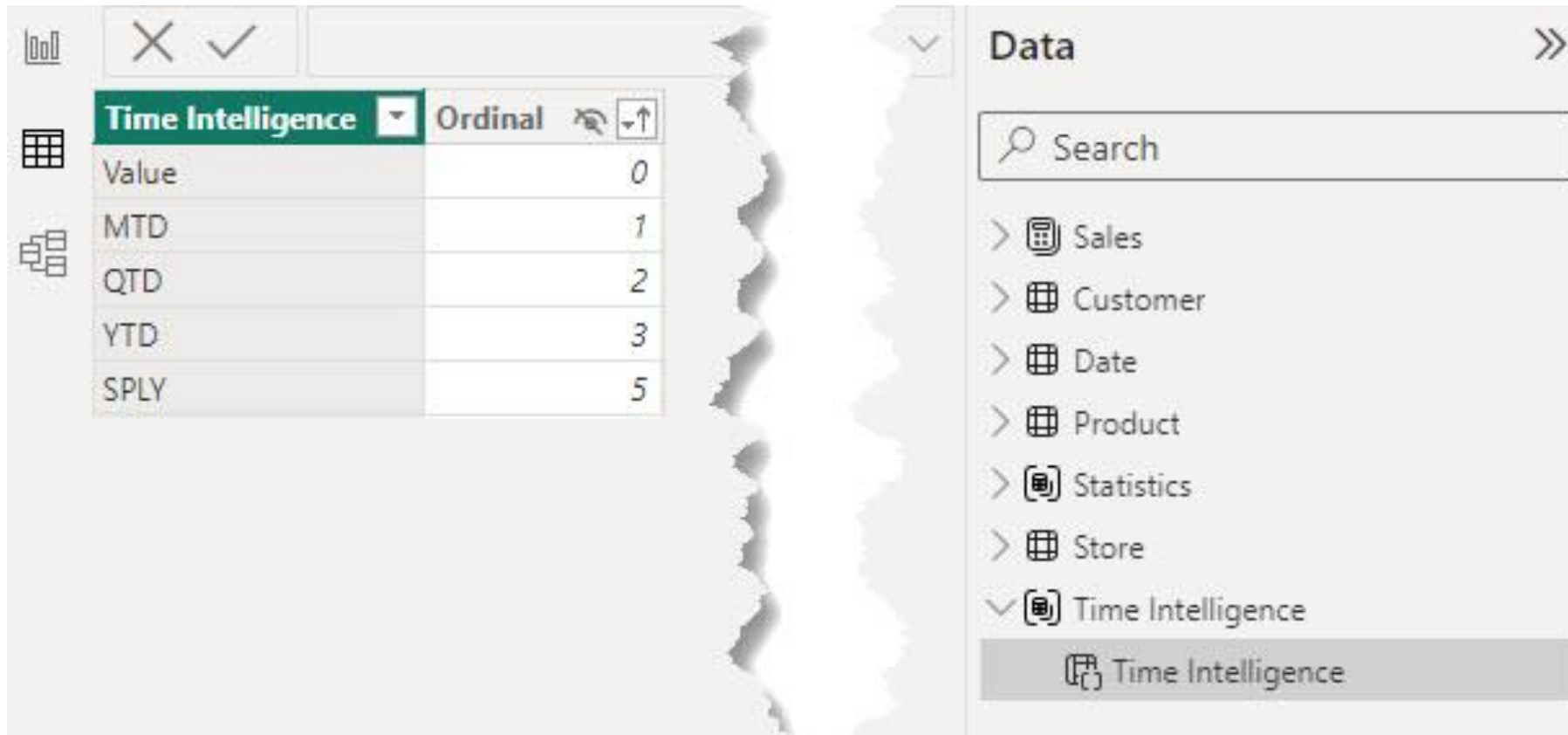
# Who's out of order

---

- Determines the order that Calculation Items are displayed
- Hidden column
  - Automatically Orders Calculation Items by this column

# Ordinal – Just a hidden column in the table

---



The image shows a screenshot of a BI tool interface, likely Microsoft Power BI. On the left, a table is displayed with a green header row. The first column is labeled 'Time Intelligence' and the second column is labeled 'Ordinal'. The table contains five rows of data. On the right, a 'Data' pane is visible, showing a search bar and a list of data sources. The 'Time Intelligence' source is highlighted at the bottom of the list.

Time Intelligence	Ordinal
Value	0
MTD	1
QTD	2
YTD	3
SPLY	5

**Data**

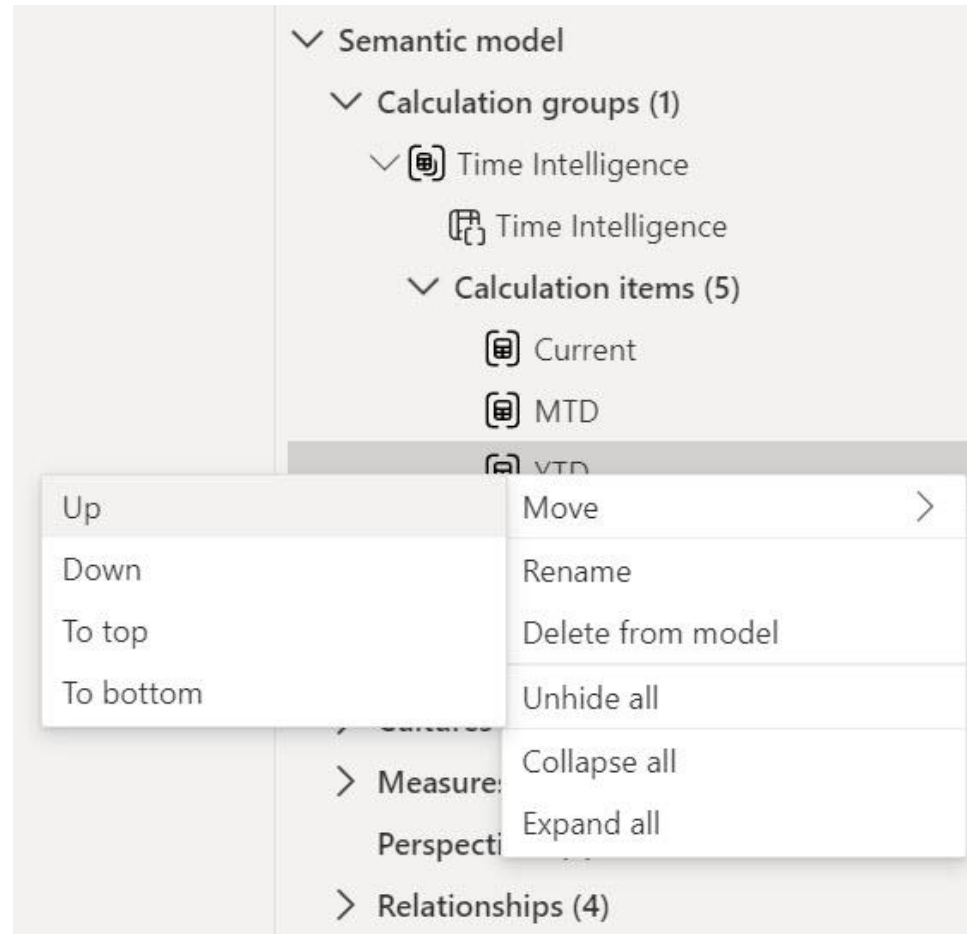
Search

- > Sales
- > Customer
- > Date
- > Product
- > Statistics
- > Store
- Time Intelligence

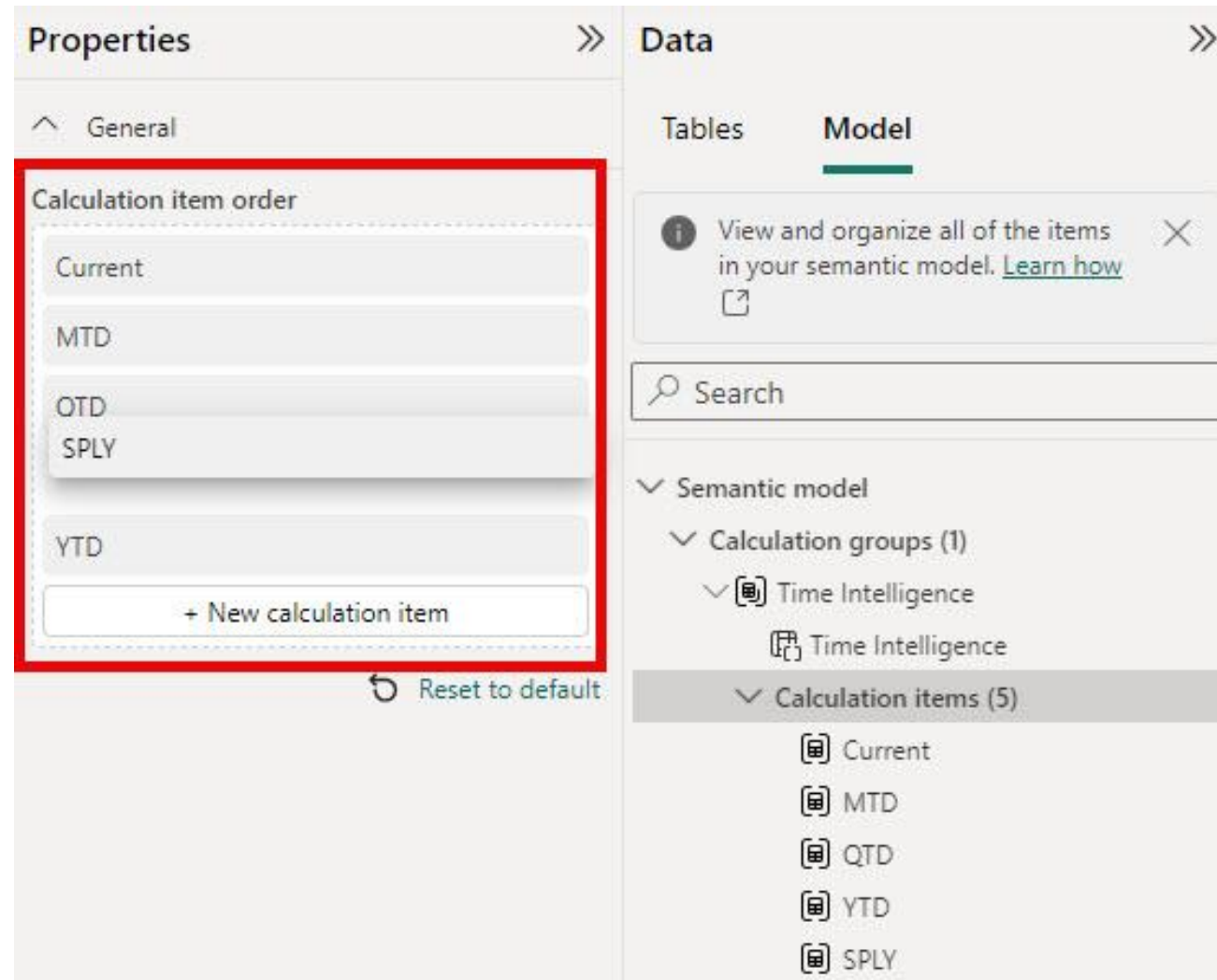
Time Intelligence

# Power BI Change Ordinal – Right Click

---

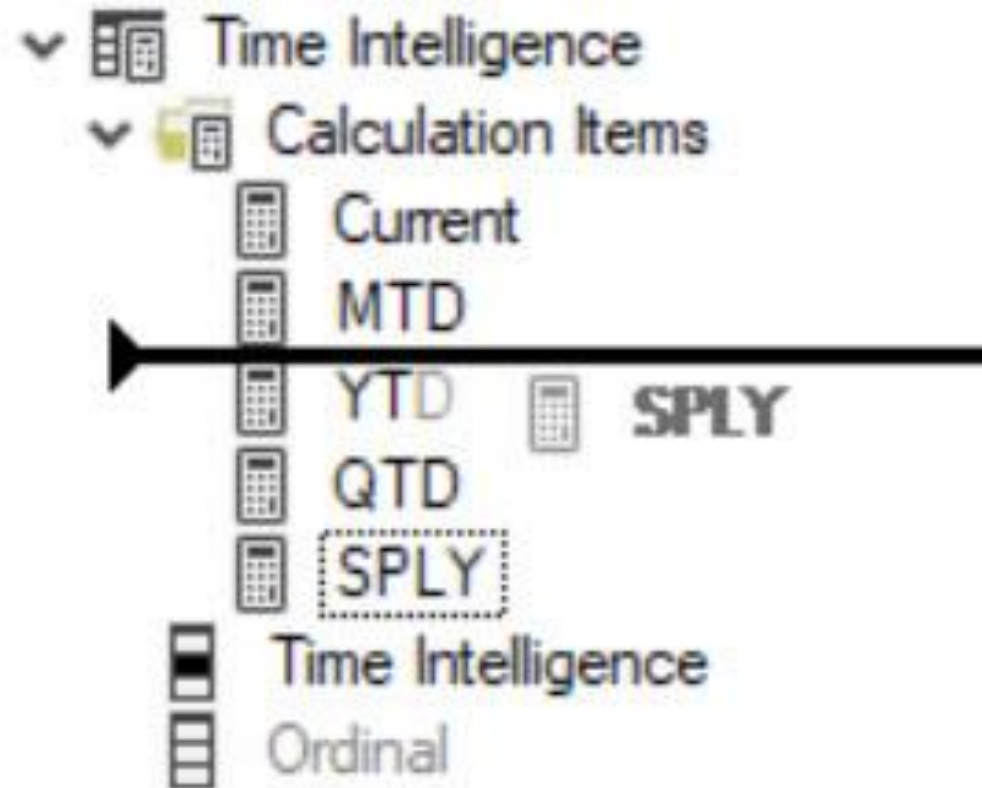


# Power BI Change Ordinal – Drag and Drop



# Tabular Editor – Drag and Drop

---



# Custom Format Strings

---

# Custom Format Strings

---

- If we don't want the format of the original measure

# Example Custom Format Strings

---

Example Measure is [Sales Amount]

CY = SELECTEDMEASURE()

PY = SELECTEDMEASURE() for SPLY

YOY = CY – PY

YOY % = DIVIDE(CY-PY, PY)

YOY % Format String = "0.00 %"



# Power BI: Format Strings

Properties

General

Name

YOY %

Formatting

Dynamic format string

Yes

Format String

"00.0 %"

Edit

Data

TablesModel

View and organize all of the items in your semantic model. [Learn how](#)

Search

Semantic model

Calculation groups (1)

Time Intelligence

Time Intelligence

Calculation items (7)

Current

MTD

QTD

YTD

SPLY

YOY

YOY %

# Dynamic Format String

---

- DAX expression to determine the format string
- `SELECTEDMEASUREFORMATSTRING()`
  - Returns measure format string
- Example Use
  - Only on these measures append this Unit of Measure

**ISSELECTEDMEASURE()**

---

# ISSELECTEDMEASURE()

---

```
IF (  
    NOT ISSELECTEDMEASURE ([Margin %]),  
    DO THIS,  
    Otherwise  
        Maybe SELECTEDMEASURE()  
)
```

# ISSELECTEDMEASURE()

---

Average =

```
IF (
    NOT ISSELECTEDMEASURE ( [Margin %] ),
    AVERAGEX (
        VALUES ( 'Date'[Year Month Number] ),
        SELECTEDMEASURE ()
    ),
    SELECTEDMEASURE ()
)
```

# Power BI Desktop Demo

---

# Agenda

---

1. Why Calculation Groups
2. How to Create
3. Calculation Items
- 4. Multiple Calculation Groups**
5. Conclusion

# How to Handle Precedence

---



# Precedence

---

Which Calculation Group gets applied first?

Is it peanut butter surrounded by chocolate or chocolate surrounded by Peanut Butter

# Precedence

---



[This Photo](#) by Unknown Author is licensed under [CC BY-NC-ND](#)

# Precedence

---

- Higher number has priority
- Set at group level
- Not at individual items

# Precedence

---

- Don't let this prevent you from trying Calculation Groups

# Stats (MAX) with higher precedence

---

YTD =

```
    CALCULATE (
        SELECTEDMEASURE (),
        DATESYTD ( 'Date'[Date] )
    ) -- Precedence 10
```

MAX =

```
    MAXX (
        VALUES ( 'Date'[Year Month] ),
        SELECTEDMEASURE ()
    ) - Precedence 20
```


---

YTD =

```
CALCULATE (  
    SELECTEDMEASURE (),  
    DATESYTD ( 'Date'[Date] )  
) -- Precedence 10
```

MAX =

```
MAXX (  
    VALUES ( 'Date'[Year Month] ),  
    SELECTEDMEASURE ()  
) - Precedence 20
```



---

```
MAXX (
    VALUES ( 'Date'[Year Month] ),
    CALCULATE (
        SELECTEDMEASURE (),
        DATESYTD ( 'Date'[Date] )
    )
)
```

# Not the result we were looking for...

---

```
MAXX (
    VALUES ( 'Date'[Year Month] ),
    CALCULATE (
        [Sales Amount],
        DATESYTD ( 'Date'[Date] )
    )
)
```



# Take two

---

YTD =

```
CALCULATE (  
    SELECTEDMEASURE (),  
    DATESYTD ( 'Date'[Date] )  
 ) -- Precedence 20
```

MAX =

```
MAXX (  
    VALUES ( 'Date'[Year Month] ),  
    SELECTEDMEASURE ()  
 ) - Precedence 10
```

# Time Intelligence with higher precedence

---

YTD =

```
CALCULATE (  
    SELECTEDMEASURE (),  
    DATESYTD ( 'Date'[Date] )  
 ) -- Precedence 20
```

MAX =

```
MAXX (  
    VALUES ( 'Date'[Year Month] ),  
    SELECTEDMEASURE ()  
 ) - Precedence 10
```

# Time Intelligence with higher precedence

---

YTD =

```
CALCULATE (  
    SELECTEDMEASURE (),  
    DATESYTD ( 'Date'[Date] )  
)
```

-- Precedence 20

MAX =

```
MAXX (  
    VALUES ( 'Date'[Year Month] ),  
    SELECTEDMEASURE ()  
)
```

- Precedence 10

# Time Intelligence with higher precedence

---

```
YTD =  
    CALCULATE (  
        MAXX (  
            VALUES ( 'Date'[Year Month] ),  
            SELECTEDMEASURE ()  
        )  
        DATESYTD ( 'Date'[Date] )  
    )
```

# Much Better

---

```
YTD =  
    CALCULATE (  
        MAXX (  
            VALUES ( 'Date'[Year Month] ),  
            [Sales Amount]  
        )  
    )  
    DATESYTD ( 'Date'[Date] )  
)
```

If you aren't sure which to set higher – test it

# Precedence

---

- May need to simulate the application of Calculation Items in DAX code
- Higher Precedence becomes the shell
  - Lower Precedence gets inserted
- Higher Precedence for Time Intelligence
  - Lower for Stats
  - i.e. YTD should be applied first and then the calculation

# Precedence and Dynamic Format Strings

---

- Precedence determines which dynamic string is used
  - Highest precedence calculation group is the one used
- If the measure has dynamic string
  - Lowest precedence to any calculation group

# Off the Rails

---

- If you start using Calculation Groups in DAX Code you need to understand the details



# Sideway Recursion

---

- Each Calculation Item can only be called once
- Benefit code runs
- Negative is that no warning is issued
- Best Practice – Do not use Calculation Groups in Measures

# Applying Multiple Calculation Items

---

- If you try to apply multiple Calculation Items it will return the original Measure
- For Example
  - Apply YTD and QTD to Sales Amount

# Agenda

---

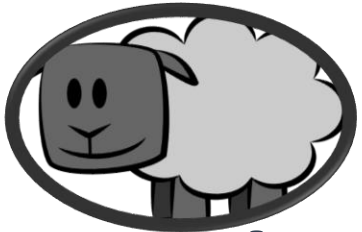
1. Why Calculation Groups
2. How to Create
3. Calculation Items
4. Multiple Calculation Groups
- 5. Conclusion**

# Worth Investing Your Time

---

- Start with Single Calculation Group
- Try Creating with Power BI Desktop and Tabular Editor
  - TE has lots of ways to speed up creating Calculation Groups

# Thank you



**The Dax Shepherd**

**Jason Romans**

**thedaxshepherd@gmail.com**

**<https://thedaxshepherd.net/>**

The DAX Shpeherd

