

Power BI Gotchas – Dan's Dirty Dozen

Dan English

Principal Program Manager @ Microsoft |
Microsoft Fabric Customer Advisory Team (CAT)



slides available @ <https://bit.ly/m365tc25powerbigotchas>



Please thank our sponsors



Submit session feedback



Dan English (he/him)

Principal Program Manager | Fabric CAT

Professional:

LinkedIn - [/in/denglishbi](https://in/denglishbi)

Twitter (X) - [@denglishbi](https://twitter.com/denglishbi)

Blog - <https://denglishbi.wordpress.com/>

Author, Speaker, former 8x Microsoft Data Platform MVP

Helped run following communities (# years)

PASSMN (4),

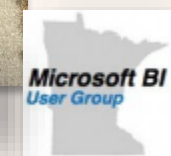
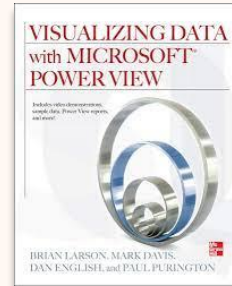
MN Microsoft BI User Group (6),

PASS Business Analytics Virtual Group (5)

Personal:

Running 🏃, MN Vikings 🏈, MN Timberwolves 🏀,

Dogs 🐕, Movies 🎬



Agenda

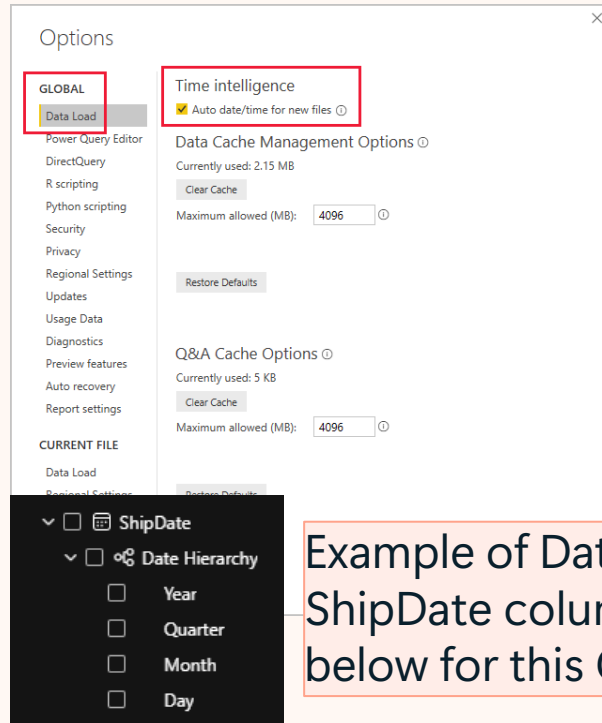
- Power BI Gotchas
 - Data Model (6)
 - DAX (3)
 - Report Design (3)
- Resources
- Wrap up and Q&A
- Session and Event Feedback

Power BI Gotchas – Dan's Dirty Dozen

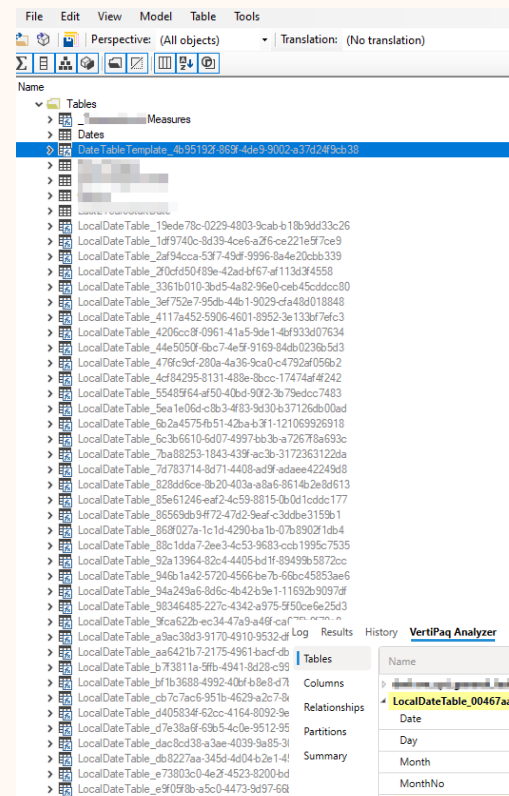
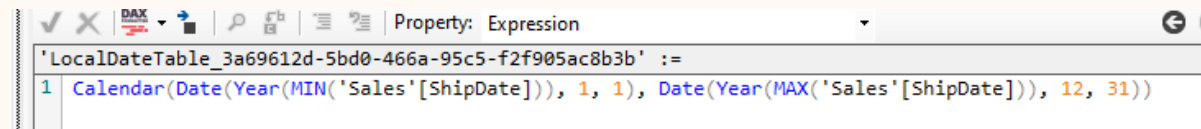


#1 Data Model

Auto Date/Time Tables



Example of Date Hierarchy built for ShipDate column and DAX logic below for this Calculated Table.



In this example the semantic model had 54 tables

- 43 auto date/time tables
- 8 dimension tables
- 1 fact table
- 2 hidden reference tables

Another example below where 2nd largest table is auto date/time table and has over 3.5M rows

Name	Cardinality	Total Size ↓	Data	Dictionary	Hier Size	Encoding	Data Type	RI Violations	User Hier Size	Rel Size	% Table	% DB
LocalDateTable_00467aab-2a1e-4e63-b28d-1ae60bf267ee	3,578,646	272,870,488	21,039,888	163,199,016	28,708,160	HASH	DateTime	1	59,923,424	0	96.64%	0.32%
Day	31	2,387,596	2,385,896	1,444	256	HASH	Int64	-	-	-	1.12%	0.00%
Month	12	1,806,932	1,789,456	17,364	112	HASH	String	-	-	-	0.85%	0.00%
MonthNo	12	1,790,936	1,789,456	1,368	112	HASH	Int64	-	-	-	0.84%	0.00%
Year	9,798	511,112	131,080	301,632	78,400	HASH	Int64	-	-	-	0.24%	0.00%
Quarter	4	331,800	314,576	17,176	48	HASH	String	-	-	-	0.16%	0.00%
QuarterNo	4	315,960	314,576	1,336	48	HASH	Int64	-	-	-	0.15%	0.00%
LocalDateTable_0a4178e4-0be8-4f47-8af2-462018e99aff	2,958,464	182,688,740	17,355,144	92,061,820	23,733,120	Many	-	1	49,538,656	0	0.28%	0.37%
LocalDateTable_9fa318af-2ce1-4636-aa15-8b3cfb4c637b	73,413	5,460,112	337,240	3,304,056	589,504	Many	-	1	1,229,312	0	0.01%	0.00%
LocalDateTable_915cb74a-9fe1-4fcc-b181-8020ba1a4960	7,305	602,208	29,768	390,904	59,200	Many	-	1	122,336	0	0.00%	0.00%
LocalDateTable_9ef6950f-50fc-46b1-957b-c6e2af61da6b	2,922	231,436	11,376	147,052	24,048	Many	-	1	48,960	0	0.00%	0.00%
LocalDateTable_4755a228-b430-4eeb-b4e0-cb67f1e1076f	1,461	131,916	6,096	89,020	12,320	Many	-	1	24,480	0	0.00%	0.00%
LocalDateTable_01cfb8ed-0fa3-4067-9e0c-f9ed9e88d85d	1,096	115,300	4,648	82,892	9,392	Many	-	1	18,368	0	0.00%	0.00%
LocalDateTable_9180a0a0-66eb-44aa-83b9-fcb12fed9430	1,096	115,300	4,648	82,892	9,392	Many	-	1	18,368	0	0.00%	0.00%
LocalDateTable_50d0fe24-b650-4ca8-8978-fc84ba395616	1,096	115,300	4,648	82,892	9,392	Many	-	1	18,368	0	0.00%	0.00%
DateTableTemplate_4065d022-de2b-441b-89f1-4f50871253f7	0	4,416	1,088	3,328	0	HASH	-	-	0	0	0.00%	0.00%

[Auto date/time in Power BI Desktop - Power BI | Microsoft Learn](#)

[Auto date/time guidance in Power BI Desktop - Power BI | Microsoft Learn](#)

#2 Data Model

Columns not needed

Table columns have two main purposes

- Reporting : report design, filtering, grouping, summarizing data
- Model structure : relationships, calculations, security roles, formatting

VertiPaq Analyzer													
Log Results History													
Tables													
Columns													
Relationships													
Name	Cardinality	Total Size ↓	Data	Dictionary	Hier Size	Encoding	Data Type				% Table	% DB	Co
General Ledger	165,705,797	19,242,818,593	2,152,823,664	15,555,244,641	1,531,572,144	Many	-	-	0 3...			94.13%	
General Ledger ID	165,705,797	16,272,894,877	655,891,304	14,291,357,189	1,325,646,384	HASH	String	-	-	-	84.58%	79.60%	

In the above example

- Fact table has a unique key ID column
- takes up 15.2GB (79.6% of the entire model size)
- column is not needed for any calculations or reporting requirements

Other common things are dimension attributes duplicated in Fact tables, ETL date/time columns, source natural keys

#3 Data Model

Data Types

Watch out for “Floating Point / Decimal Number (Double)” Data Type columns

- Use “Integer” or “Fixed Decimal” instead; can see between 10 to 60% in size reduction

Before: table size 22.7GB

Log	Results	History	VertiPaq Analyzer											
Tables	Name	Cardinality	Total Size ↓	Data	Dictionary	Hier Size	Encoding	Data Type	R.	Us...	R..	%.	% DB	C
Columns	f_data	257,127,352	24,376,942,331	7,436,209,216	14,599,737,331	2,340,975,488	Many	-	-	0	20,...		99.69%	

After: table size 8.1GB

Tables	Columns	Relationships	Partitions	Summary					
Name	Cardinality	Table Size	Col Size	Data	Dictionary	Hier Size	Encoding	Data Type	RI Violations
▶ f_data	257,054,445	8,690,467,0...	8,690,446,7...	6,682,385,0...	1,938,455,1...	69,606,528	Many	-	

Reduction: 14.6GB (64% size reduction)

Other data types to watch out for would be String and Date Time

- With String you can see larger dictionaries and hierarchies
- With Date Time watch out for precision if time is included

#4 Data Model

Bi-Directional and Many-to-Many relationships

May cause performance degradation

- ensure if they are necessary
- and they are working as intended

Row Labels	Relationship Type
Fact [Table]	
'Fact [Table] [Key] ==<=>1 'Dim [Table] [Key]	M:1 (Bid.)
'Fact [Table] [Key] ==<1 'Dim [Table] [Key]	M:1
'Fact [Table] [Key] ==<1 'Dim [Table] [Key]	M:1
'Fact [Table] [Key] ==<1 'Dim [Table] [Key]	M:1
'Fact [Table] [Key] ==<1 'Dim [Table] [Key]	M:1
'Fact [Table] [Key] ==<1 'Dim [Table] [Key]	M:1
'Fact [Table] [Key] ==<1 'Dim [Table] [Key]	M:1
'Fact [Table] [Key] ==<=>1 'Dim [Table] [Key]	M:1 (Bid.)
'Fact [Table] [Key] ==<=>1 'Dim [Table] [Key]	M:1 (Bid.)

In above example only three of the dimensions have bi-directional relationship with fact table...? Common scenario is to keep slicers in sync and filter each other in a report

[Filter slicers without using bidirectional filters in Power BI - SQLBI](#)

[Bi-directional relationship guidance - Power BI | Microsoft Learn](#)

[Many-to-many relationship guidance - Power BI | Microsoft Learn](#)

#5 Data Model

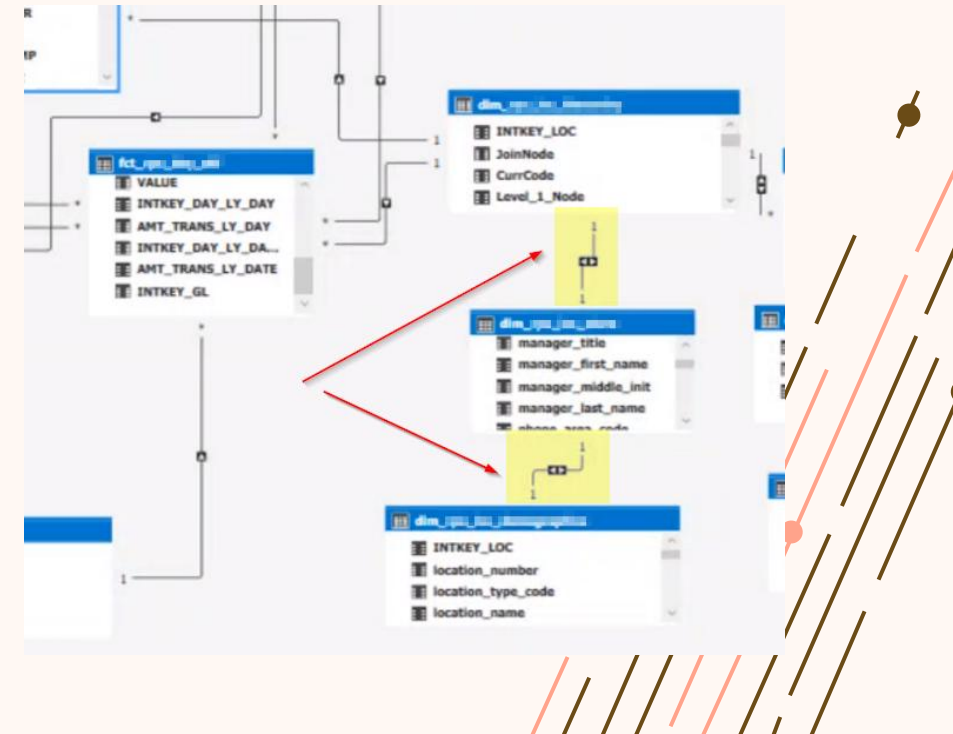
One-to-One relationships

Recommend avoid creating one-to-one relationships if possible

- Contributes to clutter in the Data pane listing more tables than necessary
- Makes it difficult for report authors to navigate and find related fields
- Limits the ability to create user-defined hierarchies for navigation
- Can lead to odd results if values missing between tables

Guidance

- Combine the queries / tables
- If missing values replace with value for reporting like N/A, Unknown, Undefined, etc.



#6 Data Model

Calculated columns

This logic should ideally be done as far upstream as possible

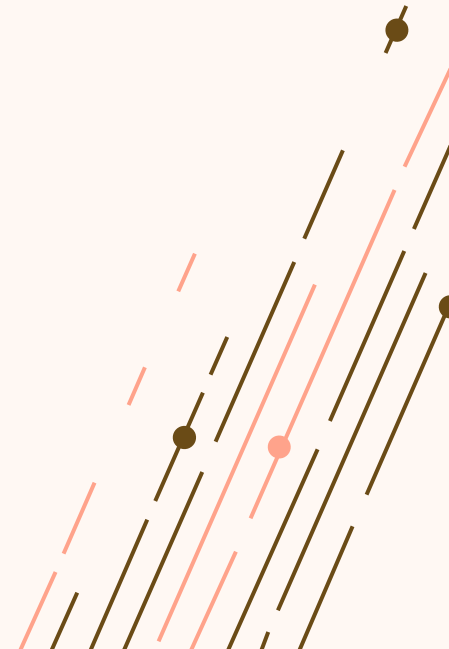
- this will reduce processing time
- these calc columns need to be recalcd for each row on refresh
- these columns will not compress as well as normal columns
- moving logic upstream makes it available for others to consume

Roche's Maxim of Data Transformation^[2] states:

”

*Data should be transformed as far upstream as possible,
and as far downstream as necessary.*

[Roche's Maxim of Data Transformation – BI Polar \(ssbipolar.com\)](http://ssbipolar.com)



#7 DAX

FILTER logic in CALCULATE

The CALCULATE function accepts a table expression returned by the [FILTER](#) DAX function, which evaluates its filter expression for each row of the **Product** table.

```
Red Sales = CALCULATE( [Sales], FILTER('Product', 'Product'[Color] = "Red") )
```

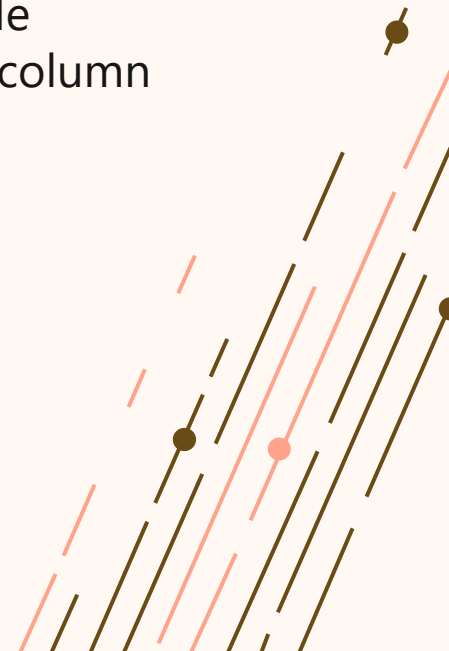
Here's an improved measure definition, which uses a Boolean expression instead of the table expression. The [KEEPFILTERS](#) DAX function ensures any existing filters applied to the **Color** column are preserved, and not overwritten.

```
Red Sales = CALCULATE( [Sales], KEEPFILTERS('Product'[Color] = "Red") )
```

Another classic example would be

```
Sales Amount = CALCULATE( SUM(FactTableName[SalesAmount]),  
    FILTER(FactTableName, FactTableName[ColumnName] = "Value") )
```

[Avoid using FILTER as a filter argument in DAX - DAX | Microsoft Learn](#)



#8 DAX

Avoid multiple references (use variables)

Measure examples where variables can reduce evaluations

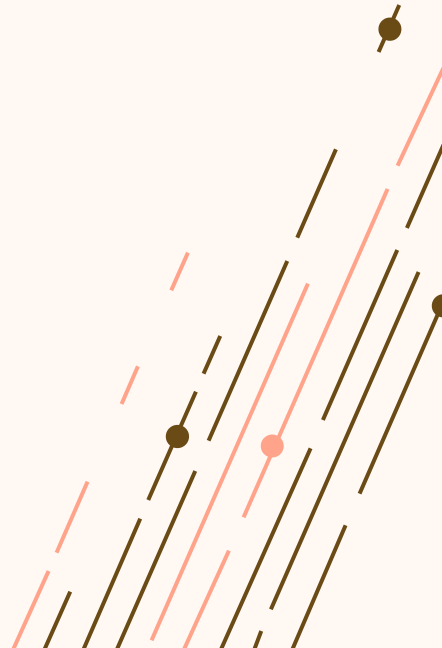
- Measure % Diff = `DIVIDE ([This Year $] - [Last Year $], [Last Year $])`
- Measure QTY = `IF([Quantity] = 0, BLANK(), [Quantity])`
- Measures % Rate = `DIVIDE([Voted Count], [Voted Count] + [Non-Voted Count])`

Use variables in DAX to

- Improve performance, readability, simplify debugging, reduce complexity
- Measure % Diff =
 `VAR LastYear = [Last Year $]`
 `RETURN DIVIDE(([This Year $] - LastYear), LastYear)`

[Use variables to improve your DAX formulas - DAX | Microsoft Learn](#)

[Measure In Power BI: Optimization Tips And Techniques \(enterprisedna.co\)](#)

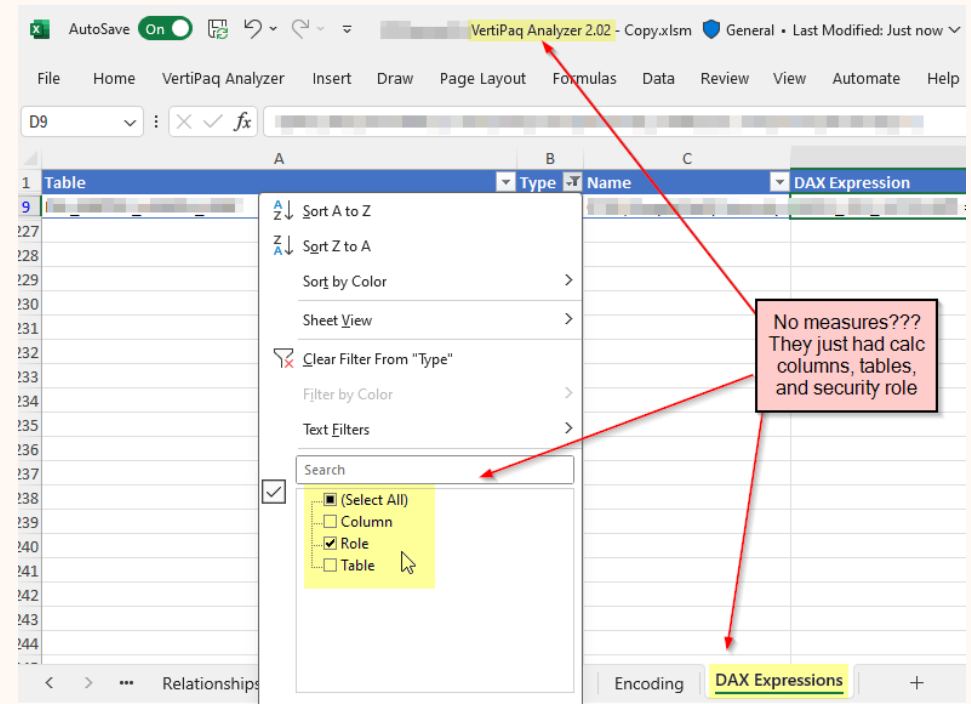


#9 DAX

No explicit measures defined

There are **no explicit measures defined in the model** 🤔 so must be using amount columns as implicit measures 🙌

- No reusable logic to reference for multiple report authors if needed
- Explicit measures would be needed for Analyze in Excel and Paginated reports
 - Implicit measure support in Excel was just added for Analyze in Excel in May 2023
 - [What's New in Excel \(May 2023\) - Microsoft Community Hub](#)



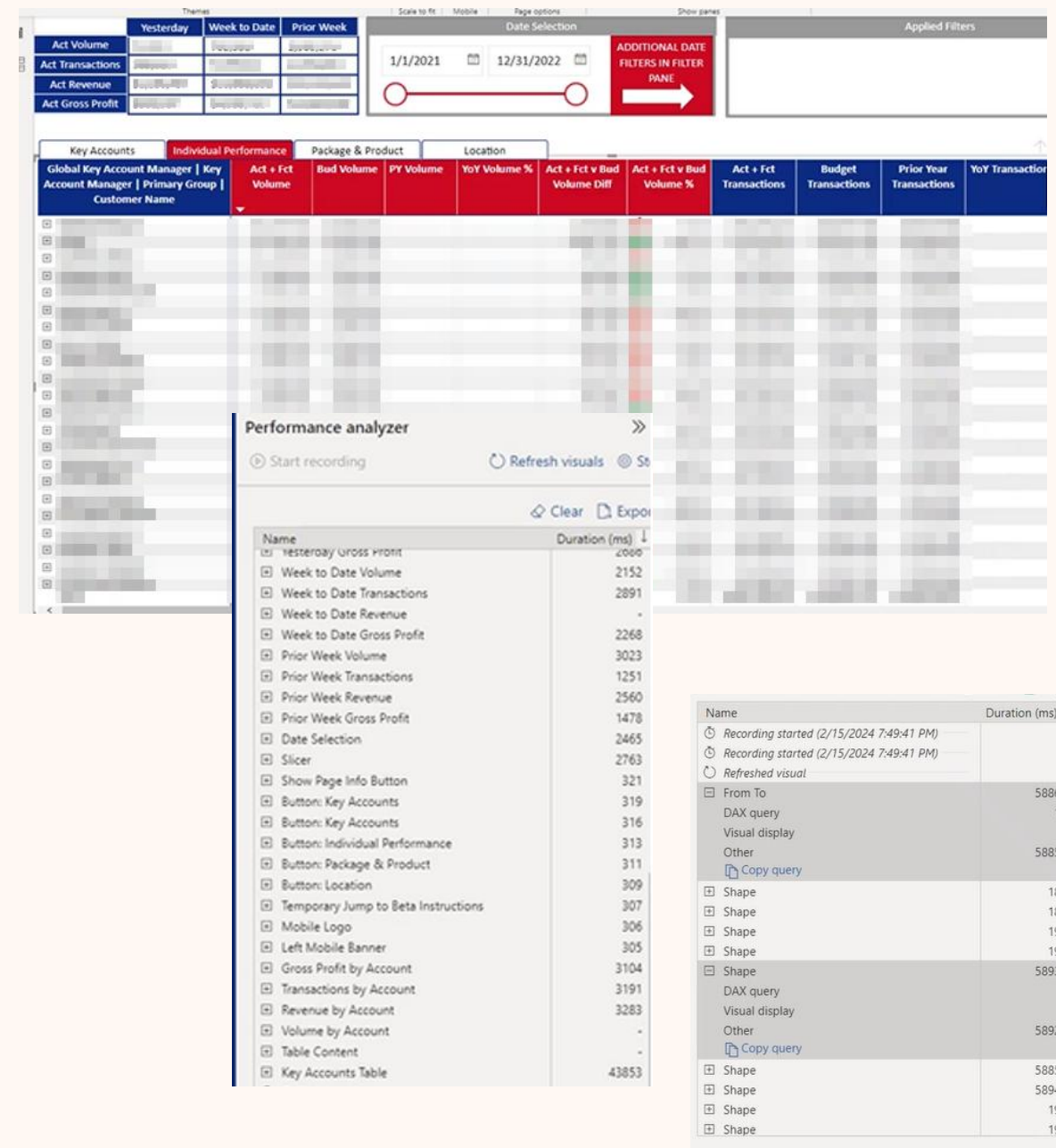
Have seen this twice in the past year and both solutions used a single wide table for all analysis with like 200 columns and no user hierarchies either.

#10 Report Design

Too many visuals

Reduce the number of visuals and items on the report canvas

- In reviewing Performance Analyzer observed approximately 30 or so items being loaded which is a lot and will cause delays in loading the report page because visuals will be waiting for other visuals to load.
- This can be seen by reviewing the Performance Analyzer information and you will notice that the 'Other' category is much larger than the actual query duration.
- Use PowerPoint template for backgrounds to reduce the # of items loading on a report like logos, layouts, etc.

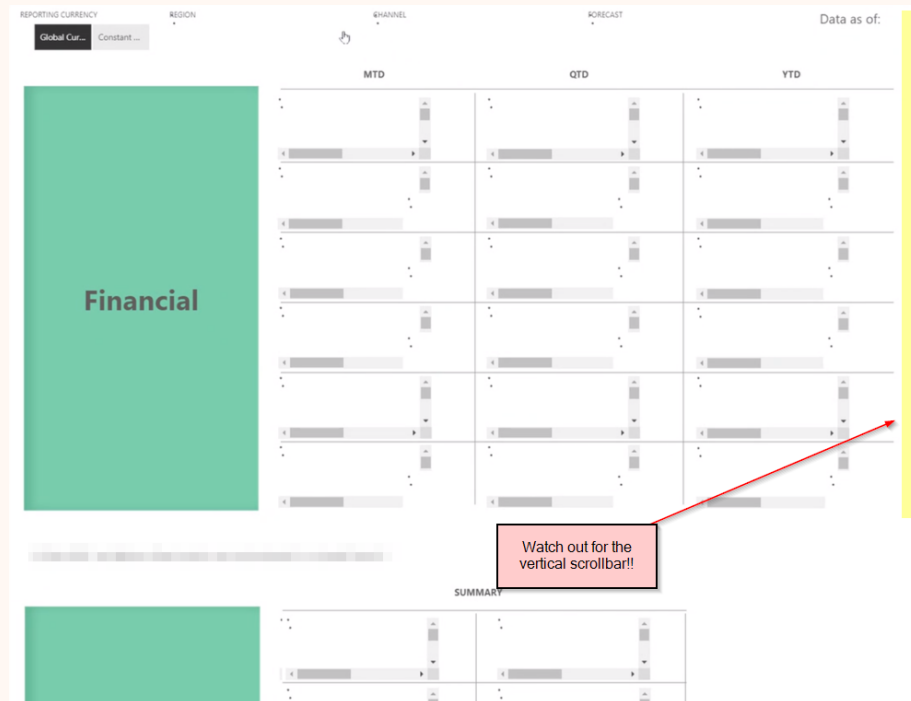


[Consolidating report elements for improved performance | Alluring Analytics \(alluringbi.com\)](https://alluringbi.com)

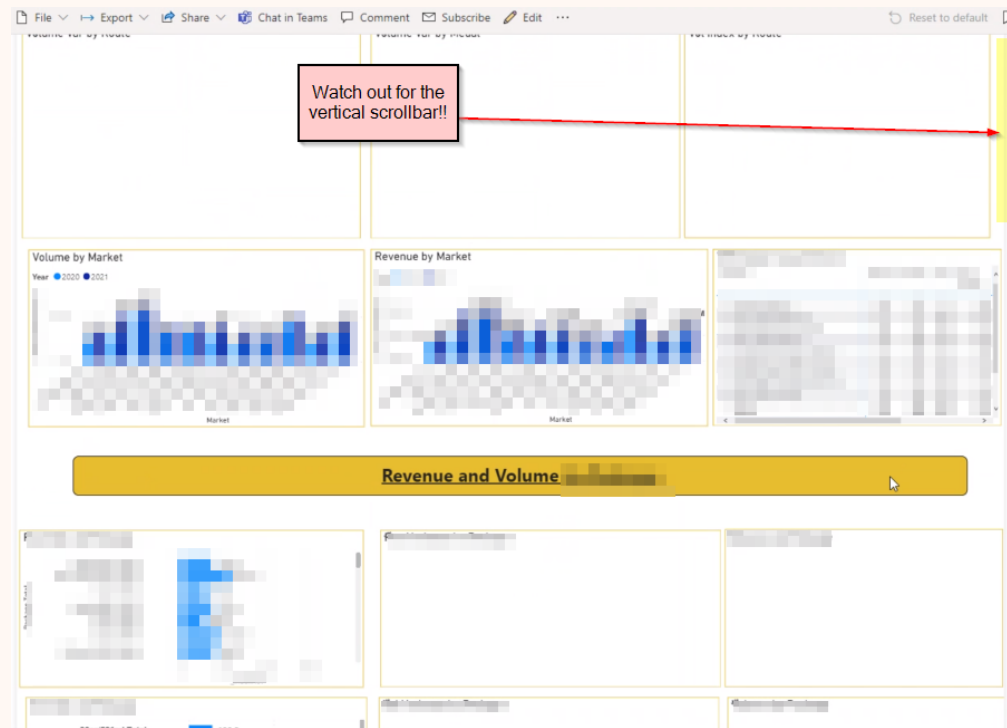
#11 Report Design

Increased the Page Height (vertical scrollbar)

- also the case of #10, too many visuals



The above report was using DirectQuery 🤖



#12 Report Design

Filtering Visuals with Measures

- [Report Analyzer \(elegantbi.com\)](http://elegantbi.com) (external tool like Best Practice Analyzer that would highlight this)
- Determine what the measure filter is attempting to do and push the logic into the actual underlying measures being referenced if you are not getting the proper results in the visual
- There are exceptions to the rule like using a measure to filter a slicer visual to eliminate the use of bi-directional filtering in the model

Date	Town	Fast Measure	Slow Measure
01/01/2018	BRISTOL	3	1.00
01/01/2018	CATERHAM	2	2.00
01/01/2018	DARTFORD	2	2.00
01/01/2018	LONDON	2	2.00
01/01/2018	MANCHESTER	2	2.00
01/01/2018	MORECAMBE	3	3.00
01/01/2018	NEWPORT	4	2.00
01/01/2018	SHEFFIELD	3	3.00
01/01/2018	SOUTHAMPTON	2	2.00
01/01/2018	SUNDERLAND	2	2.00
01/01/2018	TONYPANDY	2	2.00
02/01/2018	ANDOVER	3	3.00
02/01/2018	ATHERSTONE	2	2.00
02/01/2018	BARNOLDSWICK	2	2.00
02/01/2018	BASILDON	2	2.00
02/01/2018	BASINGSTOKE	2	2.00
02/01/2018	BATH	2	2.00
02/01/2018	BEVERLY	2	2.00
02/01/2018	BIRMINGHAM	2	2.00
02/01/2018	BOLTON	2	2.00
02/01/2018	BOSTON	2	2.00
02/01/2018	BOURNEMOUTH	2	2.00

Filters

Search

Filters on this visual

Date is (All)

Fast Measure is greater than 1

Show items when the value is greater than 1

And Or

Apply filter

```

DEFINE
    VAR __ValueFilterDM0 =
        FILTER(
            KEEPFILTERS(
                SUMMARIZECOLUMNS(
                    'Date'[Date],
                    'Property Transactions'[Town],
                    "Fast_Measure", 'Property Transactions'[Fast Measure],
                    "Slow_Measure", 'Property Transactions'[Slow Measure]
                )
            ),
            [Fast_Measure] > 1
        )

    VAR __DS0Core =
        SUMMARIZECOLUMNS(
            'Date'[Date],
            'Property Transactions'[Town],
            __ValueFilterDM0,
            "Fast_Measure", 'Property Transactions'[Fast Measure],
            "Slow_Measure", 'Property Transactions'[Slow Measure]
        )

    VAR __DS0PrimaryWindowed =
        TOPN(501, __DS0Core, 'Date'[Date], 1, 'Property Transactions'[Town], 1)

EVALUATE
    __DS0PrimaryWindowed

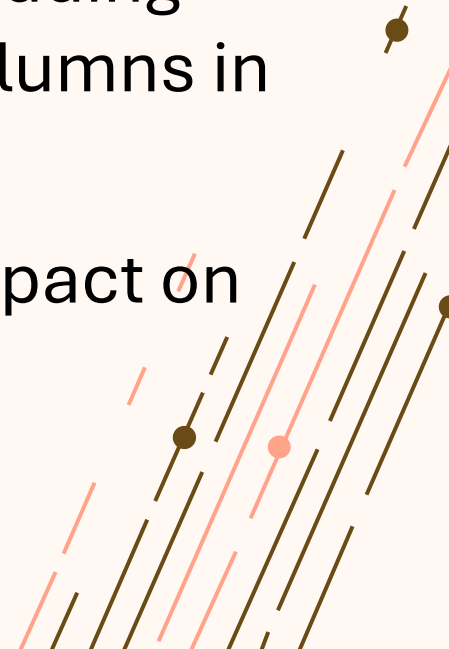
ORDER BY
    'Date'[Date], 'Property Transactions'[Town]

```

[Chris Webb's BI Blog: Performance Tuning Table Visuals With Filters Applied In Power BI](http://crossjoin.co.uk) [Chris Webb's BI Blog \(crossjoin.co.uk\)](http://crossjoin.co.uk)

Final Advice

- Anytime you build a semantic model review **Best Practice Analyzer** and **VertiPaq Analyzer** (metrics)
- Anytime you build a report review the **Performance Analyzer** information to check performance
- Review page **load times** based on each change made like adding more visuals, including filters on visuals, including more columns in tables/matrix, etc.
- As more items are added to the canvas they can have an impact on the overall load time and the load time of other visuals.



Resources



Resource Links

- [Power BI Guidance Documentation](#)
- Guy In a Cube Videos
 - [My CHECKLIST for troubleshooting Power BI Performance – YouTube](#)
 - [3 signs it's time to OPTIMIZE your Power BI report - YouTube](#)
 - [Debugging a slow Power BI report with Phil Seamark – YouTube](#)
 - [Guy in a Cube - YouTube \(best practice search\)](#)
- [Best Practice Analyzer for Tabular Editor Update 5/20](#)
- [best practices | Search Results | \(desertislesql.com\) \(3 part series\)](#)
- [10 Best practice tips to improve your Tabular Model performance | BI4ALL - Turning Data Into Insights](#)
- [Improve Power BI Performance by Optimizing DAX | MAQ Software](#)

Questions?





slides available @ <https://bit.ly/m365tc25powerbigotchass>



Thank you!

Dan English
<https://www.linkedin.com/in/denglishbi>



Submit session feedback

