

SumProduct

NEWSLETTER #79 - June 2019

www.sumproduct.com | www.sumproduct.com/thought



It's been a busy month

for SumProduct as we have been in Amsterdam, London, Manchester, Manila and Royal Tunbridge Wells, seeing clients, marketing and arranging events. Yet somehow, we still manage to put together a 9,000-page newsletter!

OK, so my father used to say, "...for the billionth time, stop exaggerating!" – but there *is* plenty to read in this June newsletter. For example, we report back on the Amsterdam Summit and remind you about the upcoming Australian one (and there are *more* coming before the end of the year too!).

We discuss tips for building a forecast model with historical data, provide all of our usual update stories (including one on PivotCharts for Excel Online), plus you can "count" on our A to Z of Excel Functions too.

Something for everyone as always. Except paracetamol.

Until next month.

Liam Bastick, Managing Director, SumProduct



Excel Summit South: Website Up, Bookings Being Taken

EXCEL SUMMIT SOUTH 2019

Brisbane | Melbourne | Perth | Sydney

The website is now truly up and running, and tickets are now available! The fourth annual Excel Summit is getting ready for action and you can be a part of the action – for a much-reduced fee based on last year's prices, thanks to the Commonwealth Bank of Australia's generous sponsorship.

The dates have been confirmed as follows:

- **Brisbane:** Monday 29 and Tuesday 30 July
- **Sydney:** Thursday 1 and Friday 2 August
- **Melbourne:** Monday 5 and Tuesday 6 August
- **Perth:** Thursday 8 and Friday 9 August.

Check out the website at www.excelsummitsouth.com for all the details, prices and registration. And here is a bonus: until Friday 14 June, you can get 10% off the prices by using the code **NEWSLETTER10**. Hopefully, we will see you there!

contact@sumproduct.com | www.sumproduct.com | +61 3 9020 2071



Amsterdam Summit

Talking of Excel Summits, the sixth running of Amsterdam was held in mid-May, and SumProduct's **Liam Bastick** was there to talk about Power Query (Get & Transform) and Financial Modelling.



Left to right: MVPs Chandoo, Liam Bastick, Charles Williams, Jan Karel Pieterse, Oz du Soleil, Henk Vlootman, Hervé Thiriez, Leila Gharani, Ingeborg Hawighorst, Gašper Kamenšek and Microsoft's Itai Goldstein.

It was extremely successful with experts flying in from Austria, Australia, France, Germany, India, Israel, the Netherlands, New Zealand, Slovenia, the UK and the USA. Subjects included decoding complex formulae, Power Pivot, Power Query, Power BI, financial modelling, charts and dashboards. And this was just for the one-day event!

One thing to note: the raffle went into legend. Attendees were picked out of an Excel randomiser (what else?) and were able to choose from an

array of prizes including our very own Introduction to Financial Modelling book, various other Excel books donated by Microsoft / the MVPs and some chocolate from overseas. As the prizes dwindled, it was noted that winners preferred the chocolate to books on PivotTables!! Has Power Pivot finally come of age..?

Historical vs. Actual vs. Forecast

Many accountants and analysts have to analyse trends using past and current performance in order to create informed forecasts for the future. It's a common problem that causes consternation and bad modelling far too frequently. However, it can be circumvented using a simple layout and application of previous functions discussed.

This requires three distinct datasets:

1. **Historical information:** data that has now been gathered that actually happened. This is often recorded in Management Information Systems, general ledgers, tax returns, *etc.* This data tends to have one thing in common: it's a matter of *fact*
2. **Actual data:** whilst historical information summarises past data that has been locked down, "actuals" may be slightly more fluid in that data may be available for the first nine months of the year (*say*) combined with three months' projections, based upon predicted orders, work schedules, hours of operation and so on
3. **Forecast projections:** these numbers are usually based upon calculations and correlations with the historical / actual data. The first two datasets may be distinct and independent, but typically, there is a relationship with at least one of the other types and forecast projections.

You may or may not agree with the above summation. That's not really the point here. The key takeaway here is that when we undertake financial modelling, we often have more than one situation that occurs over the time periods in the same row of the financial model, *e.g.*

	Date 1	Date 2	Date 3	Date 4	Date 5	Date 6	Date 7	Date 8	Date 9	Date 10	Date 11	Date 12
Line Item	Historical				Actual		Forecast					

The problem with data set out like this is two-fold:

- Less experienced modellers will use three types of data entry for the three segments, which may be a combination of formulae and / or hard code. The concern is someone else will come along and copy one formula across the entire row which will not be appropriate for the other data categories
- More experienced modellers acknowledge one consistent formula must be used across the row – but then try and do all calculations in the one cell and construct a formula that typically starts =IF(IF(... *etc.*

Neither is optimal, but the second option is nearer the truth. The problem with =IF(IF(...) is it intimidates end users who try to follow it, with a succession of nested calculations. Office 365's and Excel 2016's IFS function is even worse as people have to understand both the concept of nesting and the prevailing syntax. Stepping out seems way simpler. And it gives me an opportunity to bring together quite a few of my past articles in an applied example.

Let me explain. I am going to take start with an example with three scenarios:

- **Historical:** in my example, these will be inputs, but they could link to another file (*say*)
- **Actual:** in my example, again, these will be inputs, but this could be a calculation based on actual data for the period to date and then extrapolated (*say*)
- **Forecast:** in my example, this will be a formula.

You could have five situations, two linked, one hard coded and two formulaic – the idea would remain the same. The first thing is to collate all necessary inputs:

Assumptions

Historical v Actual v Forecast

	2015	2016	2017	2018	2019	2020	2021
Historical	55	65	75	85	95	105	115
Actual	60	69	78	87	96	105	114
Forecast		2.00%	2.50%	3.00%	3.50%	4.00%	4.50%

The plan here is to use the first row of data for historical periods, the second row for the actual period, and then use the growth rate on the previous period for forecasts. Clearly, only one value would be used in each year (column).

Next, I created a selector for which year was the actual year:

Assumptions

Years

Final Year of Historical
Actual
Forecast From

Year
2017
2018
2019

Change date here

The input cell (Actual, in yellow and using data validation) drives the other two dates, as Historical would have to be all periods up to the year before and Forecast from the following year. With more situations, there would have to be more inputs, but the general idea would remain similar.

I can now create a “flag” system (1 for on / yes, 0 for off / no) to denote which situation relates to which period:

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
17																
18																
19																
20																
21																
22																
23																
24																
25																
26																
27																
28																
29																
30																
31																
32																
33																
34																
35																
36																
37																
38																
39																
40																
41																
42																
43																
44																

The Historical flags (row 37 of the image above) are 1 for years 2015, 2016 and 2017 using the formula

$$=(G\$36<=\$G25)*1$$

in cell **G37**, for example. This checks that the years in question is less than the value in cell G25, which is 2017. Similarly, the Actual formula is

$$=(G\$36=\$G26)*1$$

and the Forecast formula is

$$=(G\$36>=\$G27)*1$$

which means all formulae are simple.

A check has been put in row 41 to ensure there is a ‘1’ (but only one ‘1’) in each column (year) too:

$$=(SUM(G37:G39)<>1)*1$$

Finally, the Choice (row 43) uses the **MATCH** function to determine where the 1 in each column is, *i.e.* 1 if it is in the first row, 2 if it is in the second, *etc.*

$$=MATCH(1,G\$37:G\$39,0)$$

MATCH(lookup_value,lookup_array,[match_type]) returns the relative position of an item in an array that (approximately) matches a specified value. It is not case sensitive.

The third argument, **match_type**, does not have to be entered, but for most situations, I strongly recommend that it is specified. It allows one of three values:

- **match_type 1 [default if omitted]**: finds the largest value less than or equal to the lookup_value – but the lookup_array must be in strict ascending order, limiting flexibility;
- **match_type 0**: probably the most useful setting, MATCH will find the position of the first value that matches lookup_value exactly. The lookup_array can have data in any order and even allows duplicates; and
- **match_type -1**: finds the smallest value greater than or equal to the lookup_value – but the lookup_array must be in strict descending order, again limiting flexibility.

When using **MATCH**, if there is no (approximate) match, #N/A is returned (this may also occur if data is not correctly sorted depending upon **match_type**).

For example:

	E	F
12	1	a
13	2	b
14	3	a
15	4	c
16	5	b
17	6	d
18	7	d
19	8	e
20	9	f
21	10	a
22	11	c

In the figure above, **MATCH("d",F12:F22,0)** gives a value of 6, being the relative position of the first 'd' in the range. Note that having **match_type** 0 here is important. The data contains duplicates and is not sorted alphanumerically. Consequently, **match_types** 1 and -1 would give the wrong answer: 7 and #N/A respectively.

Returning to the 1's and 0's section (the latter displayed as dashes in the graphic), these values have also been used to drive conditional formatting in the original input section too, viz.

Assumptions

Historical v Actual v Forecast

Historical
Actual
Forecast

2015	2016	2017	2018	2019	2020	2021
55	65	75				
			87			
				3.50%	4.00%	4.50%

To create grey cells, I have simply highlighted the entire range and use conditional formatting (in the 'Styles' section of the 'Home' tab of the Ribbon, **ALT + O + D**) as follows:

Edit Formatting Rule
?
X

Select a Rule Type:

- Format all cells based on their values
- Format only cells that contain
- Format only top or bottom ranked values
- Format only values that are above or below average
- Format only unique or duplicate values
- Use a formula to determine which cells to format

Edit the Rule Description:

Format values where this formula is true:
=G37=0

Preview:
Format...

OK
Cancel

We are pretty much done now. All that is needed is a formula for the output:

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
45																		
46																		
47																		
48																		
49																		
50																		
51																		
52																		
53																		
54																		

The formula

=CHOOSE(G\$43,G\$13,G\$14,F53*(1+G\$15))

in cell **G53** uses the value in row 13 if the choice is 1 (Historical selected), in row 14 if the choice is 2 (Actual) and multiplies the previous period by 1 plus the growth rate in row 15 if the choice is 3 (Forecast) using the **CHOOSE** function.

The **CHOOSE** function employs the following syntax to operate:

CHOOSE(index_number, value1, [value2])

As a reminder, the **CHOOSE** function has the following arguments:

- **index_number**: this is required and is used to specify which value argument is to be selected. The **index_number** must be a number between 1 and 254, or a formula or reference to a cell containing a number between 1 and 254.
 - o if **index_number** is 1, **CHOOSE** returns **value1**; if it is 2, **CHOOSE** returns **value2**; and so on
 - o if **index_number** is less than 1 or greater than the number of the last value in the list, **CHOOSE** returns the **#VALUE!** error value
 - o if **index_number** is a fraction, it is truncated to the lowest integer before being used.
- **value1, value2, ...: value 1** is required, but subsequent values are optional. There may be between 1 and 254 value arguments from which **CHOOSE** selects a value or an action to perform based on **index_number**. The arguments can be numbers, cell references, defined names, formulas, functions, or text.

The great thing about this method is regardless of whether end users know about conditional formatting, **INDEX** or **CHOOSE**, the actual final calculation is easy to follow on paper, viz.

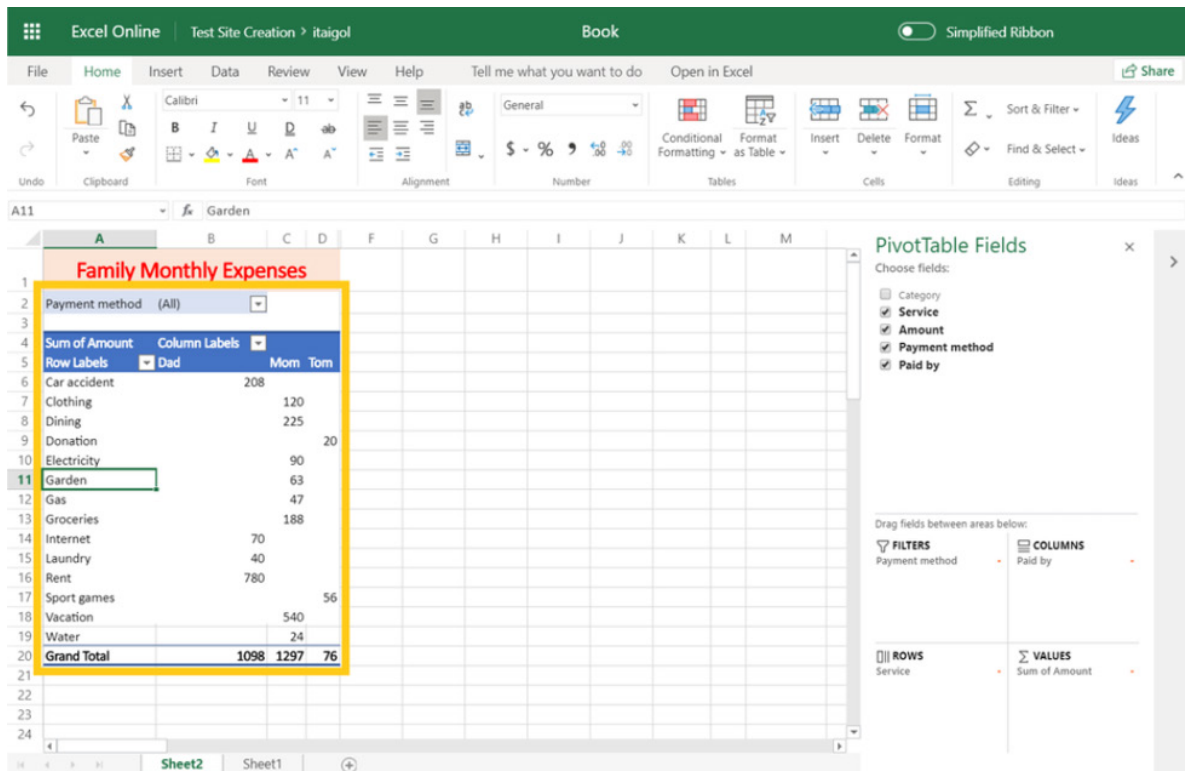
	A	B	C	D	E	F	G	H	I	J	K	L	M	N
5														
6														
7														
8														
9														
10														
11														
12														
13														
14														
15														
16														
17														
18														
19														
20														
21														
22														
23														
24														
25														
26														
27														
28														
29														
30														
31														
32														
33														
34														
35														
36														
37														
38														
39														
40														
41														
42														
43														
44														
45														
46														
47														
48														
49														
50														
51														
52														
53														
54														

And that's what financial modelling is all about!

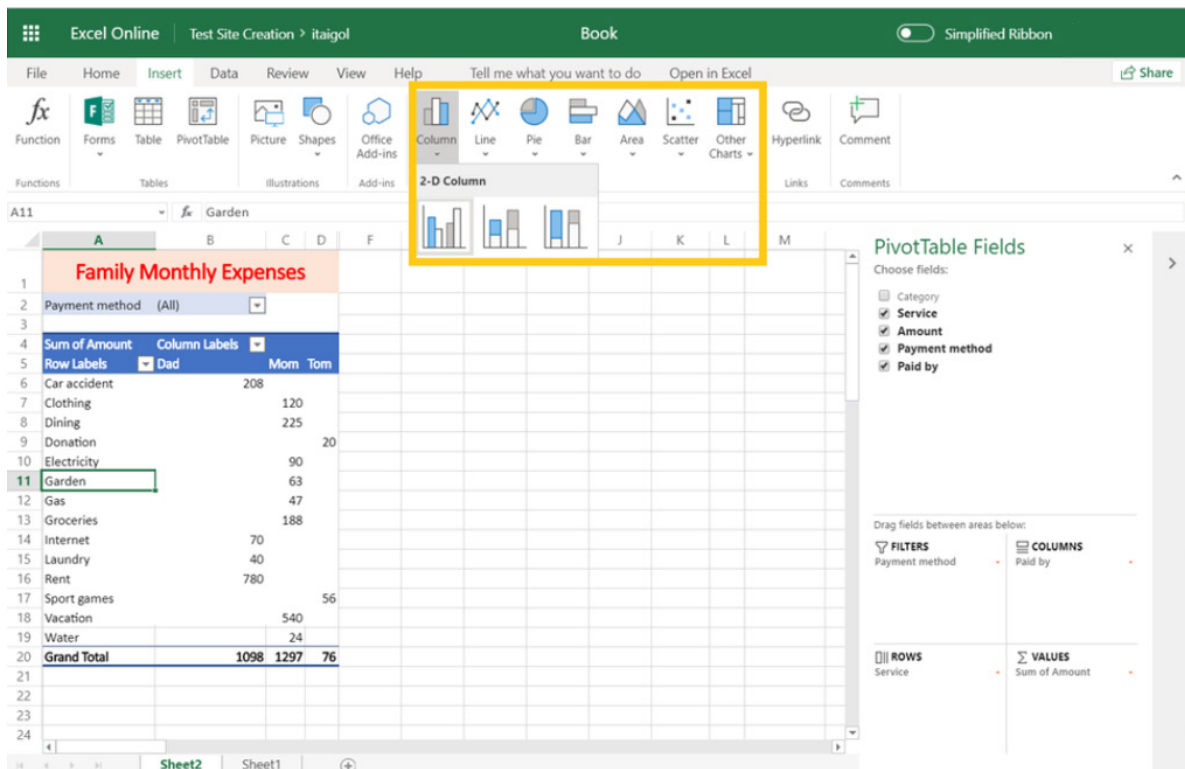
PivotCharts Now Available in Excel Online

PivotCharts – the graphical representation of PivotTables – have been a noticeable omission for Excel Online for some time now. But that's all changed now. Just like in Excel Desktop, now you can create your own PivotCharts and edit them in the Excel web application.

Go to the PivotTable you'd like to visualise or insert a new PivotTable. Then, select a cell in your PivotTable.

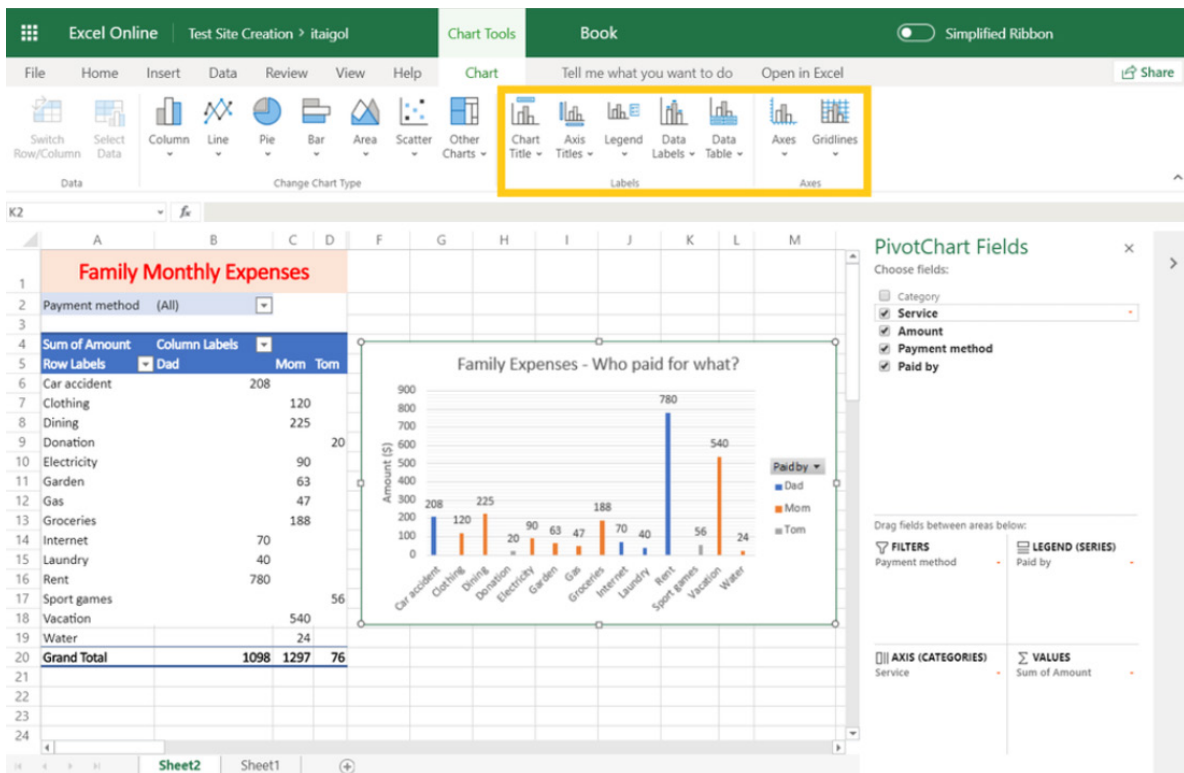


Select the Insert tab on the Ribbon, click on one of the chart buttons and choose your PivotChart type.

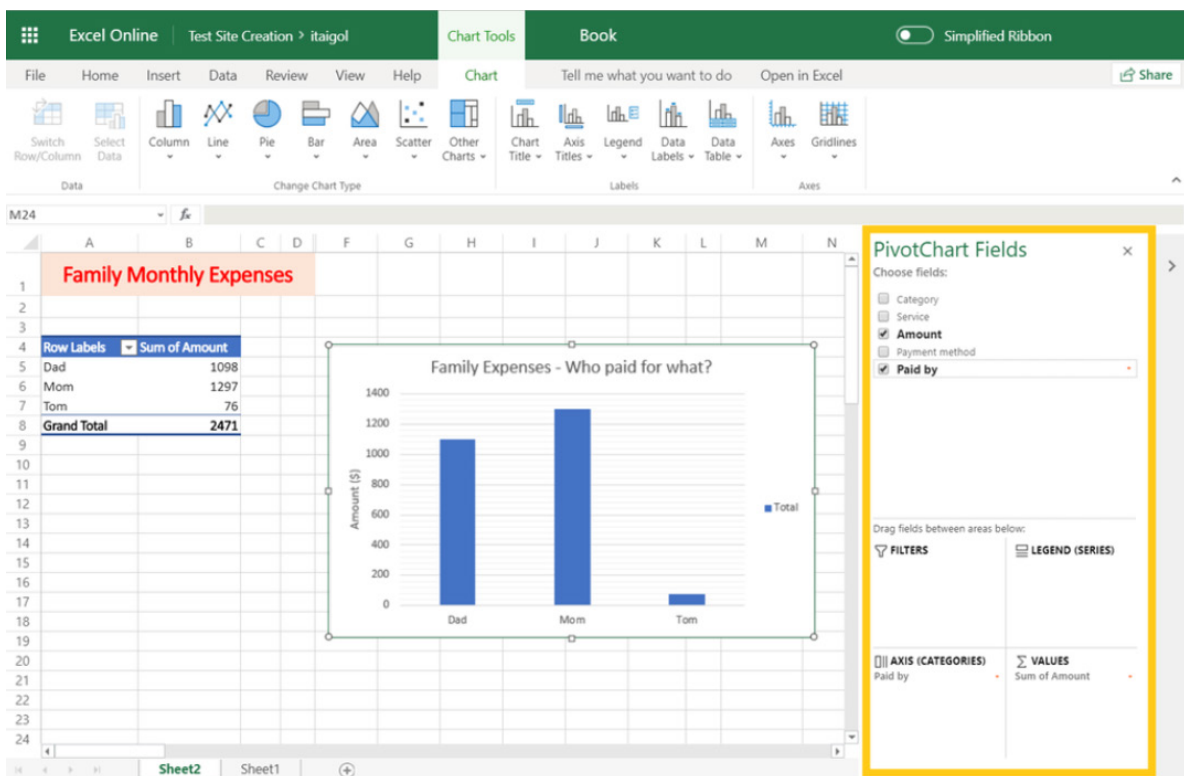


You will now have a PivotChart in Excel Online!

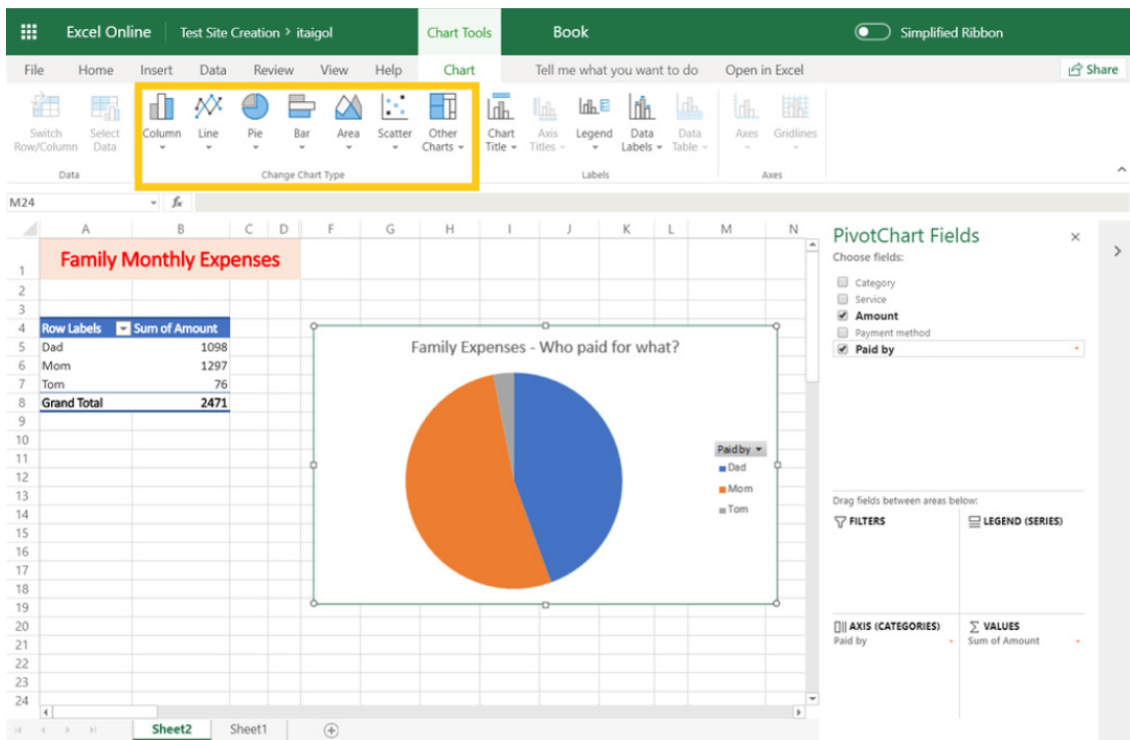
Next, you can start to play. Just like in Desktop Excel, you can utilise the options on the 'Chart Tools' contextual tab for editing titles, legend, labels, axes or gridlines.



You may also edit your PivotTable fields. The changes will be reflected immediately on the interactive PivotChart.



Finally, you can change the chart type for getting another point of view. Select the PivotChart and choose one of the other chart buttons instead.



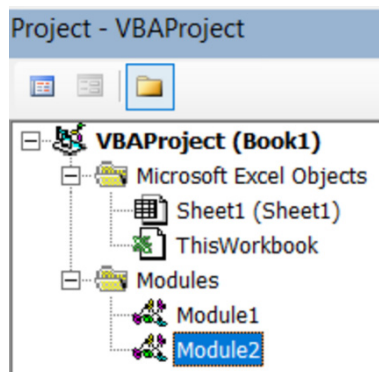
Visual Basics

We thought we'd run an elementary series going through the rudiments of Visual Basic for Applications (VBA) as a springboard for newer users. This month, we look at the concept of a variable scope at a project level.

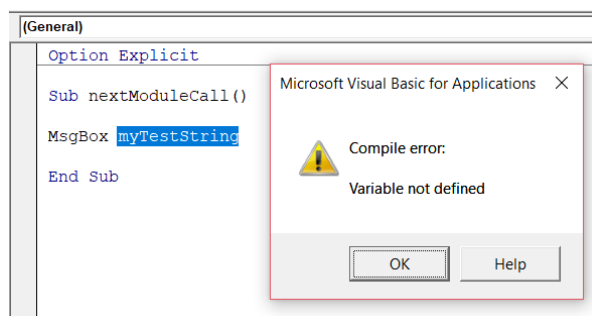
Variables can also be used in different subroutines and functions. If there are items that are known to have constant values throughout the entire workbook, they can be declared explicitly in one place for easy reference.

Project Level

Sharing variables between procedures within the same module was very simple, but what if another module wants to reuse the same variable?



Module2 wants to use **myTestString** but it is declared in **Module1**.



Upon running, it again says the 'Variable not defined'. In order to force a variable to be accessible to the entire workbook the statement, the 'Public' statement may be used.

```
(General)

Option Explicit

Public myTestString As String

Sub ScopeTest()

myTestString = "Hello World!"

End Sub

Sub NextScopeTest()

MsgBox myTestString

End Sub
```

Now this will be accessible from **Module2** so when the following code is called.

```
(General)

Option Explicit

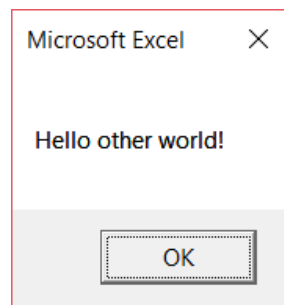
Sub nextModuleCall()

myTestString = "Hello other world!"

MsgBox myTestString

End Sub
```

The variable is used as necessary and the expected result eventuates.



More next month.

Power Pivot Principles

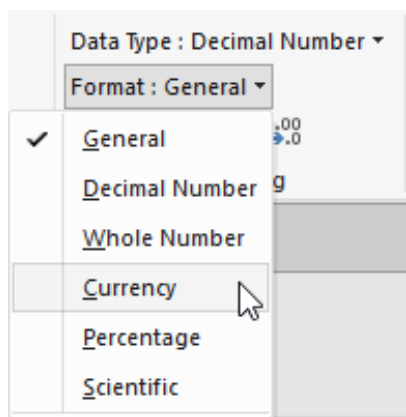
We continue our series on the Excel COM add-in, Power Pivot. This month, we continue to look at how to import data from an Access database.

Not all databases are created equal; some are perfect, some need formatting adjustments. Our Access Database from last month is not perfect – let's perform some formatting adjustments!

Rename the table 'FactInternetSales' to 'Sales' by typing it in, in the bottom tab (simply double-click or right-click on the tab). This will clearly identify each table that's imported into PowerPivot.



Some columns will have to be updated to ensure they are showing the correct Data Type and Format, e.g. a whole number, currency, text, date. We also have to specify if they are a whole number, decimal number, currency etc.



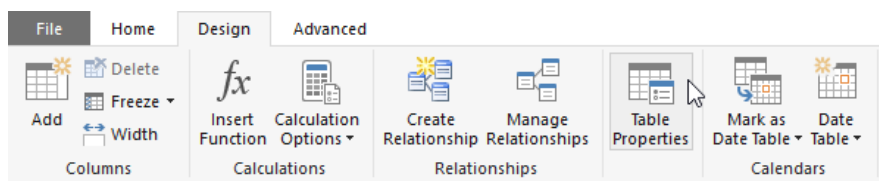
Do remember the following:

- **Formatting:** Changing the formatting changes the appearance but not the actual contents. It is merely 'cosmetic'. For example, the value '3.7' formatted as a whole number would appear as '4', but if you multiplied it by 10 you would get '37'
- **Data Type:** Changing the data type may remove information. For example, changing the Data Type from 'Decimal Number' to 'Whole Number' for the value '3.7' would round this number to '4'. Multiplying this value would give a result of '40'. When information is at risk, Power Pivot will provide a warning before the change takes place. In any case, the data may be restored simply by refreshing the source.
You cannot damage the underlying data.

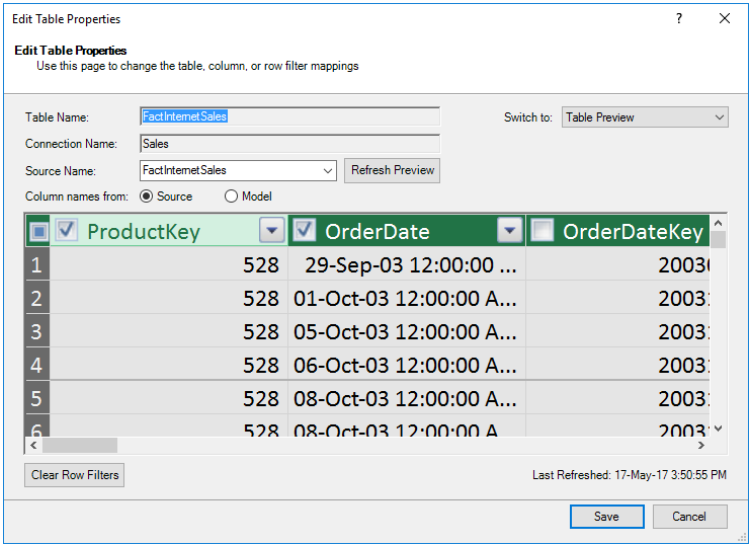
In the 'Formatting' section of the 'Home' tab select the drop-down box next to 'Data-Type'. Go through each column and format each field to ensure that it is showing the correct data type and in the correct format: for instance, the 'SalesAmount' column, the format should be changed from 'General' to 'Currency', viz.

	ProductKey	OrderDate	CustomerKey	OrderQuantity	SalesAmount
1	477	01-Aug-03 1...	16982	1	\$4.99
2	477	02-Aug-03 1...	16781	1	\$4.99
3	477	03-Aug-03 1...	21918	1	\$4.99
4	477	06-Aug-03 1...	16953	1	\$4.99
5	477	07-Aug-03 1...	17506	1	\$4.99
6	477	07-Aug-03 1...	16814	1	\$4.99
7	477	07-Aug-03 1...	18110	1	\$4.99
8	477	08-Aug-03 1...	23277	1	\$4.99
9	477	08-Aug-03 1...	17447	1	\$4.99
10	477	08-Aug-03 1...	18102	1	\$4.99
11	477	09-Aug-03 1...	18095	1	\$4.99
12	477	15-Aug-03 1...	17014	1	\$4.99
13	477	17-Aug-03 1...	17165	1	\$4.99
14	477	18-Aug-03 1...	19639	1	\$4.99
15	477	19-Aug-03 1...	18111	1	\$4.99
16	477	19-Aug-03 1...	18751	1	\$4.99
17	477	20-Aug-03 1...	17016	1	\$4.99
18	477	23-Aug-03 1...	18094	1	\$4.99
19	477	27-Aug-03 1...	18109	1	\$4.99

If a column in our original data was not originally brought in, we can bring it in by going to the 'Design' tab of the Power Pivot window and then select the 'Table Properties' on the Ribbon:



The 'Edit Table Properties' dialog box will appear, allowing us to select any columns that we have missed out earlier. Scroll across, select the 'Freight' column and click Save:



The Freight column will now appear on the sheet:

	ProductKey	OrderDate	CustomerKey	OrderQuantity	SalesAmount	Year	Month	Freight
1	477	01-Aug-03 1...	16982	1	\$4.99	2003	8	0.1248
2	477	02-Aug-03 1...	16781	1	\$4.99	2003	8	0.1248
3	477	03-Aug-03 1...	21918	1	\$4.99	2003	8	0.1248
4	477	06-Aug-03 1...	16953	1	\$4.99	2003	8	0.1248
5	477	07-Aug-03 1...	17506	1	\$4.99	2003	8	0.1248
6	477	07-Aug-03 1...	16814	1	\$4.99	2003	8	0.1248
7	477	07-Aug-03 1...	18110	1	\$4.99	2003	8	0.1248
8	477	08-Aug-03 1...	23277	1	\$4.99	2003	8	0.1248
9	477	08-Aug-03 1...	17447	1	\$4.99	2003	8	0.1248
10	477	08-Aug-03 1...	18102	1	\$4.99	2003	8	0.1248
11	477	09-Aug-03 1...	18095	1	\$4.99	2003	8	0.1248
12	477	15-Aug-03 1...	17014	1	\$4.99	2003	8	0.1248
13	477	17-Aug-03 1...	17165	1	\$4.99	2003	8	0.1248
14	477	18-Aug-03 1...	19639	1	\$4.99	2003	8	0.1248
15	477	19-Aug-03 1...	18111	1	\$4.99	2003	8	0.1248
16	477	19-Aug-03 1...	18751	1	\$4.99	2003	8	0.1248
17	477	20-Aug-03 1...	17016	1	\$4.99	2003	8	0.1248
18	477	23-Aug-03 1...	18094	1	\$4.99	2003	8	0.1248
19	477	27-Aug-03 1...	18109	1	\$4.99	2003	8	0.1248

Format it into a currency and save the file.

	ProductKey	OrderDate	CustomerKey	OrderQuantity	SalesAmount	Year	Month	Freight
1	477	01-Aug-03 1...	16982	1	\$4.99	2003	8	\$0.12
2	477	02-Aug-03 1...	16781	1	\$4.99	2003	8	\$0.12
3	477	03-Aug-03 1...	21918	1	\$4.99	2003	8	\$0.12
4	477	06-Aug-03 1...	16953	1	\$4.99	2003	8	\$0.12
5	477	07-Aug-03 1...	17506	1	\$4.99	2003	8	\$0.12
6	477	07-Aug-03 1...	16814	1	\$4.99	2003	8	\$0.12
7	477	07-Aug-03 1...	18110	1	\$4.99	2003	8	\$0.12
8	477	08-Aug-03 1...	23277	1	\$4.99	2003	8	\$0.12
9	477	08-Aug-03 1...	17447	1	\$4.99	2003	8	\$0.12
10	477	08-Aug-03 1...	18102	1	\$4.99	2003	8	\$0.12
11	477	09-Aug-03 1...	18095	1	\$4.99	2003	8	\$0.12
12	477	15-Aug-03 1...	17014	1	\$4.99	2003	8	\$0.12
13	477	17-Aug-03 1...	17165	1	\$4.99	2003	8	\$0.12
14	477	18-Aug-03 1...	19639	1	\$4.99	2003	8	\$0.12
15	477	19-Aug-03 1...	18111	1	\$4.99	2003	8	\$0.12
16	477	19-Aug-03 1...	18751	1	\$4.99	2003	8	\$0.12
17	477	20-Aug-03 1...	17016	1	\$4.99	2003	8	\$0.12
18	477	23-Aug-03 1...	18094	1	\$4.99	2003	8	\$0.12
19	477	27-Aug-03 1...	18109	1	\$4.99	2003	8	\$0.12

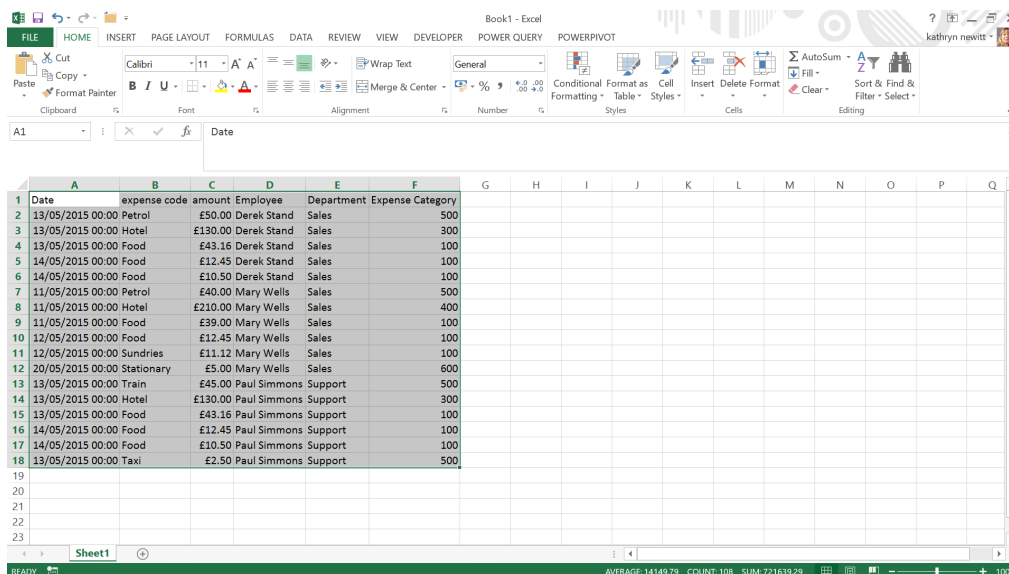
There you have it: we have just cleaned up the data from our Access database.

We'll continue next month. Stay tuned for our next post on Power Pivot. In the meantime, please remember we have training in Power Pivot which you can find out more about at www.sumproduct.com/courses/power-pivot-power-query-and-power-bi.

Power Query Pointers

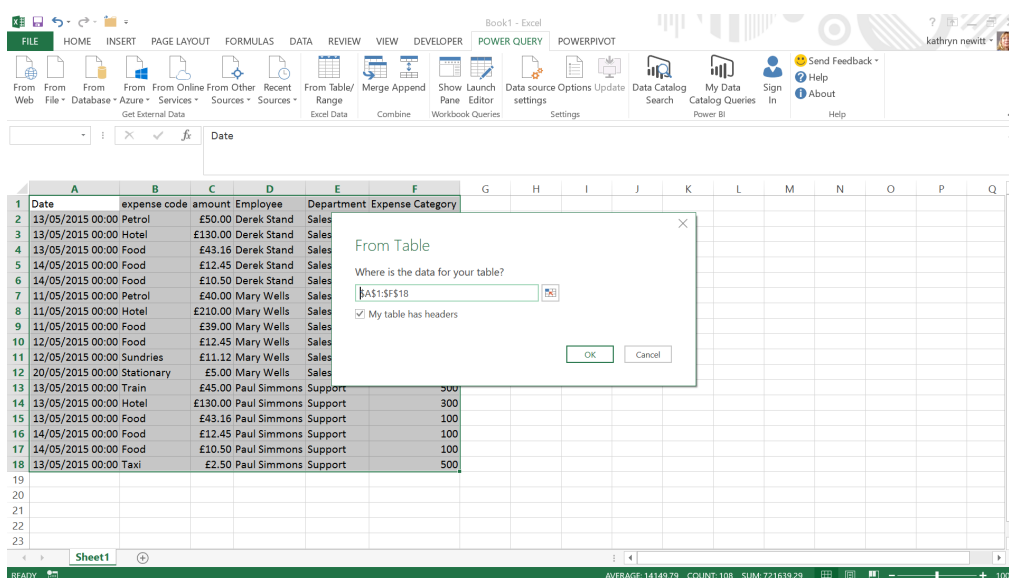
Each month we'll reproduce one of our articles on Power Query (Excel 2010 and 2013) / Get & Transform (Office 365, Excel 2016 and 2019) from www.sumproduct.com/blog. If you wish to read more in the meantime, simply check out our Blog section each Wednesday. This month, we look at how to join columns of data when the data types do not match.

Let's have some expense information in an Excel worksheet, where we'd like to combine the two columns **expense code**, which is data type 'General' and **Expense Category**, which is data type 'Number':



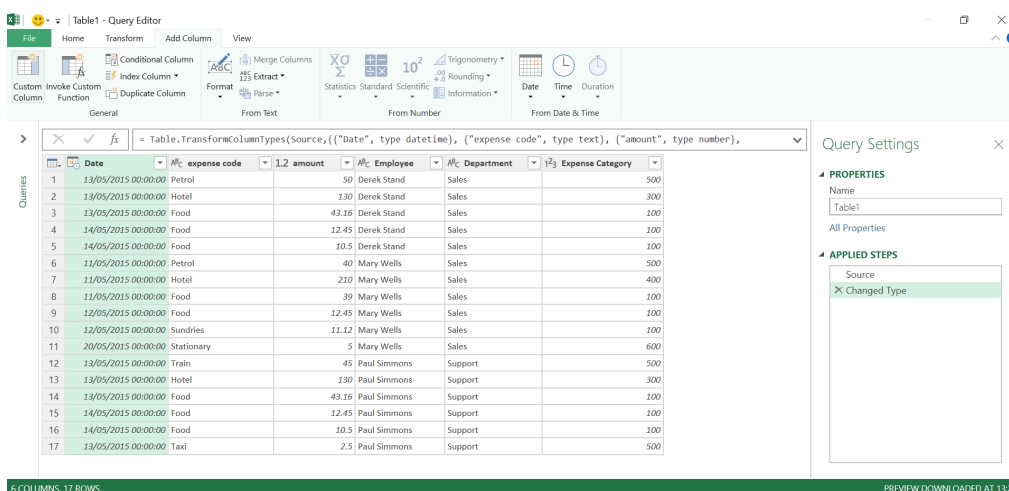
Date	expense code	amount	Employee	Department	Expense Category
13/05/2015 00:00	Petrol	£50.00	Derek Stand	Sales	500
13/05/2015 00:00	Hotel	£130.00	Derek Stand	Sales	300
13/05/2015 00:00	Food	£43.16	Derek Stand	Sales	100
14/05/2015 00:00	Food	£12.45	Derek Stand	Sales	100
14/05/2015 00:00	Food	£10.50	Derek Stand	Sales	100
11/05/2015 00:00	Petrol	£40.00	Mary Wells	Sales	500
11/05/2015 00:00	Hotel	£210.00	Mary Wells	Sales	400
11/05/2015 00:00	Food	£39.00	Mary Wells	Sales	100
12/05/2015 00:00	Food	£12.45	Mary Wells	Sales	100
12/05/2015 00:00	Sundries	£11.12	Mary Wells	Sales	100
20/05/2015 00:00	Stationary	£5.00	Mary Wells	Sales	600
13/05/2015 00:00	Train	£45.00	Paul Simmons	Support	500
13/05/2015 00:00	Hotel	£130.00	Paul Simmons	Support	300
13/05/2015 00:00	Food	£43.16	Paul Simmons	Support	100
14/05/2015 00:00	Food	£12.45	Paul Simmons	Support	100
14/05/2015 00:00	Food	£10.50	Paul Simmons	Support	100
13/05/2015 00:00	Taxi	£2.50	Paul Simmons	Support	500

We'll use Power Query to do this, so let's select all the data and go to the 'POWER QUERY' tab. In the 'Excel' section, choose 'From Table/Range':



Date	expense code	amount	Employee	Department	Expense Category
13/05/2015 00:00	Petrol	£50.00	Derek Stand	Sales	500
13/05/2015 00:00	Hotel	£130.00	Derek Stand	Sales	300
13/05/2015 00:00	Food	£43.16	Derek Stand	Sales	100
14/05/2015 00:00	Food	£12.45	Derek Stand	Sales	100
14/05/2015 00:00	Food	£10.50	Derek Stand	Sales	100
11/05/2015 00:00	Petrol	£40.00	Mary Wells	Sales	500
11/05/2015 00:00	Hotel	£210.00	Mary Wells	Sales	400
11/05/2015 00:00	Food	£39.00	Mary Wells	Sales	100
12/05/2015 00:00	Food	£12.45	Mary Wells	Sales	100
12/05/2015 00:00	Sundries	£11.12	Mary Wells	Sales	100
20/05/2015 00:00	Stationary	£5.00	Mary Wells	Sales	600
13/05/2015 00:00	Train	£45.00	Paul Simmons	Support	500
13/05/2015 00:00	Hotel	£130.00	Paul Simmons	Support	300
13/05/2015 00:00	Food	£43.16	Paul Simmons	Support	100
14/05/2015 00:00	Food	£12.45	Paul Simmons	Support	100
14/05/2015 00:00	Food	£10.50	Paul Simmons	Support	100
13/05/2015 00:00	Taxi	£2.50	Paul Simmons	Support	500

The 'From Table' sub-screen pops up to confirm the data which we want to load to a query and click 'OK'.



Date	expense code	amount	Employee	Department	Expense Category
13/05/2015 00:00:00	Petrol	50	Derek Stand	Sales	500
13/05/2015 00:00:00	Hotel	130	Derek Stand	Sales	300
13/05/2015 00:00:00	Food	43.16	Derek Stand	Sales	100
14/05/2015 00:00:00	Food	12.45	Derek Stand	Sales	100
14/05/2015 00:00:00	Food	10.5	Derek Stand	Sales	100
11/05/2015 00:00:00	Petrol	40	Mary Wells	Sales	500
11/05/2015 00:00:00	Hotel	210	Mary Wells	Sales	400
11/05/2015 00:00:00	Food	39	Mary Wells	Sales	100
12/05/2015 00:00:00	Food	12.45	Mary Wells	Sales	100
12/05/2015 00:00:00	Sundries	11.12	Mary Wells	Sales	100
20/05/2015 00:00:00	Stationary	5	Mary Wells	Sales	600
13/05/2015 00:00:00	Train	45	Paul Simmons	Support	500
13/05/2015 00:00:00	Hotel	130	Paul Simmons	Support	300
13/05/2015 00:00:00	Food	43.16	Paul Simmons	Support	100
14/05/2015 00:00:00	Food	12.45	Paul Simmons	Support	100
14/05/2015 00:00:00	Food	10.5	Paul Simmons	Support	100
13/05/2015 00:00:00	Taxi	2.5	Paul Simmons	Support	500

We want to create a 'Custom Column' (from the 'Add Column' tab) which will combine the **expense code** and **Expense Category**. The icons to the left of the column names show that these columns have different data types. Let's try simply appending the columns with an '&' viz.

There do not appear to be any syntax errors, so I go ahead and try this.

Clearly, this hasn't worked, and if you click on the 'Error' (highlighted) you will get more detail:

Expression.Error: We cannot apply operator & to types Text and Number.

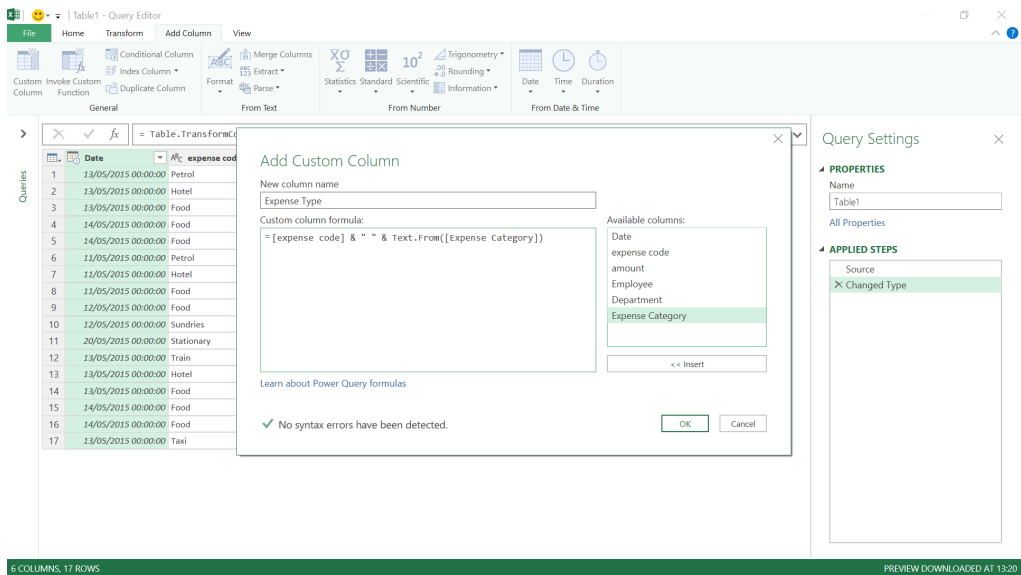
Details:

Operator=&

Left=Petrol

Right=500

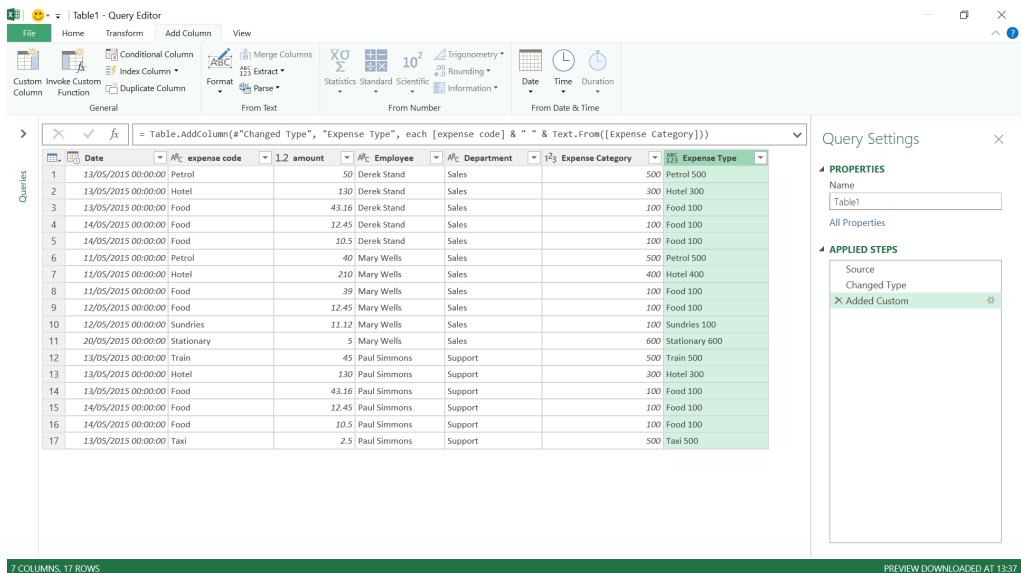
Therefore, in the custom column, we need to add some specific **M** code that will allow us to combine these expense columns. We want to convert the numerical column to text. Therefore, let's delete the 'Added Custom' step and try again...



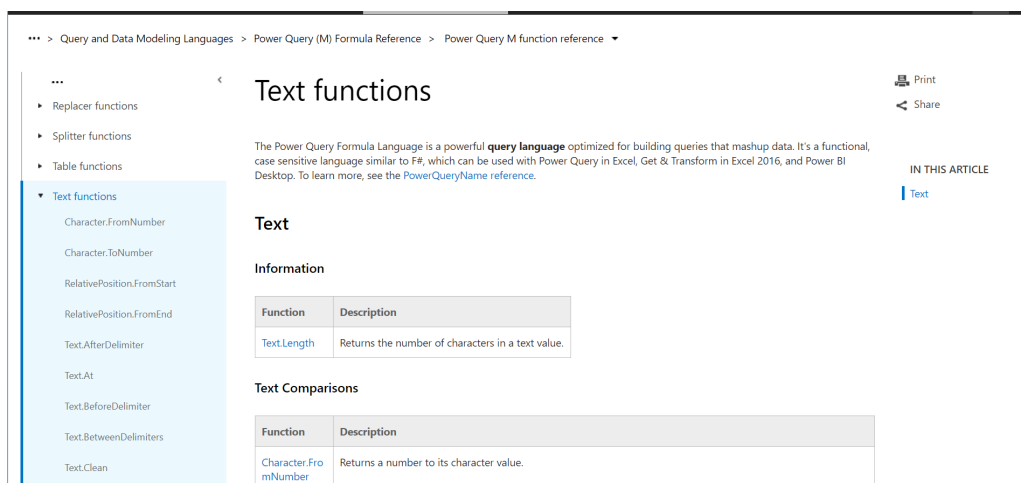
This time, the **M** code used is

```
= [expense code] & " " & Text.From([Expense Category])
```

Note that **M** code is case specific, so the capital and lowercase letters should appear exactly as in the example above. We have used the formula 'Text.From' to read the text from my numerical column. We have also added a space so that the data will be legible, and we'll click 'OK' to see the results:



The text and numerical columns have been combined to create a new text column. There are other useful formulae in **M** code which may be accessed from the 'Custom Column' screen by clicking on 'Learn more about Power Query Formulas' and exploring. Just some of the many 'Text' formulas are show below:



More next month!

Latest Updates for Power BI Desktop

On top of incremental improvements for several of the recently released features, such as the key influencers visual and the new Filter pane, the May update also sees a new 'Performance Analyzer' pane. This new pane lets you get a behind the scenes view of what's happening when your report is loading, which can help you optimise your report and improve the end user experience.

The full list of updates for May are as follows:

Reporting

- 'Performance Analyzer' pane
- Filter pane improvements (Preview)
 - o Sorting filter cards
 - o Theming support
- Table and matrix keyboard navigation
- Line data label position control
- KPI visual indicator text size control

Analytics

- Key influencers visual improvements (Preview)
 - o Binning support
 - o Formatting options
 - o Mobile support

Modelling

- Disable auto-date tables for new reports
- Update to the **ALLSELECTED** DAX function

Visualisations

- ArcGIS maps for Power BI updates
- Mapbox updates
- ValQ – Modern Digital Planning
- Advanced Graph Visual
- Craydec Control Charts
- Walkers Animated Pictogram
- Treemap Bar Chart by MAQ Software

Data Connectivity

- Essbase connector: Support for Native Query (MDX)
- Intune Data Warehouse connector
- Tenforce connector
- Roamlr connector

Other

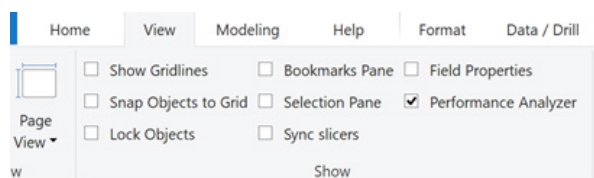
- Automatic query cancellation for Power BI Desktop.

Let's go through each new feature in turn.

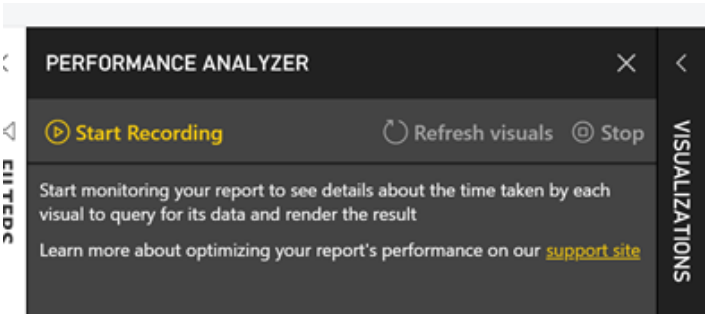
'Performance Analyzer' pane

When one of your reports is running slowly, it can be difficult to know what you can do to improve it. To help with this, Microsoft has added a new 'Performance Analyzer' pane in Power BI Desktop (it would be nice to get this for Power Query too!). You can use this pane to see where the time is being spent.

The pane is accessed through the View tab in the Ribbon:



When you first open the pane, you'll see an empty pane. All you need to do is select 'Start Recording' to see what queries are being run.



Once you start recording, any action that runs a query, such as changing a page, refreshing the visuals, or cross-highlighting, will populate the pane with information. You'll see each visual logged in load order with its corresponding load duration.

PERFORMANCE ANALYZER	
Start Recording Refresh visuals Stop	
Clear Export	
Name	Duration (ms)
Changed a slicer	-
Slicer	320
Sales and %GT Sales by Year	253
Matrix	385
Card	287
Card	268
Cross-highlighted	-
Matrix	217
Slicer	173
Card	172
Card	172
Cross-highlighted	-
Matrix	310
Slicer	118
Card	197
Card	235
Changed a filter	-
Sales and %GT Sales by Year	725
Matrix	584
Slicer	537
Card	536
Card	535

You can expand a specific visual's log to see how the time was spent divided out in three different sections:

- 1. **DAX query:** the length of time it took for Analysis Services to run the query
- 2. **Visual display:** how long it took for the visual to draw on the screen (including anything like retrieving web images or geocoding)
- 3. **Other:** covers time that the visual spent preparing queries, waiting for other visuals to complete, or doing some other background processing.

You'll also see data for how long your DirectQuery source took to return a result if you're in that mode.

Average of NSAT and Average of PurchAgain by Ye...	408
DAX Query	16
Visual display	33
Other	358
Copy query	

You can also select copy query to add the specific DAX query run for that visual to your clipboard.

```

Untitled - Notepad
File Edit Format View Help
// DAX Query
DEFINE
    VAR __DS0FilterTable =
        TREATAS({"2015"}, 'Calendar'[Year])

    VAR __DS0FilterTable2 =
        TREATAS({"France"}, 'Sales'[Country])

EVALUATE
    TOPN(
        1001,
        SUMMARIZECOLUMNS(
            'LocalDateTable_8369bfaf-356a-4b8d-a282-dd2c1868a321'[Year],
            'LocalDateTable_8369bfaf-356a-4b8d-a282-dd2c1868a321'[Month],
            'LocalDateTable_8369bfaf-356a-4b8d-a282-dd2c1868a321'[MonthNo],
            __DS0FilterTable,
            __DS0FilterTable2,
            "AverageNSAT", CALCULATE(AVERAGE('Sales'[NSAT])),
            "AveragePurchAgain", CALCULATE(AVERAGE('Sales'[PurchAgain]))
        ),
        'LocalDateTable_8369bfaf-356a-4b8d-a282-dd2c1868a321'[Year],
        1,
        'LocalDateTable_8369bfaf-356a-4b8d-a282-dd2c1868a321'[MonthNo],
        1,
        'LocalDateTable_8369bfaf-356a-4b8d-a282-dd2c1868a321'[Month],
        1
    )

ORDER BY
    'LocalDateTable_8369bfaf-356a-4b8d-a282-dd2c1868a321'[Year],
    'LocalDateTable_8369bfaf-356a-4b8d-a282-dd2c1868a321'[MonthNo],
    'LocalDateTable_8369bfaf-356a-4b8d-a282-dd2c1868a321'[Month]

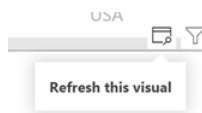
```

You can also export all the query information to a JSON file to analyse using other (third party) performance tools.



Further, if you want to dig into one slow loading visual specifically, while the Performance Analyzer is running, there will be an icon in the visual header that will let you refresh just that specific visual to see the statistics in the pane.

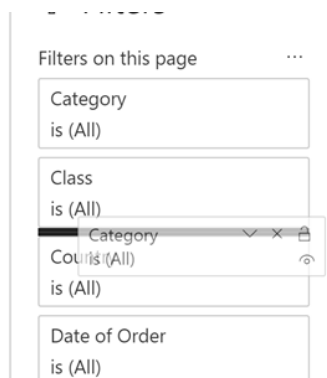
The Performance Analyzer is currently available in Power BI Desktop only and doesn't show all the time spent for R, Python, or Key Influencers visuals.



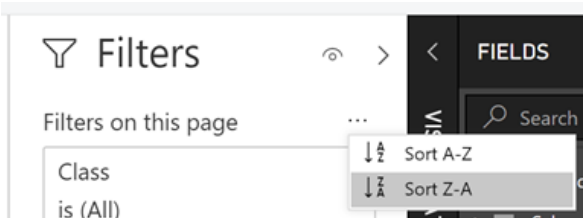
Filter pane improvements (Preview)

SORTING FILTER CARDS

One of the things that irritated users about the Filter pane was its inability to customise the sort of the Filter cards within the pane. However, with this update you can now do this. All you have to do is drag and drop Filter cards within the section it resides in (report, page, or visual) to change the order of the Filter cards.



You can also change the sort direction from the “...” menu next to the header of a given section of the Filter pane.



THEMING SUPPORT

Additionally, you can now also theme all the formatting options available for the Filter pane and cards. Here’s an example theme formatting just the pane:

```
{
  "name": "FilterPaneTheme",
  "visualStyles": {
    "aa": {
      "aa": {
        "outspacePane": [{
          "backgroundColor": {"solid": {"color": "#999999"}},
          "foregroundColor": {"solid": {"color": "#374649"}},
          "transparency": 50,
          "titleSize": 35,
          "headerSize": 8,
          "fontFamily": "Arial",
          "border": true,
          "borderColor": {"solid": {"color": "#374649"}}
        }],
        "filterCard": [
          {
            "$id": "Applied",
            "transparency": 0,
            "backgroundColor": {"solid": {"color": "#AFB5B6"}},
            "foregroundColor": {"solid": {"color": "#374649"}},
            "textSize": 15,
            "fontFamily": "Arial",
            "border": true,
            "borderColor": {"solid": {"color": "#ffffff"}},
            "inputBoxColor": {"solid": {"color": "#C8C8C8"}}
          },
          {
            "$id": "Available",
            "transparency": 40,
            "backgroundColor": {"solid": {"color": "#CCCCCC"}},
            "foregroundColor": {"solid": {"color": "#ffffff"}},
            "textSize": 10,
            "fontFamily": "Arial",
            "border": true,
            "borderColor": {"solid": {"color": "#123456"}},
            "inputBoxColor": {"solid": {"color": "#777777"}}
          }
        ]
      }
    },
    "page": {
      "aa": {
        "outspace": [{
          "color": { "solid": { "color": "#000000" } }
        ]
      }
    }
  }
}
```

Table and matrix keyboard navigation

A couple of months ago, Microsoft added support for interacting with the majority of visuals for keyboard and screen readers, and this update extends that support to the table and matrix visual. Now, when your focus is on a table or matrix, you can use the **CTRL + Right Arrow** to move focus inside to the first cell of the visual.

Country	Large	Medium	Small	Total
Germany	\$606,950	\$1,161,806	\$529,912	\$2,298,669
France	\$563,569	\$1,460,691	\$618,535	\$2,642,795
Great Britain	\$769,253	\$1,779,645	\$748,295	\$3,297,193
Australia	\$1,255,233	\$2,752,919	\$1,267,752	\$5,275,904
Canada	\$989,171	\$2,968,673	\$1,355,203	\$5,313,048
United States	\$4,863,456	\$11,689,596	\$5,260,313	\$21,813,365
Total	\$9,047,632	\$21,813,330	\$9,780,011	\$40,640,973

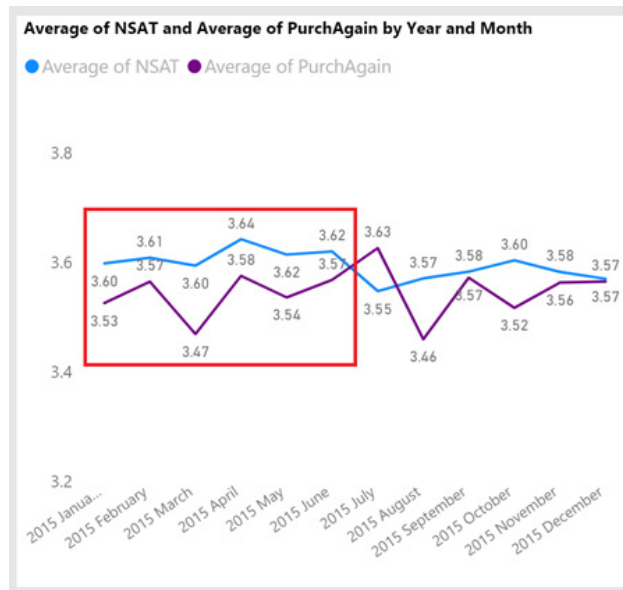
While your focus is on a cell, the screen reader will read off the relevant information, and you can use the follow keyboard shortcuts to navigate:

- **Arrow keys:** move your focus up/down/left/right on cell
- **SPACE or ENTER:** select cell
- **SHIFT + Right Arrow:** expand an expandable header
- **SHIFT + Left Arrow:** collapse an expandable header
- **CTRL key or SHIFT + F10:** open context menu.

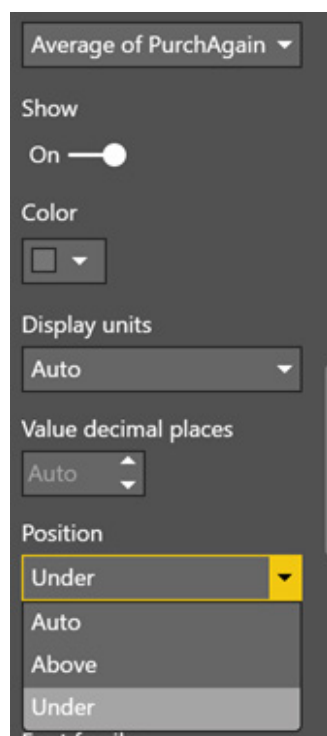
Line data label position control

When you have a line or combo chart with multiple lines overlapping with each other, it can sometimes be hard to tell which data label corresponds to which line. To help with this, Power BI Desktop has added a new position control for data labels so you can set the labels to specifically show above or below its corresponding line.

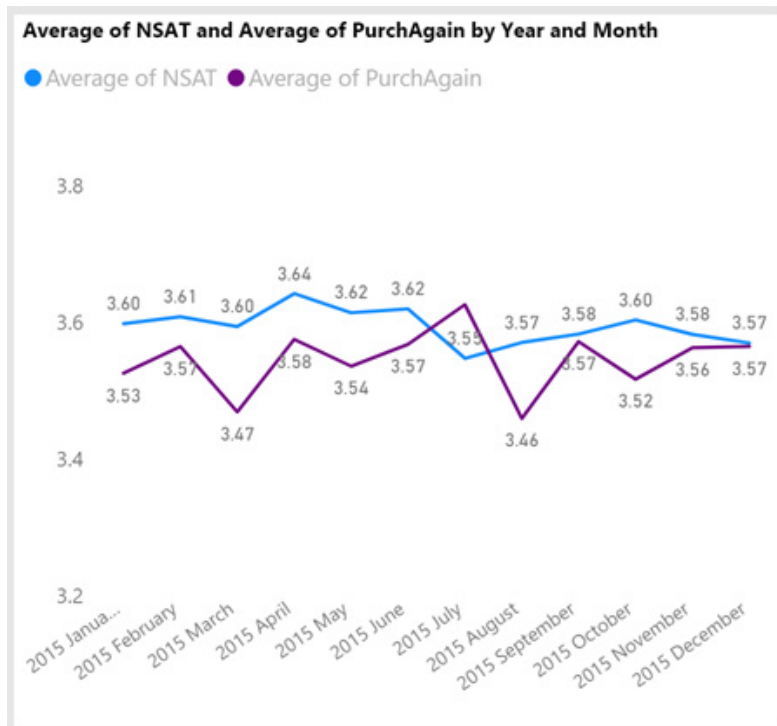
For example, the below line chart showing the average net satisfaction and likelihood to purchase again. Many of the data labels for the first several months are showing near each other with the automatic layout.



To help differentiate between the two, you can now customise the position so the labels for the PurchAgain line show below it and the labels for the NSAT show above it.

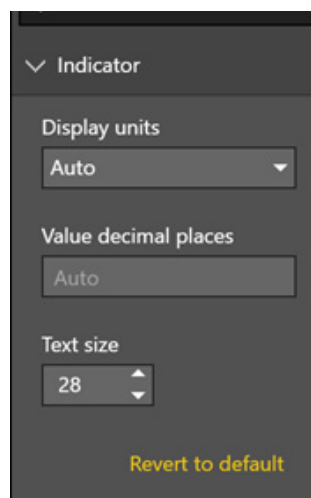


Once you do that, it's a lot clearer which label goes with which line.



KPI visual indicator text size control

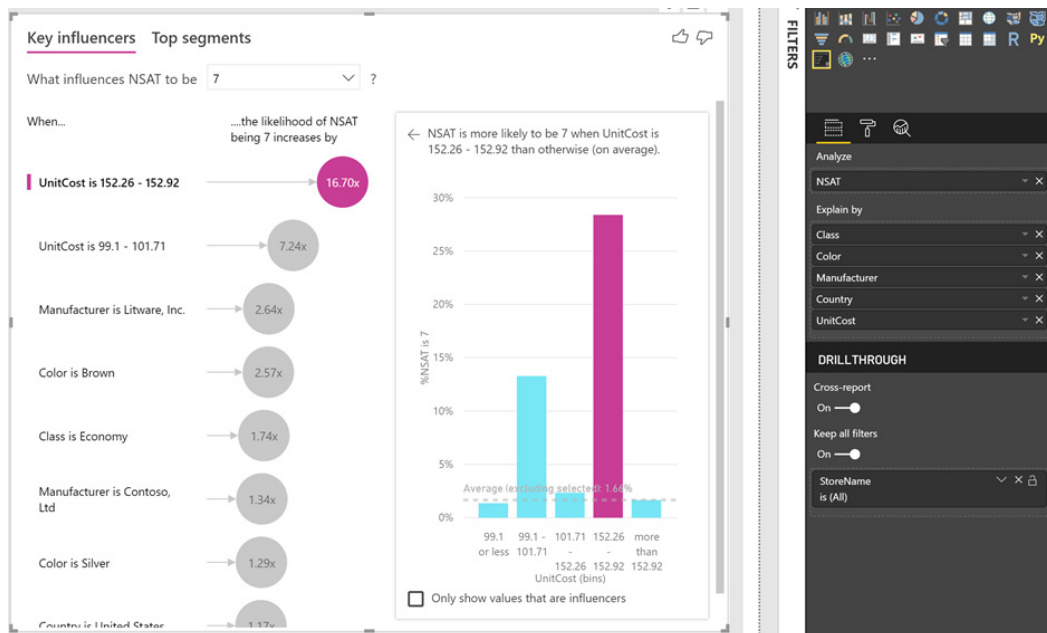
As a small formatting improvement with this update, the KPI visual has been updated to support changing the text size of the Indicator. This has been a big pain point for some people, so it's likely to be highly appreciated!



Key influencers visual improvements (Preview)

BINNING SUPPORT

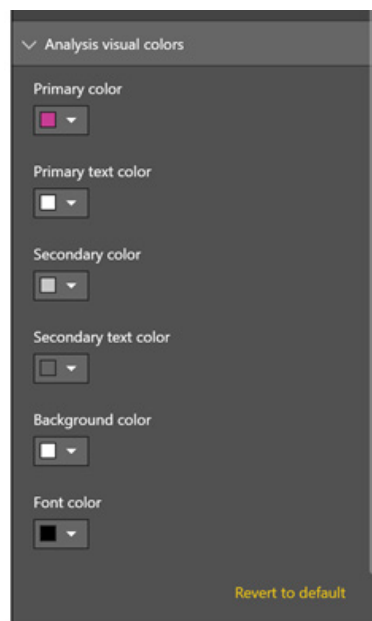
The key influencers visual now supports binning of numerical columns added to the "explain by" bucket of the field well. For example, if you add the UnitCost field to the 'Explain by' bucket to see if it affects the likelihood of a product being 7, you can see specific ranges of cost within the influencers list. In this specific example, you can actually see that when the unit costs are between \$152.26 and \$152.92, there's a high likelihood of the NSAT being 7.



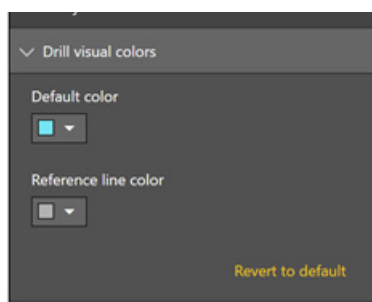
The visual's Machine Learning (ML) model will decide how to bin the data to find the most influential segments of your data. It will also decide to not bin a numeric column if it detects a strong linear relationship between the column and the target of the analysis. In this case, the visual will instead show a scatter chart demonstrating the correlation.

FORMATTING OPTIONS

This update has also added several formatting options to the key influencers visual. The Analysis visual colours are used for the bubbles section on the left half of the visual and the primary colour used for the selected influencer in the drill visual on the right half.



You can also control the drill visual's secondary colour and reference line colour.

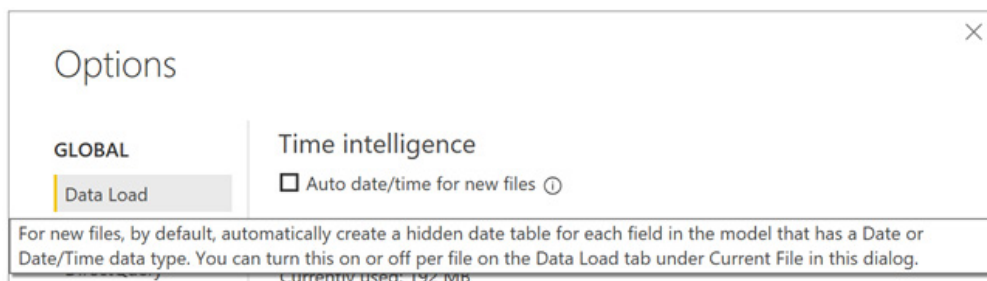


MOBILE SUPPORT

Mobile support for the Key Influencers visual has also been added. If you have this visual in a report, it should be visible and interactive, just like our other visuals, when viewing it in any of our mobile apps.

Disable auto-date tables for new reports

If you don't want to use Power BI's auto-generated date tables, and instead, for all your reports want to use your own date tables, you can now turn off the feature in the Options dialog under **Global -> Data Load -> Time intelligence**. Checking this box means that for any new files, you don't automatically create any hidden date tables.



Update to the ALLSELECTED DAX function

The **ALLSELECTED** DAX function has been extended to support multiple arguments, just like the **ALL** function.

ArcGIS maps for Power BI updates

ArcGIS Maps for Power BI is Esri's integration into the Power BI world. It provides all Power BI users with the ability to create insightful maps with their data. It is embedded within Power BI, which means you don't have to go to an app store to use it. To get started, just click the globe icon

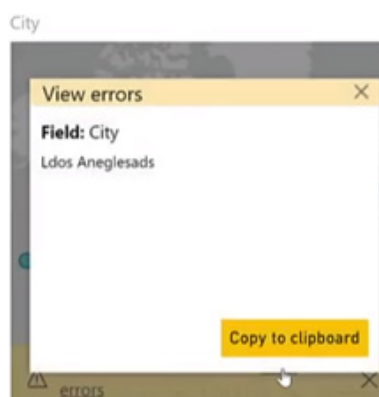


and start visualising / analysing your data spatially accordingly.

In this update, the visual has a major update that allows you to geocode with more accuracy. The first and most important step when you are analysing data spatially is geocoding. It converts your addresses and place names into coordinates on a map. Esri's geocoding engine

provides precise, accurate and reliable results for over 135 countries in key markets of the world including North America, Europe, our very own Australia, Israel, India, Brazil and Japan. To achieve this, Esri works with international distributors and partners in countries with complex addressing systems including Japan, Thailand, India, and Israel to provide the most up-to-date data. Eurovision watch out.

However, geocoding can fail if addresses in your data are misspelt. For example, if your data has "Los Angeles" instead of "Los Angeles", you will not get any results. To help you deal with such situations, the visual now has the added ability to easily spot addresses that fail to geocode for any reason as "errors". You can then simply copy these errors out of the visual and return to your data source to find and fix these errors.

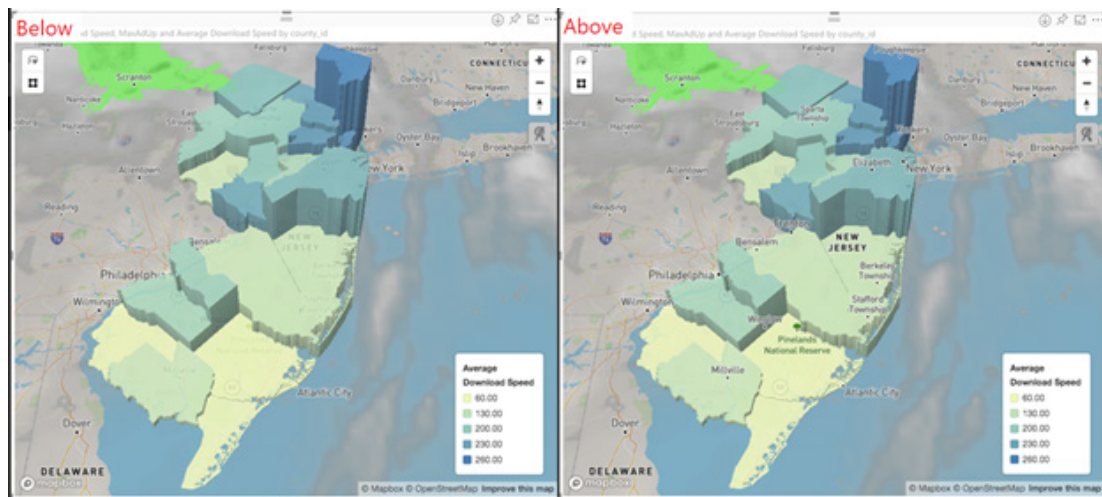


In addition, the Esri team has also added a much faster map loading experience when you add a map visual to the report.

Mapbox updates

The Mapbox team has been working to add even more value to their visual and have recently added four major features geared towards enterprise wide deployments.

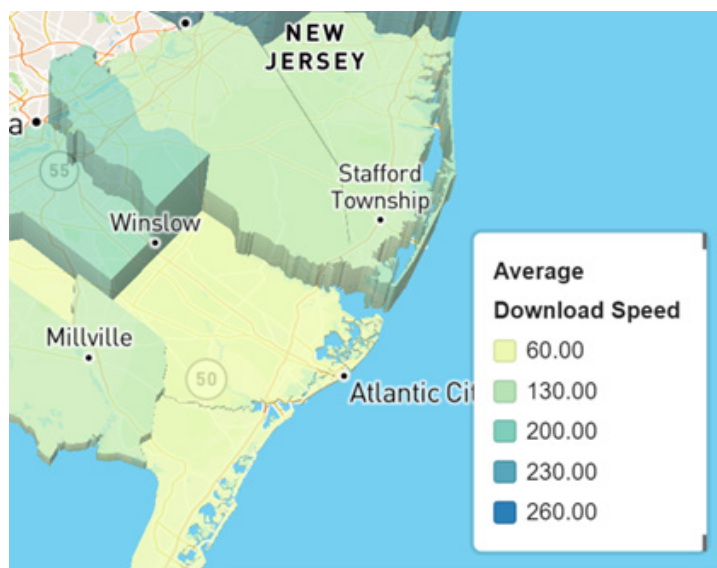
The Mapbox visual now has the option to plot the map location labels above or below the data visualised on the map.



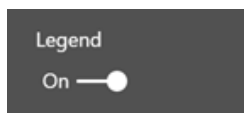
This adds more control and flexibility into the map and allows you to decide whether you want to give more emphasis to your own data or to the native mapping data. You can control this option through the 'Label position' dropdown in the 'Viz Settings' card of the Formatting pane.



You can now also add a legend to your map for Choropleth, Cluster, and Circle Layer types.



The option can be found under each layer card in the Formatting pane.



Adding a legend to your map makes it easier for your end users to understand the scale and distribution of the data you're mapping.

That's not all. You can now set the minimum, centre, and maximum value to correspond with their associated colours, where any value above the

maximum or below the minimum will be that same colour. This allows you to better account for outliers and irregular distributions of data, preventing the gradient colours from being heavily skewed.

▼ Choropleth

On ☒

Diverging

On ☒

Minimum

▼

Center

▼

Maximum

▼

Minimum

60

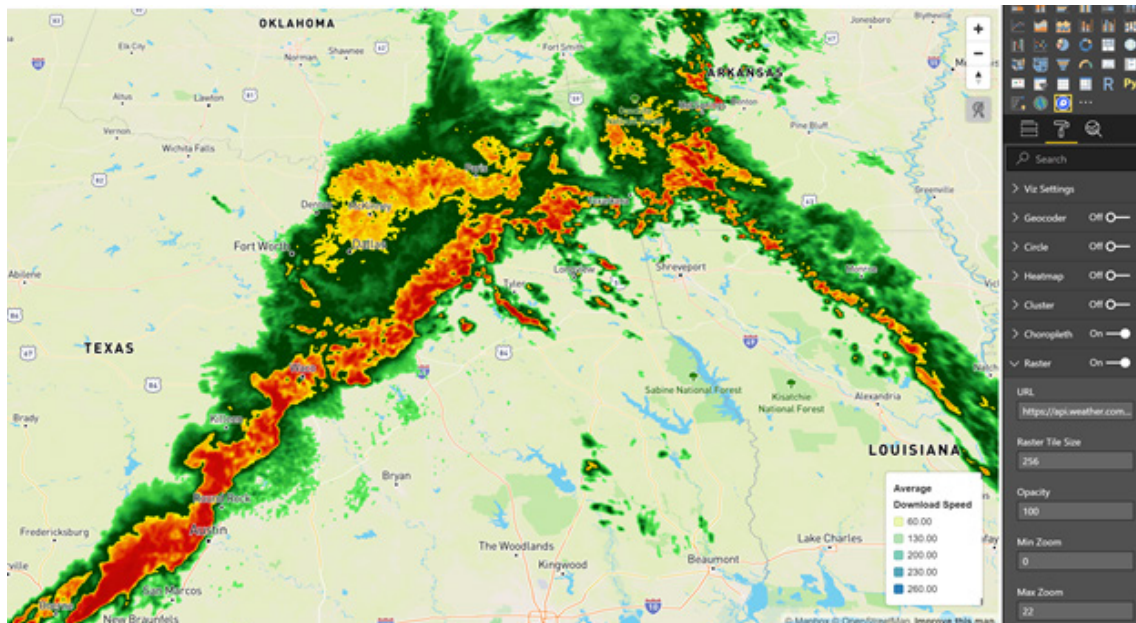
Center

200

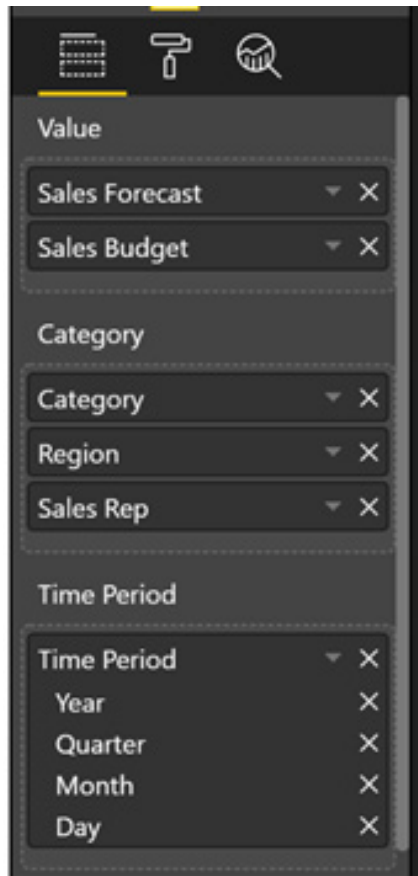
Maximum

260

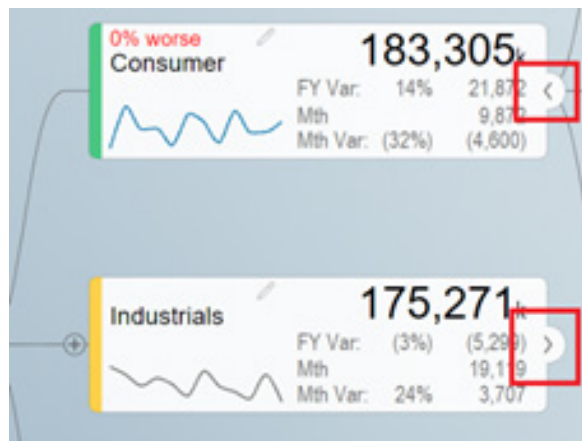
You can now also add WMS and Raster layers too, such as weather, to your Power BI Report.



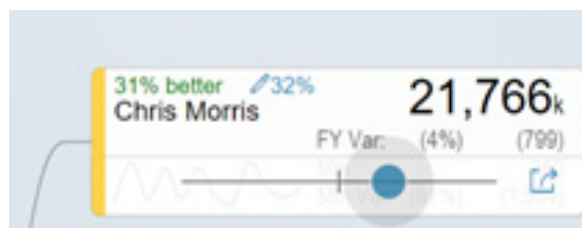
You can set this up through the Raster card of the Formatting pane.



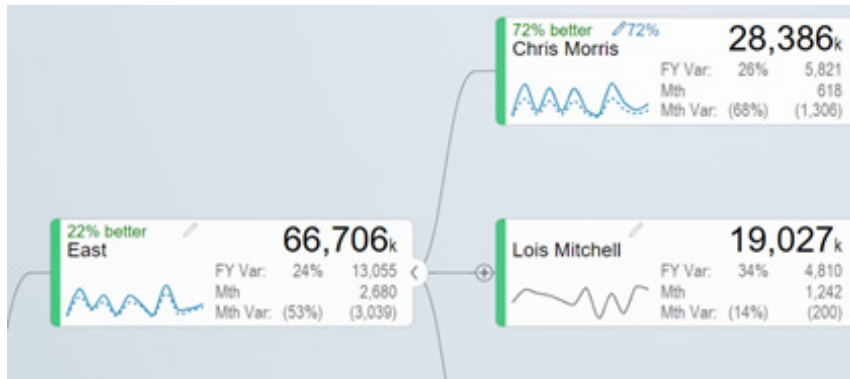
Now you can expand and collapse the nodes of your tree by clicking on the carets on the right edge of each node. As the tree grows you can zoom and pane to get the best view of the data.



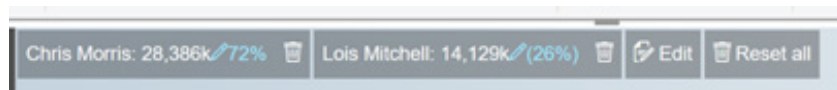
You can also simulate what-if? scenarios by dragging the slider you get when hovering over a node.



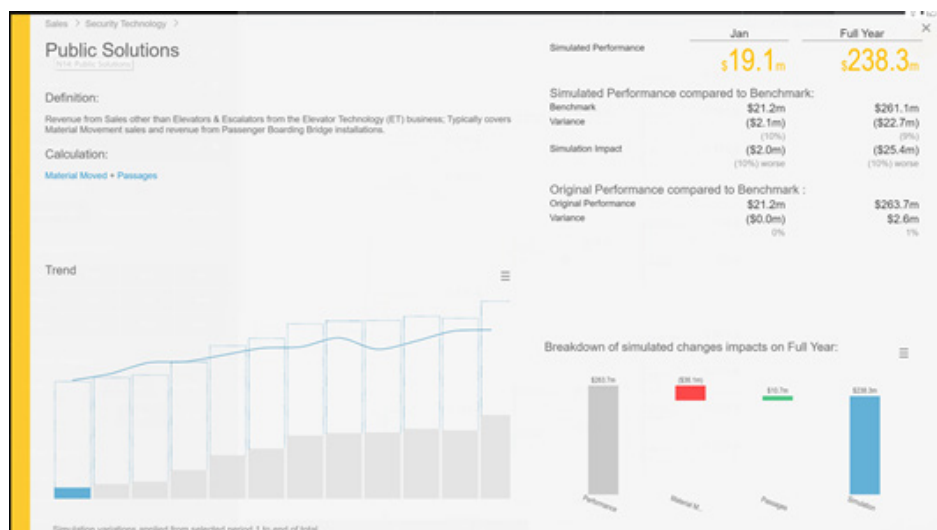
When you do this, all the nodes higher in the tree will update to reflect your change.



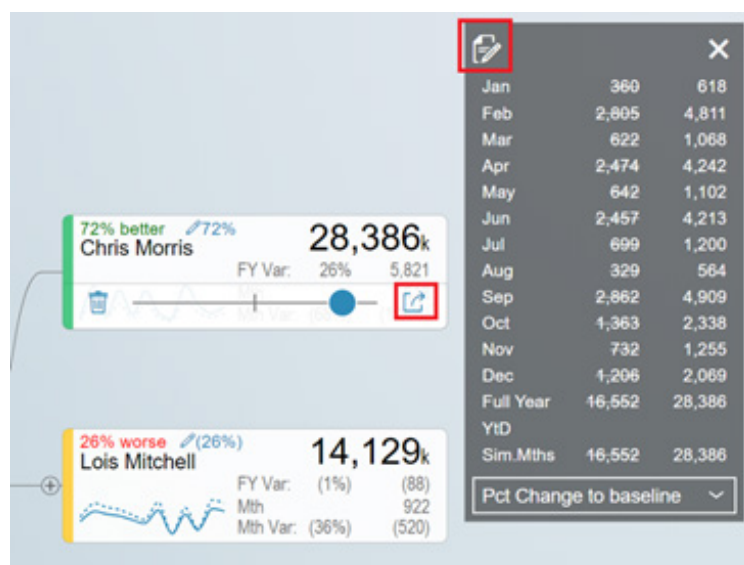
As you change more and more leaf nodes you can see all the applied changes on the top of the visual. You can also remove them easily here as well using the trash can icon.



When you click on nodes of the tree, you get a detailed pop-up giving you more information about that node. It includes information since as the trend, a waterfall chart showing on the simulated changes impact the results, and a comparison of the measures you are comparing in the visual.



You can also click on a button for any node and see the data for that node across the time period and how those numbers are impacted by the simulation.



You can also click the pencil icon from that dialog to edit the simulation manually as well.

Maintain inputs for simulation periods											
	Model	Variation	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep
Chris Morris	% Change	71.50	618k	4,811k	1,068	4,242k	1,102k	4,213k	1,200k	564k	4,909k

The pane on the left allows you to save multiple different simulations and compare them. You can also set the simulation period, change how the values are displayed and change the visual's layout.

Scenarios

1: Scenario 1

2: Scenario 2

3: Scenario 3

4: Scenario 4

Sub Models

Enterprise Sales

Components

CT Industrial

Simulation Period

Jan

Feb

Mar

Apr

May

Jun

Jul

Aug

Sep

Oct

Nov

Dec

Value Display

Month

Ytd

Full Year

Sim.Mths

0

0k

0.0m

0.0b

Key Inputs

Enterprise Sales

CT Powertrain

CT Chassis

MX Stainless Steel

\$37,962.5m

\$1,302.1m

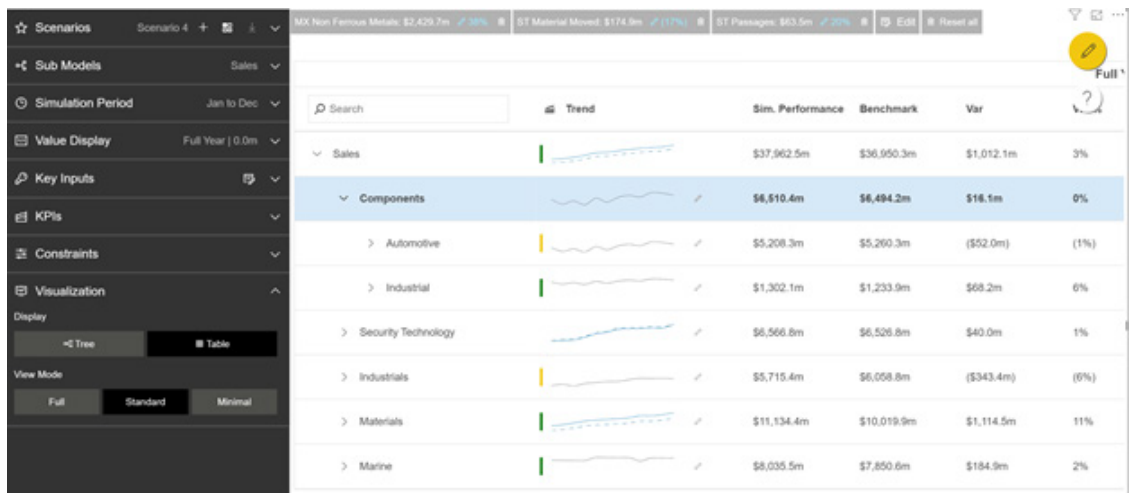
\$3,906.2m

\$2,240.8m

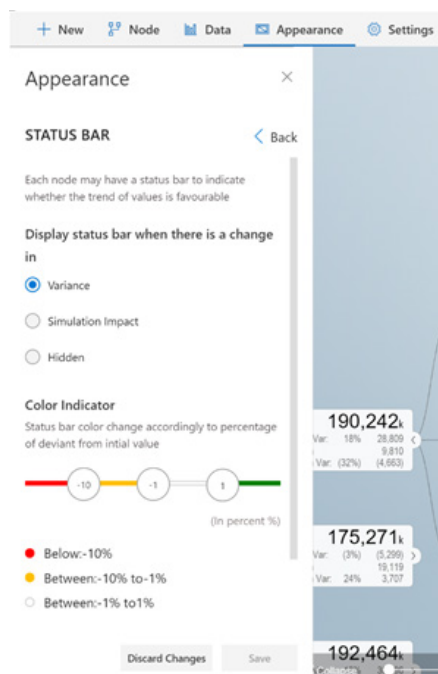
When you compare multiple scenarios, you can see the differences between each scenario and quick jump to that scenario from this view.

Sales: Full Year Scenario Comparison				
	Scenario 1: Full Year	Scenario 2: Full Year	Scenario 3: Full Year	Scenario 4: Full Year
	\$37,318.8m	\$37,024.7m	\$37,951.7m	\$37,962.5m
		MX Stainless Steel: (6%) MX Non Ferrous Metals: 38% MX Carbon Steel: (17%) ST Material Moved: (17%) ST APAC: (18%)	MX Non Ferrous Metals: 38% ST Material Moved: (17%)	MX Non Ferrous Metals: 38% ST Material Moved: (17%) ST Passages: 20%
KPIs				
Enterprise Sales	\$37,318.8m	\$37,024.7m	\$37,951.7m	\$37,962.5m
Components	\$6,510.4m	\$6,510.4m	\$6,510.4m	\$6,510.4m
Security Technology	\$6,592.1m	\$6,235.7m	\$6,556.1m	\$6,566.8m
Industrials	\$5,715.4m	\$5,715.4m	\$5,715.4m	\$5,715.4m
Materials	\$10,465.4m	\$10,527.7m	\$11,134.4m	\$11,134.4m
Marine	\$8,035.5m	\$8,035.5m	\$8,035.5m	\$8,035.5m
Key Inputs	\$37,318.8m	\$37,024.7m	\$37,951.7m	\$37,962.5m

You can also visualise the same data as a table well with sparklines if that's easier for you navigate.

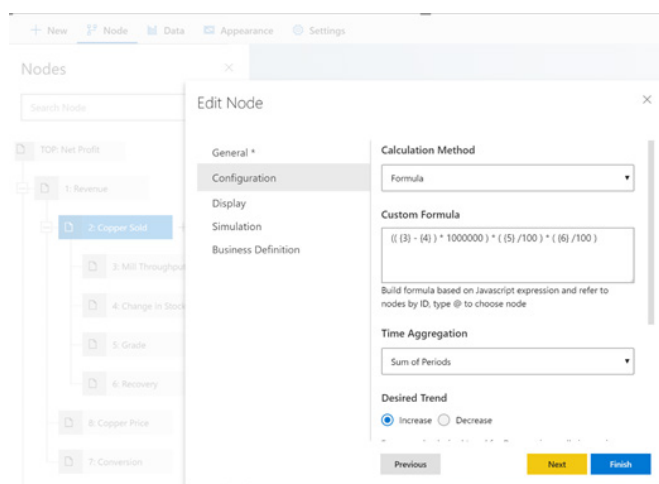


You can also use the yellow pencil icon to open and close more editing options, which include things like changing the status bar colours and adjusting the colours of the tree itself.



There are two other modes to build out your tree as well.

The second mode allows you to build out a tree from scratch, which allows you to build the tree node by node, configuring each one, either based on the data from your model or any custom calculation you want.



Alternatively, in this mode you can use the ‘Quick Edit’ option that allows you to edit all the nodes in one view.

Quick Editor

ExpandCollapseAll Columns

Unique ID	Title	Calculation Method	Formula	Business View	Increase ...	Prefix	Suffix	Aggregation
TOP	Sales	Sum Children (a+b+c+...)			<input checked="" type="checkbox"/>	\$		Sum
1	Components	Sum Children (a+b+c+...)			<input checked="" type="checkbox"/>	\$		Sum
2	Automotive	Sum Children (a+b+c+...)			<input checked="" type="checkbox"/>	\$		Sum
3	Chassis	Data Source			<input checked="" type="checkbox"/>	\$		Sum
4	Powertrain	Data Source			<input checked="" type="checkbox"/>	\$		Sum
5	Industrial	Sum Children (a+b+c+...)			<input checked="" type="checkbox"/>	\$		Sum
6	Bearings	Data Source			<input checked="" type="checkbox"/>	\$		Sum
7	Undercarriages	Data Source			<input checked="" type="checkbox"/>	\$		Sum
8	Other	Data Source			<input checked="" type="checkbox"/>	\$		Sum
9	Security Technology	Sum Children (a+b+c+...)			<input checked="" type="checkbox"/>	\$		Sum
10	Home and Business	Sum Children (a+b+c+...)			<input checked="" type="checkbox"/>	\$		Sum
11	Americas	Data Source			<input checked="" type="checkbox"/>	\$		Sum
12	Europe	Data Source			<input checked="" type="checkbox"/>	\$		Sum
13	APAC	Data Source			<input checked="" type="checkbox"/>	\$		Sum
14	Public Solutions	Sum Children (a+b+c+...)			<input checked="" type="checkbox"/>	\$		Sum
15	Material Moved	Data Source			<input checked="" type="checkbox"/>	\$		Sum
16	Passages	Data Source			<input checked="" type="checkbox"/>	\$		Sum
17	Industrials	Sum Children (a+b+c+...)			<input checked="" type="checkbox"/>	\$		Sum
18	Physical Infrastructure	Price Factors			<input checked="" type="checkbox"/>	\$		Sum

The third mode, Import an Advanced Model, allows you to either import a tree that you exported from a previous tree visual or import data from an Excel template (which you can get by emailing support@valq.com). The template is similar to the ‘Quick Edit’ experience you can get

through the second mode and is an easy way to build out the constraints and guidelines for the tree. It’s ideal for an analyst who can sit with a business user, write up the requirements in the Excel template, and just import it to automatically get a tree that meets your needs.

Value Driver Tree Definition

Define Value Driver Tree nodes and relationships

Max ID: 4

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	Name	Level	Label	Calc Meth	Form
Net Profit																	TOP	0	Net Profit	F	(1)-(5)
Revenue																	1	Revenue	F	(2) * (3) * (1-4)/100	
Quantity																	2	Quantity	L		
Price																	2	Price			
Discount																	2	Discount			
Expenses																	1	Expenses	M		
Quantity																	6	Quantity			
Cost Per Unit																	7	Cost Per Unit			
Gross Margin																	1	Gross Margin	F	((1) - (5)) * 100 / (1)	

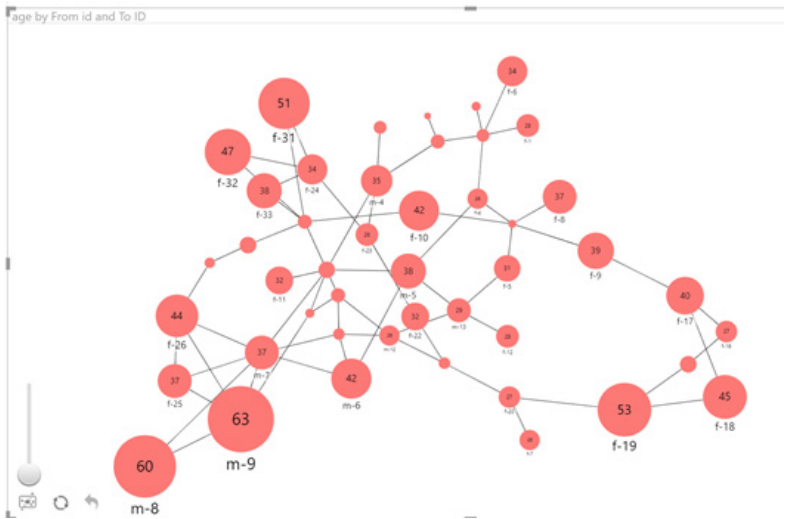
The visual comes with sample trees you can test out, and Visual BI, who made the visual, is adding more getting started content to their website as well.

Advanced Graph Visual

The Advanced Graph in-app purchase custom visual by ZoomCharts is a sophisticated network visual that lets you define and visualize one or multiple bi-directional links and format them as you choose. Some uses of this visual include analysing money flow, managing and monitoring IT infrastructure, and analysing social connections.

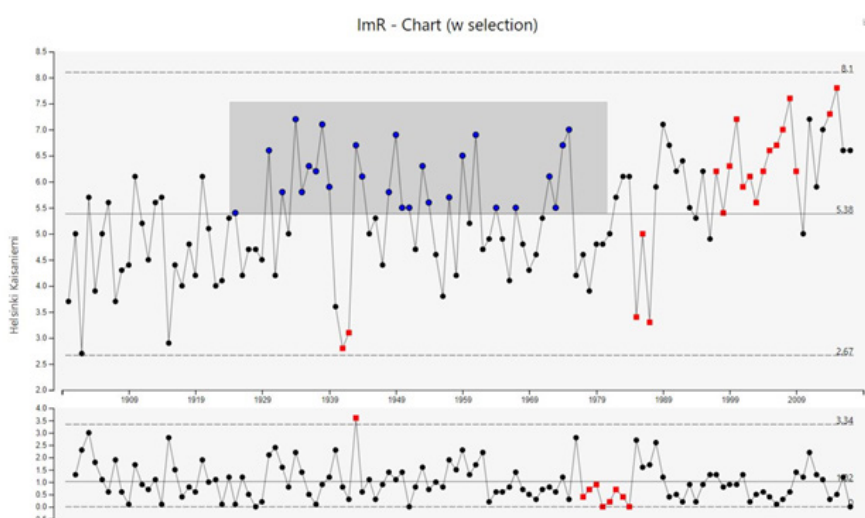
Some of the core features of this visual include:

- **Automated graph structure:** the visual builds a graph structure by identifying relationships between objects with the size of the nodes based on the value for each node
- **Force feedback layout:** the nodes repulse each out and links act as “springs”, resulting in a more efficient and beautiful layout compared to standard network graphs
- **Rectangular section:** active rectangular section mode by long pressing to select many nodes to examine closer
- **Customise links and nodes:** you can match your corporate style through a variety of formatting options.



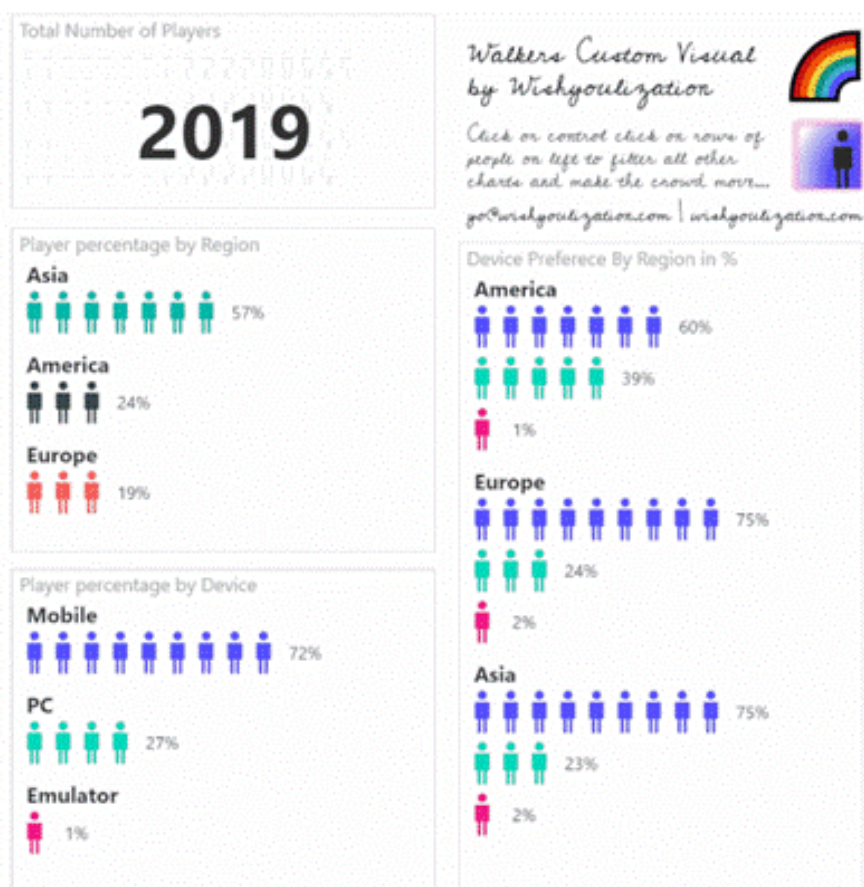
Craydec Control Charts

The Craydec Control Charts in-app purchase visual lets you analyse and track your processes and see whether they are in control. This visual also helps you predict the performance of your processes. This type of visual is commonly used as part of the Statistical Process Control (SPC) and is widely used in what's known as "process improvement paradigms". Did you manage to cross that off your Buzzword Bingo scorecard..?



Walkers Animated Pictogram

The Walkers Animated Pictogram by Wishyouulization lets you go beyond simple bar graphs to created animated pictographic bar charts. This visual can be great for building a more meaningful connection to the emotion behind the data and for making your reports more lively and animated.



Treemap Bar Chart by MAQ Software

The Treemap bar chart visual displays the breakdown of hierarchical data using colour coded proportionate rectangles. A good use case for this visual is breaking down your profit across several different categories at once as countries, regions, and cities.

Some key features of this visual include:

- Switch between a bar chart and column chart
- Add total values and data labels for all categories
- Tooltips with additional insights
- Animations to examine the visual at the granular level
- Cross-highlighting support for legend values, axis values, and the bricks
- Support for bookmarks, report page tooltips, and drillthrough.



Essbase connector: Support for Native Query (MDX)

With this month's release, Microsoft has enhanced the Essbase connector to add support for providing Native Queries (MDX) when using Import mode. Native Queries can be specified as part of the Essbase connector dialog. Note that you will need to specify your Essbase Server and Application name in order to execute native queries.

The screenshot shows the 'Essbase' connector dialog box. It contains the following fields and options:

- URL:** A text field with a dropdown menu set to 'AFC'.
- Data Connectivity mode:** Two radio buttons: 'Import' (selected) and 'DirectQuery'.
- Advanced options:** A section header with a downward arrow.
- Command timeout in minutes (optional):** A text field.
- Server:** A text field.
- Application:** A text field.
- MDX statement (optional):** A large text area containing the following MDX query:

```
SELECT
  CrossJoin ({[100-10]}), {[East],[West],[South],[Central]}}
ON COLUMNS,
  CrossJoin (
    {[Sales],[COGS],[Margin %],[Profit %]}, {[Qtr1]}
  )
ON ROWS
FROM Sample.Basic
WHERE (Budget)
```
- Buttons:** 'OK' and 'Cancel' buttons at the bottom right.

Intune Data Warehouse connector

The Intune connector allows you to connect to your Intune Data Warehouse to build reports and provide insight into your enterprise mobile environment with Power BI Desktop. This connector connects to version v1.0 of the Data Warehouse.

Tenforce connector

TenForce is a software solution able to provide an integrated overview of a company's operational reality on a global scale. Mapping all relevant workflows, processes, and actions, TenForce helps its users to smartly plan, track and manage their work, mitigate risk, ensure regulatory

compliance and improve efficiency. The TenForce (Smart)List connector allows users to connect to their TenForce installation, consume its data while respecting privileges, and analyse the results all from within Microsoft Power BI.

Roamler connector

Roamler is an online platform where qualified workers get paid to perform curated and standardised tasks in Retail, Home Installations and Healthcare. These tasks vary from On Shelf Availability audits to installing smart Thermostats. This connector allows you to unleash the task data and visualise it for better information flows.

Automatic query cancellation for Power BI Desktop

To improve the performance of reports with slower models, Power BI Desktop now automatically cancels obsolete visual queries in four scenarios:

1. When the visual generates a new query and a previous query hasn't finished, the previous query gets cancelled. This is common when changing filters, multi-selecting in slicers or cross-highlighting
2. When you hide a visual, a running query on that hidden visual gets cancelled
3. When you change pages in a report, any running queries on the previous page get cancelled
4. When you close a report, any running queries get cancelled.

This is the first version of this feature, so there are a few scenarios not supported yet:

- Automatic query cancellation in the Power BI Service, Report Server, and Embedded
- Automatic query cancellation when using an On-premise Data Gateway.

Microsoft plans to invest further in these query cancellation scenarios, so stay tuned for future updates...

That's it until the July newsletter.

Latest Power BI Service and Mobile updates

And that's not all. As usual, there's been plenty of updates for Power BI Service and Mobile too:

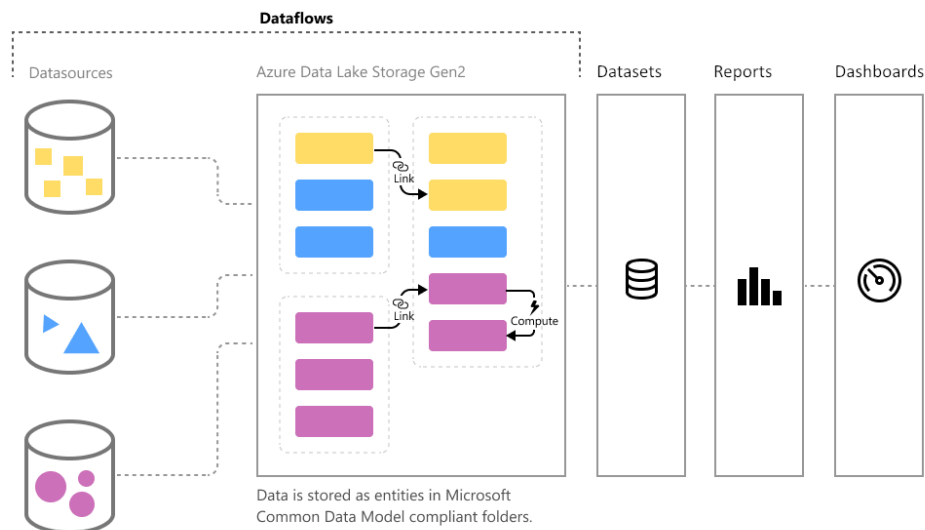
- Power BI dataflows Generally Available
- Q&A update for dashboards
- New workspaces experience Generally Available
- Paginated Report Builder
- New Power BI regions
- Additional capabilities for paginated reports
- Support for Azure Analysis Services
- Publishing within Power BI Apps
- Email subscriptions
- Oracle and Teradata support
- Power BI Premium summary and workload metrics in the Admin Portal
- Query caching in Power BI Premium
- Latest updates for on-premises data gateway
- New capabilities to data gateway management in the Power Platform Admin Center
- Power BI Home GA
- Workspace management tips and tricks
- Roadmap updates.

If you are sitting comfortably, let's begin...

Power BI dataflows Generally Available

Microsoft has recently announced that Power BI dataflows are now Generally Available. Dataflows are used to ingest, transform, integrate, and enrich big data by defining data source connections, ETL logic, refresh schedules, and more. In addition, the new model-driven calculation engine that's part of dataflows makes the process of data preparation more manageable, more deterministic, and less cumbersome for data analysts and report creators alike. Since the public Preview last year, Microsoft has shared a list of new dataflows updates, and with this Generally Available release they have added more new capabilities which include:

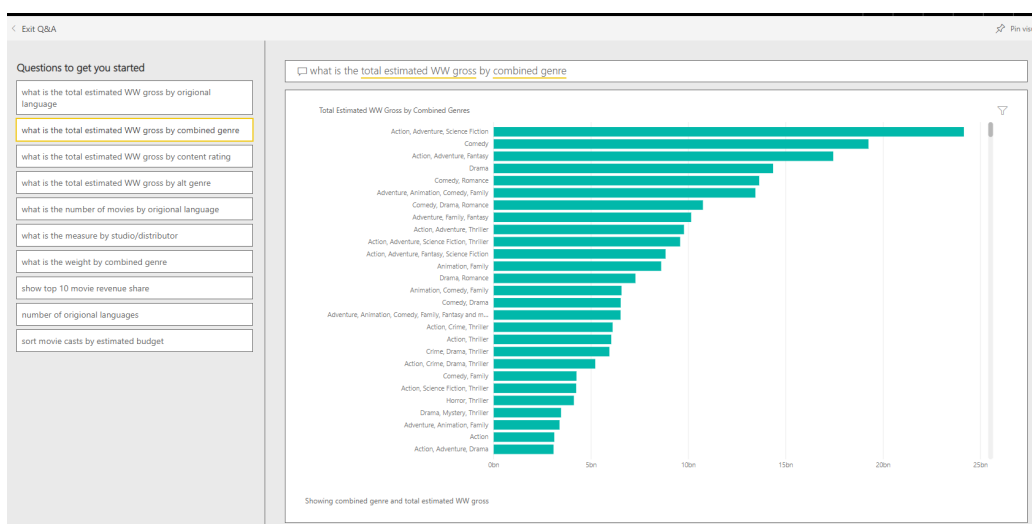
- email notifications on refresh failures
- take ownership of a dataflow
- cancel refresh
- preserving source entity relationship
- additional settings to optimise dataflow workload performance.



Q&A update for dashboards

A new update for Q&A on dashboards has been rolled out. With this revision, Microsoft has merged the existing Q&A dialog found inside reports with the dashboard experience. One of the most notable changes is the inclusion of the suggestions side pane which is available

for all users. The suggestions pane works like reports, where suggestions defined by the report author show up by default. At the same time, Q&A will also auto-generate sample suggestions to help you quickly get started and find insights inside your data.



New workspaces experience Generally Available

The new workspace experience also reached General Availability. With this announcement, the new workspace experience is now the default when creating a workspace. This will allow Power BI workspace administrators to use security groups to manage access to workspaces, enable BI teams to create workspaces without needing to create an Office 365 Group, and to provide granular workspace roles to make giving access to workspaces easier.

Create a workspace

YOU'RE CREATING AN UPGRADED WORKSPACE
Enjoy new features, better sharing options, and improved security controls.
[Revert to classic](#) | [Learn more](#)

Description

Describe this workspace

[Learn more about workspace settings](#)

Advanced

Contact list

☒ Workspace admins
☐ Specific users and groups

Enter users and groups

Workspace OneDrive

(Optional)

Dedicated capacity

☐ Off

Save

Cancel

Paginated Report Builder

The Power BI team have launched the first release of Power BI Report Builder, which is free. It's a standalone Windows Desktop application that allows you to author Paginated Reports. The application was built to serve as the primary authoring experience for paginated reports in the

Power BI, separate from SQL Server Reporting Services Report Builder. With this new tool, it provides a familiar experience but also optimises the paginated reports for publishing to the Power BI Service.

New Power BI regions

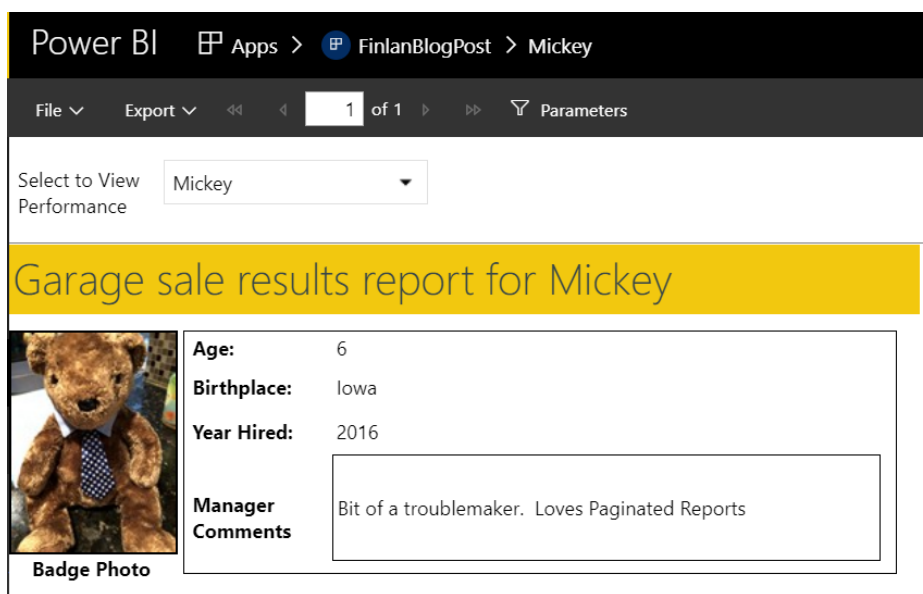
Power BI was expanded to the France Central and Korea Central regions. With the addition of these two new regions, Power BI is now available in 18 regions in Microsoft's global public cloud, in addition to six regions in their three national clouds, Power BI for US Government,

Power BI Germany, and Power BI in China (operated by 21Vianet). You can immediately leverage these by deploying your Power BI Multi-Geo Premium capacity.

Additional capabilities for paginated reports

There's been a flurry of updates to paginated reports in Power BI service, including:

- **Support for Azure Analysis Services:** paginated reports can be built on semantic models from Azure Analysis Services and Power BI datasets
- **Publishing within Power BI Apps:** paginated reports can be published from your workspaces as part of your apps in Power BI Premium

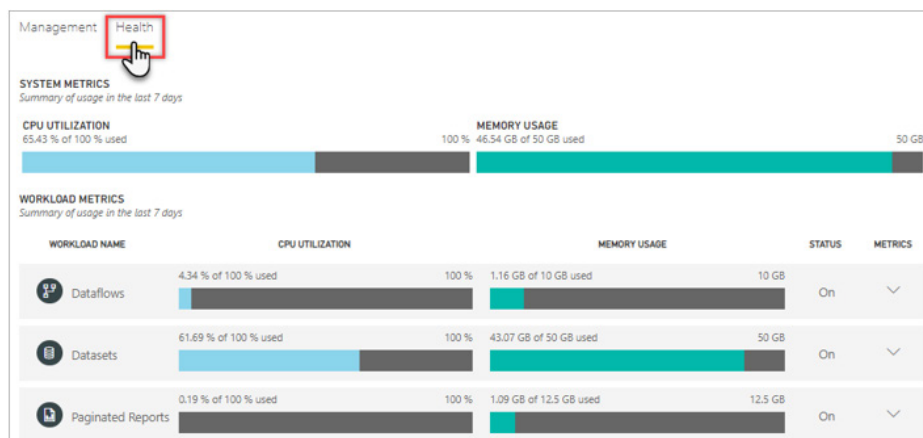


- **Email subscriptions:** you can schedule e-mail subscriptions with a PDF attachment of your full paginated report in the Power BI service. You can even set these hourly!
- **Oracle and Teradata support:** Microsoft has added Oracle and Teradata data source support to Power BI Report Builder so report authors could connect directly and begin creating paginated reports with these popular data sources.

Power BI Premium summary and workload metrics in the Admin Portal

There have been improvements made to the high-level metrics for Power BI Premium in the Admin Portal and Microsoft has also added a separate Health tab to keep track of the seven-day summary of system metrics and workload metrics. This new experience replaces the four of the following tiles that you may have seen previously (CPU, Memory

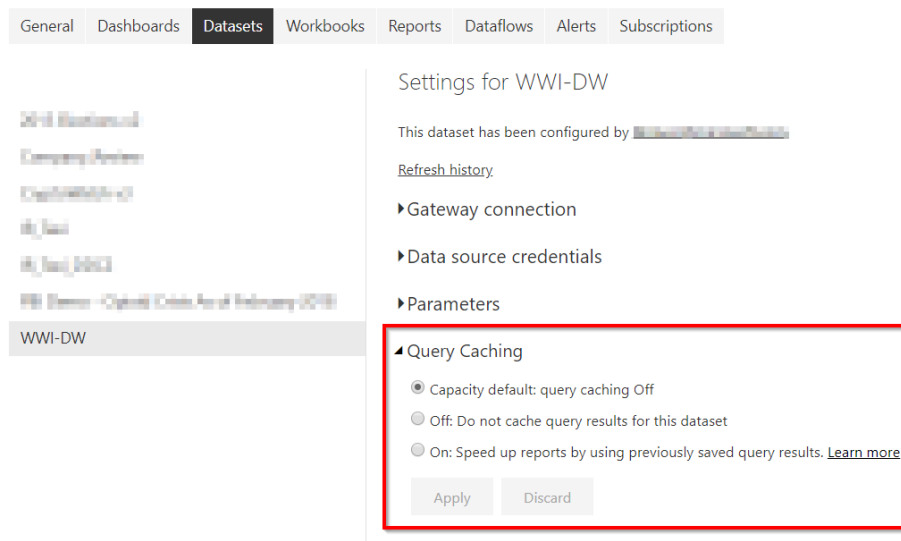
thrashing, Memory usage and DirectQuery) with a wide range of metrics measuring the usage in the past seven days to better portray the health of your capacity. It helps capacity administrators to make data driven decisions without having to leave the portal.



Query caching in Power BI Premium

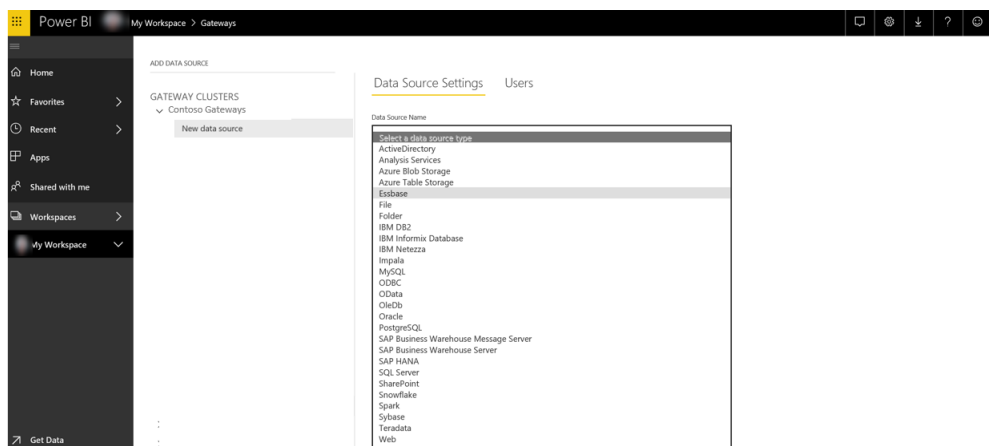
To help with the perceived performance of reports, Microsoft has announced query caching in Power BI Premium. This feature instructs the Premium capacity to use its local caching service to maintain query results, avoiding having the underlying data source compute those

results on demand. Cached query results are specific to user and report context and always respect security rules. You can control query caching behaviour on the dataset settings page in the Power BI service.

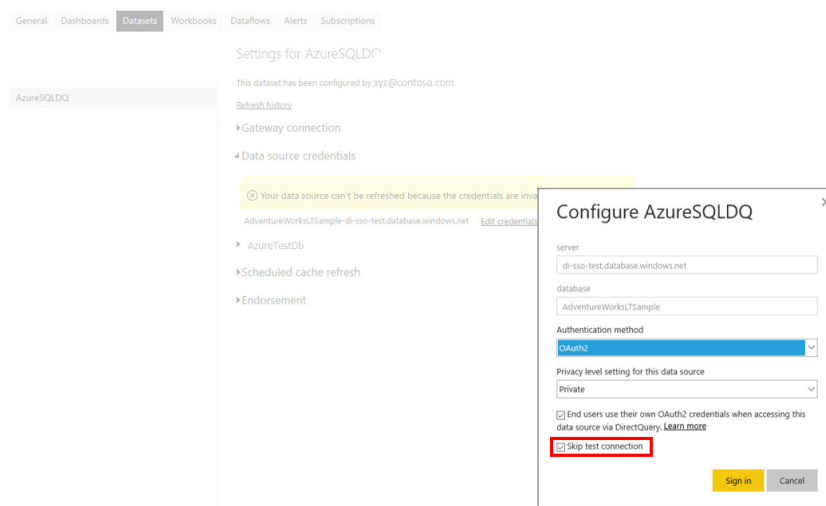


Latest updates for on-premises data gateway

A new update has been rolled out for the Power BI On-premises data gateway. This includes new connectors for Oracle Essbase and PDF support in File, and of course an updated mashup engine.



In addition, to help alleviate situations where connecting to data sources might be unavailable or slow and you keep having to retry multiple times, a new option has been added to "skip test connection" that creates a data source without running into the error.



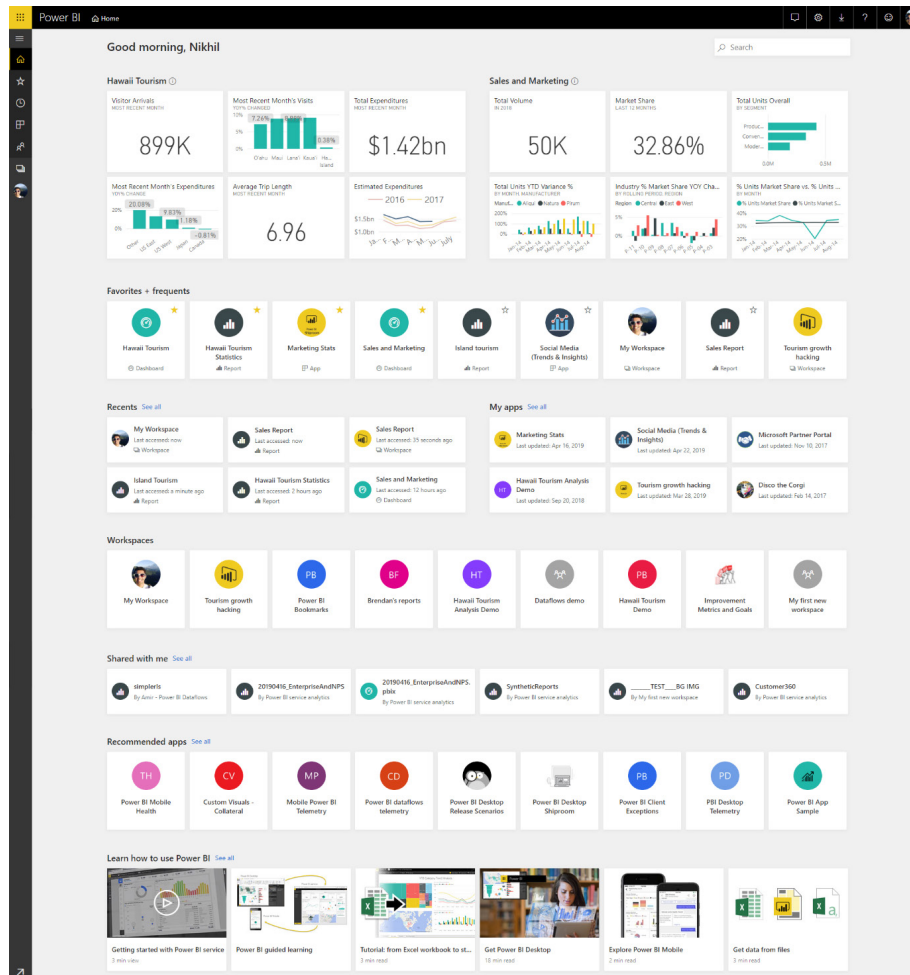
New capabilities to data gateway management in the Power Platform Admin Center

To help tenant admins have visibility into all On-premises data gateways within their organisation, two new capabilities have been added to the Data Gateway administration page which include more visibility into On-premises data gateway (Personal mode) and the ability to manage installers for On-premises data gateways.

Power BI Home GA

Microsoft has continued to improve the overall navigation experience in the Power BI service and made it easier for users to access their most important content by announcing the General Availability of Power BI Home. With this release, a separate section has been added for your

frequently used workspaces, making them accessible via 'Favorites + Frequents', and added the capability for users to 'favorite' (!) content directly on Home. Now, Home is a personalised page for both consumers and creators alike.



Workspace management tips and tricks

Microsoft has published tutorial blogs to help administrators manage workspaces more efficiently in Power BI. These include exporting the workspaces list and duplicating workspaces using Power BI cmdlets.

Roadmap updates

Here's the latest disclosures:

- **App navigation:** release date updated to May 2019
- **Shared and certified datasets:** release date updated to May 2019
- **Power BI Home:** added more details to roadmap description
- **SAP variables:** release date updated to June 2019.

More next month.

The A to Z of Excel Functions: DCOUNT

This function counts the cells that contain numbers [not text which looks like numbers] in a field (that's jargon for a column!) of records in a list or table that match one or more conditions you specify.



The **DCOUNT** function employs the following syntax to operate:

DCOUNT(database, [field], criteria)

The **DCOUNT** function has the following arguments:

- **database:** this is required and represents the range of cells that makes up the list or database. A **database** is a list of related data in which rows of related information are records and columns of data are fields. The first row of the list contains labels for each column
- **field:** indicates which column is used in the function. Make sure you enter the column label enclosed between inverted commas (double quotation marks), e.g. "Age" or "Yield", or a number (without quotation marks) that represents the position of the column within the list, that is, 1 for the first column, 2 for the second column, and so on. Microsoft's documentation states that this argument is required. We're not convinced. If **field** is omitted, **DCOUNT** counts all of the records in the table that match the criteria.
- **criteria:** is the range of cells that contains the conditions you specify.

It should be further noted that:

- You can use any range for the **criteria** argument, as long as it includes at least one column label and at least one cell below the column label for specifying the condition, e.g. if the range **G1:G2** contains the column label Income in **G1** and the amount 10,000 in **G2**, you could define the range as **MatchIncome** and use that name as the **criteria** argument in the database functions
- although the **criteria** range can be located anywhere on the worksheet, do not place the **criteria** range below the list. If you add more information to the list, the new information is added to the first row below the list. If the row below the list is not blank, Excel cannot add the new information
- make sure the **criteria** range does not overlap the list
- to perform an operation on an entire column in a database, enter a blank line below the column labels in the criteria range
- this should not be confused with the VBA function **DCOUNT**, which is used to determine the number of records that are in a specified set of records (a domain).

Please see the example below:

	A	B	C	D	E
1	Tree	Height	Height		
2	Apple	>3	<=5		
3					
4	Tree	Height	Age	Yield	Profit
5	Apple	6	20	14	105
6	Apple	4	12	10	96
7	Cherry	4.2	14	9	105
8	Apple	4.7	15	10	75
9	Pear	3	8	8	16.8
10	Apple	2.7	9	6	45
11					
12	Formula	Description	Result		
13	=DCOUNT(A4:E10,"Yield",A1:C2)	Counts the number of numeric entries in the yield column, where apple trees are over 3 metres in height, but less than or equal to 5 metres.	2		
14	=DCOUNT(A4:E10,,A1:C2)	Similar to above, this counts the number of records where the apple trees are over 3 metres in height, but are less than or equal to 5 metres in height.	2		
15					

The A to Z of Excel Functions: DCOUNTA

This function counts the nonblank cells in a field (column) of records in a list or database that match conditions that you specify.



The field argument is optional. If field is omitted, **DCOUNTA** counts all records in the database that match the criteria.

The **DCOUNTA** function employs the following syntax to operate:

DCOUNTA(database, [field], criteria)

The **DCOUNTA** function has the following arguments:

- **database:** this is required and represents the range of cells that makes up the list or **database**. A database is a list of related data in which rows of related information are records and columns of data are fields. The first row of the list contains labels for each column
- **field:** indicates which column is used in the function. This argument is optional. Make sure you enter the column label enclosed between inverted commas (double quotation marks), e.g. "Age" or "Yield", or a number (without quotation marks) that represents the position of the column within the list, that is, 1 for the first column, 2 for the second column, and so on
- **criteria:** is the range of cells that contains the conditions you specify.

It should be further noted that:

- you can use any range for the **criteria** argument, as long as it includes at least one column label and at least one cell below the column label for specifying the condition, e.g. if the range **G1:G2** contains the column label Income in **G1** and the amount 10,000 in **G2**, you could define the range as **MatchIncome** and use that name as the **criteria** argument in the database functions
- although the **criteria** range can be located anywhere on the worksheet, do not place the criteria range below the list. If you add more information to the list, the new information is added to the first row below the list. If the row below the list is not blank, Excel cannot add the new information
- make sure the **criteria** range does not overlap the list
- to perform an operation on an entire column in a database, enter a blank line below the column labels in the criteria range.

Please see the example below:

	A	B	C	D	E
1	Tree	Height	Height		
2	Apple	>3	<=5		
3					
4	Tree	Height	Age	Yield	Profit
5	Apple	6	20	14	105
6	Apple	4	12	10	96
7	Cherry	4.2	14	9	105
8	Apple	4.7	15	10	75
9	Pear	3	8	8	16.8
10	Apple	2.7	9	6	45
11					
12	Formula	Description	Result		
13	=DCOUNTA(A4:E10,"Yield",A1:C2)	Counts the number of non-blank entries in the yield column, where apple trees are over 3 metres in height, but less than or equal to 5 metres.	2		
14	=DCOUNTA(A4:E10,,A1:C2)	Similar to above, this counts the number of non-blank records where the apple trees are over 3 metres in height, but are less than or equal to 5 metres in height.	2		
15					

Criteria Examples

Typing an equal sign in a cell indicates you want to enter a formula. To display text that includes an equal sign, surround the text and the equal sign with double quotes, like so:

"=Liam"

You also do that if you're entering an expression (a combination of formulas, operators, and text) and you want to display the equal sign instead of have Excel use it in a calculation. For example:

"=>1"

Where entry is the text or value you want to find. For example:

What you type in the cell	What Excel evaluates and displays
"=Liam"	=Liam
"=>1"	>=1

- When filtering text data, Excel does not distinguish between uppercase and lowercase characters. However, you can use a formula to perform a case-sensitive search (*see below*).

The following sections provide examples of complex criteria.

Multiple criteria in one column

Boolean logic: (Salesperson = "Tim" **OR** Salesperson = "Kathryn")

To find rows that meet multiple criteria for one column, type the criteria directly below each other in separate rows of the criteria range.

e.g. In the following data range (A5:C9), the criteria range (B1:B3) displays the rows that contain either "Tim" or "Kathryn" in the **Salesperson** column (B5:B9).

	A	B	C
1	Service	Salesperson	Sales
2		=Tim	
3		Kathryn	
4			
5	Service	Salesperson	Sales
6	Consulting	Kathryn	\$ 1,744
7	Training	Tim	\$ 9,183
8	Auditing	Kathryn	\$ 2,599
9	auditing	Cecile	\$ 1,117
10			

Multiple criteria in multiple columns where all criteria must be true

Boolean logic: (Service = "Auditing" **AND** Sales > 1500)

To find rows that meet multiple criteria in multiple columns, type all of the criteria in the same row of the criteria range.

In the following data range (A5:C9), the criteria range (A1:C2) displays all rows that contain "Auditing" in the **Service** column and a value greater than \$1,500 in the **Sales** column (C5:C9).

	A	B	C
1	Service	Salesperson	Sales
2	=Auditing		>1500
3			
4			
5	Service	Salesperson	Sales
6	Consulting	Kathryn	\$ 1,744
7	Training	Tim	\$ 9,183
8	Auditing	Kathryn	\$ 2,599
9	auditing	Cecile	\$ 1,117
10			

Multiple criteria in multiple columns where any criteria can be true

Boolean logic: (Service = "Auditing" OR Salesperson = "Kathryn")

To find rows that meet multiple criteria in multiple columns, where any criteria can be true, type the criteria in different rows of the criteria range.

In the following data range (A5:C9), the criteria range (A1:B3) displays all rows that contain "Auditing" in the **Service** column (C5:C9) or "Kathryn" in the **Salesperson** column (B5:B9).

	A	B	C
1	Service	Salesperson	Sales
2	=Auditing		
3		Kathryn	
4			
5	Service	Salesperson	Sales
6	Consulting	Kathryn	\$ 1,744
7	Training	Tim	\$ 9,183
8	Auditing	Kathryn	\$ 2,599
9	auditing	Cecile	\$ 1,117
10			

Multiple sets of criteria where each set includes criteria for multiple columns

Boolean logic: ((Salesperson = "Kathryn" AND Sales >2000) OR (Salesperson = "Tim" AND Sales > 1500))

To find rows that meet multiple sets of criteria, where each set includes criteria for multiple columns, type each set of criteria in separate rows.

In the following data range (A5:C9), the criteria range (B1:C3) displays the rows that contain both "Kathryn" in the **Salesperson** column and a value greater than \$2,000 in the **Sales** column, or displays the rows that contain "Tim" in the **Salesperson** column (B5:B9) and a value greater than \$1,500 in the **Sales** column (C5:C9).

	A	B	C
1	Service	Salesperson	Sales
2		=Kathryn	>2000
3		Tim	>1500
4			
5	Service	Salesperson	Sales
6	Consulting	Kathryn	\$ 1,744
7	Training	Tim	\$ 9,183
8	Auditing	Kathryn	\$ 2,599
9	auditing	Cecile	\$ 1,117
10			

Multiple sets of criteria where each set includes criteria for one column

Boolean logic: ((Sales > 2000 AND Sales <= 3000) OR (Sales < 1500))

To find rows that meet multiple sets of criteria, where each set includes criteria for one column, include multiple columns with the same column heading.

In the following data range (A5:C9), the criteria range (C1:D3) displays rows that contain values between 2,000 and 3,000 and values less than 1,500 in the **Sales** column (C5:C9).

	A	B	C	D
1	Service	Salesperson	Sales	Sales
2			>2000	<=3000
3			<1500	
4				
5	Service	Salesperson	Sales	
6	Consulting	Kathryn	\$ 1,744	
7	Training	Tim	\$ 9,183	
8	Auditing	Kathryn	\$ 2,599	
9	auditing	Cecile	\$ 1,117	
10				

Criteria to find text values that share some characters but not others

To find text values that share some characters but not others, do one or more of the following:

- type one or more characters without an equal sign (=) to find rows with a text value in a column that begin with those characters. For example, if you type the text **Lia** as a criterion, Excel finds "Liam", "Liar" and "Lianne"
- use a wildcard character.

The following wildcard characters can be used as comparison criteria:

Use	To Find
? (question mark)	Any single character For example, sm?th finds "smith" and "smyth"
* (asterisk)	Any number of characters For example, *east finds "Northeast" and "Southeast"
~ (tilde) followed by ? , * or ~	A question mark, asterisk, or tilde For example, fy91~? finds "fy91?"

In the following data range (**A5:C9**), the criteria range (**A1:B3**) displays rows with "Co" as the first characters in the **Service** column or rows with the second character equal to "i" in the **Salesperson** column (**B5:B9**).

	A	B	C
1	Service	Salesperson	Sales
2	Co		
3		=?i*	
4			
5	Service	Salesperson	Sales
6	Consulting	Kathryn	\$ 1,744
7	Training	Tim	\$ 9,183
8	Auditing	Kathryn	\$ 2,599
9	auditing	Cecile	\$ 1,117
10			

Criteria created as the result of a formula

You can use a calculated value that is the result of a formula as your criterion. Remember the following important points:

- the formula must evaluate to TRUE or FALSE
- because you are using a formula, enter the formula as you normally would, and do not type the expression in the following way:
="= entry "
- do not use a column label for criteria labels; either keep the criteria labels blank or use a label that is not a column label in the range (in the examples below, **Calculated Average** and **Exact Match**)
- if you use a column label in the formula instead of a relative cell reference or a range name, Excel displays an error value such as #NAME? or #VALUE! in the cell that contains the criterion. You can ignore this error because it does not affect how the range is filtered
- the formula that you use for criteria must use a relative reference to refer to the corresponding cell in the first row (in the examples below, **C6** and **A6**)
- all other references in the formula must be absolute references.

The following subsections provide specific examples of criteria created as the result of a formula.

Filtering for values greater than the average of all values in the data range

In the following data range (**A5:C9**), the criteria range (**D1:D2**) displays rows that have a value in the Sales column greater than the average of all the values (**C6:C9**). In the formula, "C6" refers to the filtered column (**C**) of the first row of the data range (6).

	A	B	C	D
1	Service	Salesperson	Sales	Calculated Average
2				=C6>AVERAGE(\$C\$6:\$C\$9)
3				
4				
5	Service	Salesperson	Sales	
6	Consulting	Kathryn	\$ 1,744	
7	Training	Tim	\$ 9,183	
8	Auditing	Kathryn	\$ 2,599	
9	auditing	Cecile	\$ 1,117	
10				

Filtering for text by using a case-sensitive search

In the data range (A5:C9), the criteria range (D1:D2) displays rows that contain "Auditing" in the **Service** column by using the EXACT function to perform a case-sensitive search (A5:A9). In the formula, "A6" refers to the filtered column (A) of the first row of the data range (6).

	A	B	C	D
1	Service	Salesperson	Sales	Exact Match
2				=EXACT(A6,"Auditing")
3				
4				
5	Service	Salesperson	Sales	
6	Consulting	Kathryn	\$ 1,744	
7	Training	Tim	\$ 9,183	
8	Auditing	Kathryn	\$ 2,599	
9	auditing	Cecile	\$ 1,117	
10				

More Excel Functions next month...

Upcoming SumProduct Training Courses

Location	Course	Date	Duration
Sydney	Excel Tips & Tricks	3 Jun 2019	1 Day
Sydney	Financial Modelling	4 - 5 Jun 2019	2 Days
Sydney	Power Pivot, Power Query and Power BI	10 - 12 Jun 2019	3 Days
Sydney	Excel Tips and Tricks	15 Jul 2019	1 Day
Sydney	Financial Modelling	16 - 17 Jul 2019	2 Days
Sydney	Power Pivot, Power Query and Power BI	22 - 24 Jul 2019	3 Days
Sydney	Excel Tips and Tricks	19 Aug 2019	1 Day
Melbourne	Financial Modelling	20 - 21 Aug 2019	2 Days
Sydney	Financial Modelling	20 - 21 Aug 2019	2 Days
Melbourne	Power Pivot, Power Query and Power BI	26 - 28 Aug 2019	3 Days
Sydney	Power Pivot, Power Query and Power BI	26 - 28 Aug 2019	3 Days
Melbourne	Excel Tips and Tricks	17 Sep 2019	1 Day
Sydney	Excel Tips and Tricks	23 Sep 2019	1 Day
Sydney	Financial Modelling	24 - 25 Sep 2019	2 Days
Sydney	Power Pivot, Power Query and Power BI	7 - 9 Oct 2019	3 Days
Melbourne	Financial Modelling	14 - 15 Oct 2019	2 Days
Melbourne	Power Pivot, Power Query and Power BI	22 - 24 Oct 2019	3 Days

Sydney	Excel Tips and Tricks	4 Nov 2019	1 Day
Sydney	Financial Modelling	5 - 6 Nov 2019	2 Days
Sydney	Power Pivot, Power Query and Power BI	11 - 13 Nov 2019	3 Days
Melbourne	Excel Tips and Tricks	25 Nov 2019	1 Day
Melbourne	Financial Modelling	26 - 27 Nov 2019	2 Days
Melbourne	Power Pivot, Power Query and Power BI	9 - 11 Dec 2019	3 Days
Sydney	Power Pivot, Power Query and Power BI	9 - 11 Dec 2019	3 Days
Sydney	Excel Tips and Tricks	16 Dec 2019	1 Day
Sydney	Financial Modelling	17 - 18 Dec 2019	2 Days

Key Strokes

Each newsletter, we'd like to introduce you to useful keystrokes you may or may not be aware of. This month, we thought we would complete going through the function keys, this time with the **CTRL ALT + SHIFT** combination:

Keystroke	What it does
CTRL + ALT + SHIFT + F2	Print
CTRL + ALT + SHIFT + F4	Close application
CTRL + ALT + SHIFT + F9	Recalculate, full rebuild

There are over 540 keyboard shortcuts in Excel. For a comprehensive list, please download our Excel file at www.sumproduct.com/thought/keyboard-shortcuts. Also, check out our new daily **Excel Tip of the Day** feature on the www.sumproduct.com homepage.

Our Services

We have undertaken a vast array of assignments over the years, including:

- Business planning
- Building three-way integrated financial statement projections
- Independent expert reviews
- Key driver analysis
- Model reviews / audits for internal and external purposes
- M&A work
- Model scoping
- Power BI, Power Query & Power Pivot
- Project finance
- Real options analysis
- Refinancing / restructuring
- Strategic modelling
- Valuations
- Working capital management

If you require modelling assistance of any kind, please do not hesitate to contact us at contact@sumproduct.com.

Link to Others

These newsletters are not intended to be closely guarded secrets. Please feel free to forward this newsletter to anyone you think might be interested in converting to "the SumProduct way".

If you have received a forwarded newsletter and would like to receive future editions automatically, please subscribe by completing our newsletter registration process found at the foot of any www.sumproduct.com web page.

Any Questions?

If you have any tips, comments or queries for future newsletters, we'd be delighted to hear from you. Please drop us a line at newsletter@sumproduct.com.

Training

SumProduct offers a wide range of training courses, aimed at finance professionals and budding Excel experts. Courses include Excel Tricks & Tips, Financial Modelling 101, Introduction to Forecasting and M&A Modelling.

Check out our more popular courses in our training brochure:



Drop us a line at training@sumproduct.com for a copy of the brochure or download it directly from <http://www.sumproduct.com/training>.

Sydney Address: SumProduct Pty Ltd, Suite 803, Level 8, 276 Pitt Street, Sydney NSW 2000
New York Address: SumProduct Pty Ltd, 48 Wall Street, New York, NY, USA 10005
London Address: SumProduct Pty Ltd, Office 7, 3537 Ludgate Hill, London, EC4M 7JN, UK
Melbourne Address: SumProduct Pty Ltd, Level 9, 440 Collins Street, Melbourne, VIC 3000
Registered Address: SumProduct Pty Ltd, Level 6, 468 St Kilda Road, Melbourne, VIC 3004

contact@sumproduct.com
www.sumproduct.com
+61 3 9020 2071