We MARCH on with the updates this month

with updates for Power BI Desktop, Service, Mobile, Excel 365 and the **RANDARRAY** function. We also ask you for any questions to raise with the Microsoft Excel team as we head out to Redmond for the usual discussions, brainstorming sessions and the team-bonding partaking of fruit juices and mineral waters…

With our regular series on Power Pivot, Power Query, VBA, Keyboard Shortcuts and the A to Z of Excel Functions all ever-present, it's yet another monthly newsletter "slightly" larger than a leaflet!

Until next month.

Liam Bastick, Managing Director, SumProduct

MVP Summit

Due to Non-Disclosure Agreements being in place the Microsoft Most Valuable Professional (MVP) Summit may not be "summit we can talk about" in general. However, March heralds the 2019 get-together of Most Valuable Professionals (MVPs) [as accredited by Microsoft] to network and discuss the latest developments in their area of expertise – in our case, Excel. SumProduct has two MVPs in its rank presently – Tim Heng and Liam Bastick – with both finally getting to go this year.

Scheduled for mid - March at Redmond, we also get a chance to ask our questions to the developers. Therefore, if you have a question you'd like us to ask, drop us a line at contact@sumproduct.com.


Yes, this is the photo we always pull out at this time of year...

Excel 365 – Latest Improvements

If you keep trying to decide whether to go for the latest and greatest Excel product (Excel 2019) or instead head for its Office 365 counterpart (colloquially known as "Excel 365"), here's some sweeteners which might turn your head towards the latter.

Aside from the headline-inducing Dynamic Arrays from last year, there's several quirky new features that have flown under the radar, which are in the very latest versions of Office 365 (*e.g.* Insider Fast), but most likely won't be coming to Excel 2019.

The first is the newly positioned 'Search' box. If you recall, Excel 2016 onwards had a 'Tell me what you want to do' search box next to the 'Help' tab on the Ribbon:
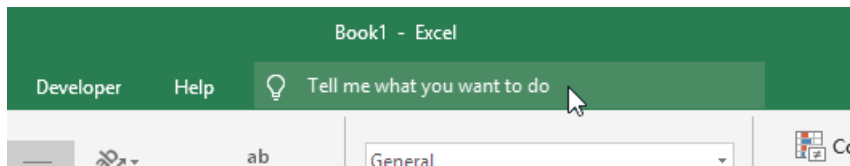
Well, it's had a facelift – not only has it gone "up" in the world (quite literally, it's at the top of the Ribbon where the filename used to be), it's become wider as well:
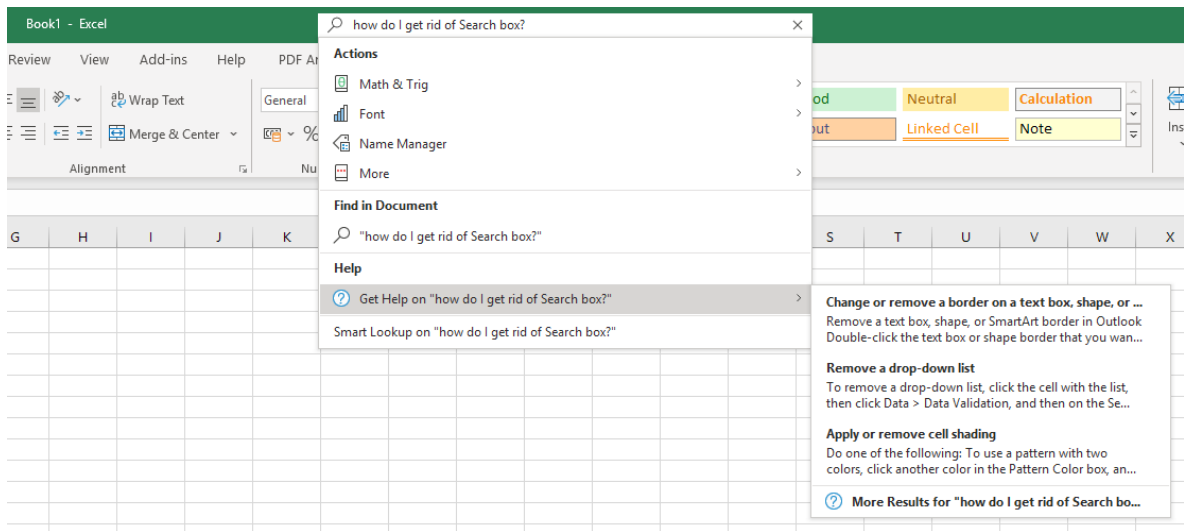

The old version had 'Actions' and 'Help'; this latest incarnation also adds 'Find in Document' to the search results too. The keyboard shortcut **ALT + Q** still activates the new search box, just like its earlier cohort.

Our initial reaction is whilst the extended results are welcomed, the revitalised search box is too big and too central. If you have a longer filename, you might not be able to see it all, for example. We'd love to see it go like Cortana, where the search box is less obtrusive, and the search

box could be switched for an icon with a simple right-click. What do you mean, we're resistant to change!?

The next improvement is welcomed, although we think the presentation could be better. If you are a typical Excel user, you may create a new Excel workbook, undertake some work and nonchalantly close it forgetting you hadn't yet saved it. Don't worry, that's happened to us all! If you do that in the latest versions of Excel 365, you'll get the following message:
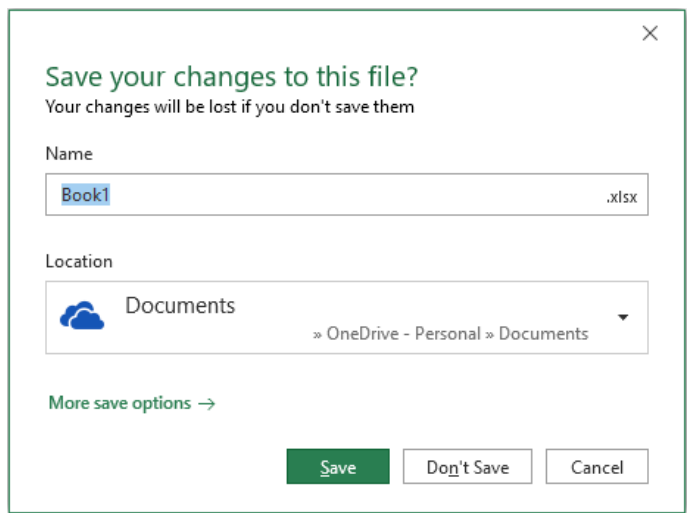

This dialog box is pretty much "in your face" and 99 times out of 98 appears to want to save everything to OneDrive, no matter which file directory you were working in (although it's easy to change by either clicking on the dropdown arrow in the 'Location' box or clicking on 'More save options').

The main criticisms here appear to be the above issue on OneDrive above

all and the fact that the dialog box does not look like 99% of all Excel dialog boxes. So, it's not just us resistant to change then.

The final improvement concerns Tooltips. If you hover over buttons / tabs / icons in Excel, the Tooltips have become significantly more verbose. For example, take a look at the verbiage concerning the Format Painter:
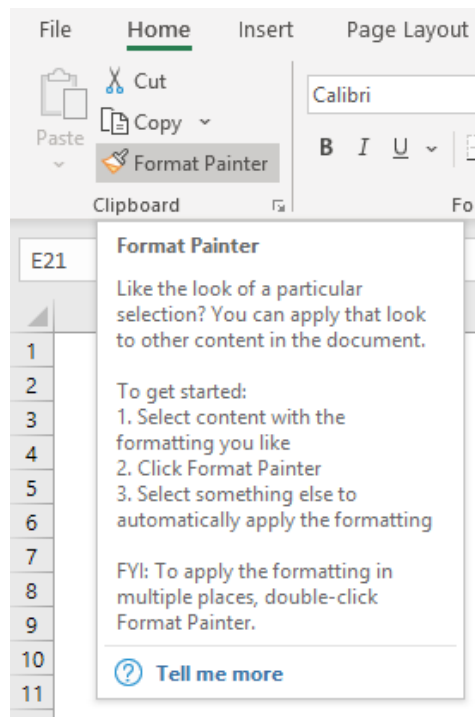
It almost makes me want to use it. 'Help' has been getting much more user friendly of late. Microsoft has a big push presently on regarding accessibility, and these new, improved Tooltips are definitely a step in the right direction.

Now, what's it to be? Excel 2019 or Excel 365?

New Improved RANDARRAY Function Coming Soon to Office 365 Excel

OK, so this is still in what Microsoft refers to as "Preview" mode, *i.e.* it's not yet "Generally Available" but it is on the outskirts of civilisation. **RANDARRAY** is still a relatively new function found in some editions of the "Office Insider" programme which is an Office 365 fast track. You can register in **File -> Account -> Office Insider** in Excel's backstage area.
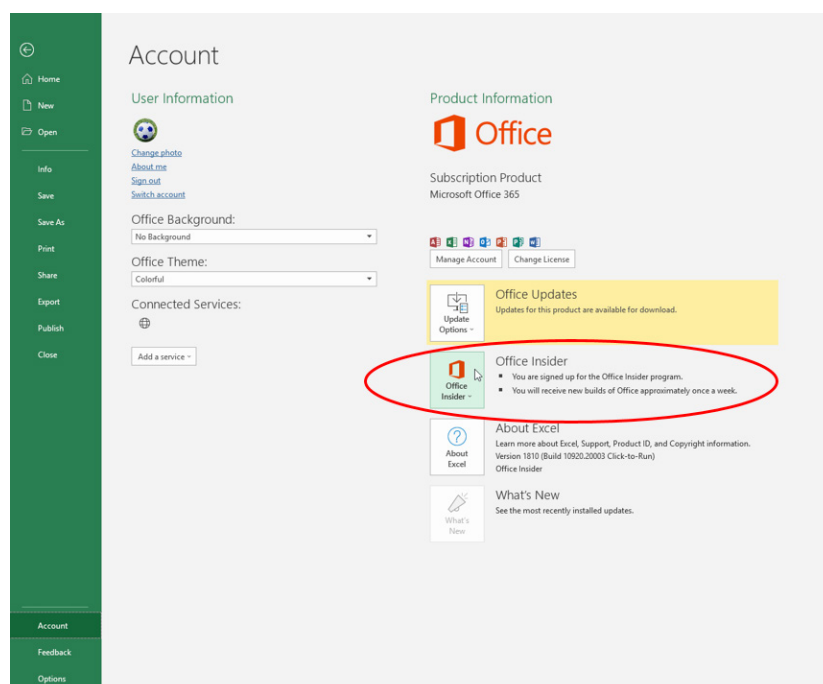
Even then, you're not guaranteed a ticket to the ball as only some will receive the new features as Microsoft slowly roll out these features and functions. Please don't let that put you off. These features <u>will</u> be with all Office 365 subscribers soon.

We first mentioned **RANDARRAY** back in September. Even though it's not yet Generally Available, it's already had a facelift. Oh yes – Microsoft is invested in these functions!

Originally, the **RANDARRAY** function returned an array of random numbers between 0 and 1. It's not clear from Microsoft, analogous to the pre-existing **RAND** function, which generates a number greater than or equal to zero and strictly less than one. However, there was a general sense of underwhelm with this function and the new and improved version has just been released. It now allows you to set you own maximum and minimum _and_ decide whether you want the values returned to be decimals (_e.g._ 17.4381672...) or integers (whole numbers).

The new syntax for the function is now as follows:

$$=RANDARRAY([rows], [columns],[min],[max],[integer]).$$

The function has five arguments, all supposedly optional (but upon testing, we weren't quite as convinced):

- **rows:** this specifies how many rows the results should spill over. If omitted, the default value is 1
- **columns:** this specifies how many columns the results should spill over. If omitted, the default value is also 1
- **min:** this is the minimum value that may be selected randomly. If this is not specified, it is assumed to be zero (0)
- **max:** this is the maximum value that may be selected randomly. If this is not specified, it is assumed to be 1
- **integer:** if this is set to TRUE, only integer outputs are allowed; the default value (FALSE) provides non-integer (decimal) results.

Other points to note:

- if **rows** or **columns** refers to a blank cell reference, this will generate the new _#CALC!_ error
- if **rows** or **columns** are entered as decimals, the values used will be truncated to the number before the decimal point (_e.g._ 3.9999999 will be treated as 3)
- if **rows** or **columns** is a value less than 1, _#CALC!_ will be returned
- if **integer** is set to TRUE and either **min** or **max** is not an integer, this will generate an _#VALUE!_ error
- **max** must be greater than or equal to **min**, else the error _#VALUE!_ is returned.

When we originally discussed the **RANDARRAY** function, we used this rather comprehensive example to create a list of random integers between two values:

Originally, the formula in cell **F44** was

$$=ROUNDDOWN(RANDARRAY(H36,H37)*(H39-H38+1),0)+INT(H38)$$

and the article explained how this worked. However, it's much easier now:

The "new improved" formula in cell **F45** (it's moved down a row due to the additional argument required in cell **H40**) is simply

$$=RANDARRAY(H36,H37,H38,H39,H40).$$

Cool, eh?

Dynamic Arrays in Action

…and another thing about Dynamic Arrays. You may recall in a previous newsletter we discussed the concept of implicit intersections. In the past, if you entered **=A$1:A$10** anywhere in rows 1 through 10, the formula would return only the value from that row. But the times, they are a-changing: the brave new world of Office 365 (albeit selected Insider recipients for the time being), typing this formula would create a Spilled Array Formula. To protect existing formulae / behaviour, Microsoft originally called upon a new function, **SINGLE**, to ensure what is known as "legacy array behaviour".

Regular readers may recall the **SINGLE** function was born to ensure that Excel would keep Excel running smoothly. The **SINGLE** function returned a single value using this implicit intersection logic.

In the initial release of Dynamic Arrays to *Office 365 Insider Fast*, when **SINGLE** referred to a range, this function would return the cell at the intersection of the row or column of the formula cell. Where there was no intersection, or more than one cell falls in the intersection, then **SINGLE** would return a *#VALUE!* error. When the supplied argument is an array, **SINGLE** returns the first item (the "top left-hand corner", namely Row 1, Column 1).

For example, originally, the two **SINGLE** formulae are supplied a range, **H13:H27**, and return the values in cells **H17** and **H22** respectively.

Well, it's all changed. Ladies and gentlemen, let me introduce you to the new implicit intersection operator, **@**.

That's right – Microsoft has changed its great collective mind and kicked **SINGLE** to the kerb for the more succinct "**@**" operator.

It's true that the **SINGLE** function was causing confusion and **@** seems less obtrusive. But there was more to it than that. Since Excel 2010, the **@** symbol has already been employed in Table references to indicate implicit intersection (and it was used there to replace previous Excel syntax too!).

For example, if you consider the Table formula **=[@Column1]**. Here, the **@** indicates that the formula should use implicit intersection to retrieve the value on the same row from **[Column1]**.

Going forward then, **@** will replace **SINGLE**, and functions that return multi-cell ranges or arrays will be prefixed ("prepended" in Microsoft-speak) with **@** if they were authored in an older version of Excel.

It is important to note that there is no change to the way your previous formula behaves – that's the entire point: it's necessary to prevent the new "spilled" behaviour. Common functions that could return multi-cell ranges include **INDEX**, **OFFSET** and good ol' User Defined Functions (UDFs). It should be noted though that a common exception is if they are wrapped in a function – often an aggregate one – that accepts an array or range (*e.g.* **SUM()** or **AVERAGE()**).

Microsoft has provided examples to try and clarify how things should work:

Original Formula	As seen in Dynamic Array Excel	Explanation
=SUM(A1:A10)	**=SUM(A1:A10)**	There is no change: no implicit intersection should occur, as the **SUM** function expects ranges or arrays.
=A1+A2	**=A1+A2**	No change: no implicit intersection should occur.
=A1:A10	**=@A1:A10**	Implicit intersection will occur.
=INDEX(A1:A10,B1)	**=@INDEX(A1,A10,B1)**	Implicit intersection could occur. The **INDEX** function can return an array or range when its second or third argument is zero (0).
=OFFSET(A1:F6,2,3)	**=@OFFSET(A1:F6,2,3)**	Implicit intersection could occur. The **OFFSET** function can return a multi-cell range; in this instance, implicit intersection would be triggered.
=MyUDF()	**=@MyUDF**	Implicit intersection could occur. User Defined Functions (UDFs) can return arrays; in this instance, implicit intersection would be triggered.

Remembering that the **@** symbol will only appear in "Dynamic Array Excel", do consider carefully the ramifications of removing these pesky little **@** critters.   Before deleting, check the function / formula immediately after the **@** symbol. With regards to this expression:

- if it returns a single value (the most common case), there will be no change by removing the **@** symbol
- if it returns a range or an array, removing the **@** will cause the formula to spill into the neighbouring cells
- if you remove the **@** and later open the workbook in an older version of Excel, it will appear as a legacy array formula (wrapped with the **CTRL + SHIFT + ENTER** braces {}): this is done to ensure previous versions of Excel will not trigger implicit intersection.

You have been warned. Don't put your spreadsheets **@** risk!! You can check out more about Dynamic Arrays at www.sumproduct.com/thought/getting-arrays-spilling-the-beans-on-seven-new-functions.

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 continue last month's discussion on running macros – this time based upon events.

Last month we took a look at a few ways to run a macro:

1. From the 'Macro' dialog box
2. Running it from the VBA Editor
3. Using a predefined keyboard shortcut
4. Anchoring the macro to a form control
5. Setting it to automatically run on a specific event.

In February's newsletter, we checked out the first four methods – this time we come in for a final approach (is this "plane speaking"?).

While most VBA scripts tend to be triggered by buttons or commands that a user can execute, it can also be programmed to run when a particular action is triggered elsewhere in the Excel workbook. These triggers can be found in the VBA Editor and navigating to a Sheet or to **ThisWorkbook**.

Worksheet events

There are a couple of key worksheet events generally used:

1. **Activate**
2. **SelectionChange.**

The **Activate** event will trigger whenever the worksheet in question is activated (*e.g.* if a user clicks on the worksheet tab in Excel).

```
Private Sub Worksheet_Activate()

    'Run some code here
    Application.Calculate


End Sub
```

This sort of event can be used to trigger an update or clean-up of data on a sheet. It is commonly used to refresh PivotTables when a sheet is selected, with the code cycling through available PivotTables and refreshing each one.

The **SelectionChange** triggers whenever a new cell is selected within the worksheet, *e.g.* when a cell is clicked on, or when the selection range is expanded or reduced. This event is a little different, in that there is also an additional parameter that becomes available, initially referred to as "ByVal Target as Range". This allows you to pass the selected area through to the VBA script being run.

Workbook events

There are several key workbook events that are commonly used in practice:

1. **Activate**
2. **BeforeClose**
3. **BeforeSave**
4. **Open.**

In particular, the **Open** event is commonly used to set up disclaimer messages, user guides and instructions, or for model clean-up. However, this relies upon the user enabling macros prior to the macro attempts to run. For a more robust process, any clean-up or disclaimer script should be written prior to the file being saved, so that any user will be required to respond to the disclaimer before the file can be used.
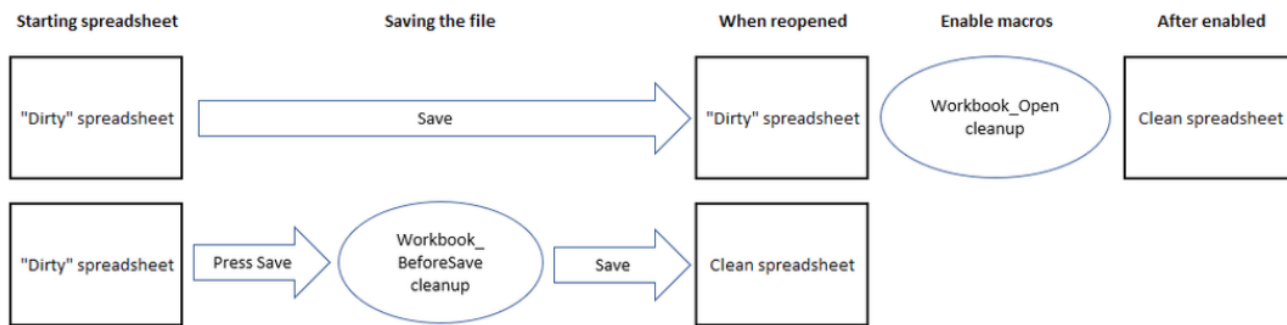
This is where the **BeforeSave** event is particularly useful. The event is triggered upon a **Save** or **SaveAs** command, and allows users to run any clean-up prior to the file being saved. The macro can be set to restore the file after it has been saved, so that the user won't need to go through the disclaimer process, resulting in a seamless save action.

Disabling events

While events can be useful, at times you may need to save the file as part of another macro script, assuming that the event will not be triggered. In these instances, you need to include the command:

Application.EnableEvents = False

This will prevent any events from being triggered until you restore the actions using:

Application.EnableEvents = True

If you decide to switch off events at the start of your macro to improve the speed and efficiency of your macro run, remember to switch it back on subsequently after your macro finishes running.

More next month.

Power Pivot Principles

We've been running the Power Query / Get & Transform series in our newsletter for some time now, so we thought it might be worth paying some attention to its sister feature, Power Pivot. This month, we look at importing data.

Power Pivot has the ability to import data from a variety of sources: *e.g.* a database, data service, Excel worksheet and even data feeds, just to name a few. Once the data has been imported, you can make changes to your data or add new data to the source data which will refresh the Power Pivot data model. We will go through some of the different methods to import data below.
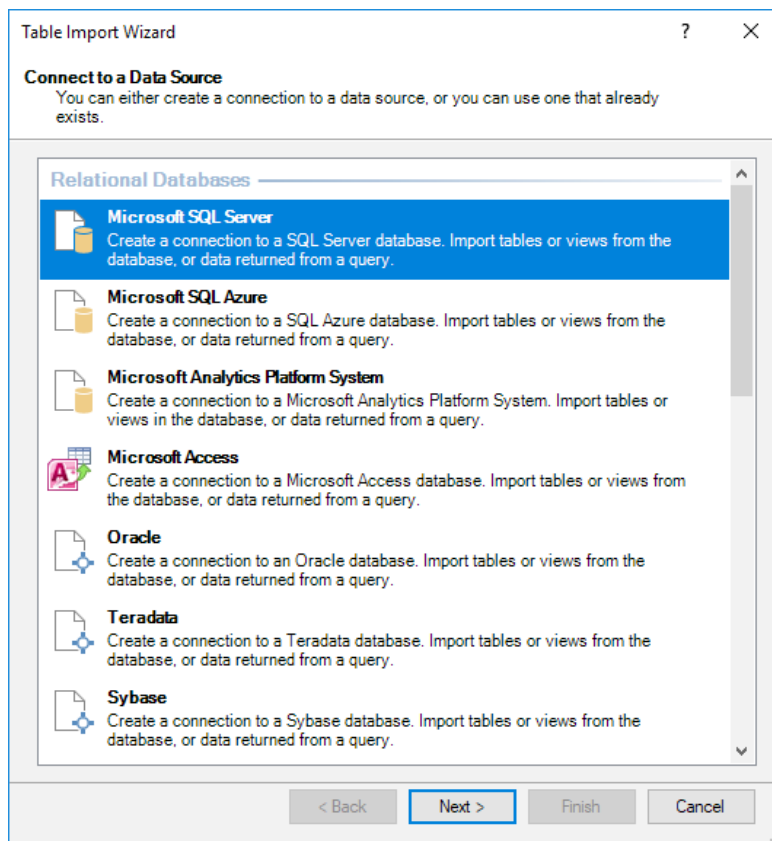
For example, you can see many of the alternatives by first clicking on the 'Manage' button on the 'Power Pivot' tab of the Ribbon:
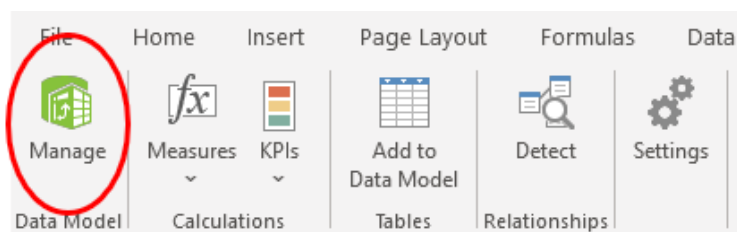
Then, on the 'Home' tab of the 'Power Pivot' window, you can select 'From Other Sources' in the 'Get External Data' section of the Ribbon:
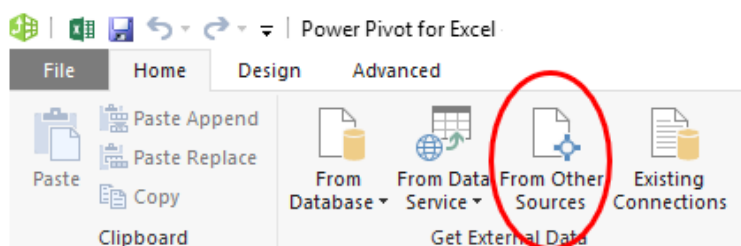
This calls the 'Table Import Wizard' where you can see many sources available for importing:


This would be the way to link to another Excel workbook for example (which would need to be closed).

### Add to Data Model from Excel

Data that has been loaded into Power Pivot is unable to be modified. That's a *good* thing. To circumvent this issue, you may load data from Excel. Loading data from a table in Excel enables you to make changes to the data in Excel, *e.g.* adding rows or columns, which in turn will cascade through to the Power Pivot file when the data is refreshed. To do this, select your Excel data and then click on 'Add to Data Model' on the 'Power Pivot' tab of the Ribbon:
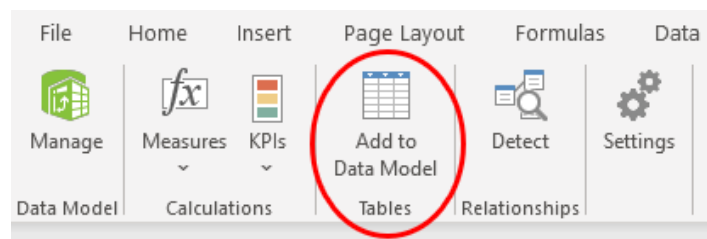
(Linking data is no longer possible as it was discovered this was causing file corruption issues.)

Copy and Paste
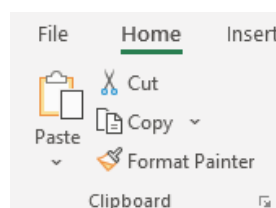
Another way to bring data into Power Pivot is by copying and pasting data. However, there are several limitations to loading data this way. For example, the columns become fixed once the data is pasted. It is not possible to add extra columns of data but you can create new calculated columns.

It is also possible to 'Paste Append' and 'Paste Replace'. If these methods are used the data source must be exactly the same as the Power Pivot file. Please be mindful of this limitation before copying and pasting data.

Database

To bring in data from a database, you will have to use the Table Import Wizard *(shown above)*. With this wizard, you are able to import data from a variety of relational databases such as SQL Server and Access Database.

Data Format

It is important to note that once you have imported your data into PowerPivot, the source type cannot be changed. If you create a table via an Excel file you cannot change it in later to be linked from a database or another source. The table will need to be deleted and then re-created from the new source.

Editing Data

Once data is loaded into Power Pivot, it is unable to be changed from the Power Pivot window. Power Pivot data is read-only, so any changes that are required will need to be made from the source data (If possible) and then refreshed. Yes, you can add / remove columns, but the data itself (if retained) may not be changed.

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 (Excel 2016) 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 extracting data by using differences in data types.

When transforming data with Power Query, the most important skill is knowing where to start, and this month we'll take a look at one such scenario.

Let's say we have received a CSV file containing expense information for a number of employees. The creator of the file has included heading information, but it is embedded in the expense data. Starting from a fresh Excel workbook, we load the data by going to the 'Power Query' tab, and choosing 'From File', then selecting the 'From Text/CSV' option:
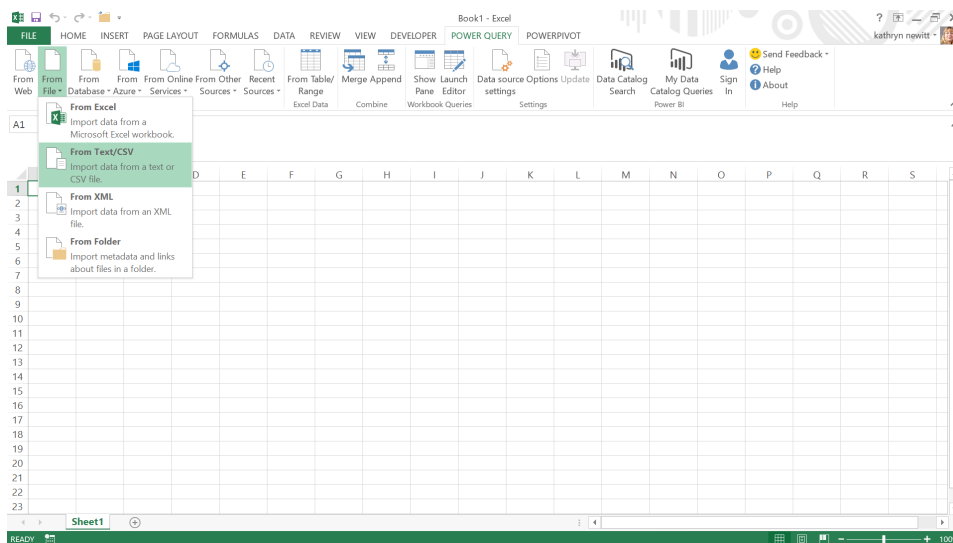
Having selected the file, let's choose to edit prior to loading so that we may access the Power Query Editor.
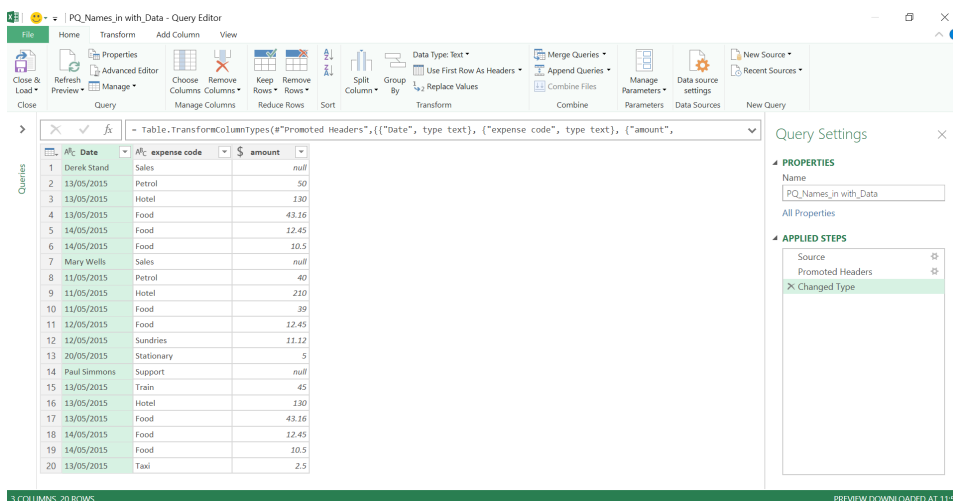
You should note that because the top row contained text for each column, Power Query has correctly assumed that this is the title row and 'Promoted Headers' accordingly which saves us renaming the columns and deleting that row. However, now we can see the problem. Since the creator of the file has included the employee and department information between the expense rows, the **Date** column has some names in it, and the **expense code** column has some departments. We need to extract the name and department data from the other rows.

In these situations, we have to look for a way of distinguishing the data that we need to extract. Our **expense code** column doesn't help us much, as departments and expense codes look fairly similar. In the **Date** column, however, the names are in a different format to the dates, and this is a reliable indicator, as I am not likely to have someone with a name that resembles a date!

We have two problems – we want to get rid of the rows that have an 'irregular' date, but we also need to extract the name data. The first step we will take, is to make a copy of the **Date** column. To do this, right-click on the **Date** column and choose 'Duplicate Column':
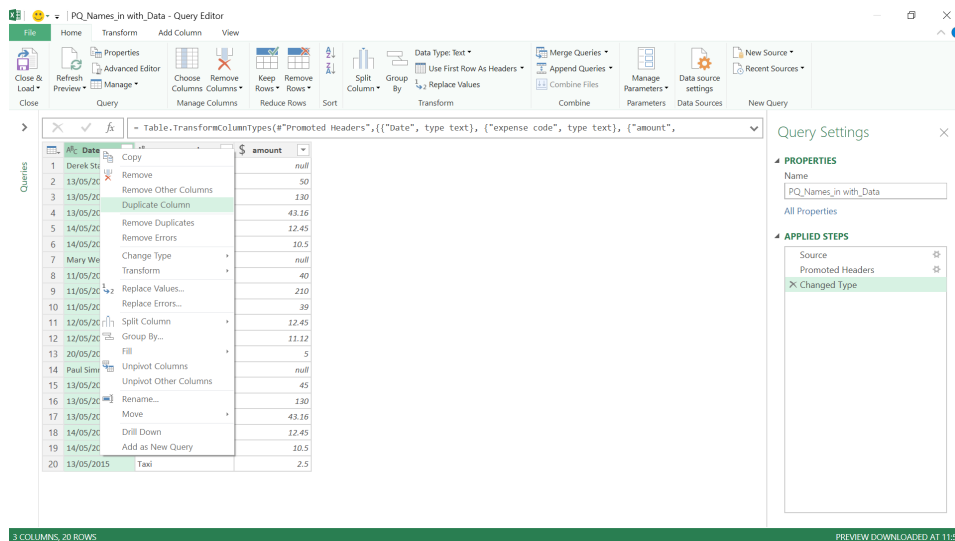
We complete this process and create a **Date – Copy** column which can be manipulated without losing data from the original column. Let's right-click the copied column and choose to change the type to 'Date', which means that the names are flagged as errors:
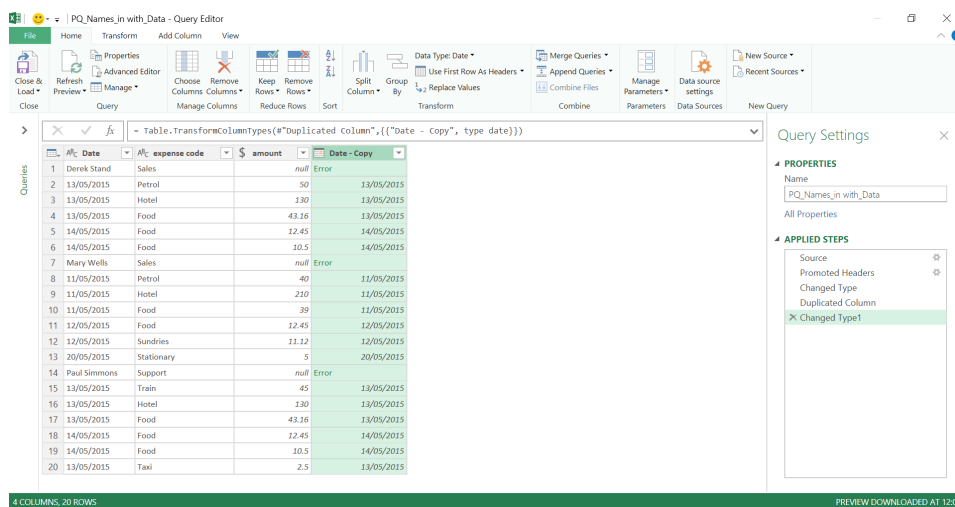
We can now reset the data in the 'Error' entries to null – just right click the copied column again and choose to 'Replace Errors' with null.
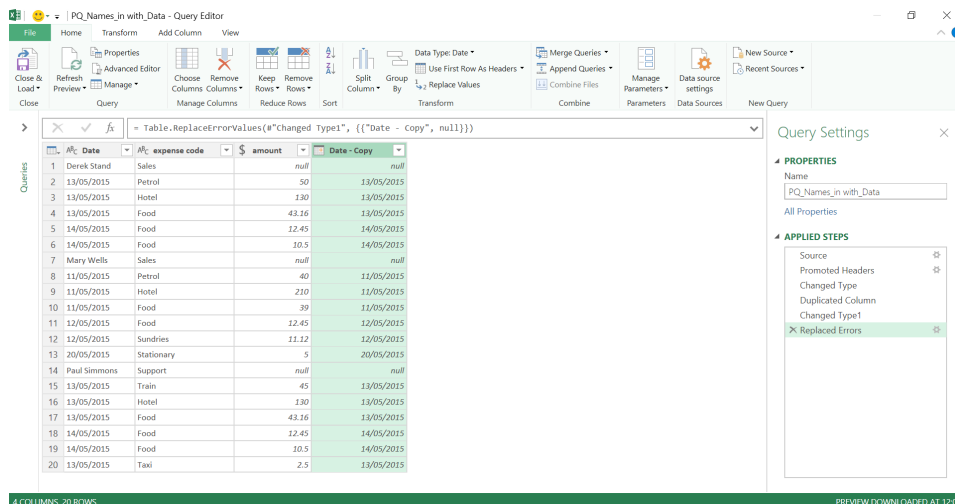
The reason that null is more useful to me than an error, is because we may use null in the conditional logic that we need to extract my name data. Therefore, we'll create a new column by choosing 'Conditional Column' from the 'General' section on the 'Add Column' tab. We want to create a column that is populated with the employee name.
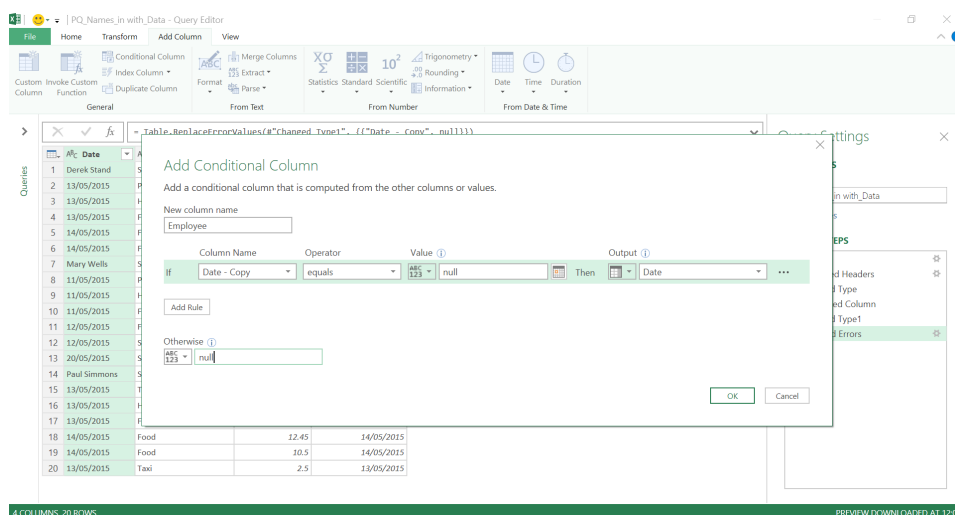

Power Query builds the logic statement, so we just need to supply the location of the data.

If Date_Copy equals null, Then Date, Otherwise null

Note that the default for the **Value** and the **Output** is a value, and since the chosen output is a column, we need to select that from the icon under **Output**, otherwise we will be inserting the value 'Date' instead of the value in **Date**. Choosing to output null when the condition is not met will help with the next step.

With this borne in mind, click 'OK' to see the new column:
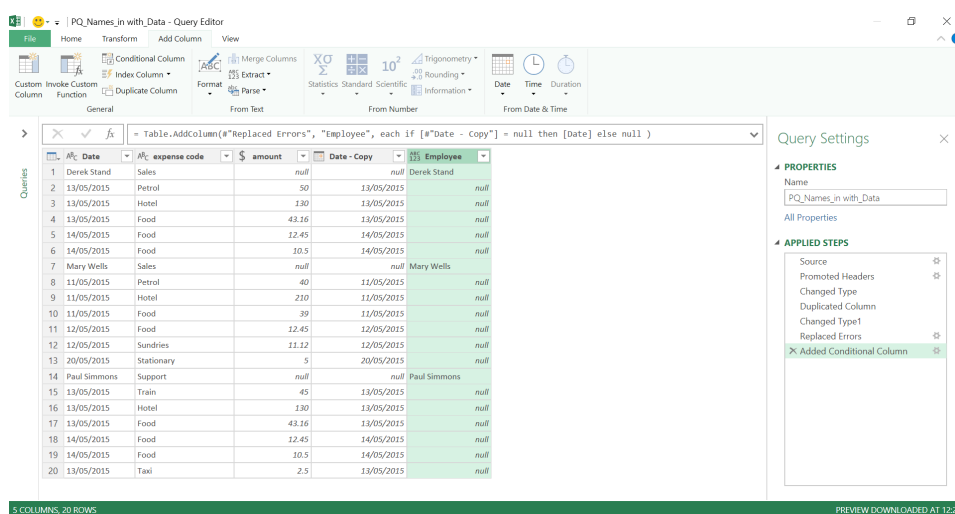
Since the value in our new column is either a name or null, we can populate the rest of the data. Right-click the new column and choose 'Fill' and then 'Down'.
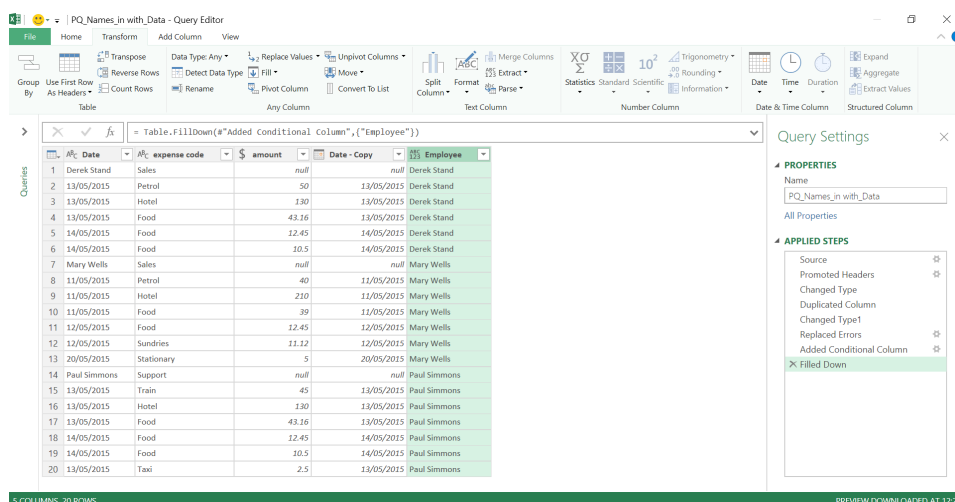
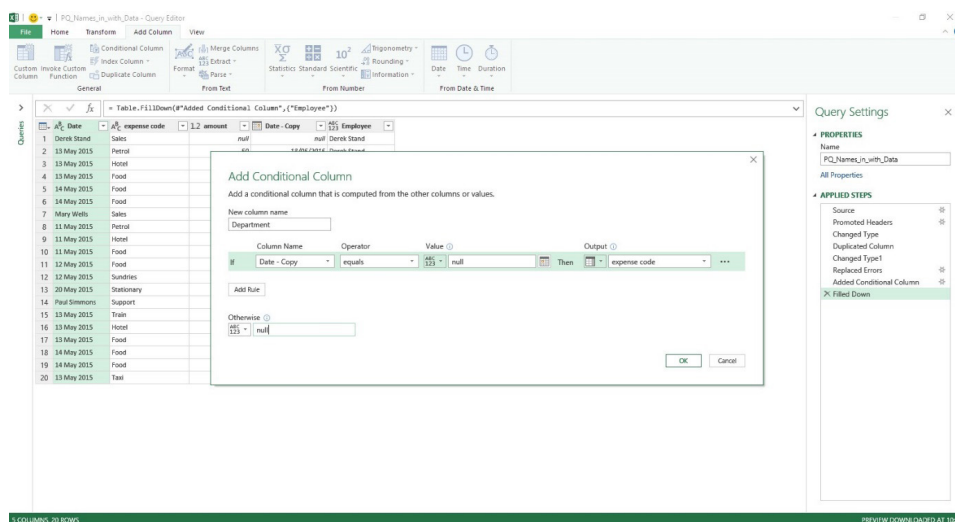We have our names; we can now create another Conditional Column to get the department information – this time our output is the ***expense code*** column:

Since the value in the new column is either a department name or null, we can populate the rest of the data. Right-click the new column and choose 'Fill' and then 'Down' as before:
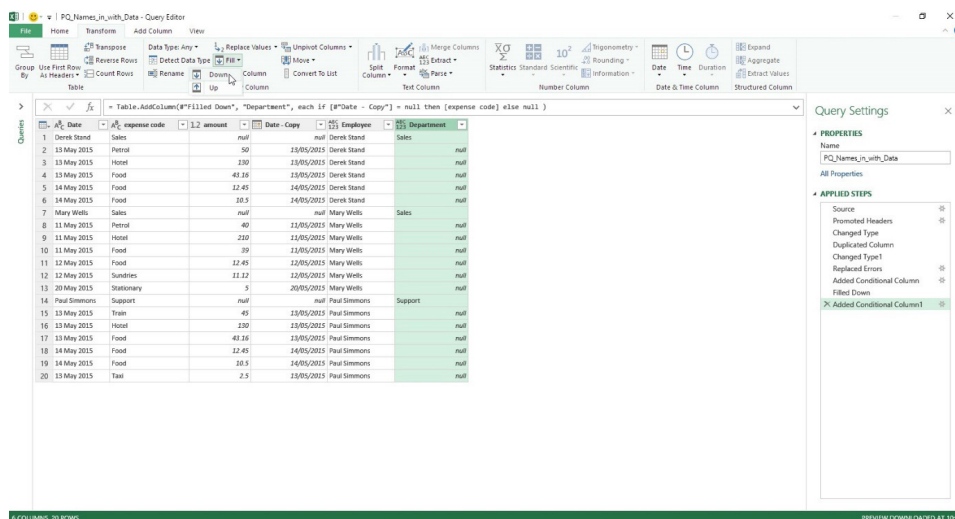
Now all we need is to get rid of our redundant columns and rows. There are a number of ways we could approach this, but since we still have nulls in ***Date – Copy***, we can use this column. Then, we use the filter icon next to the ***Date – Copy*** column to remove the null values.
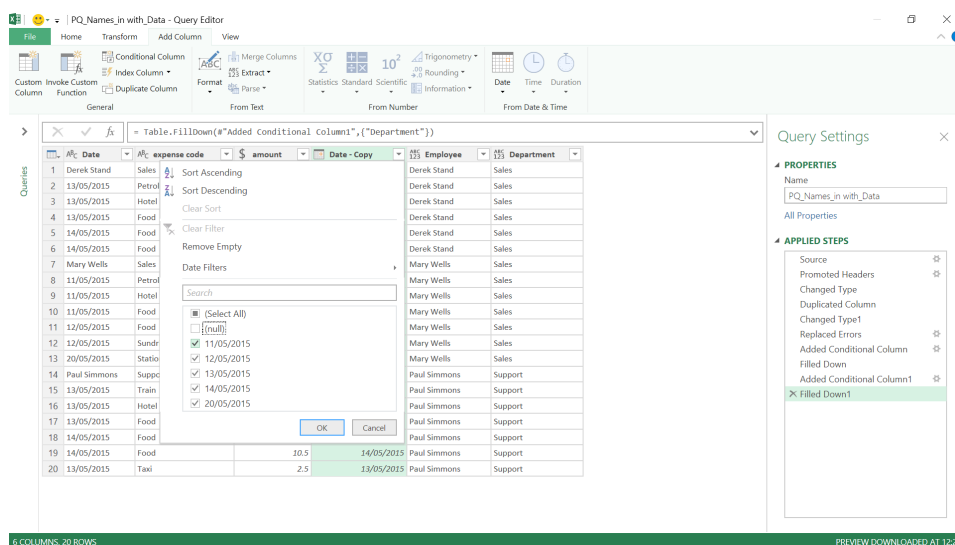
This gives us the rows we want.
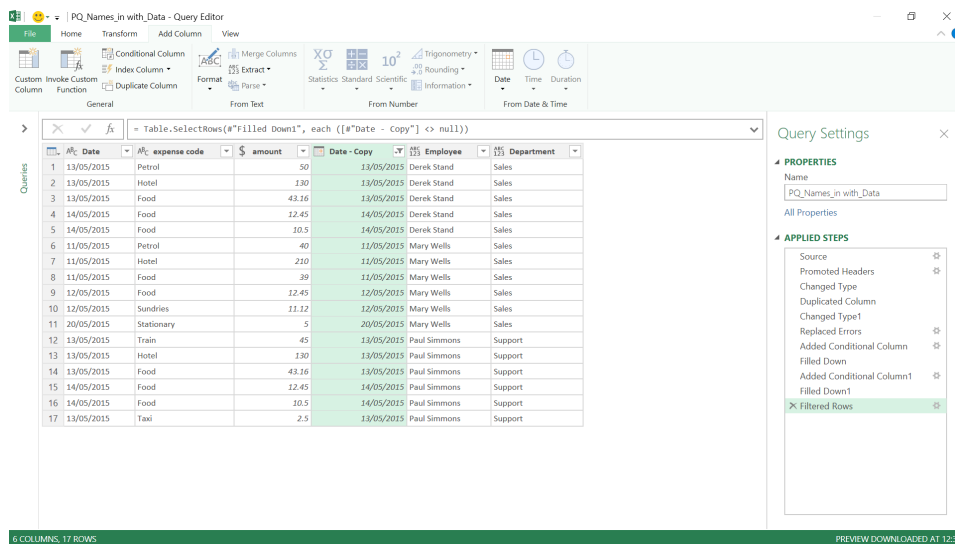
We delete **Date - Copy** and set the data types on the columns we want to keep. We are now ready to 'Close and Load' the data to an Excel worksheet.

By spotting the differences in data types in my original **Date** column, we have been able to pull out the data required and format it in a useful way.

More next month!

Latest Updates for Power BI Desktop

February's updates came out with a bang as the updates came out all guns blazing. There are some major updates for Q&A with the addition of auto-generated questions and the ability to ask Insights related questions, as well as numerous formatting and visual improvements. Here's the full list:
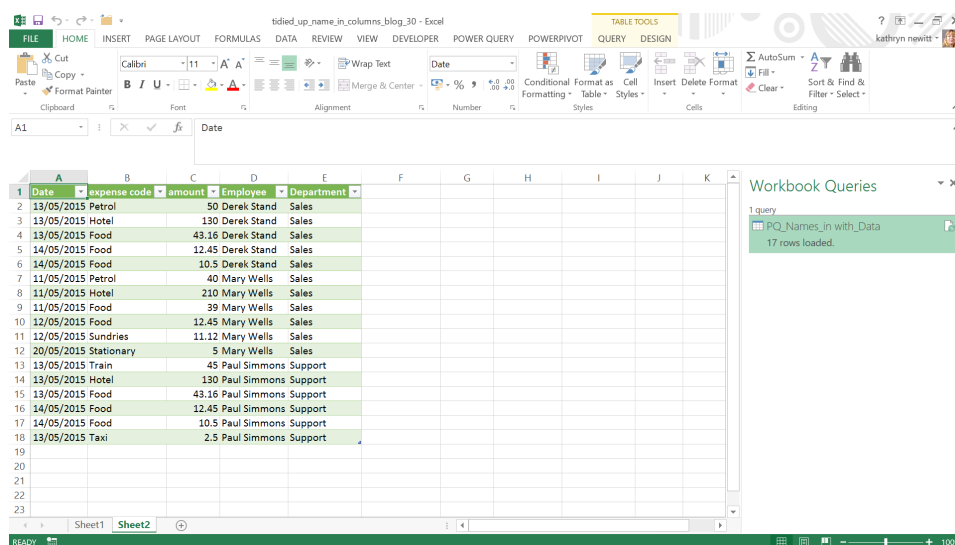
Reporting

- Updates to the new Filter pane (Preview)
    - o More Filter pane formatting
    - o Accessibility improvements for the new filtering experience
- Cross-highlight on a single point in line charts
- Word wrap on titles
- Update default visual interaction to cross-filter
- Rounded corners for visual borders

Analytics

- Key Influencers visual (Preview)
- Insights questions in Q&A
- Auto-generated suggested questions for Q&A explorer
- Improved Python and R script editor

Custom visuals

- Additional purchases for custom visuals
- Advanced Donut Visual
- Advanced Combo Visual
- Advanced TimeSeries Visual
- Advanced Network Visualization
- 3AG Systems – Line chart with absolute variance
- Summary Table
- KPImg
- Ultimate KPI Card
- Violin Plot

Data connectivity

- Microsoft Graph Security
- Guidanz' BI Connector for OBIEE
- MarkLogic
- Kronos Workforce Dimensions
- SurveyMonkey
- Qubole Presto
- Quick Base
- Sagra Emigo

Other

- Improved Live connect and Direct Query error messages.

Let's go through each new feature in turn.

More Filter pane formatting

Microsoft has added more formatting options to the updated filtering experience Preview released back in November. This update sees title text size, header text size, and font family added to the Filter pane card.
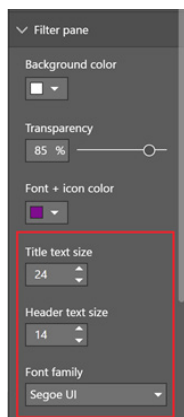
The Filter card in the Formatting pane also has new options: font + icon color, text size, font family and input box color *(sic)*.
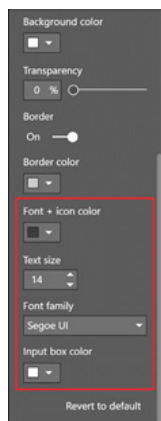
Accessibility improvements for the new filtering experience

You can now navigate through the new filter restatement experience in the visual header with a keyboard and screen reader. When the focus is on the filter icon in the visual header, you can use the 'Space' key to open the flyout and tab through the various filters applied.

From there you can the **ESC** key to close the filter restatement flyout and move focus back to the header. You can also navigate through the new Filter pane with the same keyboard and screen reader support you had in the old Filter pane.

Cross-highlight on a single point in line charts

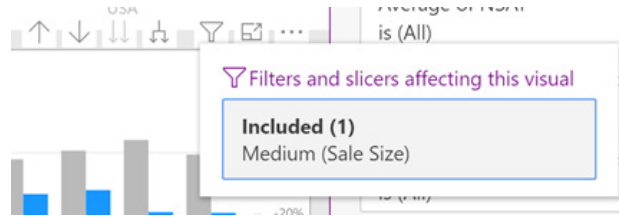You can now cross-highlight on a single point in the line chart to filter not only on the category but also the individual date too. When you click on the point, that individual point will show a marker indicating which point is the source of highlighting. If your visual is very dense, Power BI will select the closest point to your click to use as the source.
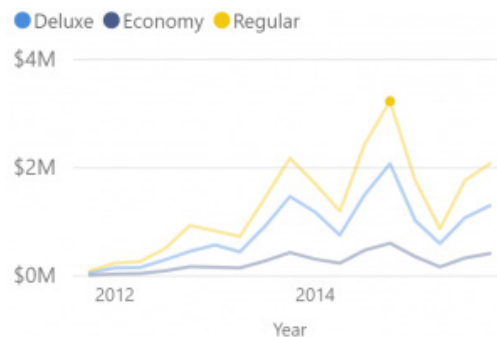
With this change, the line chart now behaves exactly the same as a stacked column chart in terms of interactivity. You can continue to use the legend when you filter just on the categorical value alone.

Word wrap on titles

There's a new formatting option for visual titles this month: word wrap. If you have titles that are currently being truncated, *e.g.*
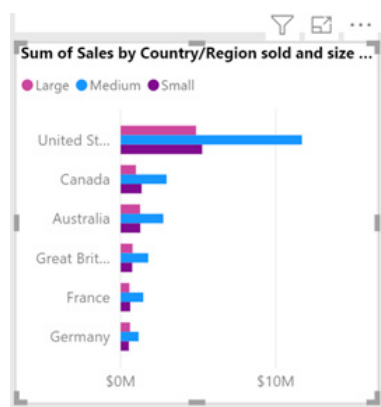
you can now turn on the word wrap option in the title card
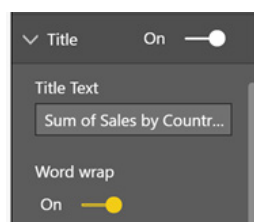
and your title will now show on two or more lines:
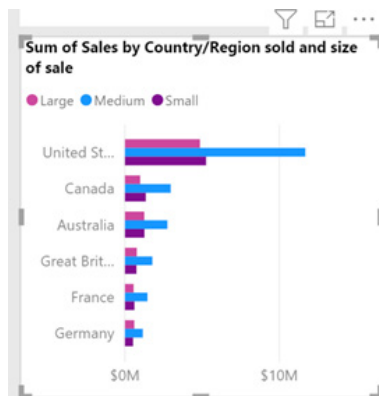
Update default visual interaction to cross-filter

If you want to quick change all your visuals to cross-filter each other instead of cross-highlight, you no longer need to manually change all your visuals one by one using the 'Edit visual interactions' feature. Now all you need to do is go to the 'Options' dialog and check the new option under **Current File -> Report Settings** and select the 'Change default visual interaction from cross highlighting to cross-filtering' checkbox:
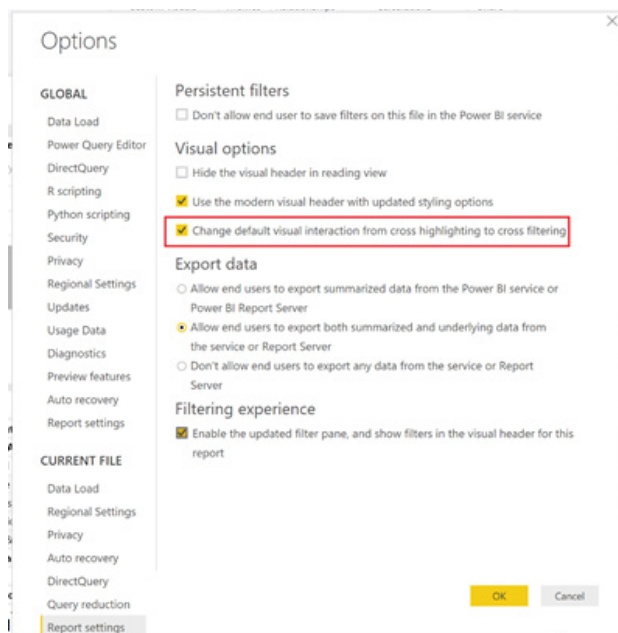
This will make the visual interaction cross-filter by default. You can still manually update visuals if you just a few to cross-highlight instead. This will also not change any visual interactions you have already set manually.

Rounded corners for visual borders

Another new formatting option this month is the ability to round the corners of your borders, a well-loved feature from Excel.
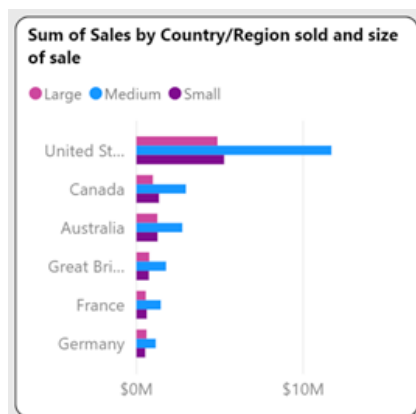
To use this option, go to the 'Border' card in the Formatting pane, and adjust the radius from zero (0) to the desired "roundness" (good word we have just made up there):
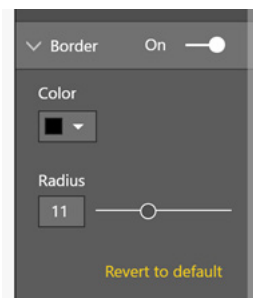
Key Influencers visual (Preview)

There's a Preview of the first AI Visualization this month, ladies and gentlemen, please welcome Key Influencers. AI Visualizations use machine learning behind the scenes in order to analyse your data and present identified insights. Once you pick a Key Performance Indicator (KPI) you wish to analyse, the Key Influencers visualization assesses what matters the most in driving your metric, as well as providing interesting segments for further investigation.

Once you turn on the Preview feature through the 'Options' dialog, you can start by adding the visual from the Visualizations pane:

From there, you can add your fields to the 'Analyze and Explain by' buckets in the field well. In the 'Analyze' bucket, you should put the column that you are interested in finding drivers for. The 'Explain by' bucket is for the different variables you want to analyse. For example, should you wish to know what drives Net Satisfaction from given data, you would place that in the 'Analyze' bucket, and then put several fields that may influence that, such as 'Manufacturer' and 'Color', in the 'Explain by' bucket.
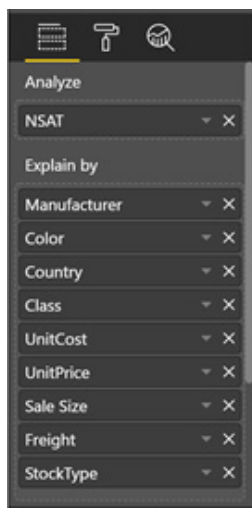

Once you add fields into both buckets, the visual will automatically analyse the data and show you the results. The visual will default to showing you the 'Key Influencers' view. Here, you may change the value you want to analyse and explore the various influencers. For example, let's say you selected 7 as the NSAT value you wish to analyse in the dropdown. Here, we can see that the biggest driver of NSATs of 7 is that the Manufacturer is 'Litware, Inc'.
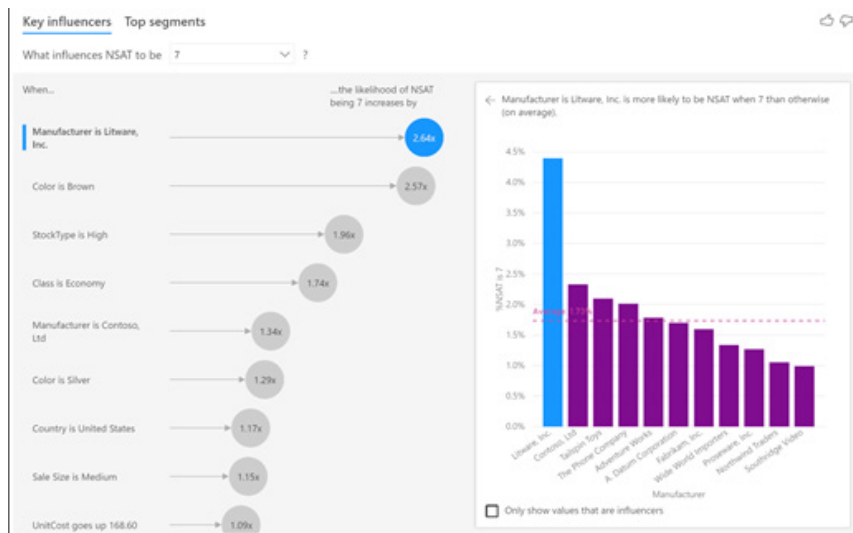
You can click through each influencer on the left side of the visual and see the results visualized on the right. You can also use the 'Only show values that are influencers' checkbox on the bottom right to filter the visual if the field you are looking at has a lot of categories that are not significant.
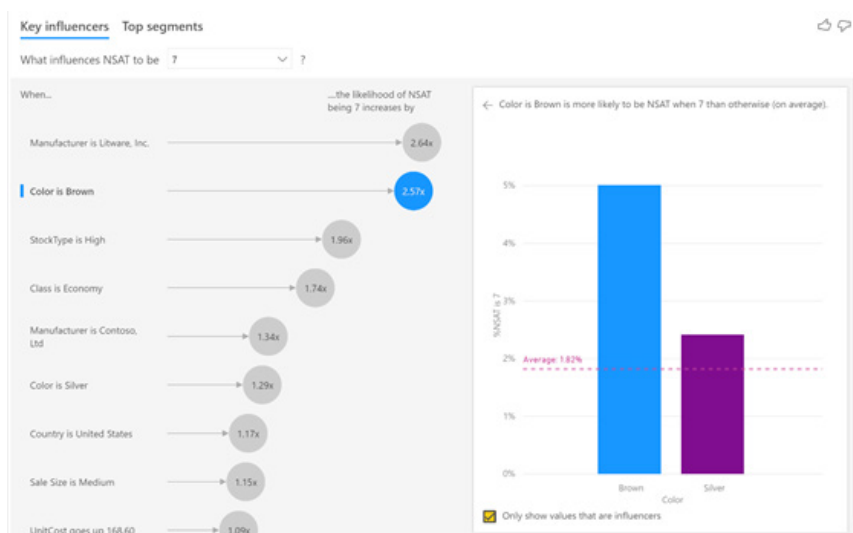
The second view of the visual is 'Top segments'. This view lets you see different segments of data points that fall into the topic you are analysing. In this case, it will show segments of products that are more likely to have a NSAT of 7. In this instance, there are two segments.
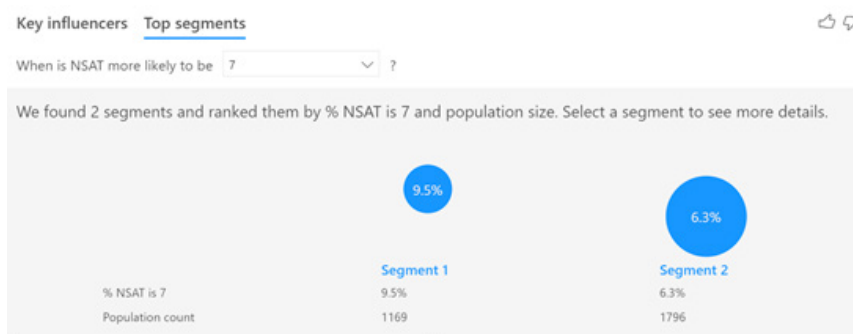
If you click on a segment, you can see the details of the segment including the number of data points in that segment, what percentage of the segment has the value of interest and key influencers for that segment. In the case of Segment 1, it has been identified that it is made up of Economy products coming from Litware, Inc. It turns out 9.5% of them have an NSAT of 7 which is eight percentage points higher than the average product in the dataset. Furthermore, 3% of the total data has those characteristics.
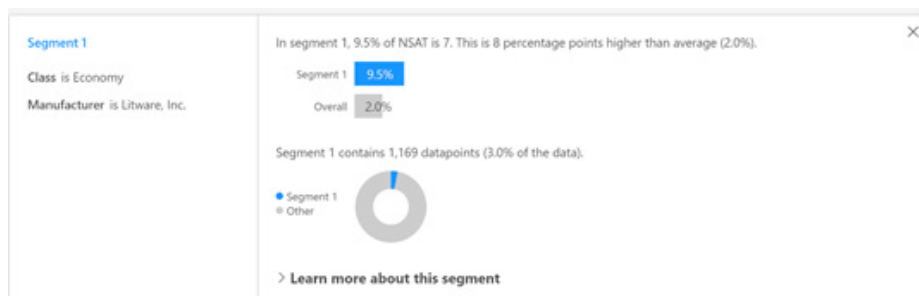
You can also use the 'Learn more about this segment' option to run insights over the segment.
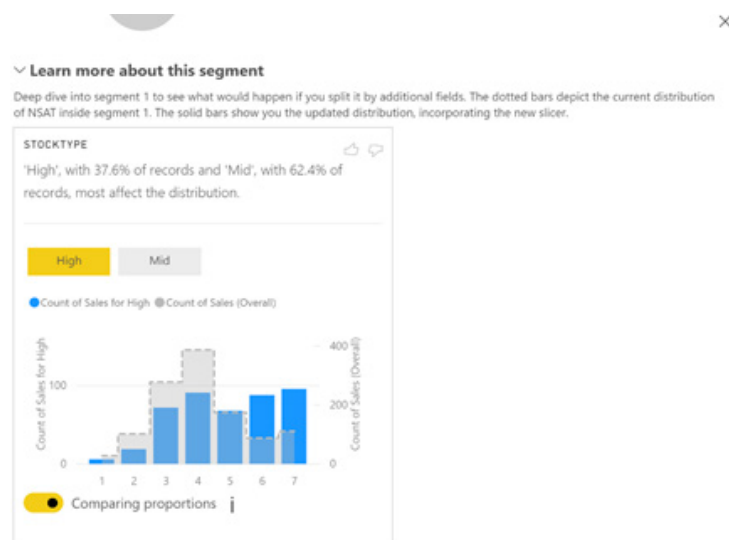
When this visual is part of your report, your end users will be able to play with all these interactions but won't be able to change the fields used for the analysis. There are limitations, but it's a fantastic start.

We are starting a blog series on this feature and you can read the first part and our first thoughts at www.sumproduct.com/blog/article/power-bi-tips/power-bi-key-influencers.

Insights questions in Q&A

You can now ask Q&A Insights related questions, such as "Explain the change of sales between 2012 and 2013" and get the same kind of insights you get through the 'Analyze' option in the context menu. Brilliant.
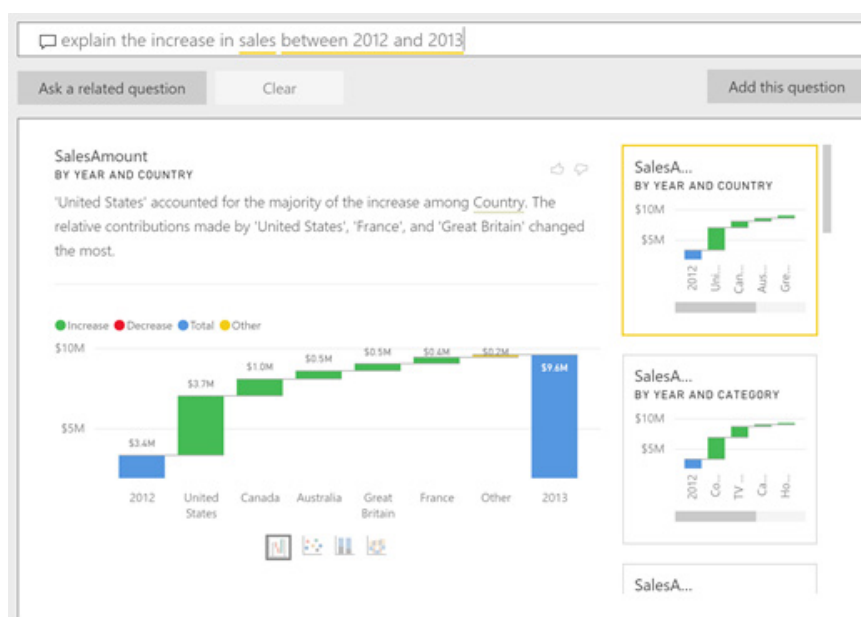
Auto-generated suggested questions for Q&A explorer

The Q&A Explorer is now auto-populated with suggested questions. This is really useful if you want some ideas of the types of questions to ask Q&A or you don't want to add specific suggested questions for your report consumers. Of course, you can still add your own suggested questions for your users should you wish.
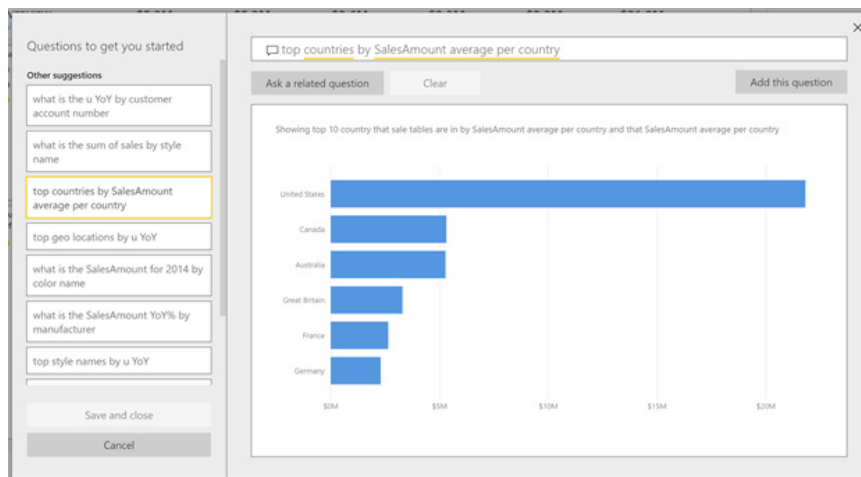
Improved Python and R script editor

The Python and R script editors now use the same script editor that the DAX formula bar uses. This means you have the same features such as intellisense, line numbers and syntax highlighting.

Additional purchases for custom visuals

Until recently, AppSource offered only Power BI visuals that had no cost associated with them. In order to promote visuals containing advanced features and support a funding model for their development, Microsoft has announced that Power BI custom visuals that may be purchased are now available in AppSource. Do I detect a sense of underwhelm from our clients...?

Any visual with "additional purchases" is marked with a price tag of 'Additional purchase may be required.' These custom visuals are still free to download, but they offer optional in-app purchases to unlock advanced features (like being able to use them).

To try and provide a more balanced assessment, it should be noted that the first set of in-app purchase ("IAP") visuals were created by ZoomCharts and are already available for download from AppSource today. They've built out these visuals based on feedback from users after launching ZoomCharts for Microsoft Power BI last year. The following are such examples, available to download for free with customisation options you may test out on a free 30-day trial. And no, we aren't on commission...

Advanced Donut Visual

ZoomCharts' Advanced Donut chart has been designed for exploring multi-level data in depth and across your particular level of interest. This visual provides navigation that allows you to interact with the chart on both desktop and mobile.

Some features of this visual include:

- **Adjustable chart types:** Select between pie, donut and gauge chart types for better data representation
- **Automatic "others" grouping:** Define the number of slices you need, and the rest will be grouped into "others"
- **Multi-level drill-down:** Touch or click the slice of your interest and drill down to the next level to easily navigate your data
- **Up to nine drill-down levels:** Add up to nine different categories and drill down for an in-depth analysis within a single chart
- **Customisable labels:** Configure inside / outside label content and appearance of the labels to suit your needs
- **Legend configuration:** Customise the appearance and placement of the legend according to your needs and preferences.

Advanced Combo Visual

ZoomCharts Advanced Combo Visual is designed for in-depth exploration of complex category-based data. This visual has navigation that again allows you to interact with the chart on any device.

Some features of this visual include:

- **Adjustable chart types:** Choose from column, line or area and customise the appearance for each individual series
- **Zoom-in / Zoom-out:** Dynamically change size of the columns by dragging chart to any side or using pinch gesture
- **Up to 12 different series:** Combine up to 12 series with unlimited categories to fully explore your data with a single chart
- **Customisable:** Customise types, colours, labels, fonts and other aspects of the chart
- **Legend configuration:** Customise placement and appearance of the legend
- **Stacks and clusters:** As well as the fact that you can adjust the chart type for any of the 12 series, you can also stack and cluster them by setting additional properties.

Advanced TimeSeries Visual

You can explore time-based data with the ZoomCharts Advanced TimeSeries Visual. This visual has many features, including:

- **Adjustable chart types:** Choose from column, line or area and customise the appearance for each individual series
- **Time unit aggregation:** When data is aggregated on the visual's side, you have multiple options as to how to aggregate data at larger units and display it on the chart
- **Up to 12 different series:** Combine up to 12 series with unlimited categories on a single chart
- **Time unit drill-down / drill-up:** Interact with any part of the chart to drill-down to the desired time span
- **Predefined time spans:** Interact with any part of the chart to drill-down to desired time span
- **Customisable:** Customise chart types, colours, labels, fonts and other aspects of the chart for each series independently
- **Legend configuration:** Customise placement and appearance of the legend.

Advanced Network Visualization

ZoomCharts' Advanced Network Visual enables you to explore and filter your data using a network layout. This visual has many features, including:

- **Customisable vertices / nodes:** Customise colour, radius, labels, size, image and shape of your vertices
- **Customisable edges / links:** Edges can be customised to have the colour, type, decorations, thickness and data-driven labels
- **Images support:** You can emphasis the vertices accordingly in your report
- **Up to nine categories:** Add up to nine categories of vertices
- **Customisable labels:** Configure inside and outside label content and appearance
- **Legend configuration:** Legend can be shown or hidden and positioned on top or bottom, right or left.

3AG Systems – Line chart with absolute variance

The Line Chart with Absolute Variance custom visual by 3AG Systems takes two series and automatically calculates the absolute variance between them. It plots the two series in a line chart and shows absolute variance on top of the line for each category.

This visual has a lot of customisation options including inversing the colours for red and green, adjusting values to thousands (K), millions (M), and billions (B), drilling-down on hierarchical data, and more.

You can use this visual to compare "Actual vs. Planned", "Actual vs. Previous Year", "Forecast vs. Planned", and "Forecast vs. Previous Year".

Actual vs. Previous Year

Forecast vs. Previous Year
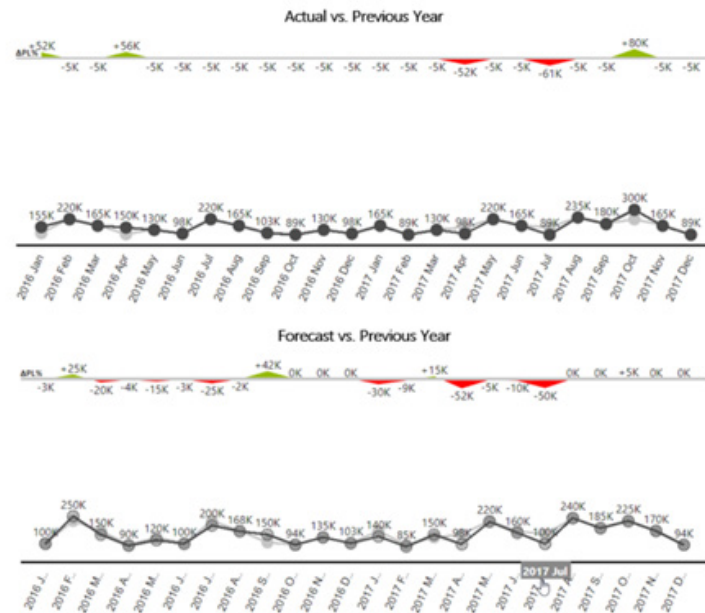
Summary Table

The Summary Table custom visual by Fredrik Hedenström is useful for creating income statement reports. It lets you create a summary table where each row can have custom styling:

	Amount	Budget
Operating revenues	$61,949	$62,257
Other Revenues	$5,990	$6,004
SUM OF REVENUES	$67,939	$68,261
Operating expenses	$6,360	$6,515
Direct expenses	$1,750	$1,758
Other expenses	$1,127	$1,117
SUM OF COSTS	$9,237	$9,390
Interest income	$2,972	$2,961
Interest expenses	$1,659	$1,691
SUM OF FINANCIAL INVESTMENTS	$1,313	$1,270
BUSINESS RESULTS	$60,015	$60,141

KPImg

The KPImg visual by Aritz Francoy shows a Key Performance Indicator (KPI) as a percentage from a value over a target. You can also visualise this with a time series, which will be shown as an area chart under the percentage. The visual also lets you set an image to show if you are over or under a target percentage.

NET SALES	GROSS PROFIT	CORE SAR	△ Net Sales CE PY%	EBIT in % of NS 3rd
3,945M	67M	-24M	45%	282%
vs. Tar:+1,183M / +43%	vs. Tar:-404M / -86%	vs. Tar:+143M / +86%	vs. Tar:-3M / -100%	vs. Tar:+0M / +43%
vs. PY:+3,550M / +900%	vs. PY:-0M / -0%	vs. PY:-0M / -0%	vs. PY:+0M / +900%	vs. PY:+0M / +900%

CORE OIE	CORE EBIT	EBIT Margin in %	GP aft. Var. NS 3rd %	CORE SAR in % of NS
7M	394M	211%	345%	43%
vs. Tar:-42M / -86%	vs. Tar:-2,367M / -86%	vs. Tar:+0M / +43%	vs. Tar:+0M / +43%	vs. Tar:+0M / +43%
vs. PY:0M / 0%	vs. PY:-0M / -0%	vs. PY:+0M / +900%	vs. PY:+0M / +900%	vs. PY:+0M / +900%

Depreciation & Amortization	CORE EBITDA	OIE in % of NS 3rd	RONOA in %	EBITDA % of NS 3rd
6M	36M	51%	244%	318%
vs. Tar:-36M / -86%	vs. Tar:-216M / -86%	vs. Tar:+0M / +43%	vs. Tar:+0M / +43%	vs. Tar:+0M / +43%
vs. PY:0M / -0%	vs. PY:0M / 0%	vs. PY:+0M / +900%	vs. PY:+0M / +900%	vs. PY:+0M / +900%

Violin Plot

The violin plot custom visual by Daniel Marsh-Patrick combines a box plot and a kernel density plot into one visual. A box plot lets you see basic distribution information about your data, such as median, mean, range and quartiles, but doesn't show you how your data looks throughout its range. If you have a multimodal distribution (multiple peaks), or some confusion as to where things are clustered, then it's not easy to figure this out.

By adding a kernel density plot, you can see the variations in your data across its distribution. It works like a histogram, but uses kernel smoothing to provide a smoother curve where noise might otherwise be present.

Features include:

- Split and color by categories
- Four different kernels
- Sampling resolution
- Estimated/manual KDE bandwidth
- Customizable box plot.
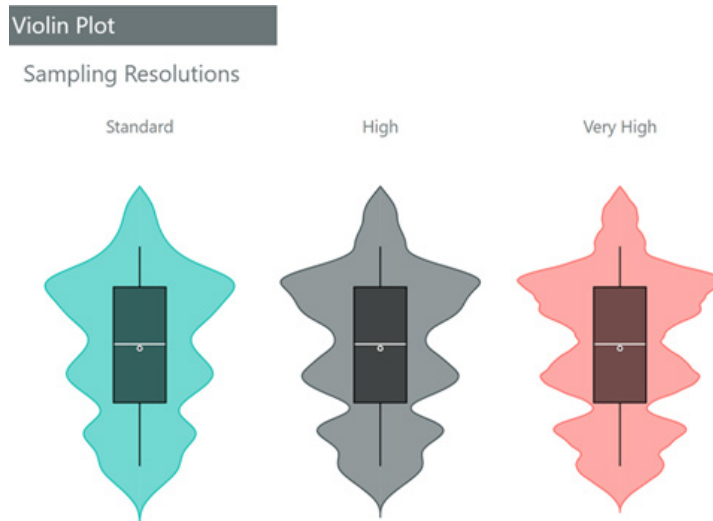
Microsoft Graph Security

The Microsoft Graph Security connector for Power BI, along with a sample dashboard and template, is now available to enable you to bring insights into your reporting solutions and get a holistic view of security and risk across your enterprise. The connector reduces the time and resources required to integrate multiple data sources, simplifying the creation of reports across multiple security solutions.
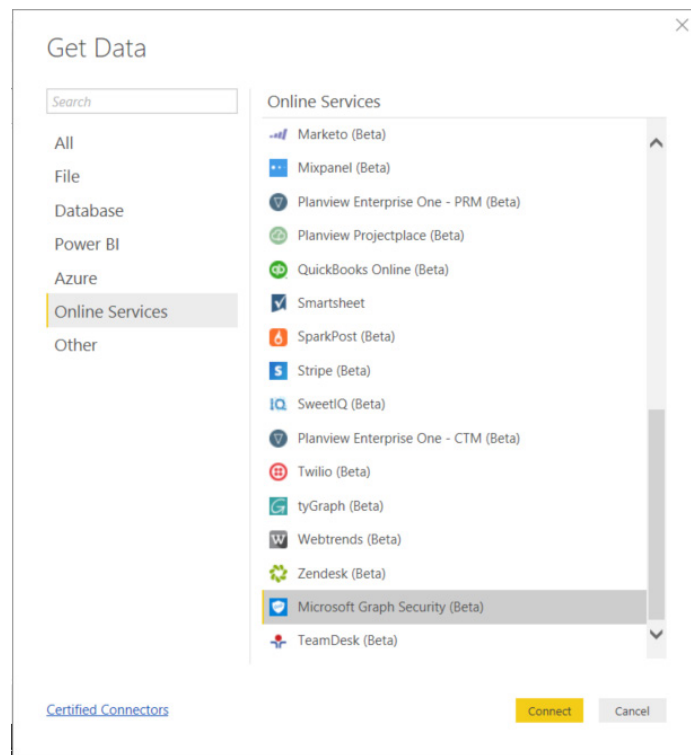
Guidanz' BI Connector for OBIEE

The BI Connector for OBIEE helps organisations to transition from traditional BI platforms to modern visualisation tools. With BI Connector, users can shorten the time to analysis, increase ROI from BI investments, and enhance data governance. Guidanz' BI Connector allows for the staging of data from OBIEE for visualisation in Power BI.
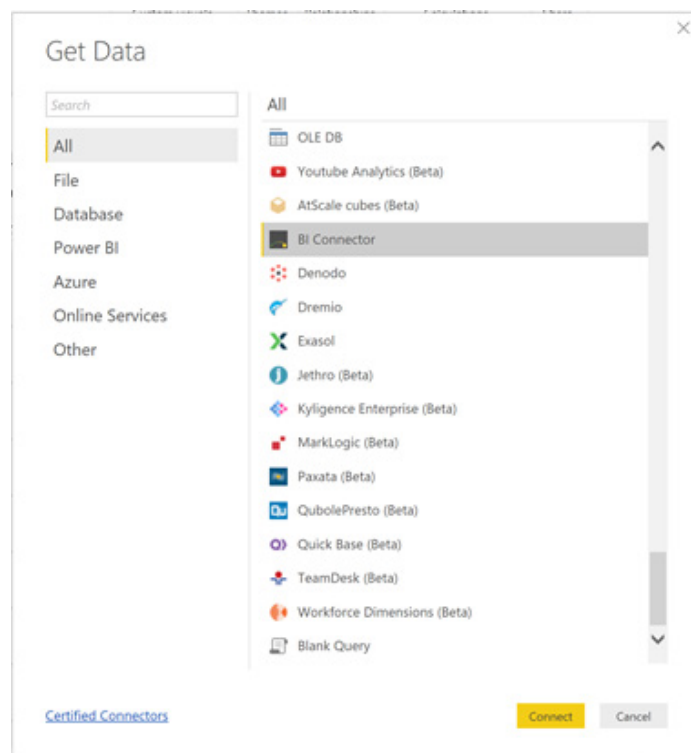
MarkLogic

MarkLogic is a database designed for NoSQL speed and scale, without sacrificing the enterprise features required to run mission-critical, operational applications. Using a multi-model approach, MarkLogic provides flexibility to integrate and store all of your data, and then view that data as documents, as a graph, or as relational data.
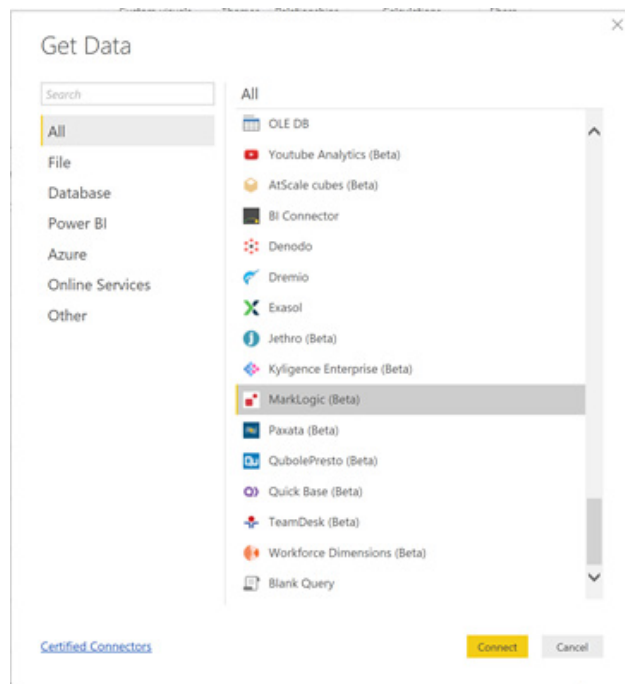
Kronos Workforce Dimensions

Powered by the Kronos D5™ platform, Workforce Dimensions™ offers employee experience and detailed levels of operational insight into managing your workforce.

SurveyMonkey

Every day, 20 million questions are answered using the SurveyMonkey platform. SurveyMonkey helps organisations measure, benchmark and act on opinions by using their People Powered Data survey platform.

Qubole Presto

Qubole offers Presto-as-a-service on Microsoft Azure and AWS to handle ad hoc queries across petabytes of data. Presto is a distributed ANSI SQL engine used for processing big data *ad hoc* queries at large scale and speed. Qubole's Presto connector for Power BI allows users to run fast interactive analytics on federated data sources. This can include data in a data lake, relational databases such as SQL server, MySQL, and Postgres, NoSQL databases such as Cosmos DB, MongoDB and Cassandra, and real-time data streaming applications such as Kafka.

Using Qubole's connector, Power BI users can now select tables across any of these catalogues and create and execute queries across these different data sources for reporting, data exploration,and *ad hoc* analytics.
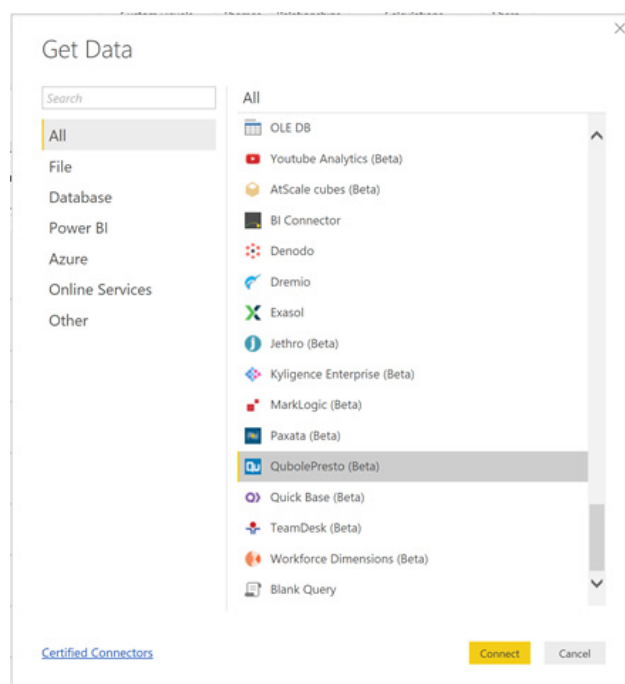
Quick Base

Quick Base allows you to create solutions to your business challenges by making it easy to manage data and automate processes. With Quick Base, custom applications can be built quickly.
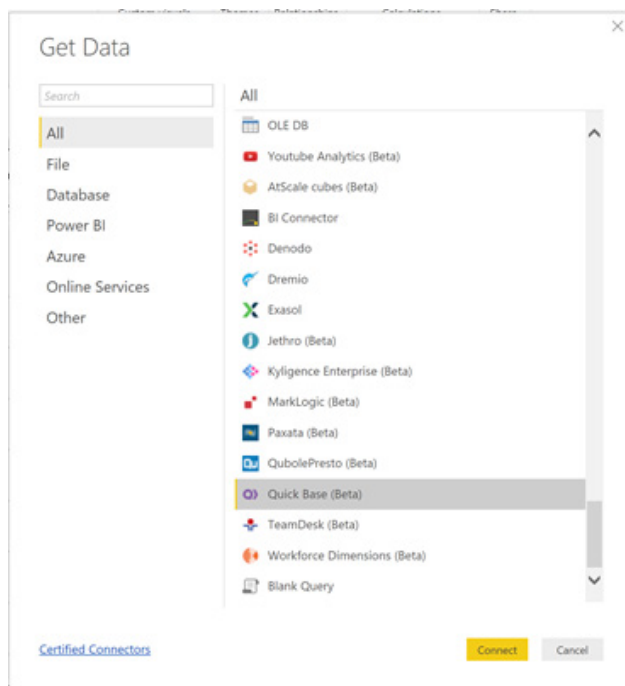
Sagra Emigo

Emigo supports sales organisations and field teams management. It automates the daily activities of field professionals. Inventory data, orders, answers from surveys or all sorts of other digital forms are available for managers and other teams within organisation, through the Biqsens Power BI app as well as through Emigo BI Connector for Power BI Desktop.

Improved Live connect and Direct Query error messages.

There's now a friendlier error message when there is an error with Live connect or Direct Query connection when loading a model. The friendly message is followed by a more technical message from the provider. This message may help you debug the issue easier on your side, and if you contact support, the message details will help the support team find the issue quicker as well.

For example, here's a sample message if there's a network issue and you fail to connect to your Direct Query server:
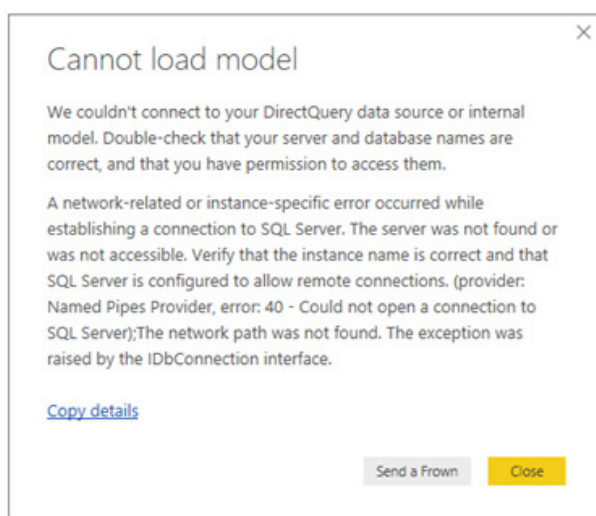
More next month no doubt.

Power BI Service and Mobile Latest Updates

Of course, that's not all. Here's the latest updates for Power BI Service and Mobile:

- Secure-embed for Power BI reports
- Dataflows updates
- Personal bookmarks
- Custom visual monetisation
- On-premises data gateway management
- Power BI Report Server update
- E-mail subscription time-based scheduling
- Report enhancements for mobile
- Presentation mode for Power BI Windows app.

Let's take a look at each in turn.

Secure-embed for Power BI reports

It's getting easier to embed Power BI reports to internal sites within your organisation. With the new 'Embed' option found in the 'File' menu for each report, you can securely embed the content within internal portals, whether it's hosted on-premises or cloud, without needing any advanced coding skills. The best part is that reports embedded in this way respect all item permissions set in Power BI and data security through row-level security (RLS). The 'Embed' option also supports basic URL Filters and URL settings, which allow you to build experiences into your portals using rudimentary HTML and JavaScript skills.
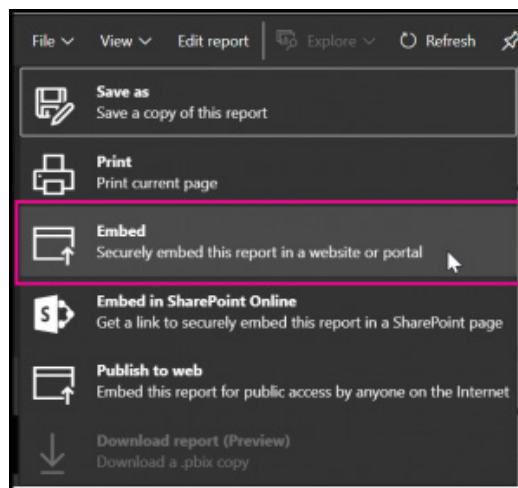
Dataflows updates

In November of last year, Microsoft announced the public Preview of Power BI dataflows. Given community feedback, Microsoft has added some upgrades:

- **Data connectors:** User Interface support has been added for new connectors such as MySQL, Teradata, Azure Data Explorer and many more have been improved
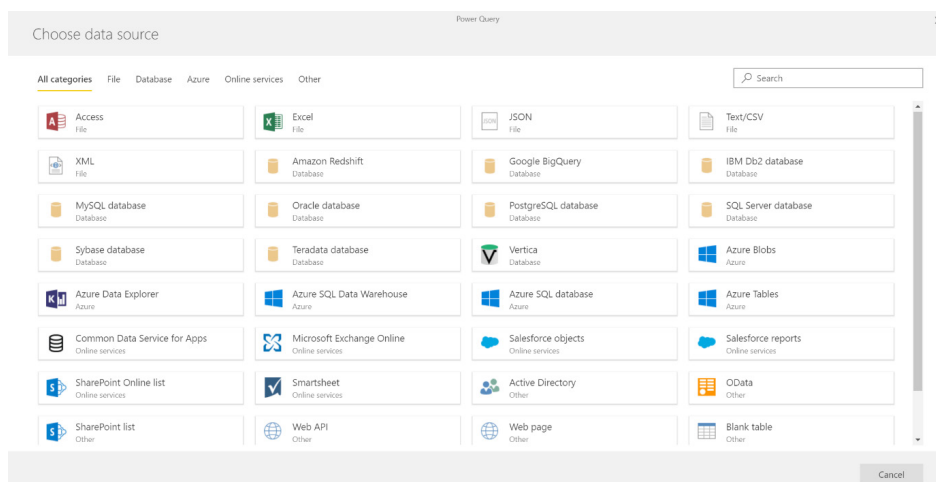
- **Native SQL queries:** Support has been added to run native SQL database queries within Power BI. saving you considerable time if you are already using them elsewhere

- **Power Query Online transformation support:** Support has also been added for all transformations in the Power Query Online engine, which is particularly useful if you are already using Power BI Desktop.

Personal bookmarks

Microsoft has brought the power of bookmarking to report consumers in the service. With the release of personal bookmarks, the ability has been added for end users to create their own set of bookmarks for the reports they have access to. Personal bookmarks allow users to capture various states of a report page (including filters, slicers, and state of visuals), give them friendly names, and later return to each state with just one click. In addition, users can also mark any bookmark as their default, so their favourite bookmark gets applied each time they open the report.

Custom visual monetisation

Customer visuals are core to building effective and intuitive storytelling in Power BI. The community of developers has grown substantially since its launch in 2015 with hundreds of visuals available for download. In January, Microsoft announced the option for developers to monetise on the custom visuals they have built with in-app purchases. A visual with additional purchases is similar to any other in-app purchase add-ins (IAP) in the AppSource marked with a price tag of 'Additional purchase may be required.' IAP custom visuals can be downloaded for free by any user, however advanced features can be purchased for a cost. We've already mentioned these in the Power BI Desktop update *(please see above)*.

On-premises data gateway management

To help you gain visibility and manage all the on-premises data gateways within your organisation, a public Preview of the Data Gateway management feature has been launched by Microsoft in the Power Platform admin portal.
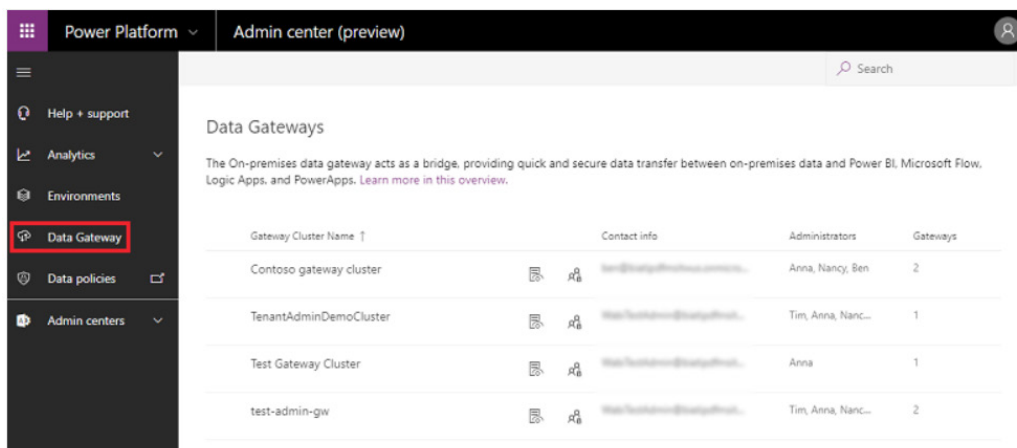
Power BI Report Server update

The Power BI Report Server continues to be updated. The latest features include copy and pasting between pbix files, expand and collapse on matrix row header, and row-level security support.

E-mail subscription time-based scheduling

To help users stay on top of their data, the e-mail subscriptions feature in Power BI has been updated by introducing time-based scheduling for reports and dashboards. With this update, users can now setup a subscription to run daily, weekly, or if you'd prefer, only on certain days of the week. This new capability is available on all Power BI dashboards and reports, in both Pro and Premium subscriptions.

Report enhancements for mobile

For Power BI mobile apps, the following new capabilities have been incorporated:

- Filter is available for landscape reports
- Modern visual header support in mobile reports
- Portrait report layout in tablets (iOS and Android)
- Support for report query string.

Presentation mode for Power BI Windows app

To help you leverage Power BI dashboard and reports when presenting, collaborating, and having day-to-day discussions at work, Microsoft has added a new presentation mode capability in the Windows 10 app. Now, all you need to do is tap the full screen button in the menu bar of the app and this will take you into a view where you only have your visuals and data.

That's it until the April newsletter.

The A to Z of Excel Functions: DAVERAGE

Just the one this month, as it's quite a long one...

This function averages the values 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 **DAVERAGE** function employs the following syntax to operate:

<div align="center">DAVERAGE(database, field, criteria)</div>

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

- **database:** this is 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
- **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 our example below:

	A	B	C	D	E
1	**Tree**	**Height**			
2	=Apple	>3			
3					
4	**Tree**	**Height**	**Height**		
5	Apple	>3	<=5		
6					
7	**Tree**	**Height**	**Age**	**Yield**	**Profit**
8	Apple	6	20	14	105
9	Apple	4	12	10	96
10	Cherry	4.2	14	9	105
11	Apple	4.7	15	10	75
12	Pear	3	8	8	16.8
13	Apple	2.7	9	6	45
14					
15	**Formula**	**Description**	**Result**		
16	*=DAVERAGE(A7:E13,"Yield",A1:B2)*	The average yield of apple trees over 3 metres in height, but less than or equal to 5 metres.	11		
17	*=DAVERAGE(A7:E13,"Yield",A4:C5)*	The average yield of apple trees over 3 metres in height, but less than or equal to 5 metres.	10		
18	*=DAVERAGE(A7:E13,3,A7:E13)*	The average age of all trees recorded.	13		
19					

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:

=''= entry ''

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).

More Excel Functions next month...

	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				

Upcoming SumProduct Training Courses

Location	Course	Date	Duration
Melbourne	Power Pivot, Power Query and Power BI	13 - 15 Mar 2019	3 Days
Sydney	Excel Tips & Tricks	1 Apr 2019	1 Day
Sydney	Financial Modelling	2 - 3 Apr 2019	2 Days
Sydney	Power Pivot, Power Query and Power BI	8 - 10 Apr 2019	3 Days
London	Financial Modelling	1 - 2 May 2019	2 Days
Auckland	Financial Modelling	6 - 7 May 2019	2 Days
Wellington	Financial Modelling	9 - 10 May 2019	2 Days
Melbourne	Excel Tips & Tricks	21 May 2019	1 Day
Melbourne	Financial Modelling	22 - 23 May 2019	2 Days
Melbourne	Power Pivot, Power Query and Power BI	28 - 30 May 2019	3 Days
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

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 continue going through the function keys, this time with the **ALT + SHIFT** combination:

Keystroke	What it does
ALT + SHIFT + F1	Insert new worksheet
ALT + SHIFT + F2	Save
ALT + SHIFT + F4	Close application
ALT + SHIFT + F10	Show On-Object User Interface (OOUI)
ALT + SHIFT + F11	Show Script Editor

There are over 540 keyboard shortcuts in Excel. For a comprehensive list, please download our Excel file a 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.