



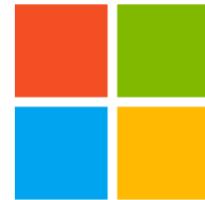
# A Power BI Documentation System

Not Just an Afterthought



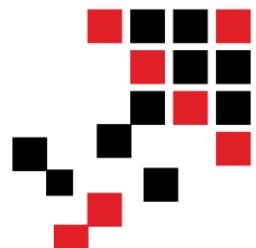


Special thanks to Fabric and Power BI Team at



**Microsoft**

This Summit presented to you by



**RADACAD**





# David Kofod Hanna



Senior Advisor, Data Storytelling @ Twoday  
+200 courses as Academy Trainer and 10 years as consultant  
Microsoft Data Platform MVP, Certified Trainer in Microsoft and Tabular Editor



Passionate about guiding self-service Power BI  
developers for more enterprise manageable concepts in a consumable and  
practical way



Born on beautiful “Sunshine island”: Bornholm  
Lives in Denmark, Silkeborg with wife and 3 kids  
Love football and running half-marathons







# Who loves documentation?





# Who writes documentation?





# Who **updates** documentation?





# Who reads documentation?





# Who can find documentation?





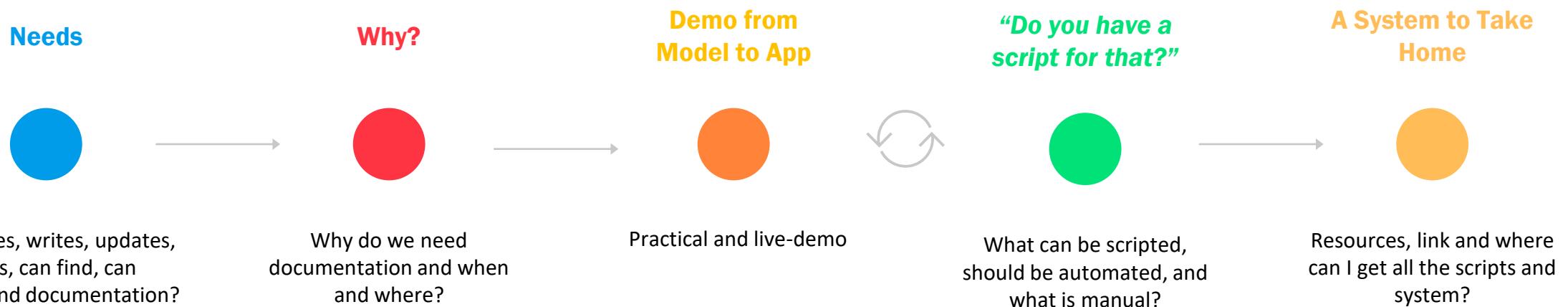
# Who can understand documentation?





# A Power BI Documentation System

– Not Just an Afterthought





# Example of project

Requirements

Approve

Develop

Validate

Test

Training

Documentation

Requirements

Approve

Develop

Validate

Test

Training

Documentation





# Why documentation?

Transparency

Reproducibility

Collaboration

Compliance

Scalability

Easier debugging





**AND NOW I'M LEAVING THE COMPANY...**





✍️ Please write my  
documentation





*“You do not rise to the level of  
your goals, you fall to  
the level of your systems.”*

- James Clear, Atomic Habits





# Power BI Documentation System – from Model to App



Power Query  
{M}



Semantic Model



Power BI Desktop



App



Purpose for docs:  
Clear, Concise,  
Complete, Consistent,  
Correct, Consumable

Documentation and  
transparency from  
model to app

From report consumers  
to report developers to  
semantic model  
developers and data  
ingest

Best Practice Analyzer  
Rules to highlight  
violations and severity  
of best practices

Tabular Editor 2/3 for  
Semantic Model and  
DAX scripts

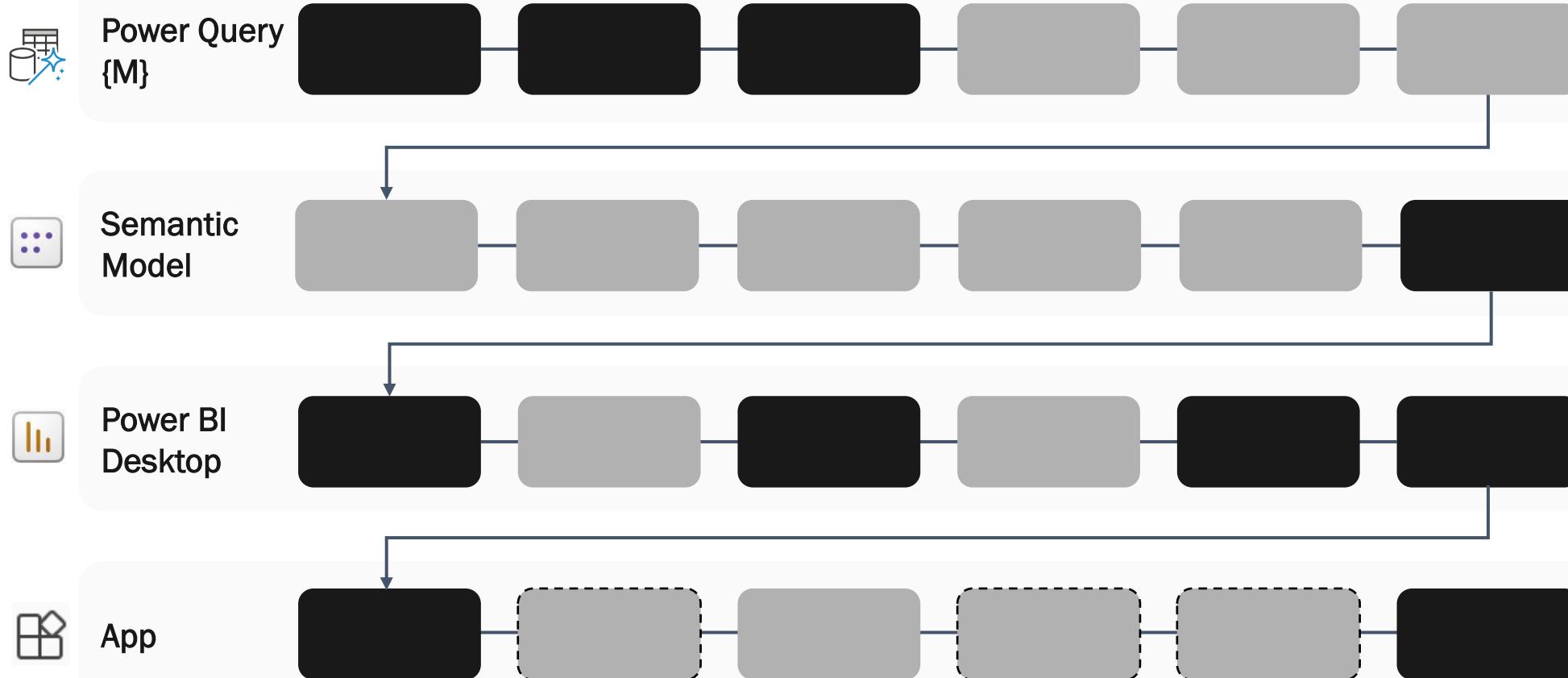
Semantic Link Labs in  
Fabric Notebooks for  
automation like never  
before





Power BI Documentation System – from Model to App

Automated or Script      Manual      Requires Fabric SKU



**Purpose for docs:**  
*Clear, Concise,  
Complete, Consistent,  
Correct, Consumable*

Documentation and transparency from model to app

From report consumers  
to report developers to  
semantic model  
developers and data  
ingest

## Best Practice Analyzer Rules to highlight violations and severity of best practices

## Tabular Editor 2/3 for Semantic Model and DAX scripts

## Semantic Link Labs in Fabric Notebooks for automation like never before

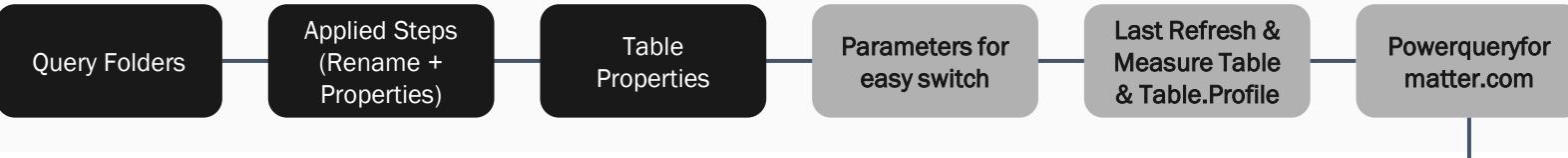




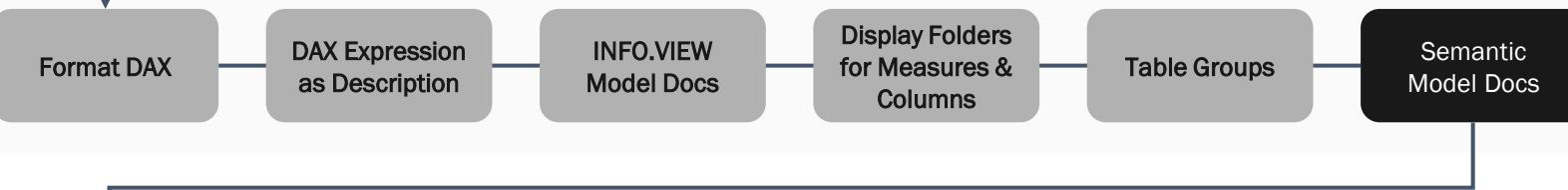
# Power BI Documentation System – from Model to App

Automated or Script    Manual    Requires Fabric SKU

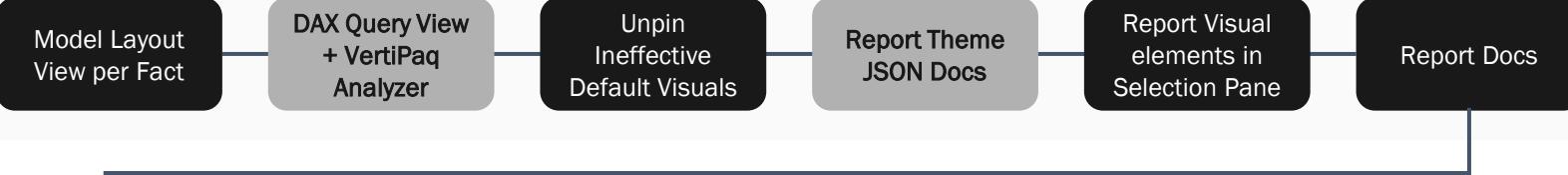
## Power Query {M}



## Semantic Model



## Power BI Desktop



## App



Purpose for docs:  
Clear, Concise,  
Complete, Consistent,  
Correct, Consumable

Documentation and  
transparency from  
model to app

From report consumers  
to report developers to  
semantic model  
developers and data  
ingest

Best Practice Analyzer  
Rules to highlight  
violations and severity  
of best practices

Tabular Editor 2/3 for  
Semantic Model and  
DAX scripts

Semantic Link Labs in  
Fabric Notebooks for  
automation like never  
before





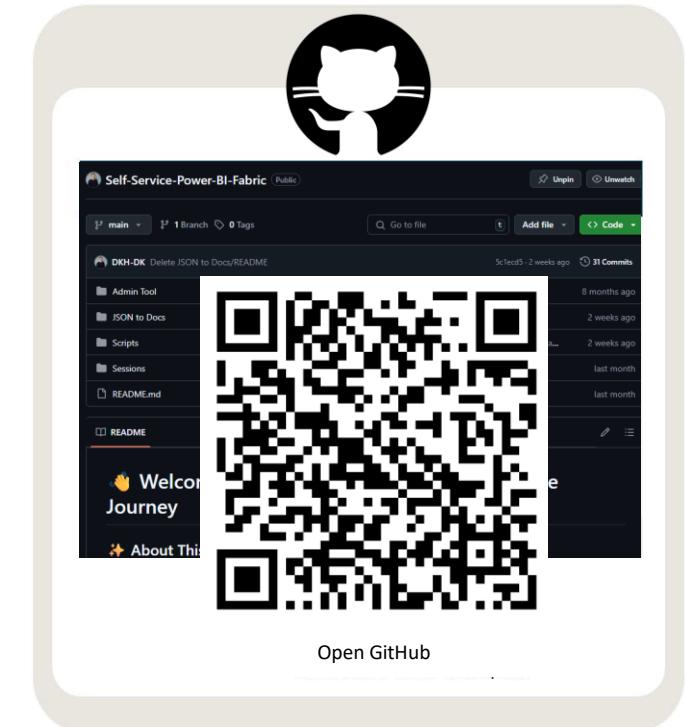
# A Power BI Documentation System

– Not Just an Afterthought

# Released on GitHub and LinkedIn article

#DocumentationCanBeFun

1. Create M Parameter	C# Script Source File
2. Create Global Measure Table	C# Script Source File
3. Create Last Refresh	C# Script Source File
4. Format Power Query (M)	C# Script Source File
5. Format DAX Measures	C# Script Source File
6. Measure DAX Expression as Description	C# Script Source File
7. Model Documentation DAX Script	TE3DAXS File
7. Model Documentation DAX	Text Document
8. Display Folders for Measures & Columns	C# Script Source File
9. Create Table Groups TE3	C# Script Source File
9. Table Groups in Power BI Desktop with INFO.VIEW DAX	File
9. Table Groups in Power BI Desktop with INFO.VIEW DAX Script	TE3DAXS File
10. Best Practice Analyzer Rules incl. John Kerski PQ Doc Rules	JSON Source File
11. Measure Dependency	DAX Query File
11. Model Issues	DAX Query File
11. Model Summary	DAX Query File
11. VertiPaq Column	DAX Query File
11. VertiPaq Memory Size	DAX Query File
11. VertiPaq Partition	DAX Query File
11. VertiPaq Relationship	DAX Query File
11. VertiPaq Table	DAX Query File
12. DHK Self-Service Report Theme - Raw Template	JSON Source File
12. JSON to Power BI Docs - Raw Template	Microsoft.MicrosoftPowerBIDesktop
13. Report Analysis Notebook - Michael Kovalsky	Jupyter Source File
14. Measure Maze Dependency Sandeep Pawar	Jupyter Source File
15. Design Document - Sample Fragment 01 - General and Scope - V0.1	Microsoft Word Document
15. Design Document - Sample Fragment 02 - Workflow Issues and Business Rules - V0.2	Microsoft Word Document
15. Design Document - Sample Fragment 03 - Data - V0.3	Microsoft Word Document
15. Design Document - Sample Fragment 04 - Reports - V0.4	Microsoft Word Document
MsBIP - Power BI Documentation System - David Kofod Hanna - May 2025	Microsoft PowerPoint Presentation



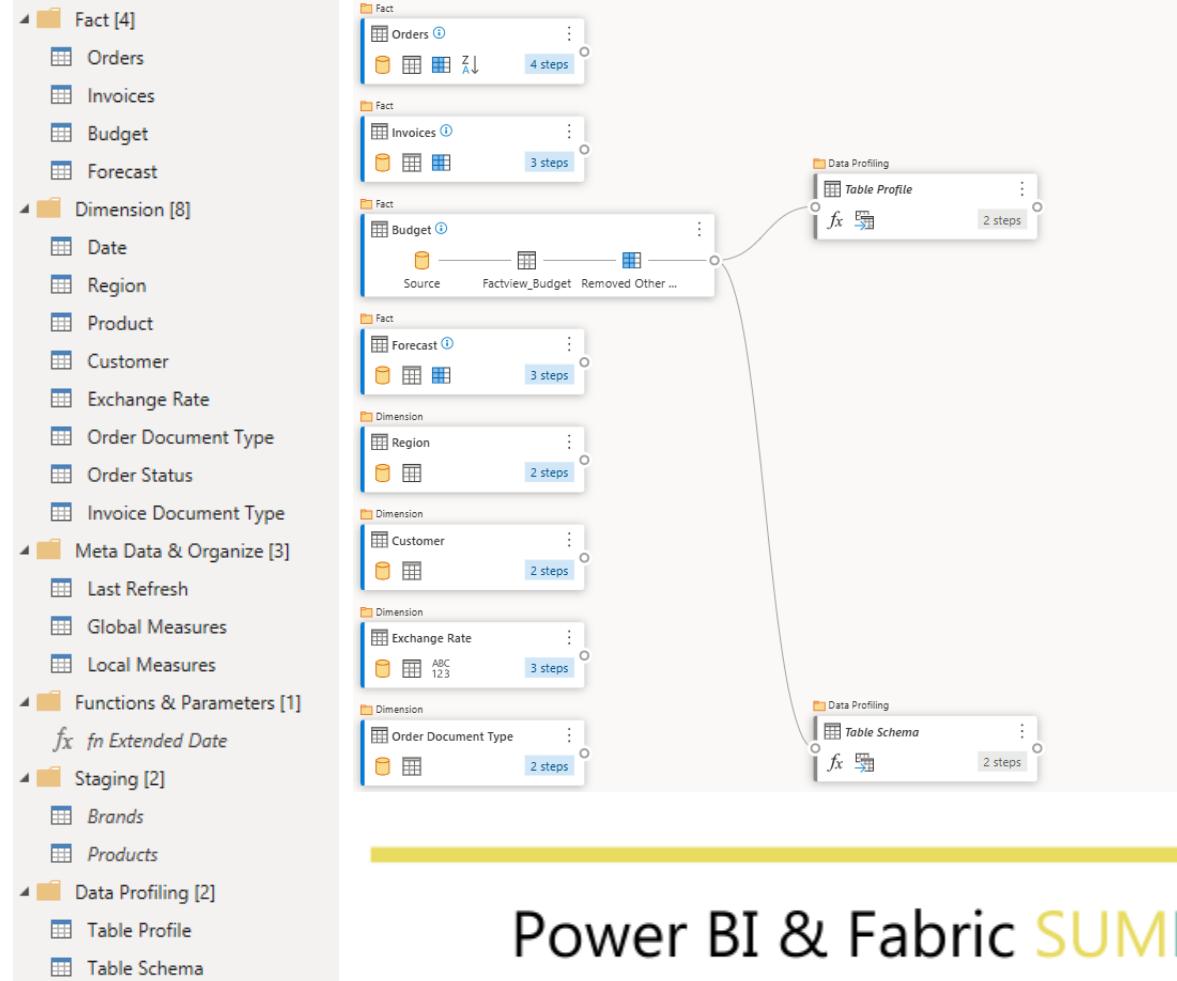
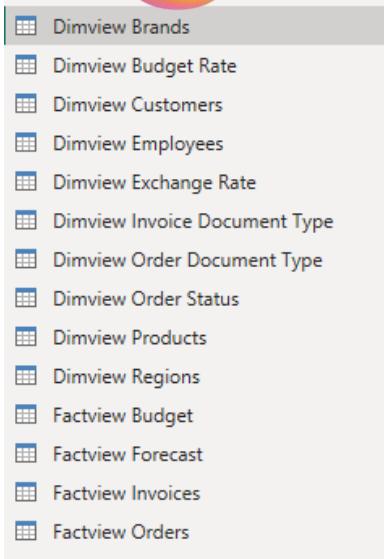
The GitHub repository screenshot shows the main page of the "Self-Service-Power-BI-Fabric" repository. It features a large GitHub logo at the top. Below it, there's a "Welcome Journey" message and an "About This Project" link. The repository stats show 1 branch, 0 tags, and 31 commits. A QR code is displayed prominently in the center. On the right side, there's a timeline of commits from 8 months ago to last month.

Open GitHub





# Query folders by type – and Diagram View in Dataflow Gen2





# Applied Steps & Properties in step



## APPLIED STEPS

Source	...
Navigation	...
Promoted Headers	...
Changed Type	...
Added Conditional Column	...
Changed Type1	...
Added Conditional Column1	...
Added Conditional Column2	...
Added Conditional Column3	...
Changed Type2	...
Inserted Sum	...
Renamed Columns	...
Merged Queries	...
Expanded Accuracy Master	...
Added Conditional Column4	...
Changed Type3	...
Filtered Rows	...



## APPLIED STEPS

Source	...
Navigation	...
Promoted Headers	...
Changed Type	...
Filtered Rows on Nulls from Excel	...
DQ Check Column: Firstname = Null	...
DQ Check Column: Lastname = Null	...
DQ Check Column: Mobilephone = Null	...
DQ Check Column: e-Mail = Null	...
DQ Check Column Sum of Nulls above	...
Merged Queries Accuracy Master for c... <sup>①</sup>	...
Expanded Accuracy Master for clientna...	...
DQ Check Column: Clientname = Client...	...
Changed Type Wholenumber for DQ c...	...

View Native Query  
Diagnose  
Properties...

### 💡 My Standard:

**Filter as early as possible (rows and columns)**

**Combine similar steps** into one – instead of Changed Type1, Changed Type2, Changed Type3

**Rename steps** that are important for yourself or colleagues to understand

- Added Conditional Column
- Merge Queries
- Append Queries

Add more documentation for tooltip info at steps by right-clicking step and choose **Properties**– if you are needing more space than in the step name.

The #"" in the steps can be avoided if you don't use space

**Query Folding** for SQL DB and OData connection and always **Roche's Maxim of Data Transformation**.

 **David Kofod Hanna** • You  
Senior Advisor | Academy | Speaker 3d ...

Not an excuse, but curious if the Power Query engine will optimize this in the engine call?

Like | Reply · 2 replies | 1,863 impressions

 **Chandeep Chhabra** • Author  
Power BI Trainer and Consultant 3d ...

David Kofod Hanna Very hard to convincingly say - "Power Query engine will take care of it"





# Table properties – add it once and see it everywhere





## Table properties – add it once and see it everywhere

**Queries [22]**

- Dimension [8]
  - Customer
    - Copy
    - Paste
    - Delete
    - Rename
    - Enable load
    - Include in report refresh
    - Duplicate
    - Reference
    - Move To Group
    - Move Up
    - Move Down
    - Create Function...
    - Convert To Parameter
    - Advanced Editor
    - Properties...

**Query Properties**

Name: Customer

Description: From CRM, ERP and ServiceNow - combined and validated by Data Governance team

Enable load to report

Include in report refresh

OK Cancel

### In Power Query Editor

**Queries [22]**

- Dimension [8]
  - Product
  - Customer
  - Ex: From CRM, ERP and ServiceNow - combined and validated by Data Governance team
  - Order Document Type

### In Power BI Desktop

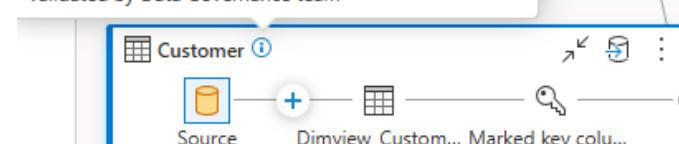
Name: Customer  
Storage mode: Import  
Data refreshed: 5/16/2025, 8:06:53 PM  
Description: From CRM, ERP and ServiceNow - combined and validated by Data Governance team

Data

- Units
- User Configuration
- Customer
- Date

### In Dataflows in the service

From CRM, ERP and ServiceNow - combined and validated by Data Governance team



# Power E

### In TMDL

Apply Preview

```
1 createOrReplace
2
3
4
5
6
7
8
9
10
11
12
/// From CRM, ERP and ServiceNow - combined and validated by Data Governance team
table Customer
lineageTag: 726a20c4-ae07-450e-8d9f-e2b51dfa4e91
column 'Customer Key'
dataType: string
isHidden: false
isAvailableInMdx: false
displayFolder: Key
```

### In INFO.VIEW DAX Functions

Model Documentation by INFO.VIEW Functions

Location	Name	Type	Expression	Description
Budget		Table	From IBM TM1	
Customer		Table	From CRM, ERP and ServiceNow - combined and validated by Data Governance team	
Date		Table	Defined from Melissa de Korte Extended Date Function	
Exchange Rate		Table	From purchased source from web	
Global Measures		Table	Store Global measures from Semantic Model Developer	
Invoice Document Type		Table	MS F&O	
Last Refresh		Table	Last refresh of semantic model - not the underlying job and pipelines	
Local Measures		Table	Store Local Report Level Measures	
Order Document Type		Table	MS F&O	
Order Status		Table	MS F&O	

### In OneLake catalog

Semantic model: Power BI Docs - System

Open

Overview Lineage Monitor Permissions

Location: Power BI & Fabric Showcase Refreshed: 19/05/25, 13:35:28 Owner: David Kofod Hanna

#### Tables

Name	Type	Description
> Customer	Table	From CRM, ERP and ServiceNow - combined and validated by ...
> Date	Table	Derived from Melissa de Korte Extended Date Function
> Exchange Rate	Table	From purchased source from web
> Global Measures	Table	Store Global measures from Semantic Model Developer



# Model Docs DAX INFO.VIEW

## Model Documentation

by INFO.VIEW Functions

Filter by keyword →

Column Measure Relationship Table

Location	Name	Type	Expression	Description
Global Measures	Global Measures	Measure	"Locate them here my semantic model developer friend"	
Global Measures	Sum of Delivery Cost	Measure	SUM('Invoices'[Delivery Cost])	This measure is the sum of column 'Invoices'[Delivery Cost]
Global Measures	Sum of Forecast (EUR)	Measure	SUM('Forecast'[Forecast (EUR)])	This measure is the sum of column 'Forecast'[Forecast (EUR)]
Global Measures	Sum of Freight	Measure	SUM('Invoices'[Freight])	This measure is the sum of column 'Invoices'[Freight]
Global Measures	Sum of Late Delivery Penalti...	Measure	SUM('Invoices'[Late Delivery Penalties])	This measure is the sum of column 'Invoices'[Late Delivery Penalties]
Global Measures	Sum of Net Invoice COGS	Measure	SUM('Invoices'[Net Invoice COGS])	This measure is the sum of column 'Invoices'[Net Invoice COGS]
Global Measures	Sum of Net Invoice Cost	Measure	SUM('Invoices'[Net Invoice Cost])	This measure is the sum of column 'Invoices'[Net Invoice Cost]
Global Measures	Sum of Net Invoice Quantity	Measure	SUM('Invoices'[Net Invoice Quantity])	This measure is the sum of column 'Invoices'[Net Invoice Quantity]
Global Measures	Sum of Net Invoice Value	Measure	SUM('Invoices'[Net Invoice Value])	This measure is the sum of column 'Invoices'[Net Invoice Value]
Global Measures	Sum of Net Order Quantity	Measure	SUM('Orders'[Net Order Quantity])	This measure is the sum of column 'Orders'[Net Order Quantity]
Global Measures	Sum of Net Order Value	Measure	SUM('Orders'[Net Order Value])	This measure is the sum of column 'Orders'[Net Order Value]
Global Measures	Sum of Total Budget	Measure	SUM('Budget'[Total Budget])	This measure is the sum of column 'Budget'[Total Budget]
Local Measures	Local Measures	Measure	"Locate them here my report developer friend"	
Table Group DAX	Number of tables	Measure	COUNTROWS(Table Group DAX)	

## INFO.VIEW DAX Functions make this possible

INFO.VIEW.COLUMNS()

INFO.VIEW.MEASURES()

INFO.VIEW.TABLES()

INFO.VIEW.RELATIONSHIPS()

```

1 Model Documentation =
2 VAR _columns =
3   SELECTCOLUMNS(
4     FILTER(
5       INFO.VIEW.COLUMNS( ),
6       [Table] <> "Model Documentation" && NOT ( [IsHidden] )
7     ),
8       "Type", "Column",
9       "Name", [Name],
10      "Description", [Description],
11      "Location", [Table],
12      "Expression", [Expression]
13    )
14 VAR _measures =
15   SELECTCOLUMNS(
16     FILTER(
17       INFO.VIEW.MEASURES( ),
18       [Table] <> "Model Documentation" && NOT ( [IsHidden] )
19     ),
20       "Type", "Measure",
21       "Name", [Name],
22       "Description", [Description],
23       "Location", [Table],
24       "Expression", [Expression]
25    )
26 VAR _tables =
27   SELECTCOLUMNS(
28     FILTER(
29       INFO.VIEW.TABLES( ),
30       [Name] <> "Model Documentation" && [Name] <> "Calculations"
31       && NOT ( [IsHidden] )
32     ),
33       "Type", "Table",
34       "Name", [Name],
35       "Description", [Description],
36       "Location", BLANK( ),
37       "Expression", [Expression]
38    )
39 VAR _relationships =
40   SELECTCOLUMNS(
41     INFO.VIEW.RELATIONSHIPS( ),
42       "Type", "Relationship",
43       "Name", [Relationship],
44       "Description", BLANK( ),
45       "Location", BLANK( ),
46       "Expression", [Relationship]
47    )
48 RETURN
49 UNION( _columns, _measures, _tables, _relationships )

```





## Display folders for columns



Factview Invoices
Billing Date
Σ Billing Document Line Item Number
Σ Billing Document Number
Billing Document Type Code
Customer Key
Σ Delivery Cost
DWCreatedDate
Σ Freight
Σ Late Delivery Penalties
Local Currency
Σ Net Invoice COGS
Σ Net Invoice Cost
Σ Net Invoice Quantity
Σ Net Invoice Value
OTD Indicator
Σ Overdue Payment Penalties
Product Key
Ship Date
Σ Taxes & Commercial Fees



Invoices
> Attributes
> Dates
> Flags
> Key
> Numeric

```
//Go through each table in the model
foreach(var table in Model.Tables)
{
    if(table.Name != "Date")
    {
        //First look at columns
        foreach( var column in table.Columns)
        {
            var keySuffix = "Key";
            var columnDataType = column.DataType.ToString();
            //DWCreatedDate column should be hidden in a seperate folder
            if( column.Name == "DWCreatedDate")
            {
                column.DisplayFolder = "Attributes\\Metadata";
                column.isHidden = true;
            }
            //Numeric columns should not be aggregated and float (double) data type should not be used
            if(column.DataType == DataType.Double || column.DataType == DataType.Decimal || column.DataType == DataType.Int64)
            {
                column.DisplayFolder = "Numeric";
                column.SummarizeBy = AggregateFunction.None;
                if(column.DataType == DataType.Double)
                {
                    column.DataType = DataType.Decimal;
                }
            }
            //Boolean data types into their own folder
            if(column.DataType == DataType.Boolean)
            {
                column.DisplayFolder = "Flags";
            }
            if(column.DataType == DataType.String)
            {
                column.DisplayFolder = "Attributes";
            }
            //Keys go into their own display folder, should not be aggregated and hidden.
            if(column.UsedInRelationships.Any())
            {
                column.DisplayFolder = "Key";
                column.SummarizeBy = AggregateFunction.None;
                column.isHidden = true;
            }
            //Date keys get their own folder and other dates go in attributes
            if( column.DataType == "DateTime" && column.Name != "DWCreatedDate")
            {
                if(column.UsedInRelationships.Any())
                {
                    column.DisplayFolder = "Key";
                    column.isHidden = true;
                }
                else{
                    column.DisplayFolder = "Dates";
                }
            }
        }
    }
}
```





# Display folders in Global and Local Measure Groups

> Global Measures  
Info   
Semantic Model Developer's DAX Measures

> Local Measures  
Info   
Local Report Level Measures

✓	□ i. Base
>	□ Budget
>	□ Forecast
>	□ Invoice
>	□ Order
>	□ ii. KPI Targets
>	□ iii. Comparison
✓	□ iv. Time Intelligence
>	□ i. MTD
>	□ ii. QTD
>	□ iii. YTD
>	□ iv. LY
>	□ v. LY %
>	□ vi. MoM
>	□ vii. QoQ
>	□ viii. YoY
>	□ v. Ratio
✓	□ vi. Report Layout
>	□ i. Last Refresh
>	□ ii. HEX Colors
>	□ iii. Dynamic Titles
>	□ iv. SVG's
>	□ v. Cascading Slicers (Alberto)

To create subfolder use \

Properties  
Display folder  
iv. Time Intelligence\iii. YTD

To order folders by ...

Number	Roman Numeral
1	i
2	ii
3	iii
4	iv
5	v
6	vi
7	vii
8	viii
9	ix
10	x





## Display folders

If you use consistent naming of measures

- ✓ Local Measures
- > Partitions
  - Σ CDF Revenue
  - Σ DEV Revenue vs. Budget
  - Σ DRL Drillthrough title
  - Σ IDC Arrow Up
  - Σ IDC Arrow Down
  - Σ IDX Revenue vs Budget
  - Σ LBL Label for Card Visual
  - Σ RLT Card Visual Revenue
  - Σ RLV Card Visual Revenue
  - Σ RLD Card Visual Revenue
  - Σ STL for Revenue
  - Σ TTL for Revenue
  - Σ TXT last refresh

It's easier to search for and structure in display folders

- ✓ Local Measures
  - > Partitions
  - ✓ Color
    - Conditional formatting (CDF)
  - ✓ Comparison
    - Deviation (DEV)
    - Indicator (IDC)
    - Index (IDX)
  - ✓ Dynamic Text
    - Drill text (DRL)
    - Label (LBL)
    - Subtitle (STL)
    - Title (TTL)
    - Text (TXT)
  - ✓ Card Visual
    - Reference Label Title (RLT)
    - Reference Label Value (RLV)
    - Reference Label Detail (RLD)



"There's a script for that"

```
// Prefix-to-Metadata Mapping Table
// -----
// Each entry contains:
//   - Prefix = Expected start of the measure name
//   - Group1 = Top-level display folder name
//   - Group2 = Second-level display folder name
//   - Desc = Human-readable description
// -----
var mappings = new [
  {
    Prefix: "CDF", Group1: "Color", Group2: "Conditional formatting (CDF)", Desc: "Conditional formatting" },
  { Prefix: "DEV", Group1: "Comparison", Group2: "Deviation (DEV)", Desc: "Deviation" },
  { Prefix: "DRL", Group1: "Dynamic Text", Group2: "Drill text (DRL)", Desc: "Drill text (enable/disable)" },
  { Prefix: "IDC", Group1: "Comparison", Group2: "Indicator (IDC)", Desc: "Indicator" },
  { Prefix: "IDX", Group1: "Comparison", Group2: "Index (IDX)", Desc: "Index" },
  { Prefix: "LBL", Group1: "Card Visual", Group2: "Label (LBL)", Desc: "Label" },
  { Prefix: "RLT", Group1: "Card Visual", Group2: "Reference Label Title (RLT)", Desc: "Reference Label Title" },
  { Prefix: "RLV", Group1: "Card Visual", Group2: "Reference Label Value (RLV)", Desc: "Reference Label Value" },
  { Prefix: "RLD", Group1: "Card Visual", Group2: "Reference Label Detail (RLD)", Desc: "Reference Label Detail" },
  { Prefix: "STL", Group1: "Dynamic Text", Group2: "Subtitle (STL)", Desc: "Subtitle" },
  { Prefix: "TTL", Group1: "Dynamic Text", Group2: "Title (TTL)", Desc: "Title" },
  { Prefix: "TXT", Group1: "Dynamic Text", Group2: "Text (TXT)", Desc: "Text" },
  { Prefix: "HEX", Group1: "Color", Group2: "HEX Color (HEX)", Desc: "HEX Color (HEX)" },
  { Prefix: "SVG", Group1: "Card Visual", Group2: "Scalable Vector Graphics (SVG)", Desc: "Scalable Vector Graphics (SVG)" }
];
// -----
// Main Processing Loop
// -----
// For each measure:
// 1. Check whether its name starts with a known prefix.
// 2. When found, set its display folder
//   - Set DisplayFolder = "Group1\Group2"
//   - Set Description
// -----
foreach (var measure in Model.AllMeasures)
{
  foreach (var map in mappings)
  {
    // Prefix matching supports "ABC_" or "ABC"
    if (measure.Name.StartsWith(map.Prefix + "_", StringComparison.OrdinalIgnoreCase) ||
        measure.Name.StartsWith(map.Prefix, StringComparison.OrdinalIgnoreCase))
    {
      measure.DisplayFolder = map.Group1 + "\\" + map.Group2;
      measure.Description = map.Desc;
    }
    break; // Stop searching once mapping is found
  }
}
```





## Table Groups in Tabular Editor 3



Semantic model

- > Calculation groups (1)
- > Cultures (1)
- > Measures (58)
- Perspectives (0)
- > Relationships (11)
- Roles (0)
- > Tables (16)
  - > Ex.00
  - > Exercises
  - > Time Intelligence
  - > Date
  - > Dimension Field Parameter
  - > Fact Population
  - > Fact Sales
  - > Fact Sales Budget
  - > Fact Sales MTD example
  - > Model Documentation
  - > Numeric Parameter
  - > Product



```
// ***** CREATE TABLE GROUPS *****
// Loop through all tables in the model:
foreach (var table in Model.Tables)
{
    if (table is CalculationGroupTable)
    {
        // Assign table group for calculation groups:
        table.TableGroup = "Calculation Groups";
    }
    else if (!table.UsedInRelationships.Any() && table.Measures.Any(m => m.IsVisible))
    {
        // Tables containing visible measures, but no relationships to other tables:
        table.TableGroup = "Measure Groups";
    }
    else if (table.UsedInRelationships.All(r => r.FromTable == table) && table.UsedInRelationships.Any())
    {
        // Tables exclusively on the "many" side of relationships:
        table.TableGroup = "Facts";
    }
    else if (!table.UsedInRelationships.Any() && table is CalculatedTable && !table.Measures.Any())
    {
        // Tables without any relationships that are calculated tables
        // and do not have measures:
        table.TableGroup = "Parameter Tables";
    }
    else if (table.UsedInRelationships.Any(r => r.ToTable == table))
    {
        // Tables on the "one" side of relationships:
        table.TableGroup = "Dimensions";
    }
    else
    {
        // All other tables:
        table.TableGroup = "Misc";
    }
}
```



Tables

- > Calculation Groups
  - > Units
  - > User Configuration
- > Dimensions
  - > Customer
  - > Date
  - > Invoice Document Type
  - > Order Document Type
  - > Order Status
  - > Product
  - > Region
- > Facts
  - > Budget
  - > Exchange Rate
  - > Forecast
  - > Invoices
  - > Orders
- > Measure Groups
  - > Global Measures
  - > Local Measures
  - > Table Group DAX
- > Misc
  - > Last Refresh
  - > Table Profile
  - > Table Schema
- > Parameter Tables





# Table Group in Power BI Desktop with INFO.VIEW

Semantic model

- > Calculation groups (1)
- > Cultures (1)
- > Measures (58)
- Perspectives (0)
- > Relationships (11)
- Roles (0)
- Tables (16) 
  - >  Ex.00
  - >  Exercises
  - >  Time Intelligence
  - >  Date
  - >  Dimension Field Parameter
  - >  Fact Population
  - >  Fact Sales
  - >  Fact Sales Budget
  - >  Fact Sales MTD example
  - >  Model Documentation
  - >  Numeric Parameter
  - >  Product
  - >  Region Country
  - >  Retailer



Description	Type	Order	Table Name
Classified as Measure Group	Measure Group	1	Exercises
Classified as Measure Group	Measure Group	1	Ex.00
Fact with many-side relationship	Fact	2	Fact Sales
Fact with many-side relationship	Fact	2	Fact Sales Budget
Fact with many-side relationship	Fact	2	Fact Population
Fact with many-side relationship	Fact	2	Fact Sales MTD example
Dimension with one-side relationship	Dimension	3	Sales Size
Dimension with one-side relationship	Dimension	3	Retailer
Dimension with one-side relationship	Dimension	3	Product
Dimension with one-side relationship	Dimension	3	Date
Dimension with one-side relationship	Dimension	3	Region Country
Dynamic calculation items	Calculation Group	4	Time Intelligence
Dynamic switch between measure or attributes	Field Parameters	5	Dimension Field Parameter
Dynamic slider for end users to select	Numeric Parameter	6	Parameter
Documentation with INFO.VIEW functions	Model Documentation	7	Model Documentation

## Semantic Model Information

### Table Groups in Power BI Desktop

Using INFO.VIEW DAX Functions & some creativity

Type	Number of tables
Calculation Group	1
User Configuration	1
Dimension	7
Customer	1
Date	1
Invoice Document Type	1
Order Document Type	1
Order Status	1
Product	1
Region	1
Fact	6
Budget	1
Customer	1
Exchange Rate	1
Forecast	1
Invoices	1
Orders	1
Measure Group	6
Global Measures	1
Last Refresh	1
Local Measures	1
Table Profile	1
Table Schema	1
Units	1

Total

21

### Rule Specifications

① Measure Group

An "empty" table to store my measures

If not below tables, no relationships, hidden columns, visible measure

② Fact

Fact table with many-side relationship \*-->

SELECTCOLUMNS ( INFO.VIEW:RELATIONSHIPS(), Table name from [FromTable] )

③ Dimension

Dimension table with one-side relationships 1-->

SELECTCOLUMNS ( INFO.VIEW:RELATIONSHIPS(), Table name from [ToTable] )

④ Calculation Group

Dynamic calculation items with SELECTEDMEASURE()

IF (INFO.VIEW:TABLES() [CalculationGroupPrecedence] >= 1)

⑤ Field Parameter

Dynamic switch between measure or attributes

IF (INFO.VIEW:TABLES() CONTAINSSTRING([Expression], "NAMEOF") )

⑥ Numeric Parameter

Dynamic slider for end users to select

IF (INFO.VIEW:TABLES() CONTAINSSTRING([Expression], "GENERATE") )

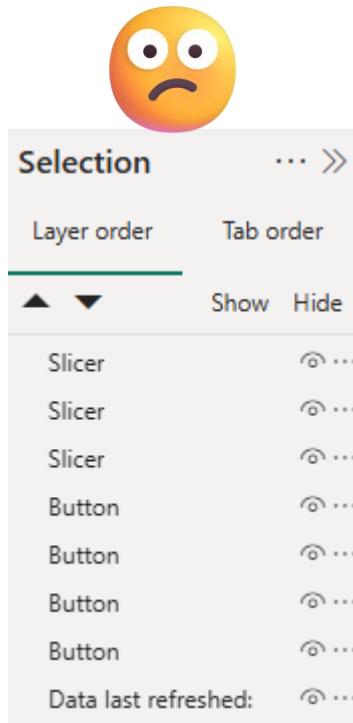
⑦ Model Documentation

Documentation with INFO.VIEW functions

IF (INFO.VIEW:TABLES() CONTAINSSTRING([Expression], "INFO.VIEW") )



## Report Visual Elements





## Report Visual Elements – especially important if using bookmarks

The screenshot shows the 'Selection' pane on the left and the 'Bookmarks' pane on the right. The 'Selection' pane includes 'Layer order' and 'Tab order' sections, with 'Slicer Pane' expanded to show 'Slicers', 'Slicer buttons', and 'Slicer shapes'. The 'Bookmarks' pane shows a list under 'Page 4: Location': 'Slicer Pane closed', 'Navigation extended', 'Navigation collapsed', and 'Slicer Pane opened' (which is highlighted). Below this is a 'Power BI User Adoption' report visualization with two bar charts showing User Engagement Rate (UER) by location.

Location	User Engagement Rate (UER) %	User Engagement Rate (UER) %
Stock	3%	91%
Austria	14%	86%
Japan	11%	89%
Copenhagen	32%	68%
Kuala Lumpur	22%	77%
Vietnam	100%	

Location	User Engagement Rate (UER) %	User Engagement Rate (UER) %
Copenhagen	32	67
Austria	16	83
(Blank)	11	88
Kuala Lumpur	6	94
Vietnam	5	

David Kofod Hanna: <https://www.linkedin.com/pulse/introducing-bookmark-killer-david-kofod-hanna-90quf/>



### 💡 My Standard:

#### For non-visual objects:

Use the visual object as Prefix + “-” and then a meaningful name for the object “Slicer – Year” or “Image – Logo”

Relevant for: Slicer, Textbox, Shapes, Images, Button, Page Navigator, Bookmark Navigator, etc.

For **visual elements** the name of the object is the Title in the visual, thus can't be prefixed with Bar chart, etc.

Group visual elements into e.g.

“Slicer Pane”,  
“Title, Navigation & Footer”,  
“KPI’s”  
“Graphs”.

Layer order from top left corner to bottom right corner  
(F or Z structure)

For use cases where only **Bookmarks** can solve the report user need and counting the total cost of ownership, use Selected Visuals and the Group of visuals.

Group Bookmarks together and name it prefix of page no and page name “**Page 4: Sales Region Analysis**”





# DAX Query View for testing

**DAX Query View**

DAX queries will be saved to your model. They won't be visible when published in the Power BI service. [Learn more](#).

**Run** | Update model with changes (0)

```

1 // Author - Hariharan Rajendran
2 // Name - Model_Summary_in_DAX_Query_View
3 // Version - 1.0
4 // Contact - https://www.linkedin.com/in/hariharan/
5 // This query will provide details about your model like property, value and comment to execute the DAX query to see the complete list
6 // Go to the properties
7 EVALUATE
8 VAR _tableCount = COUNTROWS(INFO.TABLES())
9 VAR _columnCount = COUNTROWS(INFO.COLUMNS())
10 VAR _calculatedColumnCount = COUNTROWS(FILTER(
11     INFO.COLUMNS(),
12     [Type] = 2
13 ))
14 VAR _Measures = COUNTROWS(INFO.MEASURES())
15 VAR _Relationships = COUNTROWS(INFO.RELATIONSHIPS())
16 VAR _Composite = IF(
17     COUNTROWS(UPPERCASE(
18         SELECTCOLUMNS(VTBL.TABLES),
19         [StorageMode]
20     )) > 3,
21     "yes",
22     "No"
23 )
24 VAR _daxTables = COUNTROWS(FILTER(

```

**Model Summary**

[Property]	[Value]	[Comment]
Tables	21	EVALUATE INFO.TABLES()
Columns	197	EVALUATE INFO.COLUMNS()
--CalculatedColumns	0	EVALUATE FILTER(INFO.CALCULATEDCOLUMNS())
--DirectColumns	197	EVALUATE FILTER(INFO.DIRECTCOLUMNS())
Measures	17	EVALUATE INFO.MEASURES()
Relationships	13	EVALUATE INFO.RELATIONSHIPS()
IsCompositeModel	No	EVALUATE INFO.VIEWTYPE()
DAXTables	2	EVALUATE FILTER(INFO.DAXTABLES())
IsPartitioned	No	EVALUATE FILTER(INFO.ISPARTITIONED())
Perspectives	0	EVALUATE FILTER(INFO.PERSPECTIVES())
Calculation Groups	2	EVALUATE FILTER(INFO.CALCULATIONGROUPS())
Roles	0	EVALUATE FILTER(INFO.ROLES())
PBIDesktopVersion	2.142.1277.0 (25.04)+014fde45...	EVALUATE FILTER(INFO.PBIDESKTOPVERSION())

## Model Issues

[Property]	[Value]	[Comment]	[Status]
1 Is Partition Required?	Yes	EVALUATE VAR _dt_partition	Fail
2 Dedicated Date Table	No	EVALUATE FILTER(INFO.DEDICATED_DATE_TABLE)	Fail
3 Is there any Redundant Columns	Yes	EVALUATE FILTER(GROUPBY(COLUMNS))	Fail
4 Are descriptions added?	No	EVALUATE FILTER(INFO.DESCRIPTIONS)	Fail
5 Calculated Columns		EVALUATE FILTER(INFO.CALCULATED_COLUMNS)	Pass
6 Local Date Tables (Auto Time Intelligence)	0	EVALUATE FILTER(INFO.LOCAL_DATE_TABLES)	Pass
7 In Active Relationships	0	EVALUATE FILTER(INFO.IN_ACTIVE_RELATIONSHIPS)	Pass
8 Bi-directional Relationships	0	EVALUATE FILTER(INFO.BI_DIRECTIONAL_RELATIONSHIPS)	Pass
9 Many to Many Relationships	0	EVALUATE FILTER(INFO.MANY_TO_MANY_RELATIONSHIPS)	Pass

## Measure Dependency

[MeasureName]	[Expression]	[SourceMeasure]	[SourceExpression]	[Type]	[ReferenceTable]	[ReferenceObject]
1 Net Invoice Value YTD	CALCULATE([Sum of Net Invoice Value], DATESYTD('Invoices'))	Sum of Net Invoice Value	SUM('Invoices'[Net Invoice Value])	TABLE	Invoices	Invoices
2 Net Invoice Value YTD	CALCULATE([Sum of Net Invoice Value], DATESYTD('Invoices'))	Sum of Net Invoice Value	SUM('Invoices'[Net Invoice Value])	COLUMN	Invoices	Net Invoice Value
3 Net Invoice Value MTD	CALCULATE([Sum of Net Invoice Value], DATESMTD('Invoices'))	Sum of Net Invoice Value	SUM('Invoices'[Net Invoice Value])	TABLE	Invoices	Invoices
4 Net Invoice Value MTD	CALCULATE([Sum of Net Invoice Value], DATESMTD('Invoices'))	Sum of Net Invoice Value	SUM('Invoices'[Net Invoice Value])	COLUMN	Invoices	Net Invoice Value
5 Net Invoice Value QTD	CALCULATE([Sum of Net Invoice Value], DATESQTD('Invoices'))	Sum of Net Invoice Value	SUM('Invoices'[Net Invoice Value])	TABLE	Invoices	Invoices
6 Net Invoice Value QTD	CALCULATE([Sum of Net Invoice Value], DATESQTD('Invoices'))	Sum of Net Invoice Value	SUM('Invoices'[Net Invoice Value])	COLUMN	Invoices	Net Invoice Value
7 Number of tables	COUNTROWS(Table Group DAX)			CALC_TABLE	Table Group DAX	Table Group DAX
8 Sum of Total Budget	SUM('Budget'[Total Budget])			TABLE	Budget	Budget
9 Sum of Total Budget	SUM('Budget'[Total Budget])			COLUMN	Budget	Total Budget
10 Sum of Net Order Quantities	SUM('Orders'[Net Order Quantities])			TABLE	Orders	Orders

## VertiPaq Memory Size

[Size in MB]
1017.25

Visible in .PBIP and for easy copy/paste

System.SemanticModel > DAXQueries >

<b>.pbi</b>	<b>.pbi</b>	File folder
<b>DAXQueries</b>	<b>Measure Dependency</b>	DAX Query File
<b>.platform</b>	<b>Model Issues</b>	DAX Query File
<b>definition.pbism</b>	<b>Model Summary</b>	DAX Query File
<b>diagramLayout</b>	<b>VertiPaq Columns</b>	DAX Query File
<b>model</b>	<b>VertiPaq Memory Size</b>	DAX Query File
	<b>VertiPaq Partitions</b>	DAX Query File
	<b>VertiPaq Relationships</b>	DAX Query File
	<b>VertiPaq Table</b>	DAX Query File





## Run DAX Query and save as delta table in Lakehouse 😳

### DAX Query x Semantic Link Labs x DAX INFO Functions

Purpose of notebook is to collect, analyze and store semantic model INFO functions and DAX Queries in a Lakehouse and share with semantic model or report owners or "crazy solution documenters"

This can be run in a Python notebook

#### Credit to following

- Michael Kovalsky - Semantic Link Labs
- List of all DAX INFO functions
- Fourmoo on Python Notebook advantages vs spark notebooks
- Hariharan Rajendran's DAX Query on Models Issues and Model Summary
- Power BI Tips on Save to Delta table with Python
- Reza Rad's take on Power BI model analysis using DAX INFO functions
- David Kofod Hanna's GitHub on Power BI Documentation System

If you feel frustrated like me, not all DAX Info functions can be used like INFO.VIEW functions by adding a calculated DAX table.

Then we have been forced to used DAX Query View.

1 Storage Table Column = `INFO.STORAGECOLUMN()`

The function 'INFO.STORAGECOLUMN()' cannot be used in the expression of a measure, calculated column, or calculated table defined in the model.

Save your frustration, because I found a way to store DAX Query results back to a lakehouse 😳

#### Manual DAX Query

```
In [ ]: # Define DAX Query to execute  
dax_query = """  
    EVALUATE  
    INFO.TABLES()  
    """  
  
In [ ]: # Run DAX Query against Semantic Model  
df_result_manual = fabric.evaluate_dax(  
    dataset=SemanticModelName,  
    dax_string=dax_query,  
    workspace=WorkspaceName  
)  
  
# Display the result  
display(df_result_manual)  
  
In [ ]: labs.save_as_delta_table(  
    dataframe=df_result_manual,  
    delta_table_name="daxquery",  
    write_mode="overwrite", # or "append" based on your requirement  
    merge_schema=False, # Set to True if you want to merge schema  
    schema=None, # Provide schema if needed  
    lakehouse=LakehouseName,  
    workspace=None  
)
```

#### Automated DAX Query loop and save to delta table

```
In [ ]: # Defining DF of all INFO DAX Functions and descriptions  
  
dataDAX = [  
    ("INFO.ALTERNATEOFDEFINITIONS()", ""),  
    ("INFO.ANNOTATIONS()", "Returns a list of all annotations in t"),  
    ("INFO.ATTRIBUTEHIERARCHIES()", "Represents the TMSHEMA_ATTR"),  
    ("INFO.ATTRIBUTEHIERARCHYSTORAGES()", ""),  
    ...]
```





# JSON Report Theme – For the Nerdy Data Storytellers

Home / JSON to Power BI

## JSON Report Theme to Power BI

Power BI custom report themes provide granular control over many aspects of a report theme. It's crucial to build a solid, documented and adopted in an organization for consistency and efficiency.

### Theme colors in Power BI

Power BI's data color system lets you define up to 8 colors that map directly to your data points. Along with these colors, Power BI automatically generates 5 different shades of each color that can be dynamically applied throughout the file.

### KPI & Divergent Colors

These properties set the status colors used by the waterfall chart and the KPI visual or set the various gradient colors in the conditional formatting dialog box.

Use them to refer in conditional formatting measures where you can simply in the measure, instead of hard-coding a HEX, use "good", "neutral", "bad", "minimum" or "maximum"

### Structural Colors

These color classes set the structural colors for elements in the report, such as axis gridlines, highlight colors, and background colors for visual elements.

Active Report Theme: DKH Self-Service Report Theme

Try yourself! Download at [GitHub](#)

**Theme Colors 1-8**  
For data visualization

Theme	HEX	ID
Blue	#367cff	1
Orange	#fa8100	2
Cyan	#6cc6c3	3
Purple	#aa77dd	4
Red	#d14576	5
Brown	#b26d6d	6
Dark Blue	#8b9fd4	7
Yellow	#eae5c9	8

Remember to think of brand guidelines together with enough categorical values along side thinking contrast and call-to-action and WCAG and color-blindness.

**KPI Colors**  
For KPI performance status context

KPI	HEX	Property
Green	#37a78f	Good
Grey	#f2f2f2	Neutral
Red	#a74d37	Bad

**Divergent Min & Max Colors**  
For divergent heat map

Divergent	HEX	Property
Orange	#fa8100	Minimum
Grey	#f2f2f2	Center
Blue	#367cff	Maximum
Grey	Null	

Example measure diff Color =  
IF(  
    [Revenue LY %] > 0,  
    "good",  
    "bad"  
)

**Structural Colors**  
Non-data ink for colors except the data colors

Structural	HEX	Property
Background	#485257	Background
BackgroundLight	#F1F3F4	BackgroundLight
BackgroundNeutral	#F1F5F4	BackgroundNeutral
Foreground	#485257	Foreground
ForegroundNeutralSecondary	#485257	ForegroundNeutralSecondary
ForegroundNeutralTertiary	#606E74	ForegroundNeutralTertiary
Hyperlink	#e6e6e6	Hyperlink
TableAccent	#e6e6e6	TableAccent
Visited Hyperlink		Visited Hyperlink

Any formatting elements that aren't included in the JSON file revert to their default values and settings.

Visual Styles  
15 of 52 visual styles defined in theme

Image	VisualStyleAttribute	Defined in JSON Theme	Type
textbox	textbox	textbox	Object
tableEx	tableEx	tableEx	Visual
slicer	slicer	slicer	Visual
shape	shape	shape	Object
report	report	report	Object
pivotTable	pivotTable	pivotTable	Visual
pageNavigator	pageNavigator	pageNavigator	Object
page	page	page	Object
multiRowCard	multiRowCard	multiRowCard	Visual
kpi	kpi	kpi	Visual
image	image	image	Object
cardVisual	cardVisual	cardVisual	Visual
bookmarkNavigator	bookmarkNavigator	bookmarkNavigator	Object
advancedSlicerVisual	advancedSlicerVisual	advancedSlicerVisual	Visual
actionButton	actionButton	actionButton	Object
aiNarratives	aiNarratives	aiNarratives	Visual
areaChart	areaChart	areaChart	Visual
azureMap	azureMap	azureMap	Visual
barChart	barChart	barChart	Visual
card	card	card	Visual
clusteredBarChart	clusteredBarChart	clusteredBarChart	Visual
clusteredColumnChart	clusteredColumnChart	clusteredColumnChart	Visual
columnChart	columnChart	columnChart	Visual
decompositionTreeVisual	decompositionTreeVisual	decompositionTreeVisual	Visual
donutChart	donutChart	donutChart	Visual
filledMap	filledMap	filledMap	Visual
filter	filter	filter	Object
funnel	funnel	funnel	Visual
gauge	gauge	gauge	Visual
group	group	group	Object
hundredPercentStackedAreaChart	hundredPercentStackedAreaChart	hundredPercentStackedAreaChart	Visual
hundredPercentStackedBarChart	hundredPercentStackedBarChart	hundredPercentStackedBarChart	Visual
hundredPercentStackedColumnChart	hundredPercentStackedColumnChart	hundredPercentStackedColumnChart	Visual

Visual Styles

Be aware of the new preview visuals as the report theme schema can change more often on these.

**Recommendation:**  
Set Visual Styles for frequently used visuals like

- Card New Visual
- Matrix
- Table
- Line chart
- Column chart
- Bar chart
- Slicer
- Action button
- Page Navigator

Attribute Property Value  
Filter by keyword All Filter by keyword

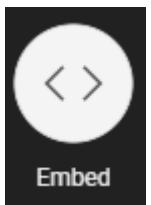
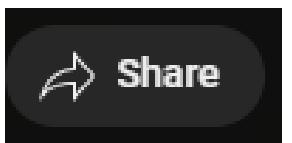
**Detail Properties of Visual Styles in Your JSON Theme**  
Click on a Visual Style to filter

Image	Name	Attribute	Property	Value
bookmarkNa	sid	fill	selected	
vigator		shape	default	
		Text	selected	
		border	#AEBBD	
		bottom	*	0
		color	Border	#AEBBD
		fillColor	fill	#F1F3F4
		fontColor	Text	#30373A
		fontFamily	Text	Segoe UI Semibold
		fontSize	Text	10,5
		left	*	0
		radius	Border	8
		right	*	0
		roundEdge	shape	8
		show	Border	FALSE
		fill	TRUE	
		Outline	FALSE	
		tileShape	shape	rectangle





## Embed videos



Medieval helpdesk with English subtitles

Watch on YouTube It's closed and everything's save inside it.

Watch Later Share

Embed Video

```
<iframe width="560" height="315" src="https://www.youtube.com/embed/pQHX-SjgQvQ?si=n2JMTj9g4HiOfptp" title="YouTube video player." frameborder="0" allow="accelerometer; autoplay; clipboard-write; encrypted-media; gyroscope; picture-in-picture; web-share" referrerpolicy="strict-origin-when-cross-origin" allowfullscreen></iframe>
```

Start at 0:01

EMBED OPTIONS

Copy

[Helpdesk support back in the day of the middle age](#)



### New link

To include a link to an item or website, enter the URL.

\* required fields

#### Link name \*

Medieval Help Desk

#### URL address \*

<https://www.youtube.com/embed/pQHX-SjgQvQ?si=n2JMTj9g4HiOfptp>

Include https:// at the beginning of the URL

#### Link behavior

Open link in new browser tab

Embed linked content into app

Create

Cancel





## Embed PowerPoint, Excel, PDF

MsBIP - Power BI Documentation System - David Kofod Hanna -

File Home Insert Draw Design Transitions Animations Slides

- New
- Open
- Share
- Create a copy
- Embed this presentation in your blog or website...
- Print
- Rename
- Move file
- Version history
- Info

Embed this presentation in your blog or website...

- Share with people
- Copy link to this presentation
- Embed this presentation
- Manage Access

### Embed

Preview

Dimensions

Width (px): 476

Height (px): 288

Interaction

Use default autoadvance settings from the file

Embed Code

```
<iframe src="https://twodaygroup-my.sharepoint.com/personal/david_hanna_twoday_com/_layouts/15/Doc.aspx?sourcedoc={d687580d-e075-431a-bf09-...>
```

This file will only be accessible to people with permissions.

Copy Close





# Design Documents – Samples

alexbadiu-insightsinmotion / **PBI-Documentation**

- Design Document - Sample Fragment 01 - General and Scope - V0.1.docx
- Design Document - Sample Fragment 02 - Workflow Issues and Business Rules - V0....
- Design Document - Sample Fragment 03 - Data - V0.3.docx
- Design Document - Sample Fragment 04 - Reports - V0.4.docx
- Design Document - Sample Fragment 05 - Validation - V0.5.docx
- Design Document - Sample Fragment 06 - Deployment - V0.6.docx
- Design Document - Sample Validation Spreadsheet - V0.5.xlsx

*Power BI Documentation – Design Document  
Sample Fragment 04 – Reports*

---

### Table of Contents

1.	Introduction .....	3
2.	Scope of Work.....	3
3.	Workflow .....	3
4.	Issues .....	3
5.	Business Rules.....	3
6.	Data.....	3
7.	Reports.....	3
7.1.	Common .....	3
7.1.1.	Theme.....	3
7.1.2.	Filters .....	4
7.1.3.	Navigation.....	5
7.1.4.	Page Header.....	6
7.1.5.	Page Footer.....	7
7.1.6.	Slicers.....	7
7.2.	Specific .....	8
7.2.1.	Semantic Model (specific).....	8
7.2.2.	AR01 – All Invoices.....	9
7.2.3.	AR02 – Current Invoices .....	10
7.2.4.	AR03 – Upcoming Invoices .....	10
7.2.5.	AR04 – Historical Invoices .....	11

The items that will be included are described below.			
ID	Name (Category / Subcategory)	Design / Selected / Unselected / Hover	
N-1	Invoices	DESIGN:	<ul style="list-style-type: none"> <li>Type=button</li> <li>Shape=any, with border=off</li> <li>Action=page navigation (subcategory 1)</li> </ul>
		DEFAULT (selected):	<ul style="list-style-type: none"> <li>Font=Segoe UI, white, 10 pt</li> <li>Background=dark blue</li> <li>Navigation=page, Invoices-All</li> </ul>
		DEFAULT (unselected):	<ul style="list-style-type: none"> <li>Font=Segoe UI, medium grey, 10 pt</li> <li>Background=medium blue</li> <li>Navigation=page, Invoices-All</li> </ul>
		HOVER:	<ul style="list-style-type: none"> <li>Font=Segoe UI, dark grey, 11 pt</li> <li>Background=medium grey</li> <li>Navigation=page, Invoices-All</li> </ul>
N-2	Invoices / All	(same as N-1 above)	
N-3	Invoices / Current	(same as N-1 above, but with adjusted page navigation and selected and unselected defaults reversed)	
N-4	Invoices / Upcoming	(same as N-1 above, but with adjusted page navigation and selected and unselected defaults reversed)	
N-5	Invoices / Historical	(same as N-1 above, but with adjusted page navigation and selected and unselected defaults reversed)	

7.1.6. Slicers		
The slicers that will be included on each page of each all report are described below.		
ID	Slicer	Design Features / Data / Notes
n/a	General	<p>Title=Segoe UI Semibold, 10 pt Values=Segoe UI, 8 pt Style=dropdown Selection=multi-select; CTRL off; Select All off Header icons=off Search box=enabled</p>
S-1	Fiscal year	Data=Dates[Fiscal Year] Notes=search box unavailable as numeric data
S-2	Fiscal quarter	Data=Dates[Fiscal Quarter] Notes=search box unavailable as numeric data
S-3	Date range	Type=between Slider=on, responsive off Data=Dates[Date]
S-4	Province	Type=text Data=Countries[Province]





## Embed Notebooks for technical documentation

```
1 #Author and MVP: Sandeep Pawar  
2 https://fabric.guru/measure-maze-visualizing-measure-dependencies-using-semantic-link-network-analysis
```

+ Code + Markdown

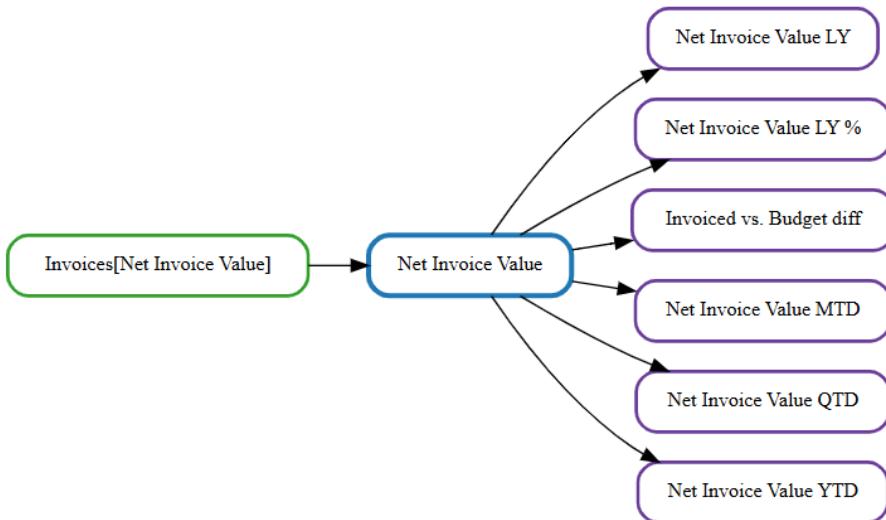
```
1 #Install Measure Maze in a Fabric Notebook  
2 !pip install https://github.com/pawarbi/MeasureMaze/raw/main/measuremaze-0.0.1-py3-none-any.whl --q
```

✓

```
1 #Change the layout  
2 all_objects.plot('Net Invoice Value', layout='horizontal')
```

[7]

✓ - Command executed in 375 ms by David Kofod Hanna on 1:02:35 PM, 8/12/25

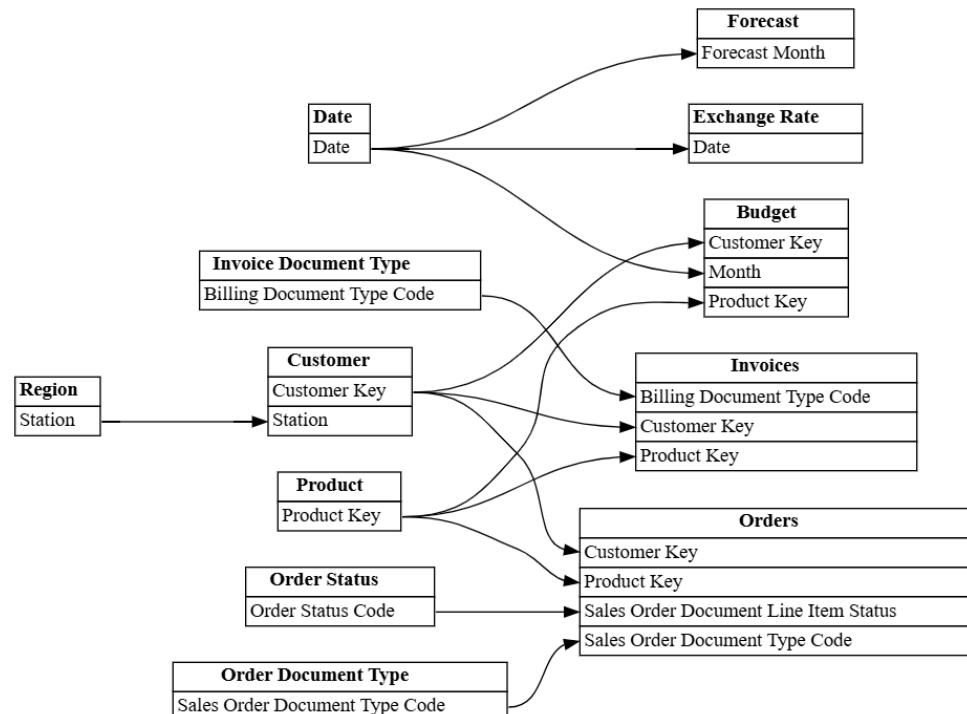


```
1 #Semantic Link can also be used to plot relationships between tables.
```

```
2  
3 from sempy import fabric  
4 from sempy.relationships import plot_relationship_metadata  
5  
6  
7 relationships = fabric.list_relationships(workspace=ws, dataset=ds)  
8 plot_relationship_metadata(relationships)  
9
```

[18]

✓ - Command executed in 1 sec by David Kofod Hanna on 1:02:59 PM, 8/12/25





## T HTML Documentation script – Kurt Buhler x Tabular Editor

 **Documentation System Devon**

**DATA GOBLINS SEMANTIC MODEL CHECKLIST**

Enhanced with Tabular Editor 3 • Generated: August 12, 2025

---

**Model: Model**

 1,07 GB Size (sum of table sizes)	 21 Tables	 196 Columns	 31 Measures
--	--	--	--

**Top 5 Tables by Size**

Invoices	499,40 MB
Orders	465,53 MB
Budget	91,92 MB
Product	24,96 MB
Model Documentation	5,09 MB

**Top 5 Columns by Size**

Orders[Net Order Value]	233,52 MB
Budget[Total Budget]	87,20 MB
Invoices[Net Invoice Value]	68,87 MB
Invoices[Delivery Cost]	59,22 MB
Invoices[Net Invoice COGS]	52,48 MB

This checklist has been automatically analyzed based on your current semantic model structure and contains pre-checked items where applicable.

**How to Use This Enhanced Checklist**

- **Print as PDF:** Use Ctrl+P and select 'Save as PDF' to create a form-fillable document

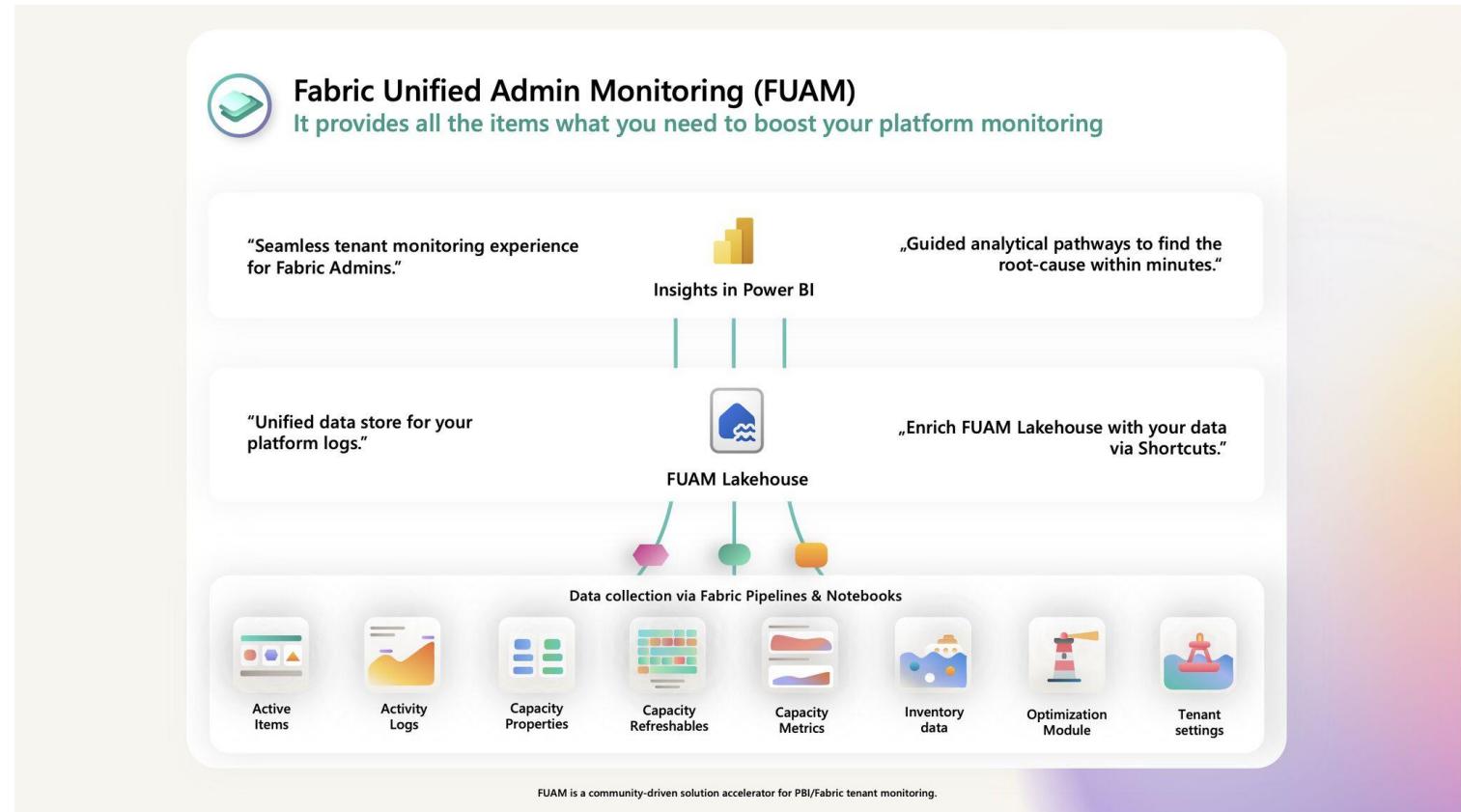
Kurt Buhler & Tabular Editor: <https://tabulareditor.com/blog/power-bi-semantic-model-checklist>





## All into a Lakehouse + GA of PBIP and PBIR in 2026

- Semantic Link Labs
- Dataflow Gen2
- DAX Queries
- FUAM – Fabric Unified Admin Monitoring
- Copilot – Now with F2 SKU
- Copilot to be trained on Model.Bim file and .PBIR
- .PBIR meta data
- Report Best Practice Analyzer
- Best Practice Analyzer Semantic Model
- VertiPaq Analyzer
- Usage Metrics for activity logs
- Data Agents and Copilot on top of your data in Fabric





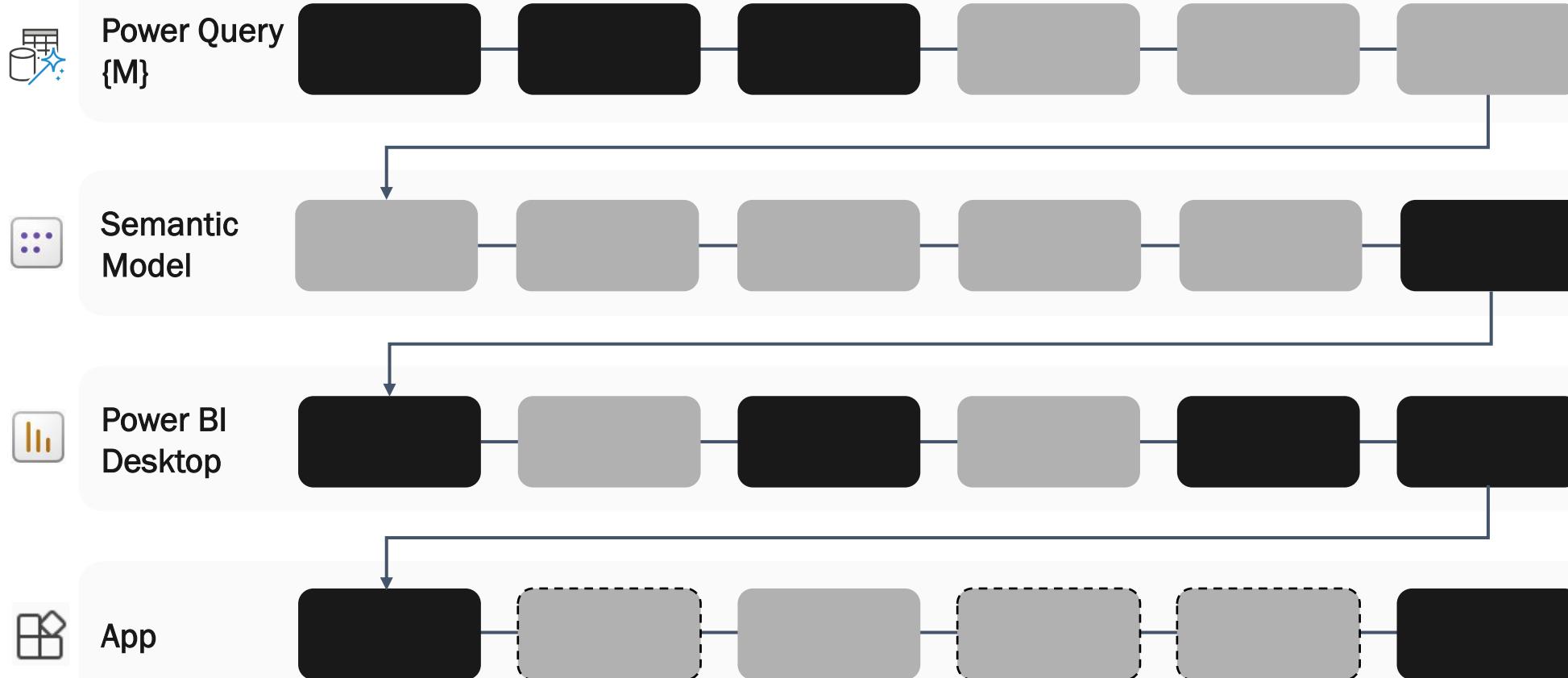
✍️ Please write my  
documentation





# Power BI Documentation System – from Model to App

Automated or Script      Manual      Requires Fabric SKU



**Purpose for docs:**  
*Clear, Concise,  
Complete, Consistent,  
Correct, Consumable*

## Documentation and transparency from model to app

From report consumers  
to report developers to  
semantic model  
developers and data  
ingest

## Best Practice Analyzer Rules to highlight violations and severity of best practices

## Tabular Editor 2/3 for Semantic Model and DAX scripts

# Semantic Link Labs in Fabric Notebooks for automation like never before

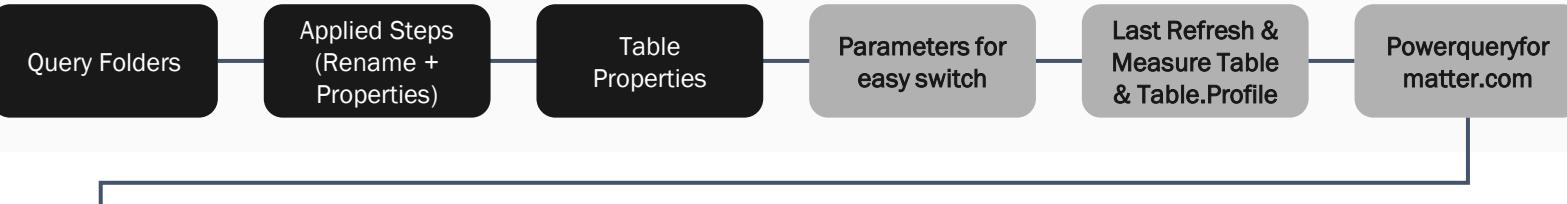




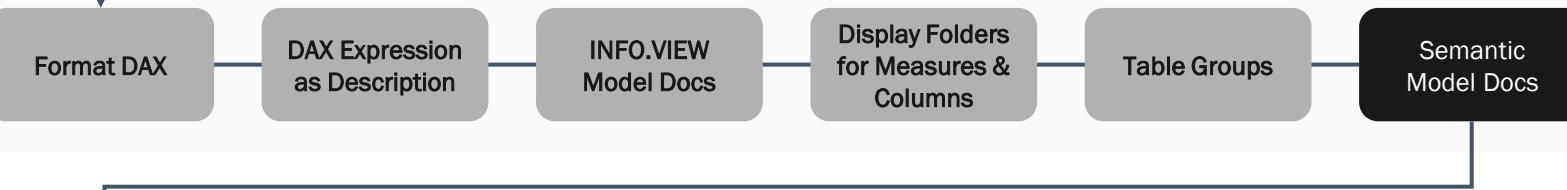
# Power BI Documentation System – from Model to App

Automated or Script    Manual    Requires Fabric SKU

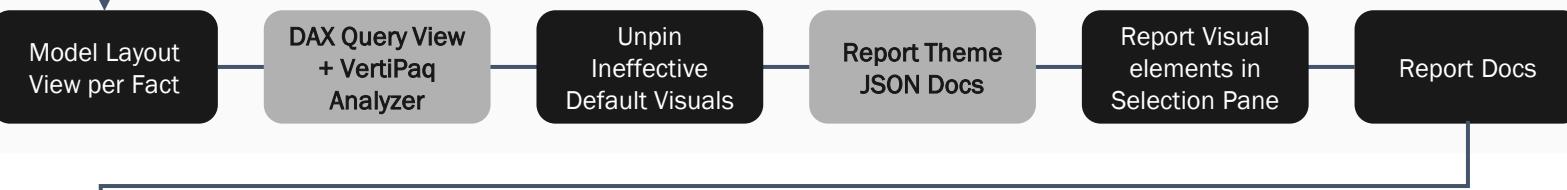
## Power Query {M}



## Semantic Model



## Power BI Desktop



## App



Purpose for docs:

Clear, Concise,  
Complete, Consistent,  
Correct, Consumable

Documentation and  
transparency from  
model to app

From report consumers  
to report developers to  
semantic model  
developers and data  
ingest

Best Practice Analyzer  
Rules to highlight  
violations and severity  
of best practices

Tabular Editor 2/3 for  
Semantic Model and  
DAX scripts

Semantic Link Labs in  
Fabric Notebooks for  
automation like never  
before



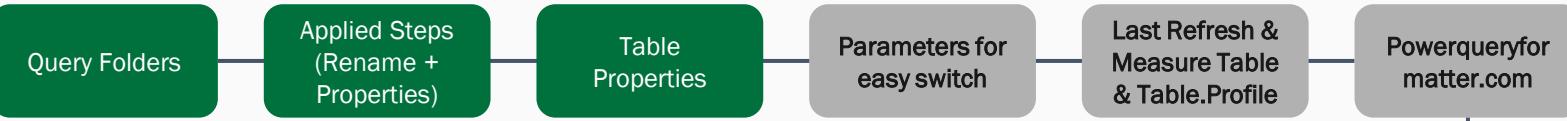


# Power BI Documentation System – from Model to App

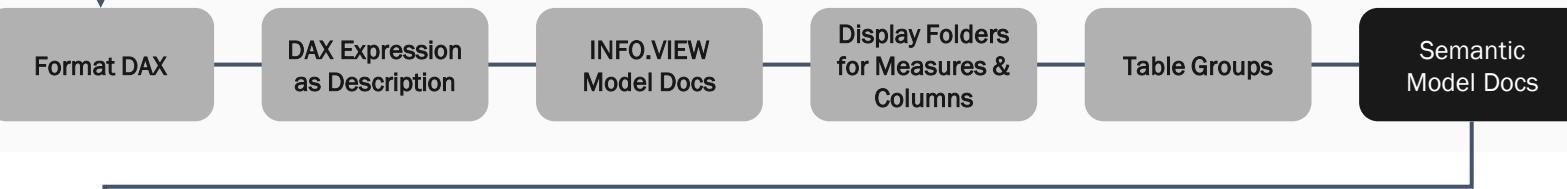
Automated or Script    Manual    Requires Fabric SKU    GitHub Copilot with MCP server Agent Mode and .PBIP



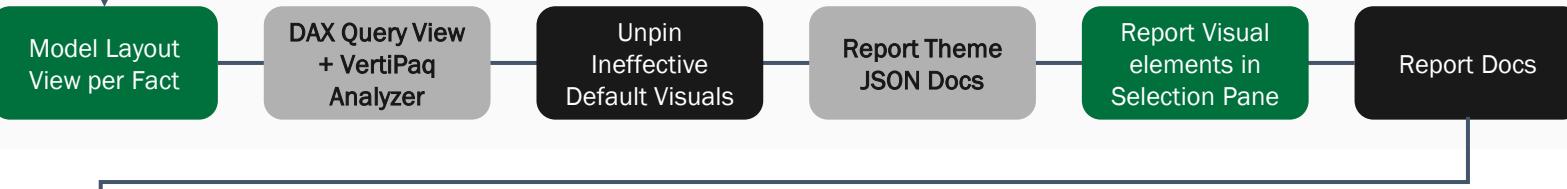
## Power Query {M}



## Semantic Model



## Power BI Desktop



## App



Purpose for docs:

Clear, Concise,  
Complete, Consistent,  
Correct, Consumable

Documentation and  
transparency from  
model to app

From report consumers  
to report developers to  
semantic model  
developers and data  
ingest

Best Practice Analyzer  
Rules to highlight  
violations and severity  
of best practices

Tabular Editor 2/3 for  
Semantic Model and  
DAX scripts

Semantic Link Labs in  
Fabric Notebooks for  
automation like never  
before





*“You do not rise to the level of  
your goals, you fall to  
the level of your systems.”*

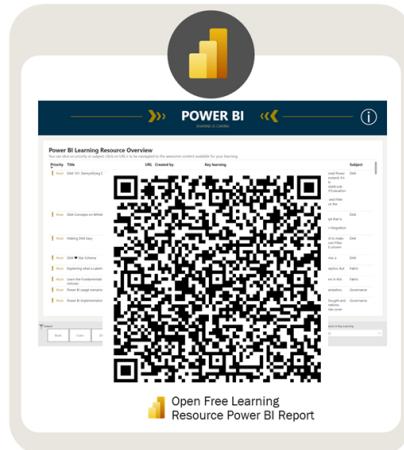
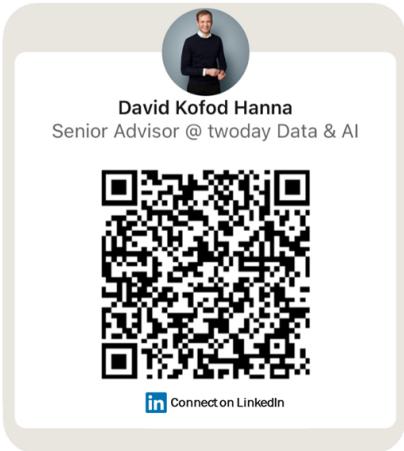
- James Clear, Atomic Habits





# Thank you

- Connect to me at:



- Stay online for my live Q&A sessions

