

Power BI 2023 Update

Mark Wilcock

New in 2023 (and 2024) – a personal choice

Stuff I can demo

- New card visual (multi-cards, indicators, actions,...)
- Better charts (dynamic formatting, on-object formatting, leader lines, smoothed and stepped lines, sparklines)
- Data Modelling and DAX:
 - WINDOW() function (Dec 2022),
 - calculation groups (Oct 2023),
 - semantic model tree (Oct 2023),
 - DAX Query Pane,
 - visual calculations

Demos and resources

- PBI 2023 Update PBIX (*)
 - New card visuals
 - Calculation groups
 - WINDOW() function: comparison with older approach
 - Chart improvements
- Rainfall.PBIX (DAX query pane use case) (*)
- Scorecard.PBIX : dynamic formats use case (* after YouTube video Jan 2024)

** indicates I will make these available after the presentation*

New Card Visual (Phase 1 demo)

- Phases
 1. (June 2023) multi-cards, layout features, dynamic formats
 2. (Nov 2023?) indicators
 3. actions
 4. trends
 5. small multiples
- Will replace “old” card, KPI and multi-card visuals
- Links:
 - Phase 1 [\(4\) New Card Visual Tutorial - Power BI \(June 2023 Update\) – YouTube](#)
Power BI Park (18 minutes)
 - Linked In Post: Armand (PBI: Core Visuals team) “Sneak Preview of Nov/Dec features)
 - Miguel Myers (PM) video tutorial (June 2023)

New Card Visual: Phase 1

New Card Visual

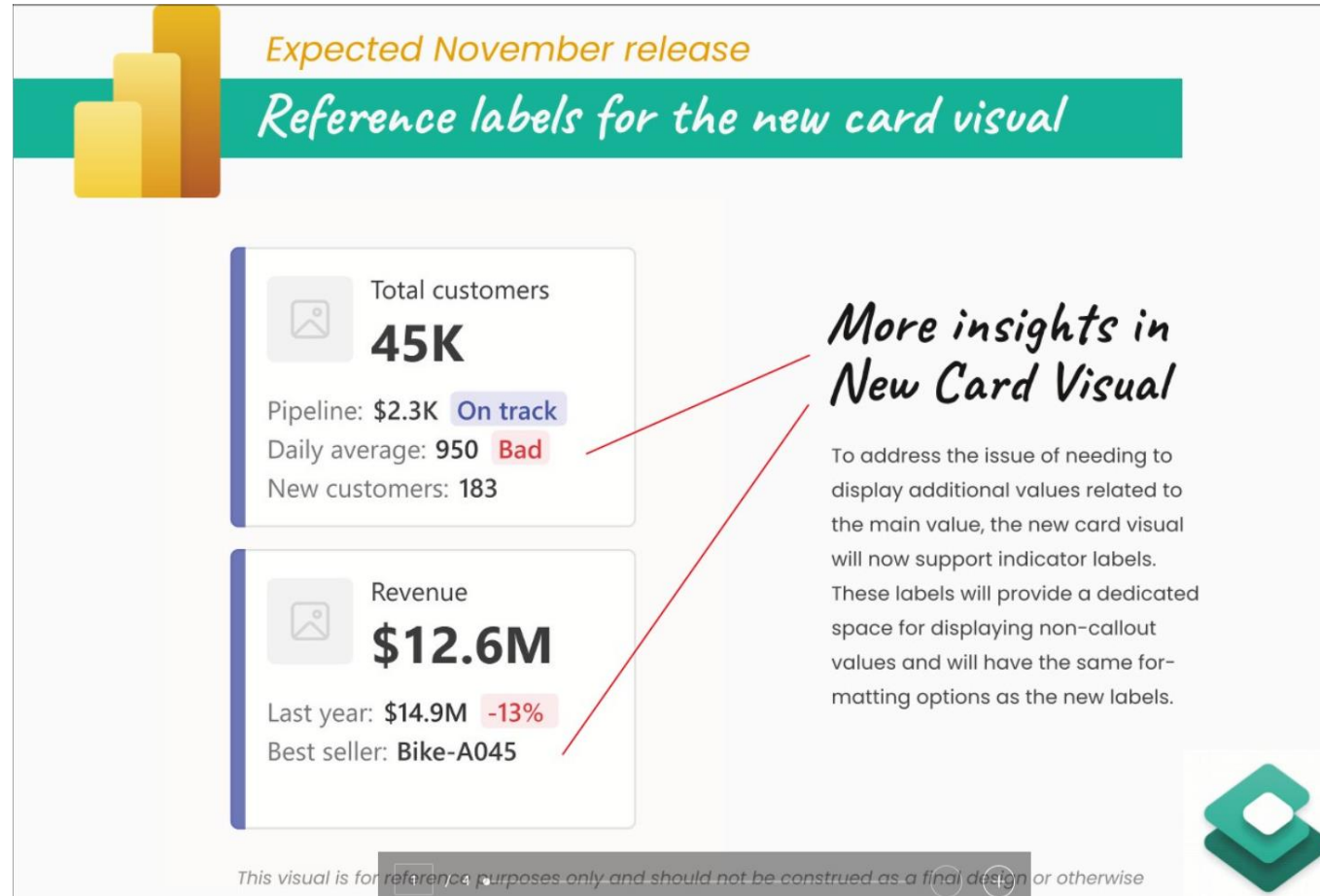
This is Phase 1 (multi-cards)



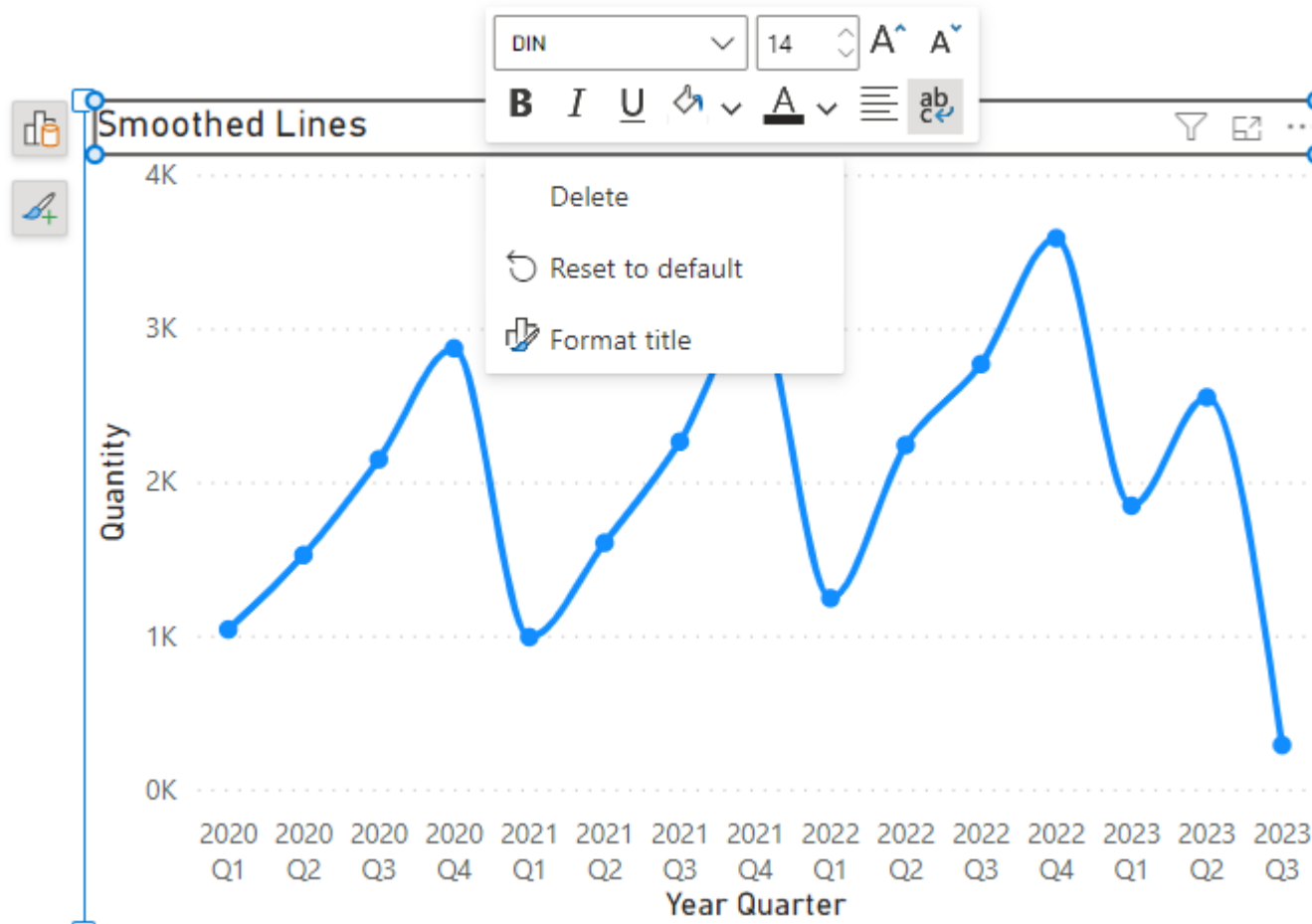
New Card Visual: Snapshots

<https://www.linkedin.com/feed/update/urn:li:activity:7125198405177339905/>

LinkedIn, Armand van Amersfoort, Nov 2023



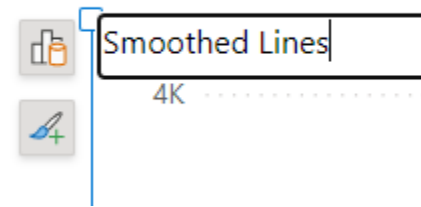
On Object Formatting



Add to your visual

- ☒ Title
- ☒ X-axis
- ☒ Y-axis
- ☒ Secondary y-axis
- ☐ Zoom slider
- ☐ Markers
- ☒ Data labels
- ☐ Series labels

More options



Build a visual

Visual types

● Off Suggest a type

X-axis

Year Quarter

+Add data

Y-axis

Quantity

+Add data

Secondary y-axis

+Add data

Legend

+Add data

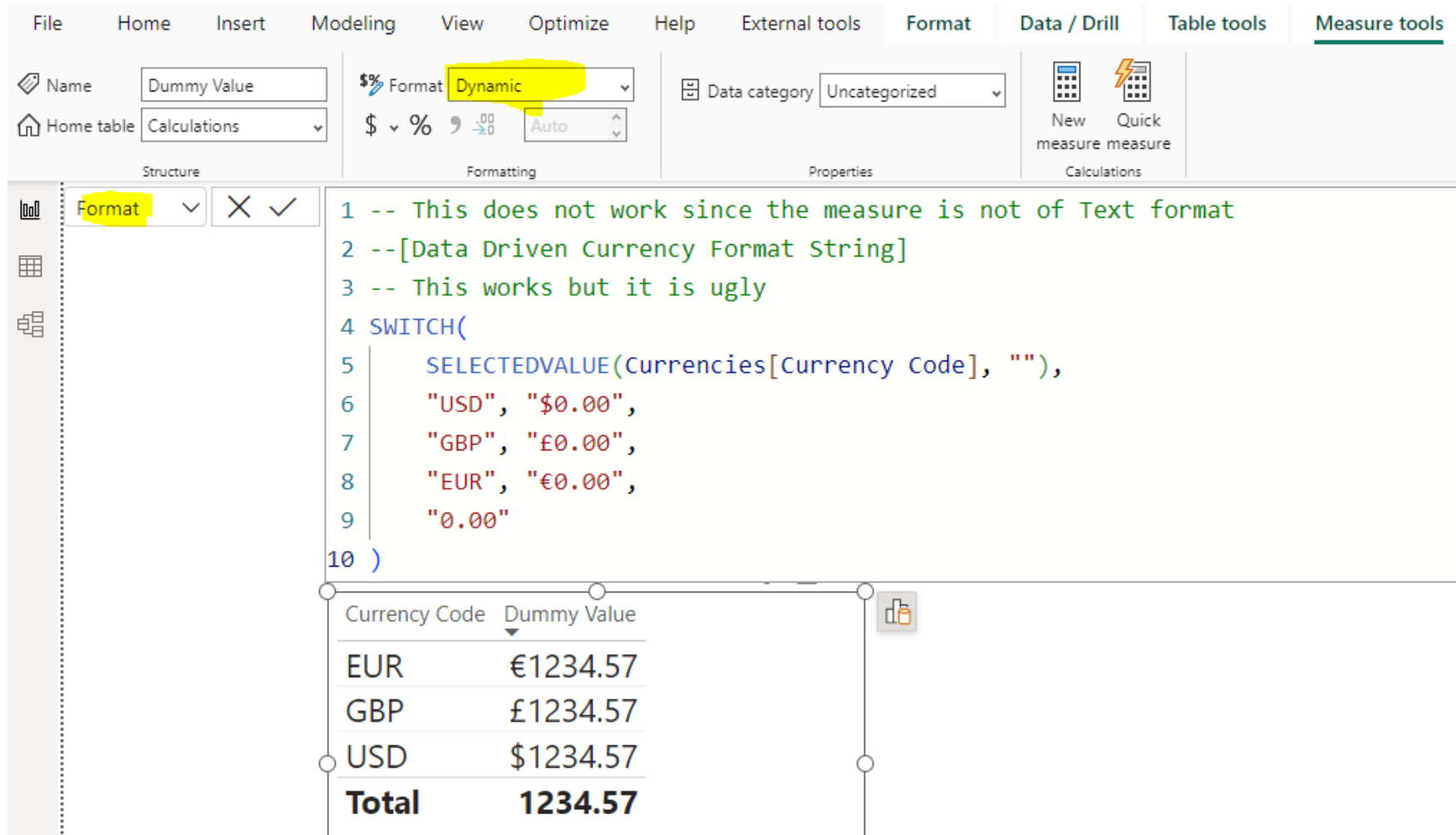
Small multiples

+Add data

Tooltips

+Add data

Dynamic formats



The screenshot displays the Microsoft Power BI interface, specifically the 'Format' tab. The 'Name' field is set to 'Dummy Value', and the 'Home table' is 'Calculations'. The 'Format' dropdown is set to 'Dynamic'. The 'Data category' is 'Uncategorized'. The 'Format' dropdown is highlighted in yellow.

The DAX measure is defined as follows:

```
1 -- This does not work since the measure is not of Text format
2 --[Data Driven Currency Format String]
3 -- This works but it is ugly
4 SWITCH(
5     SELECTEDVALUE(Currencies[Currency Code], ""),
6     "USD", "$0.00",
7     "GBP", "£0.00",
8     "EUR", "€0.00",
9     "0.00"
10 )
```

The resulting table shows the following data:

Currency Code	Dummy Value
EUR	€1234.57
GBP	£1234.57
USD	\$1234.57
Total	1234.57

Dynamic formats

Some uses of dynamic formats

- Use in-house formats, e.g. m rather than the default M for millions
- Scorecard situations, e.g. dataset has indicator, date, value columns

Data Modelling and DAX

We will cover

- WINDOW() function
- Calculation groups, (and semantic model tree)
- DAX query pane
- Visual queries

DAX WINDOW() function

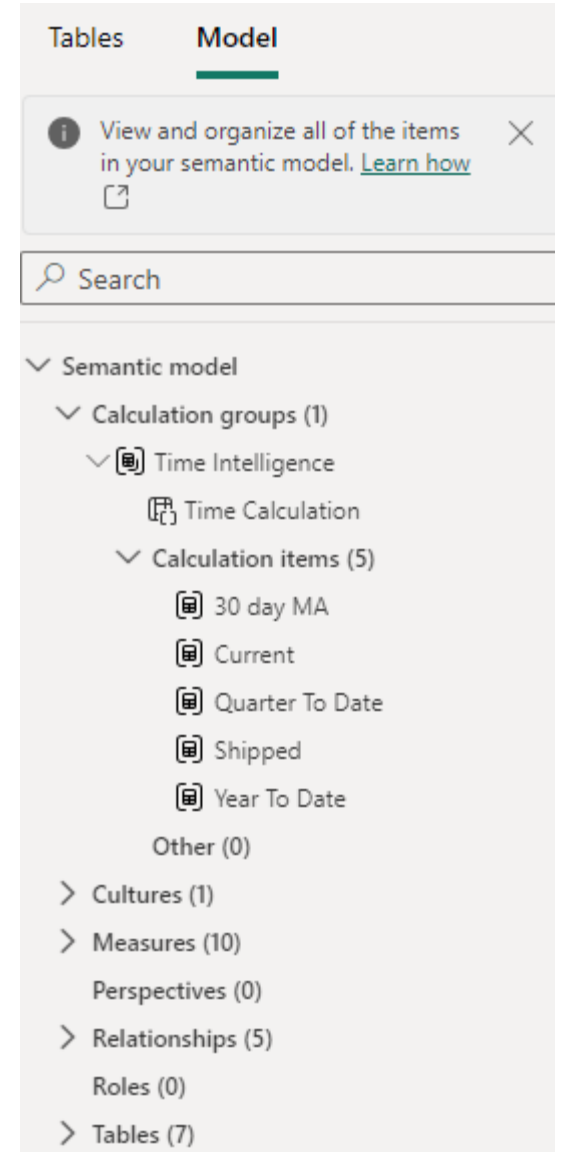
- Simpler, more versatile and more efficient than traditional methods for time intelligence measures
- Many people already know window functions from SQL or Python

```
1 Running Sales (window) =  
2 SUMX(  
3     WINDOW(  
4         1, ABS, 0, REL,  
5         ALLSELECTED(Dates[Date]),  
6         ORDERBY(Dates[Date])  
7     ),  
8     [Sales]  
9 )
```

```
1 Running Sales (traditional) =  
2 VAR maxDate = MAX(Dates[Date])  
3 RETURN  
4 CALCULATE(  
5     [Sales],  
6     --REMOVEFILTERS(Dates),  
7     Dates[Date] <= maxDate  
8 )
```

Calculation Groups (October 2023)

- Useful for large models with many base measures e.g. [Sales], [Profit], [Cost],..., all with time intelligence derivatives e.g. YTD, QTD, 10 day moving average, Q on Q,...
- Avoids building hundreds of measures
- Create these in new “Semantic Model” tree



DAX Query Pane (coming soon)

- EVALUATE DAX queries and DEFINE measures
- Think SSMS for DAX or DAX Studio , if you know these
- Can see all measure definitions in one place
- Can see definitions of a selected measure and all the component measures
- Later: can edit DEFINE measure and save back to model
- Code first approach to DAX
- Example of use: Grab DAX query from Performance Inspector and run it

DAX Query Pane (Snapshot)

Empower every BI professional to do more with Microsoft Fabric | OD06

File Home Help External tools

Paste Cut Copy Get data Excel workbook Data hub SQL Server Enter data Dataverse Recent sources Transform data Refresh data Manage relationships New measure Quick measure New column New table

Clipboard Data Queries Relationships Calculations

Run Copilot

```
1 DEFINE
2     MEASURE 'Pick a sales measure'[Total sales amount $] = SUM ( 'Sales'[Sales Amount] )
3
4     MEASURE 'Pick a sales measure'[Order cost $] = SUM ( 'Sales'[Total Product Cost] )
5
6     MEASURE 'Pick a sales measure'[Profit margin $] = [Total sales amount $] - [Order cost $]
7
8     MEASURE 'Pick a sales measure'[Profit margin %] = DIVIDE ( [Profit margin $], [Total sales amount $] )
9 // show me profit margin % by fiscal year
10 EVALUATE
11     SUMMARIZECOLUMNS(
12         'Date'[Fiscal Year],
13         "Profit margin %", [Profit margin %]
```

Results Result 1 of 1 Copy Open with

	Date[Fiscal Year]	[Profit margin %]
0	FY2018	0.12723470999221562
1	FY2019	0.10883441022176717
2	FY2020	0.11194344005606567

DAX query pane: useful to explain this

```
1 Average Monthly Rainfall =  
2 -- loop through the rows in the virtual table created below  
3 -- and get the average of the month average values  
4 AVERAGEX(  
5     -- create a virtual table with a row for every month  
6     VALUES(Rainfall[Month]),  
7     -- add a column with the average rainfall in each month  
8     [Average Daily Rainfall]  
9 )
```

DAX query

```
1 EVALUATE VALUES(Rainfall[Month])
2
3 EVALUATE
4 ADDCOLUMNS(
5     VALUES(Rainfall[Month]),
6     "@Rainfall", [Average Daily Rainfall]
7 )
8
9 DEFINE TABLE SlimMonthlyTable = VALUES(Rainfall[Month])
10 EVALUATE SlimMonthlyTable
11
12 DEFINE VAR SlimMonthlyTable = VALUES(Rainfall[Month])
13 EVALUATE ADDCOLUMNS(
14     SlimMonthlyTable,
15     "@Rainfall", [Average Daily Rainfall]
16 )
17
18 DEFINE VAR SlimMonthlyTable = VALUES(Rainfall[Month])
19 VAR FullMonthlyTable =
20 ADDCOLUMNS(
21     SlimMonthlyTable,
22     "@Rainfall", [Average Daily Rainfall]
23 )
24 EVALUATE
25 {AVERAGEX(
26     FullMonthlyTable,
27     [@Rainfall]
28 )}
29 }
30
```

log Results History

Month	@Rainfall	
Month 01 Jan	4.33333333333333	
Month 02 Feb	10.5	

Showing DAX Studio (external tool) until
DAX query pane in Power BI Desktop is
released

DAX query pane: example use case demo

- Data: “Rainfall” table with columns: Date, Value and one row for some date
- Objective: Calculate the average of the average monthly rainfall over the year
- *Not* the same as the average (annual rainfall)
- Challenge: double aggregation, average rows for each month, then average the monthly values
- In a measure, AVERAGEX over a virtual table (monthly averages)
- Virtual table created by table function, VALUES() and never seen

Visual Calculations (March 2024?)

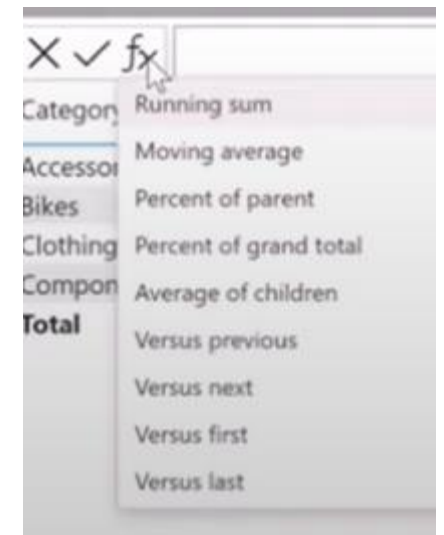
- Write simple DAX calculations defined on a specific visual
- Can refer to fields on the visuals, executed in scope of the visual
- Use visual matrix with rows columns
- Standard template for typical calculations: e.g. running sum
- Some new high level DAX visual functions:
 - Positional: FIRST(), PREVIOUS(), NEXT(), LAST()
 - Functions to expand groups if two fields in rows area: EXPAND(), XOLLAPSE()
- Links
 - Power BI 2023 Release 2 Wave Plan [Understanding visual calculations | Microsoft Learn](#)
 - Jay ter Heerdt (PM) presentation to user group (1 hour video) [\(4\) Visual Calculations - Improvements to doing calculations in Power BI - Jay ter Heerdt - YouTube](#)

Visual Calculations: Why?

- What problem does it solve: DAX becomes a vertical learning curve. Simple problems e.g. running totals that are easy in Excel are hard in DAX

Date	Value	Running Sum
13/11/2023	3	3
14/11/2023	4	=D5+C6
15/11/2023	9	16
16/11/2023	4	20
17/11/2023	4	24
18/11/2023	5	29
19/11/2023	6	35
20/11/2023	3	38
21/11/2023	3	41
22/11/2023	3	44

Visual calcs: Snapshots (Microsoft PM video)



Visual calcs: Snapshots (Microsoft PM video)



Visual calcs vs measures vs calc columns



Calculated column

- Defined on a table
- Works on a row-by-row basis (row context)
- Computed at dataset refresh (for import tables) or query refresh (for DirectQuery tables)
- Result persisted (for import tables)



Measure

- Defined in the data model
- Works on sets of rows (filter context)
- Computed at query execution



Visual calculations

- Defined on a visual
- “Visible context”
- Computed at query execution
- Can refer to visual structure

Visual calcs: Snapshots (Microsoft PM video)



Visual calculations are easy and flexible

- WYSIWYG
- Point-and-click provided, if you want
- Just “visible context”
- High-level functions for common business calculations
- Refer to visual structure

Visual calcs: Snapshots (Microsoft PM video)



Example: Axis

`FIRST([Sales Amount], Rows)` →
for each *Row*, retrieve the first Sales Amount from the

1 Calculation = <code>FIRST([Sales Amount], Rows)</code>						
Fiscal Year	FY2018		FY2019		FY2020	
Category	Sales Amount	Calculation	Sales Amount	Calculation	Sales Amount	Calculation
Accessories	\$36,814.85	36,814.85	\$138,901.55	138,901.55	\$1,096,341.49	1,096,341.49
Bikes	\$22,590,983.47	36,814.85	\$28,544,881.62	138,901.55	\$43,484,661.12	1,096,341.49
Clothing	\$66,327.53	36,814.85	\$757,224.19	138,901.55	\$1,294,061.73	1,096,341.49
Components	\$1,166,765.32	36,814.85	\$4,629,101.14	138,901.55	\$6,003,210.20	1,096,341.49
Total	\$23,860,891.17	23,860,891.17	\$34,070,108.50	34,070,108.50	\$51,878,274.54	51,878,274.54

Visual Calculations: Importance!

- Impact: Microsoft PM says his expectation that “people mostly using visual calcs and only calc columns [...] and measures when necessary... most calcs will be visual calcs”
- Performance: better than measures, since run on aggregated data
- Unintended consequences?
 - Measures: write once, use everywhere
 - Visual calcs: write every time, use once

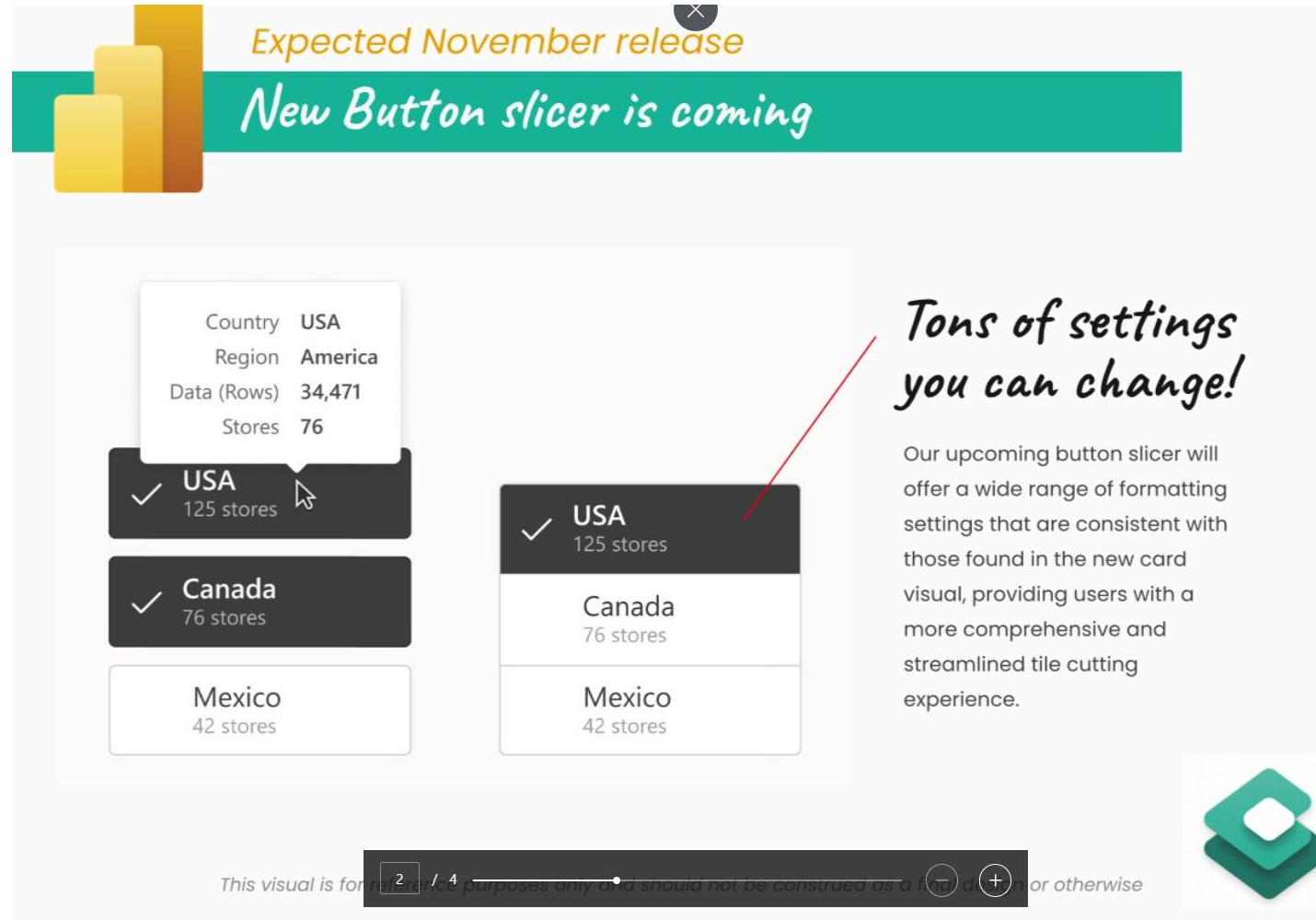
Other stuff in the pipeline

- Different style of bars: outline rather than solid. For example use solid for actuals and outline for forecast
- Better data labels: add trend as well as value
- Technique:| Using measures as data label (Helen Wall reference)
- In Preview
- Line charts: leader lines, smoothed and stepped lines
- Sparklines

New button slicer: Snapshots

<https://www.linkedin.com/feed/update/urn:li:activity:7125198405177339905/>

LinkedIn, Armand van Amersfoort, Nov 2023



The image is a screenshot of a LinkedIn post from Armand van Amersfoort, dated November 2023. The post features a banner at the top with the text "Expected November release" and "New Button slicer is coming". Below the banner, there are two visualizations of a button slicer. The left visualization shows a list of countries: USA (125 stores), Canada (76 stores), and Mexico (42 stores). The right visualization shows a similar list but with a different layout. A red arrow points from the text "Tons of settings you can change!" to the right visualization. Below the visualizations, there is a footer with the text "This visual is for reference purposes only and should not be construed as a final design or otherwise".

Expected November release

New Button slicer is coming

Country USA
Region America
Data (Rows) 34,471
Stores 76

✓ USA 125 stores

✓ Canada 76 stores

Mexico 42 stores

✓ USA 125 stores

Canada 76 stores

Mexico 42 stores

Tons of settings you can change!

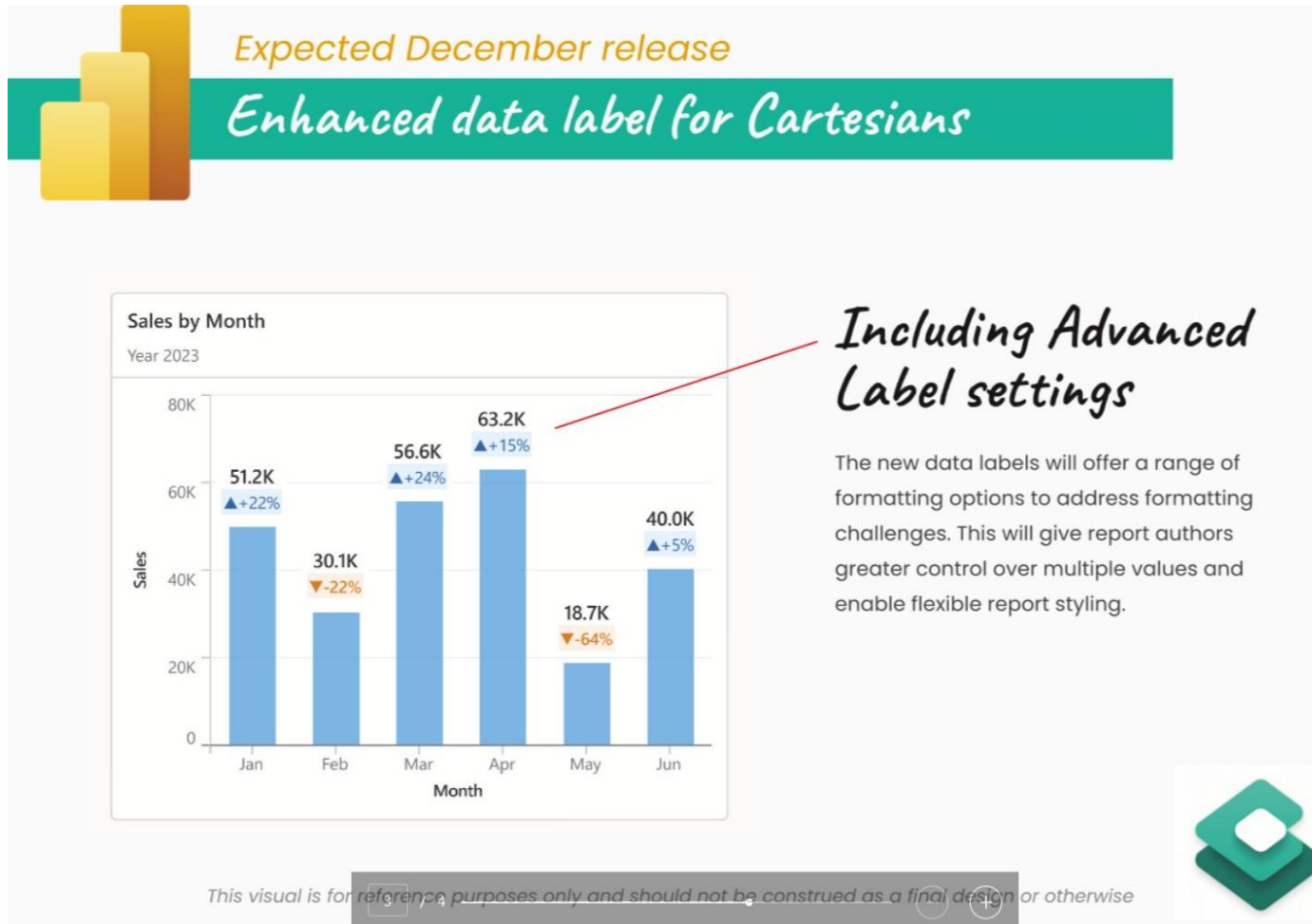
Our upcoming button slicer will offer a wide range of formatting settings that are consistent with those found in the new card visual, providing users with a more comprehensive and streamlined tile cutting experience.

This visual is for reference purposes only and should not be construed as a final design or otherwise

Better data labels: Snapshots

<https://www.linkedin.com/feed/update/urn:li:activity:7125198405177339905/>

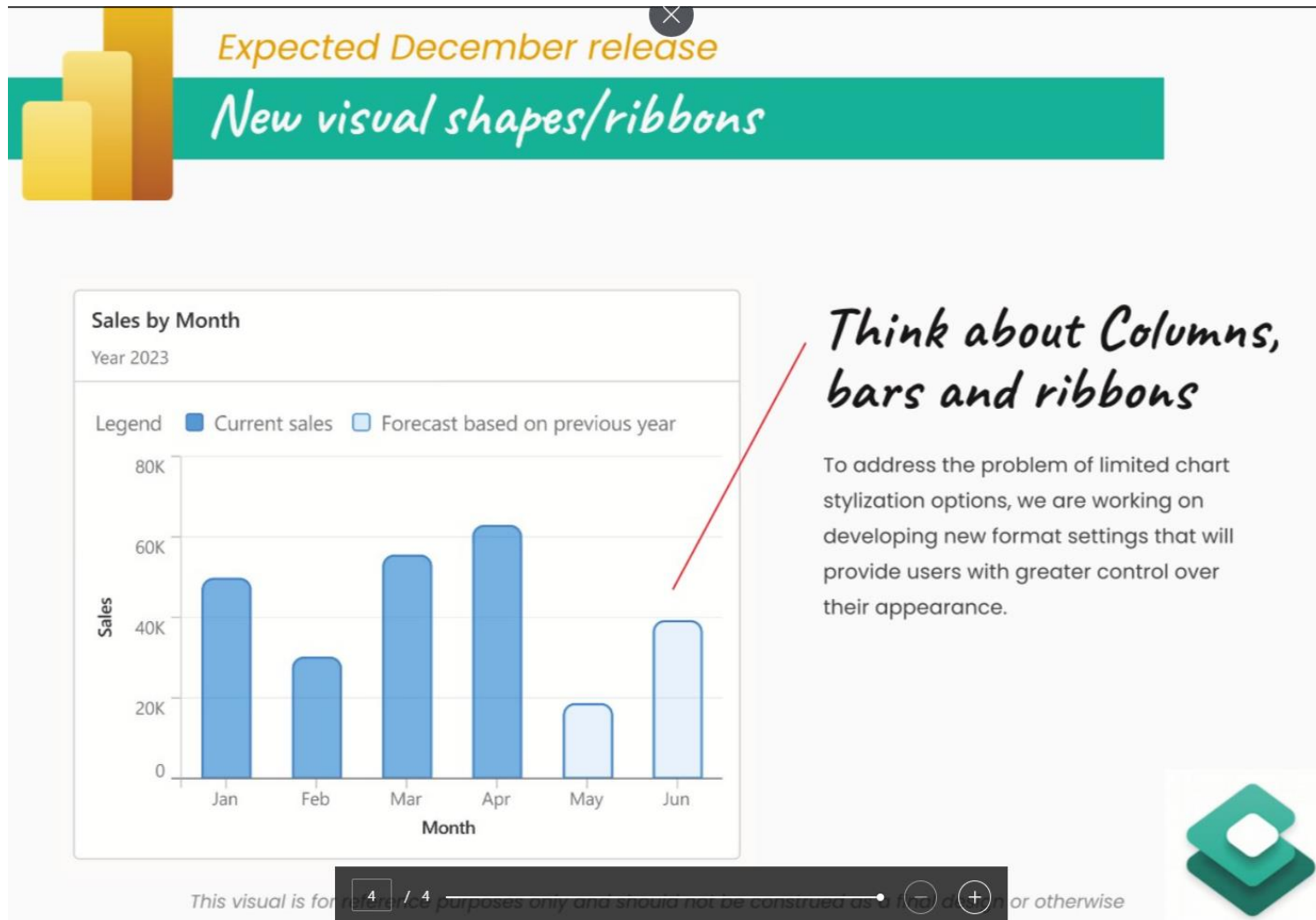
LinkedIn, Armand van Amersfoort, Nov 2023



New visuals shapes

<https://www.linkedin.com/feed/update/urn:li:activity:7125198405177339905/>

LinkedIn, Armand van Amersfoort, Nov 2023



Not in Power BI Desktop but...

PBI Explorer

PBI Explorer is a **free** tool to



Take full ownership of your Power BI reports

PBI Explorer gives you a code-first view of your PBIX/PBIP reports. All visual properties, settings, and relationships are fully exposed in Tabular Editor style.



Inspect differences between report versions

PBI Explorer provides a git-like diff view between two reports, highlighting what's been modified or removed.



Understand how the data model is used by reports

PBI Explorer provides a view of all elements from the data model used in the report.



View difficult-to-see report features

PBI Explorer exposes all report features that are difficult to inspect visually (like bookmarks, filters, etc).

PBI Explorer is free and will always be free.

The screenshot displays the PBI Explorer application interface, titled "PBI Explorer by Tabular Tools (BETA)". The main window shows the "Summary" tab for the report "Superstore Star - Example Solution.pbix", last modified on 2023-11-01 at 08:55:20. The summary includes counts for Pages (25), Visuals (1.7), Fields (3.2), and Filters (5). It also lists the top 3 pages by visual count, top 3 visuals by type, top 3 fields used, and filter types. The "Issues" section shows 76 best practice issues detected, categorized by severity. The "Themes" section shows the current theme is "(Base theme)". The "Misc" section lists various report features like Tooltips, Bookmarks, Layouts, and Custom visuals. On the right, the "Report Explorer" and "Data Model Explorer" panels are visible, showing a hierarchical view of the report's structure and data model elements.

Summary **Details**

Superstore Star - Example Solution.pbix 2023-11-01 08:55:20 5.46

Pages	Visuals	Fields	Filters
25	1.7	3.2	5

Top 3 pages by visual count:

Page	Visual Count
1. Explain the Increase	7
2. viz: Field Parameters	6
3. City DT	5

Top 3 visuals by type:

Visual	Count
1. Stacked Column Chart	11
2. Table	9
3. Clustered Column Chart	6

Top 3 fields used:

Field	Count
1. [Sales]	25
2. 'Dates'[Year]	10
3. 'Geography'[Region]	5

Filter types:

Filter Type	Count
Report-level filters:	0
Page-level filters:	3
Visual-level filters:	2

Issues **76**

Best Practice issues detected:

Severity	Count
Severe:	6
Medium:	69
Low:	1

Themes **1**

Current theme: (Base theme)

Misc

Feature	Count
Tooltips	0
Bookmarks	0
Layouts	1
Custom visuals	0

Report Explorer **Data Model Explorer**

Object	Type
Report	Report
Sales By Order and Ship Date	Page
Discounted Sales	Page
Analysis By City	Page
d7c30afcad9452e102a3	Table
4809a515a04654939900	Table
c3978a0856206a3ebe8a	Action Butt...
01eb0d494929db497c90	Textbox
LASTDATE...	Page
Bi-directional relationships	Page
Explain CALCULATE	Page
Cumulative and YTD	Page
Previous YTD	Page
Year on Year	Page
21f748128e0c3626da27	Line and Cl...
f041a59a0ebe58f9ca73	Textbox
Big Ticket revenue	Page
Cohort Analysis	Page
QTD Implementations	Page
Sales for new customers	Page
By Number of Orders	Page
Show label of latest value only	Page
Days To Ship Analysis	Page
City DT	Page
Viz: Decomposition Tree	Page
viz: Field Parameters	Page
viz: Sparkline	Page
viz: Small Multiples	Page
viz: not concatenated labels	Page
Explain the Increase	Page
Subcat > 5%	Page

Inspecting Superstore Star - Example Solution.pbix

I should have paid more attention to ..

- Code first improvements e.g. PBIP projects,
- DAX measure suggestions (Copilot for DAX is coming)
- *What else?*