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

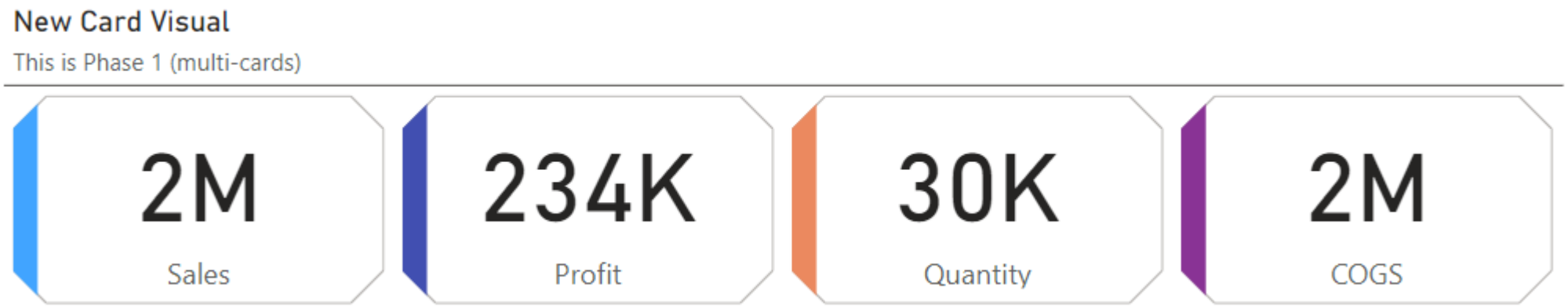
- PBI 2023 Update PBIX (*)
  - New card visuals
  - Calculation groups
  - WINDOW() function: comparison with older approach
  - Chart improvements
- Rainfall.PBIX (DAX query pane use case) (*)
- Scorecard.PBIX : dynamic formats use case (* after YouTube video Jan 2024)

** 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)

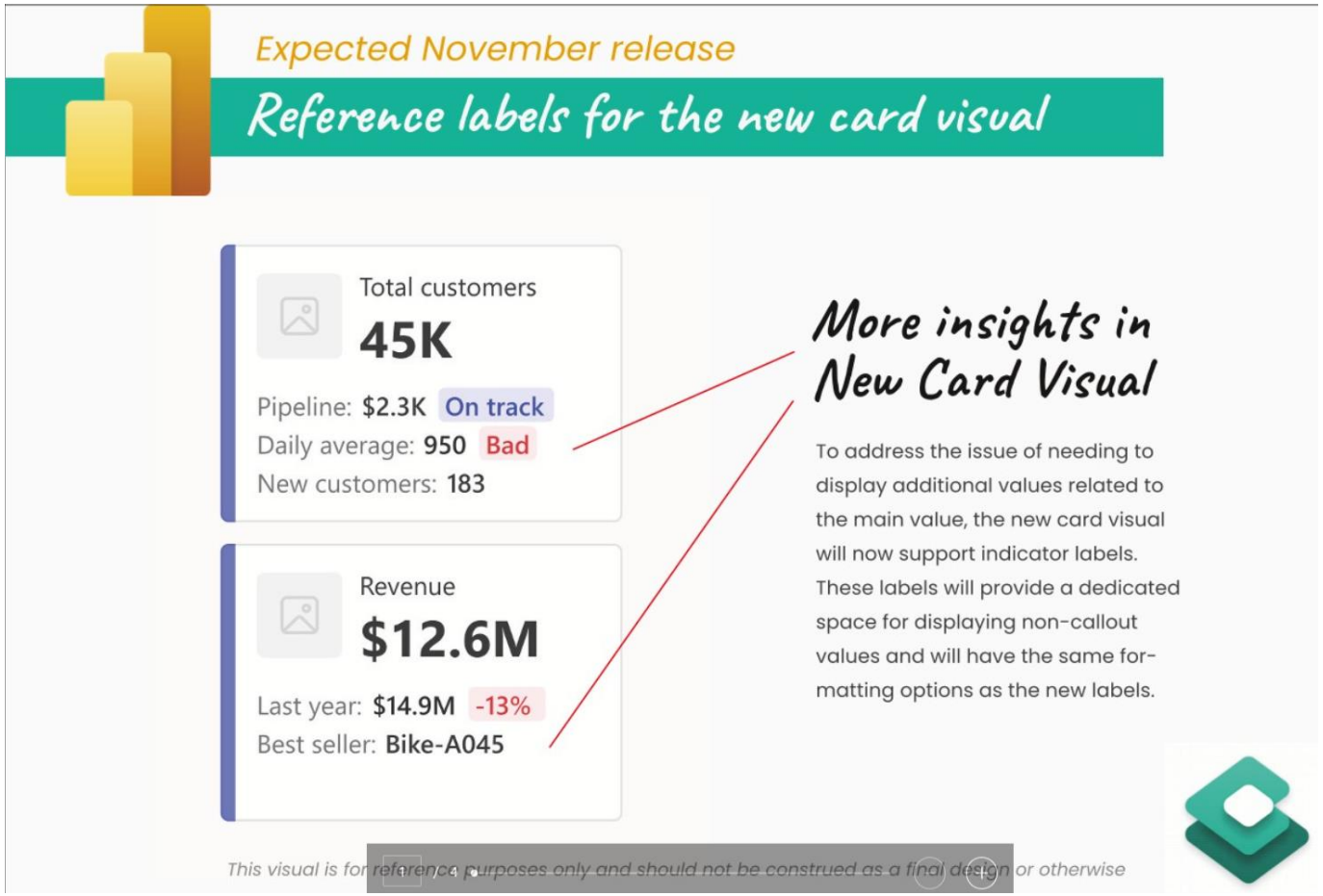
New Card Visual: Phase 1



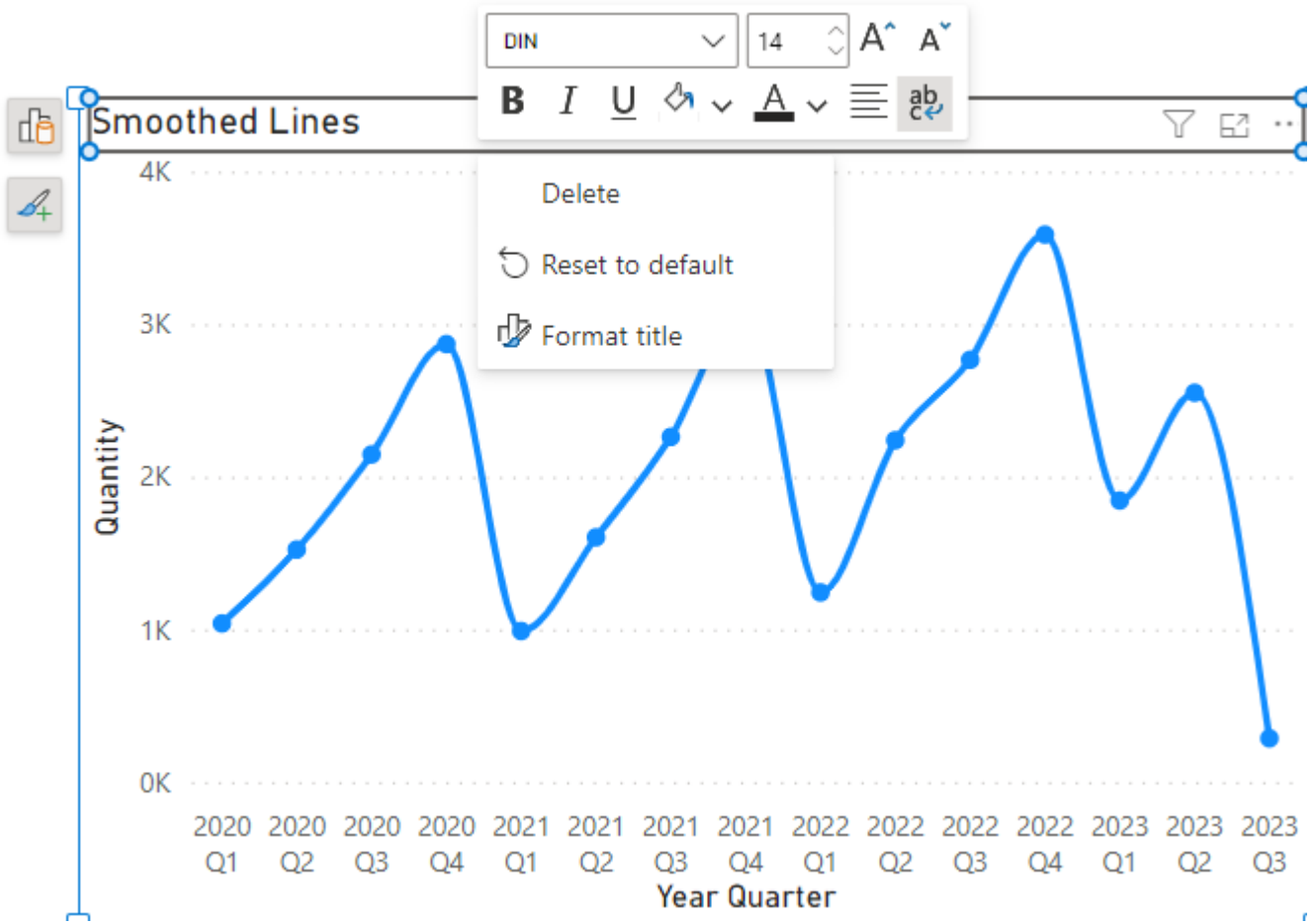
New Card Visual: Snapshots

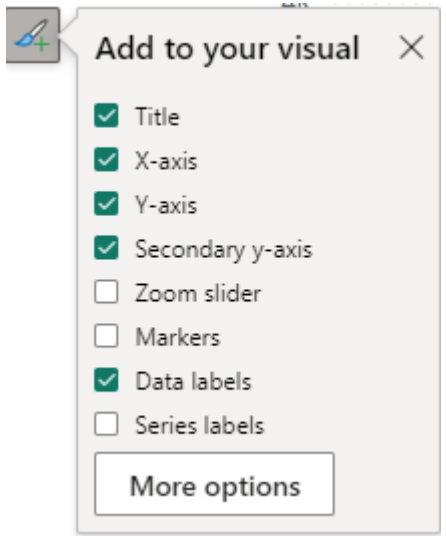
https://www.linkedin.com/feed/update/urn:li:activity:7125198405177339905/

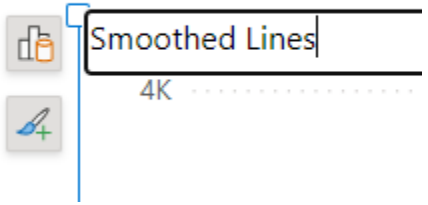
LinkedIn, Armand van Amersfoort, Nov 2023

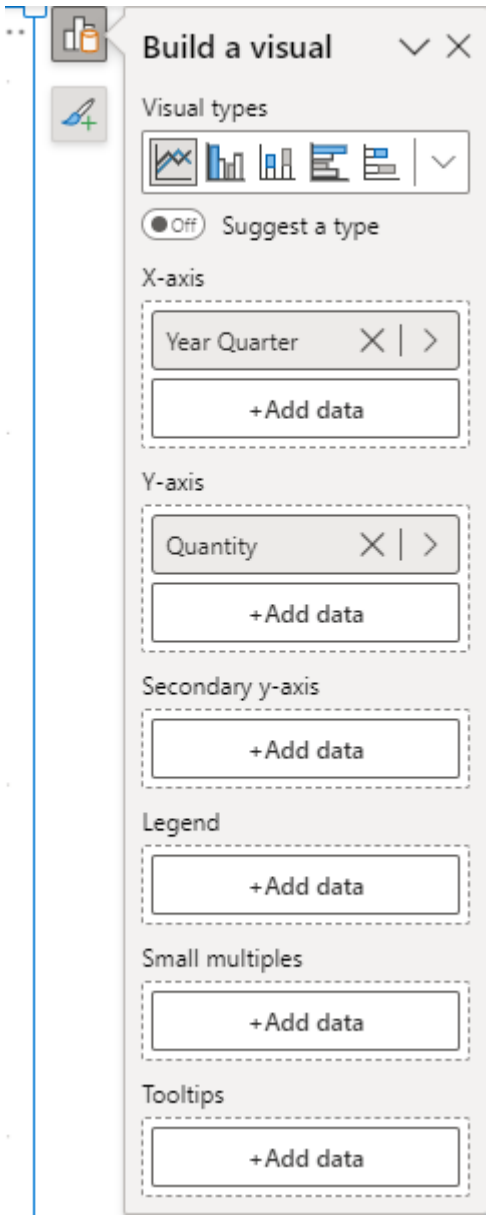


On Object Formatting

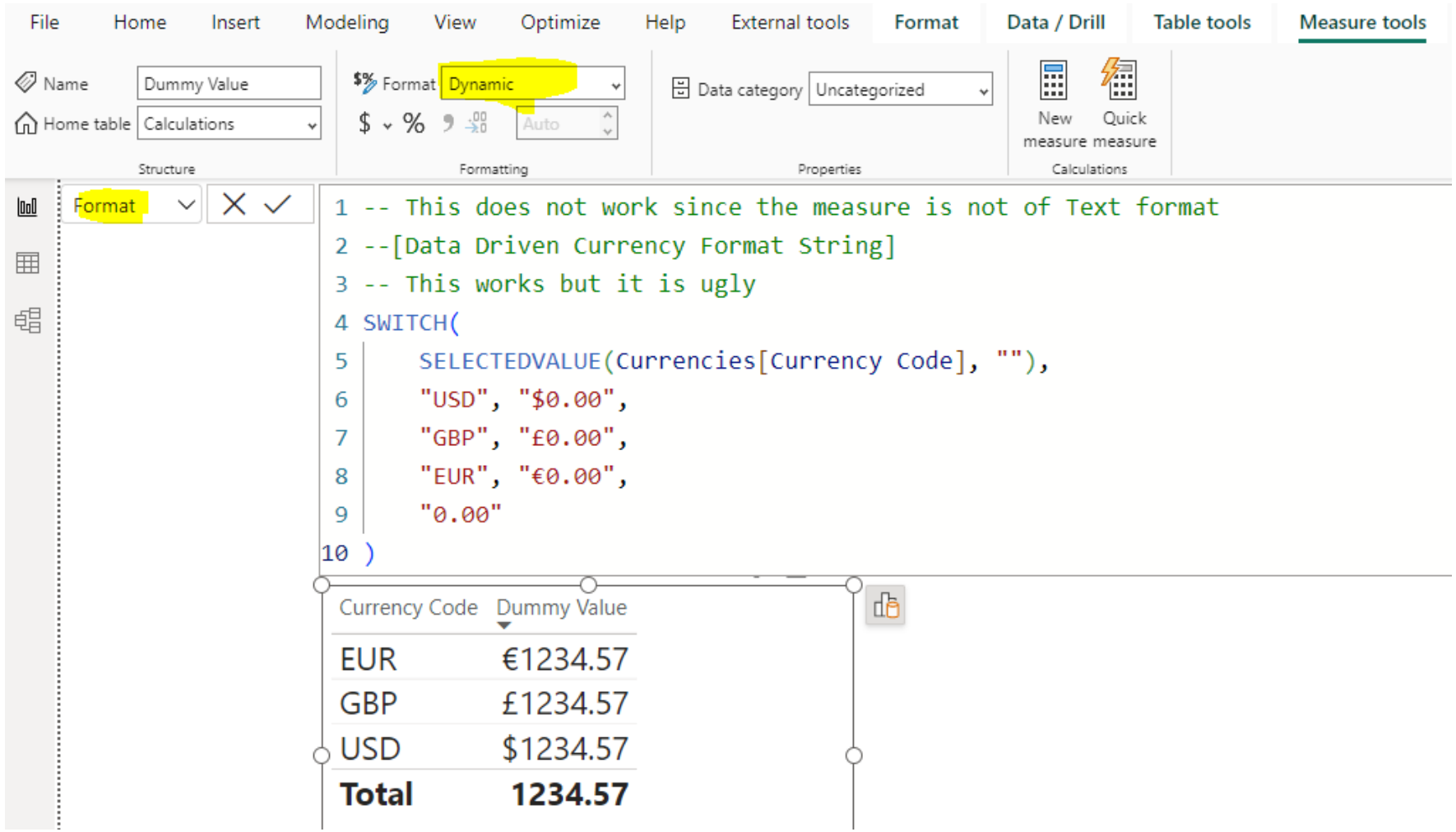








Dynamic formats



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

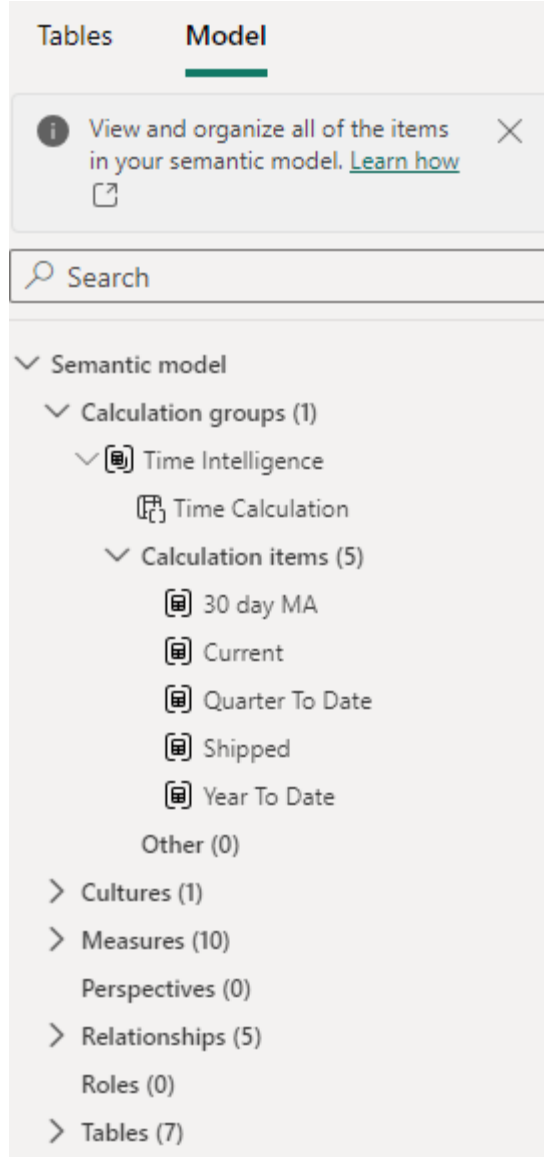
- Simpler, more versatile and more efficient than traditional methods for time intelligence measures
- Many people already know window functions from SQL or Python

```
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)

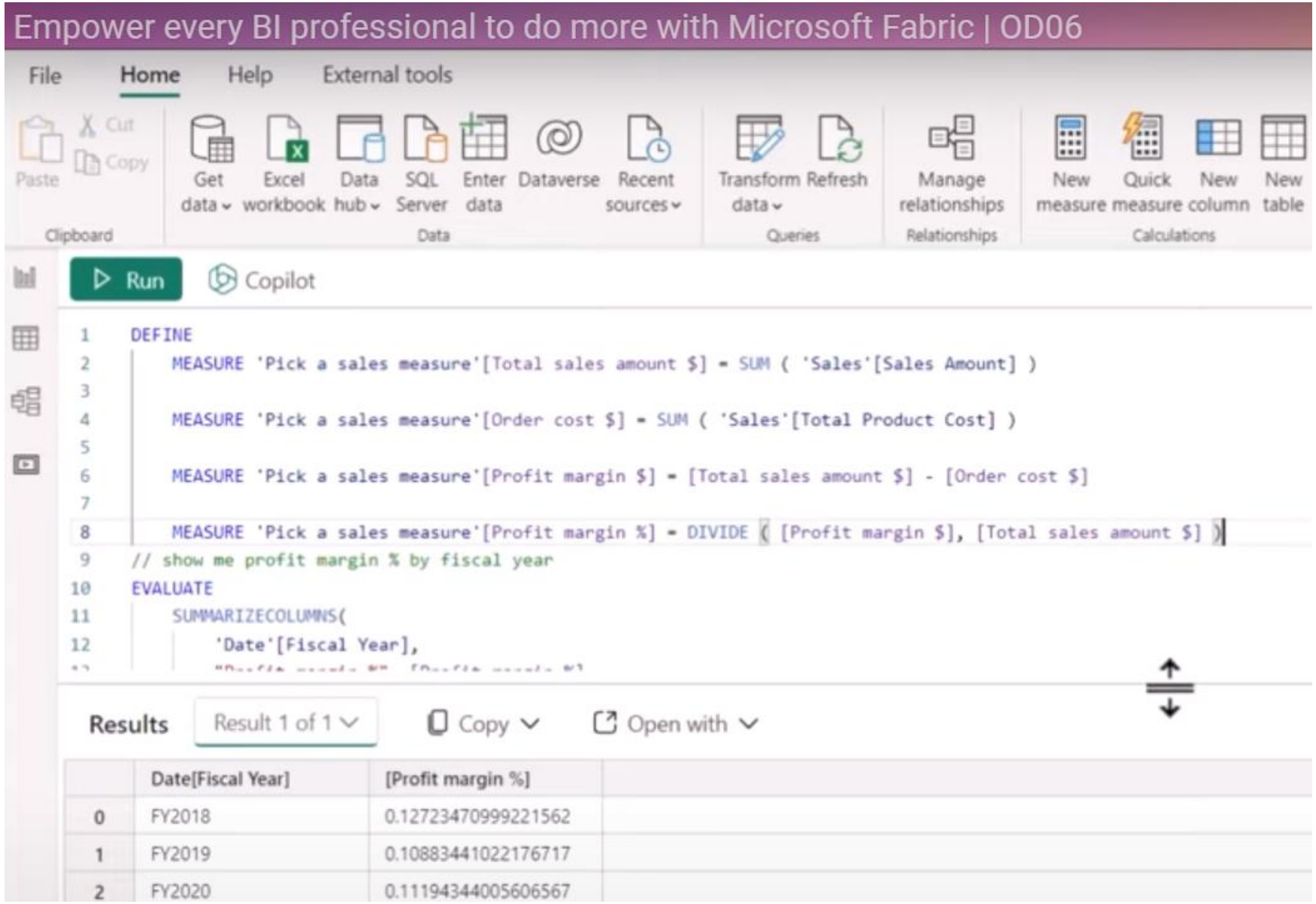
- Useful for large models with many base measures e.g. [Sales], [Profit], [Cost],…, all with time intelligence derivatives e.g. YTD, QTD, 10 day moving average, Q on Q,…
- Avoids building hundreds of measures
- Create these in new “Semantic Model” tree



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)



Ignite Power BI Developer Experiences talk by Microsoft PMs, Christian Wade, Rui Romano and Zoe Douglas (1 hour YouTube video) https://youtu.be/OdkS7DF7ElY?si=u8xWZv5nWjq3gdXy

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

```
EVALUATE VALUES(Rainfall[Month])

EVALUATE
ADDCOLUMNS(
    VALUES(Rainfall[Month]),
    "@Rainfall", [Average Daily Rainfall]
    )

DEFINE TABLE SlimMonthlyTable = VALUES(Rainfall[Month])
EVALUATE SlimMonthlyTable

DEFINE VAR SlimMonthlyTable = VALUES(Rainfall[Month])
EVALUATE ADDCOLUMNS(
    SlimMonthlyTable,
    "@Rainfall", [Average Daily Rainfall]
    )

DEFINE VAR SlimMonthlyTable = VALUES(Rainfall[Month])
VAR FullMonthlyTable =
ADDCOLUMNS(
    SlimMonthlyTable,
    "@Rainfall", [Average Daily Rainfall]
    )
EVALUATE
{AVERAGEX(
    FullMonthlyTable,
    [@Rainfall]
    )
}
```

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?)

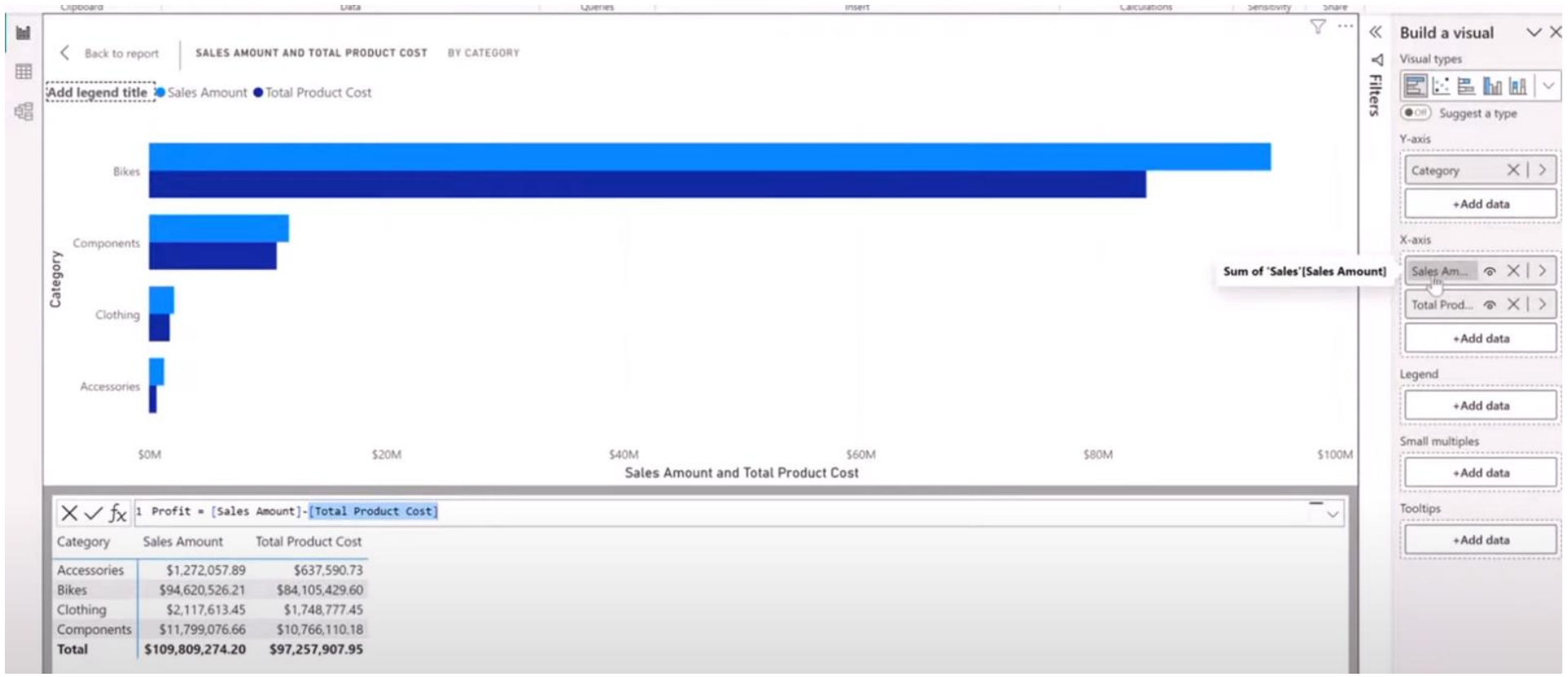
- Write simple DAX calculations defined on a specific visual
- Can refer to fields on the visuals, executed in scope of the visual
- Use visual matrix with rows columns
- Standard template for typical calculations: e.g. running sum
- Some new high level DAX visual functions:
  - Positional: FIRST(), 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

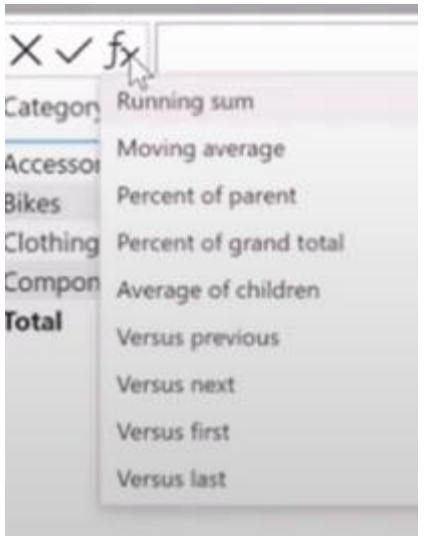
Visual Calculations: Why?

- 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

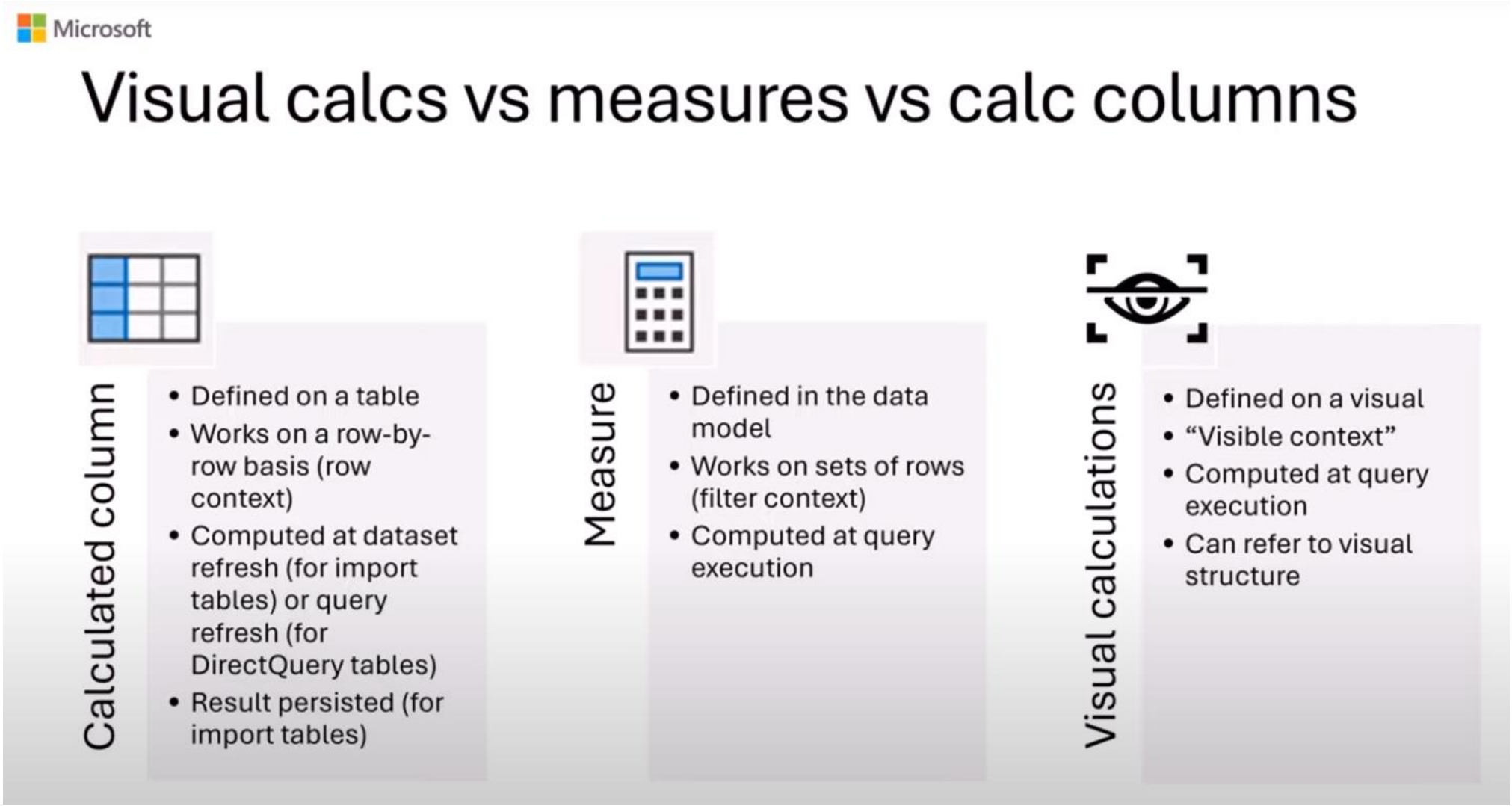
Date	Value	Running Sum
13/11/2023	3	3
14/11/2023	4	=D5+C6
15/11/2023	9	16
16/11/2023	4	20
17/11/2023	4	24
18/11/2023	5	29
19/11/2023	6	35
20/11/2023	3	38
21/11/2023	3	41
22/11/2023	3	44

Visual calcs: Snapshots (Microsoft PM video)

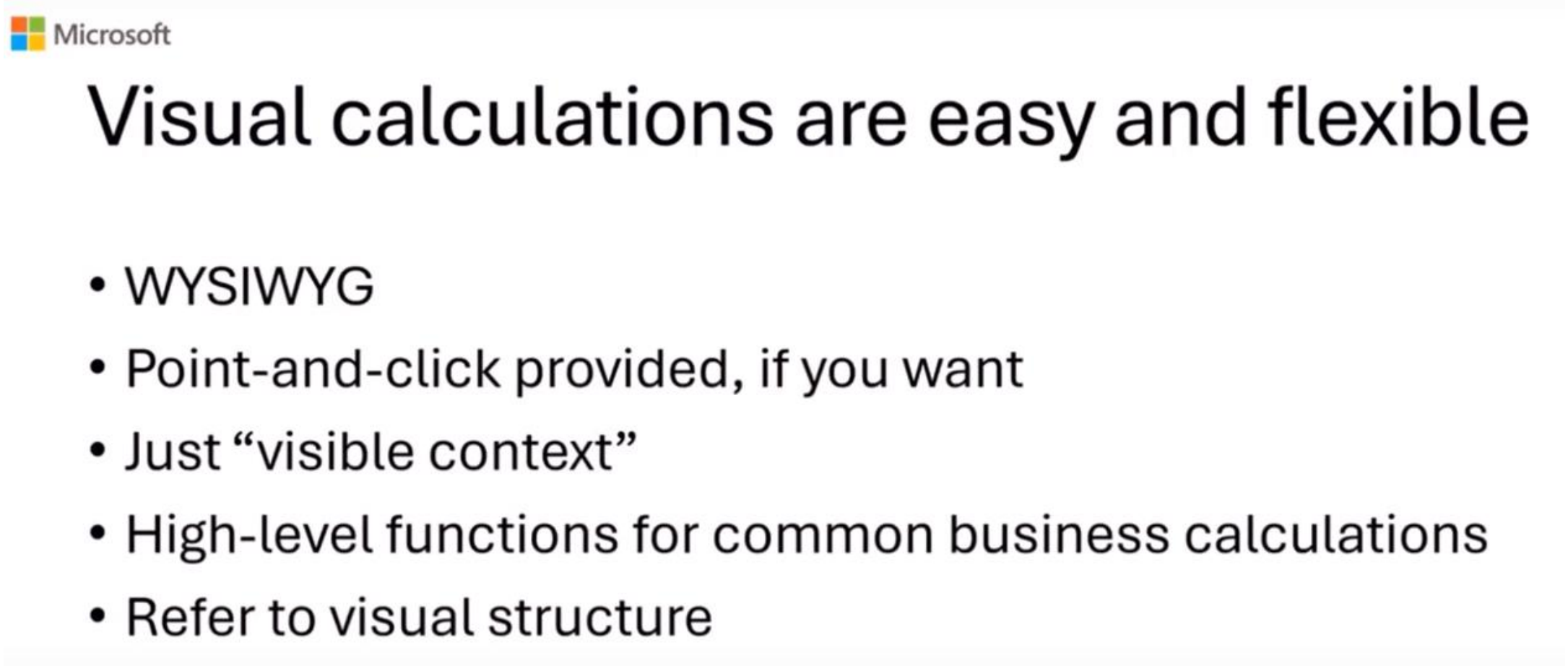




Visual calcs: Snapshots (Microsoft PM video)



Visual calcs: Snapshots (Microsoft PM video)

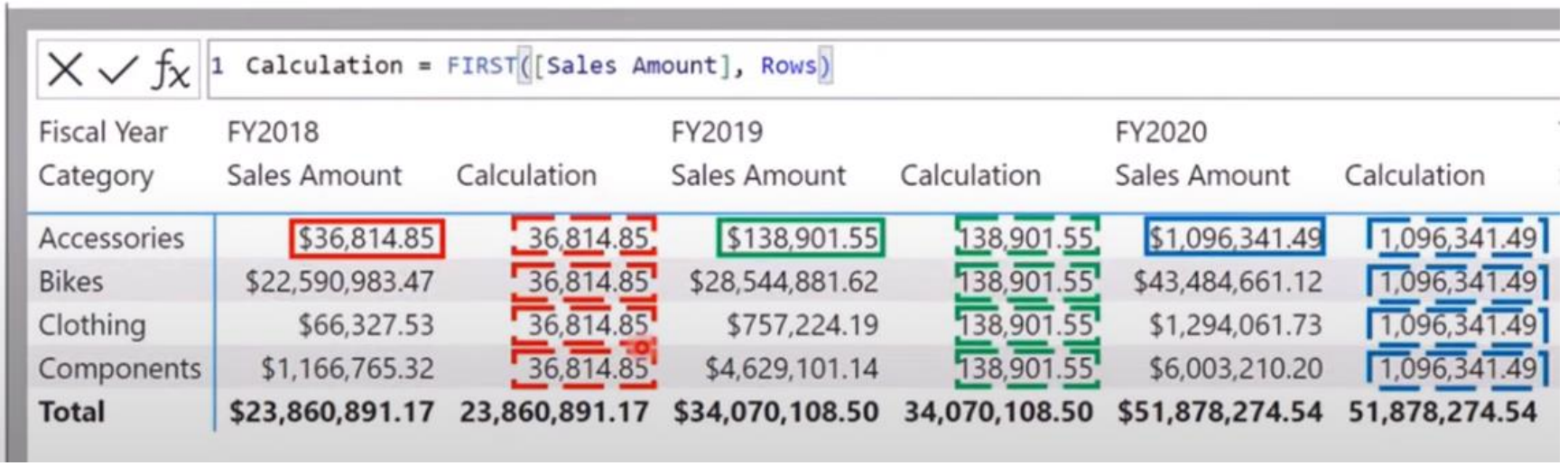


Visual calcs: Snapshots (Microsoft PM video)

Microsoft

Example: Axis

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



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 then 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

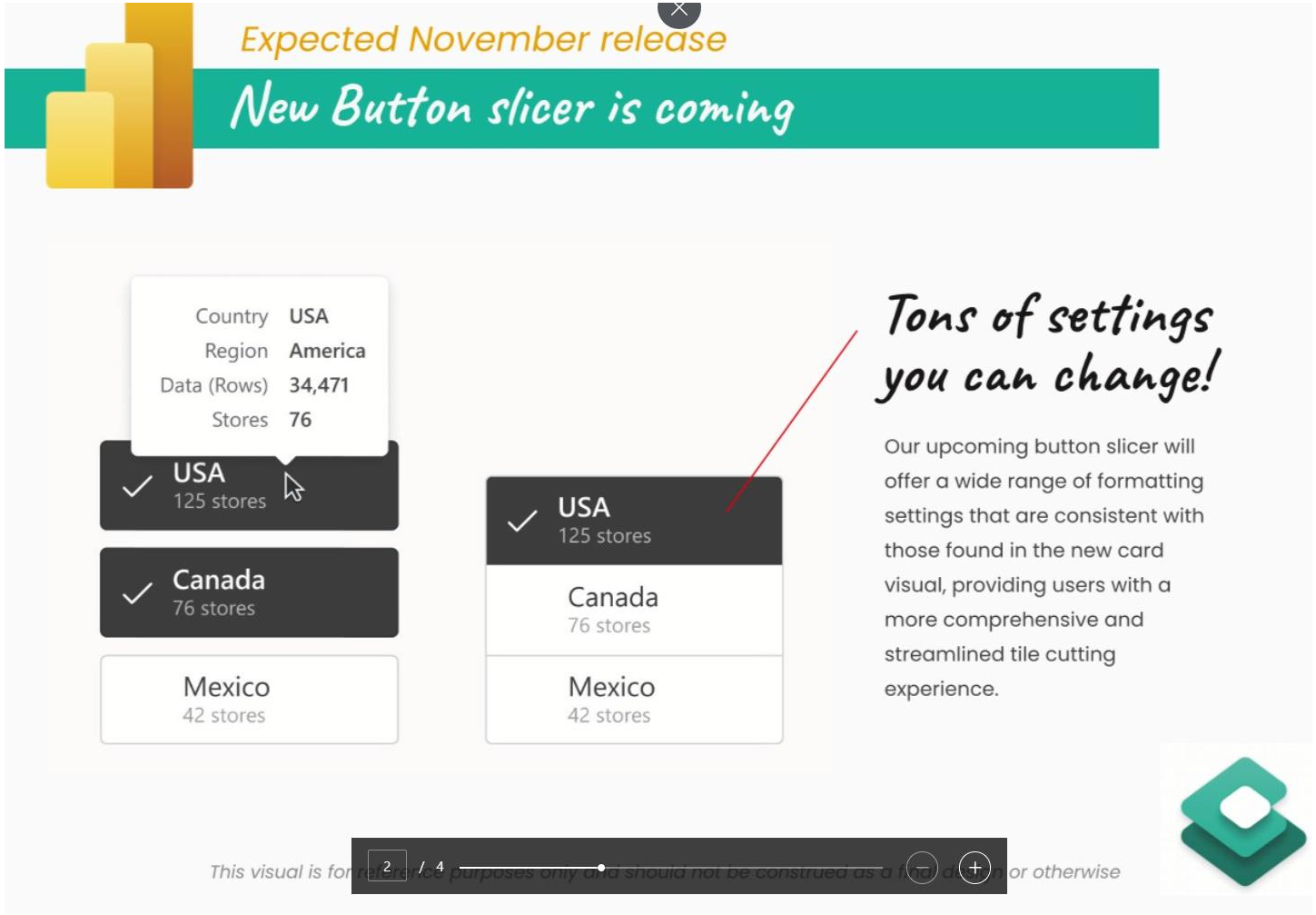
- Different style of bars: outline rather than solid. For example use solid for actuals and outline for forecast
- Better data labels: add trend as well as value
- Technique:| Using measures as data label (Helen Wall reference)

- 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/

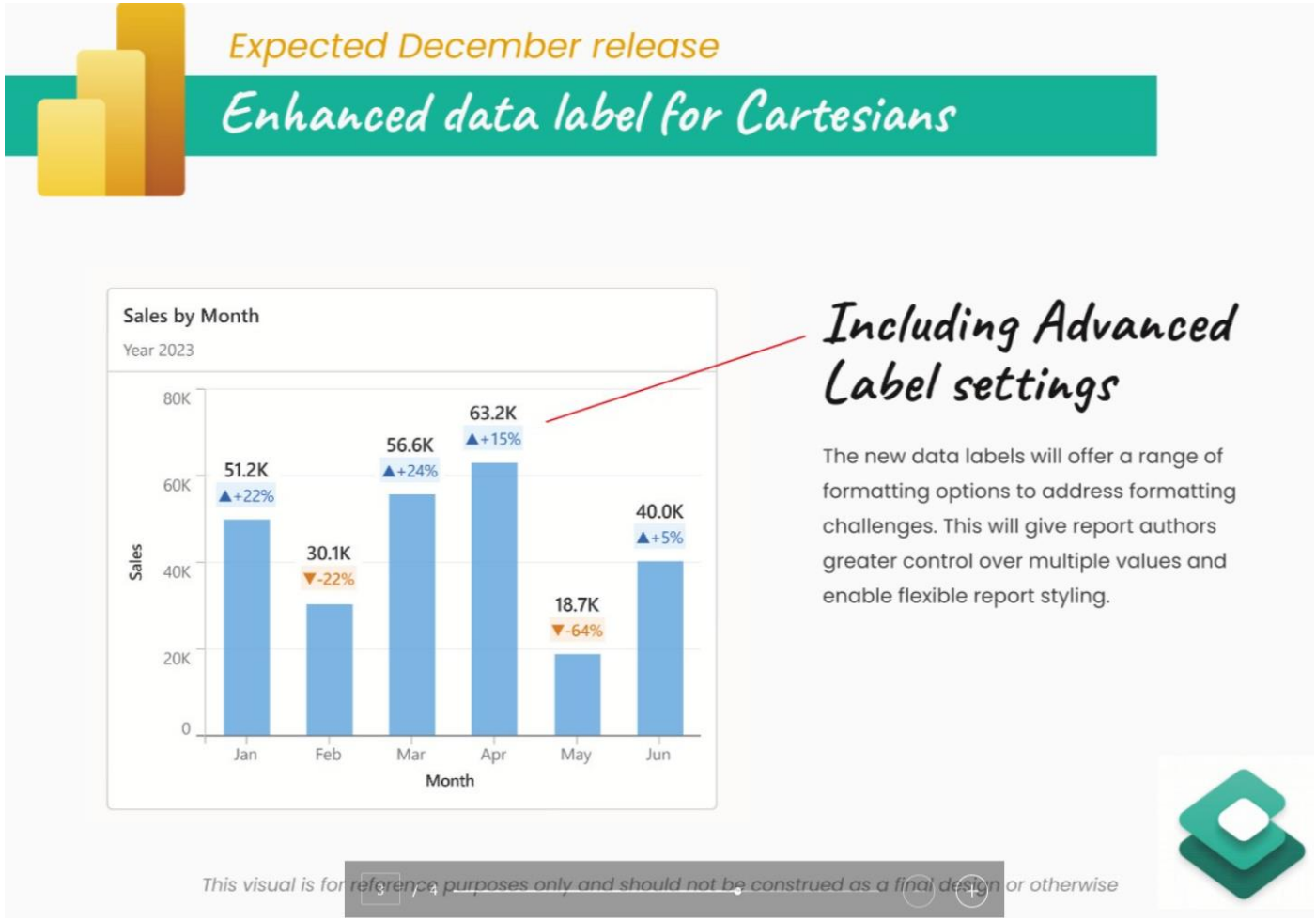
LinkedIn, Armand van Amersfoort, Nov 2023



Better data labels: Snapshots

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

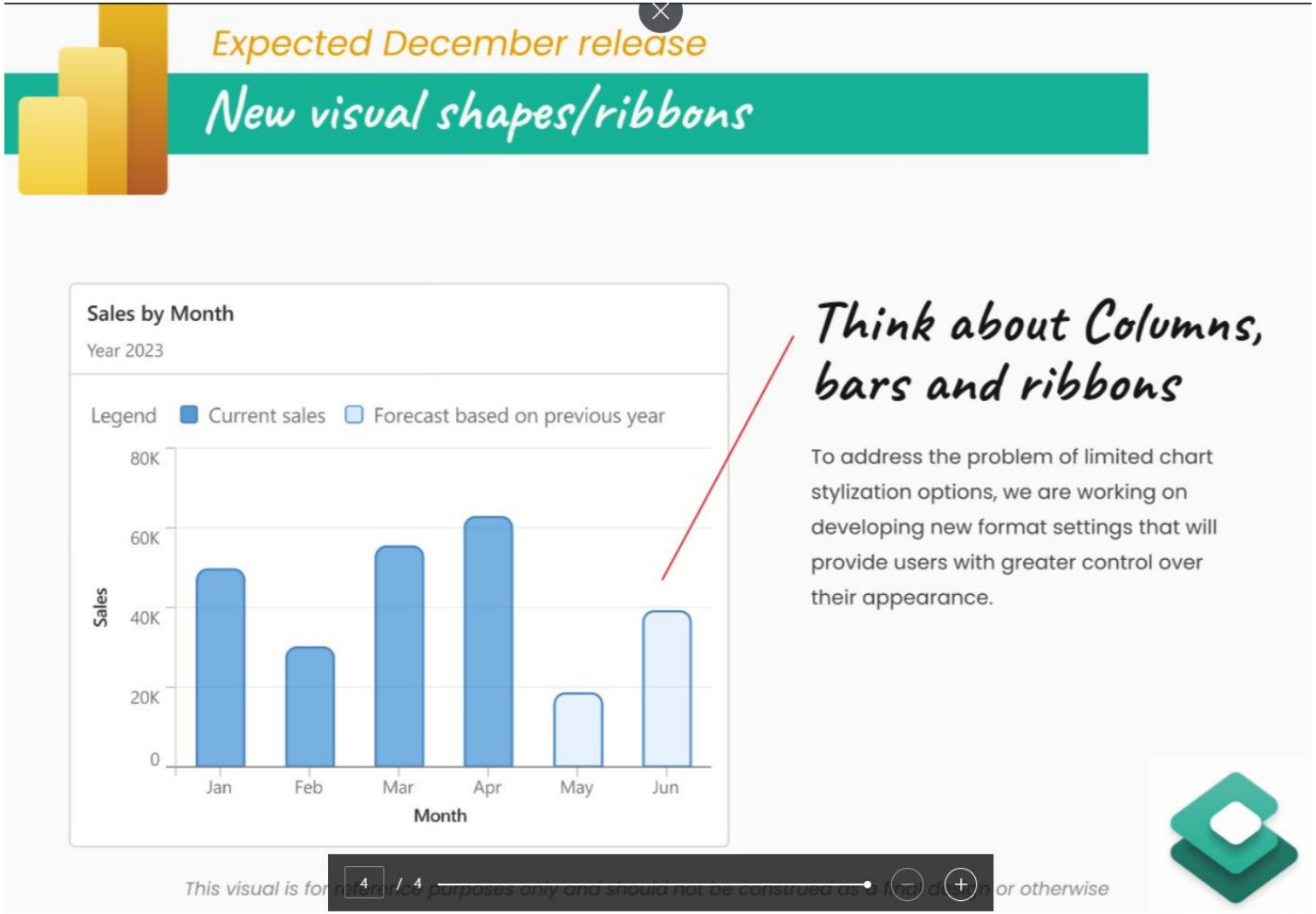
LinkedIn, Armand van Amersfoort, Nov 2023



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.

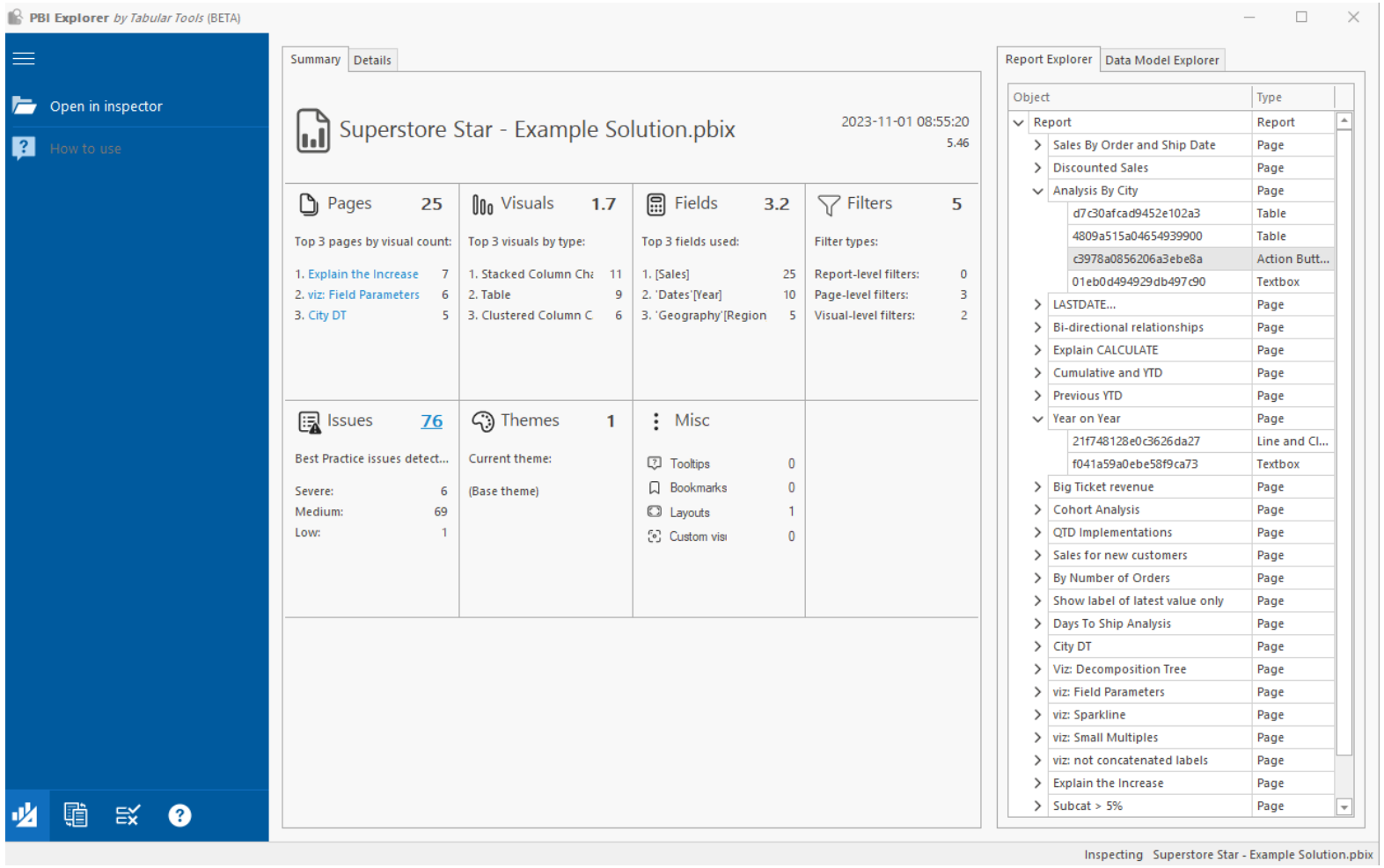
Understand how the data model is used by reports

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.



I should have paid more attention to ..

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