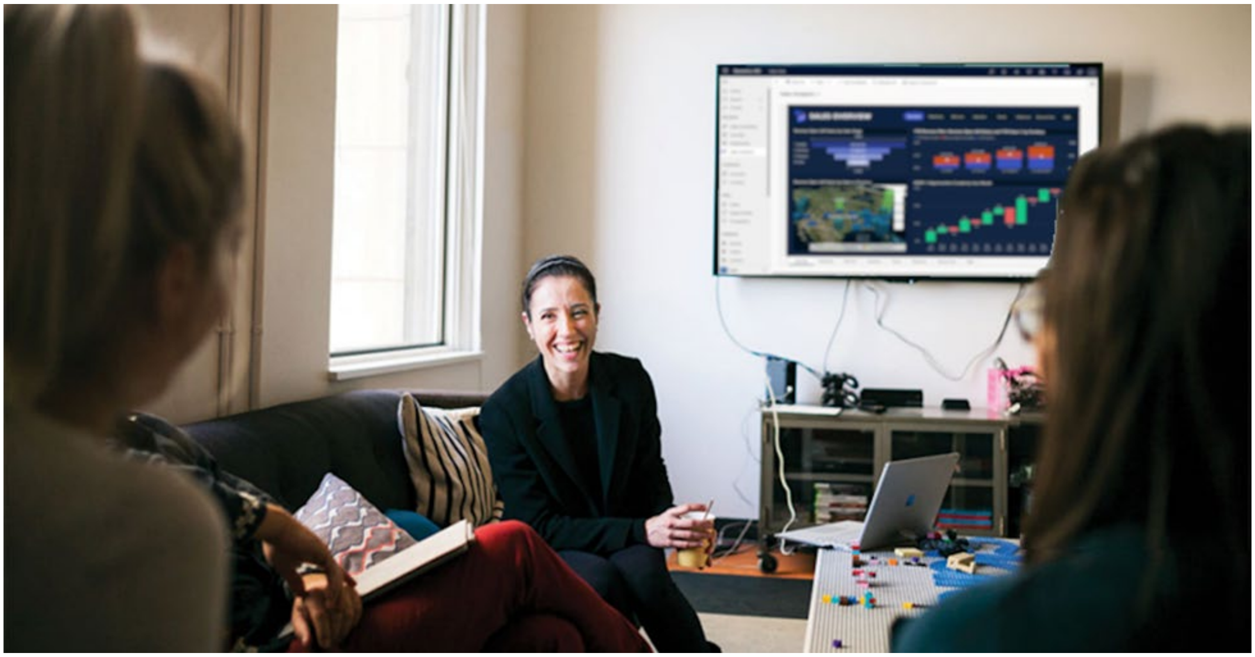
Dynamics 365
Technology Specialist Environment



Analytics & Reporting Demo – Dataverse TDS Endpoint

Authored by:

Scott Sewell
Principal Program Manager
Fabric CAT

Version 4

Contents

Introduction

These assets are designed to help demonstrate the power of using Power BI analytics and reporting to expose the value of the data in a Dynamics 365 CE implementation. The sample data is sourced from Dataverse via the TDS endpoint.

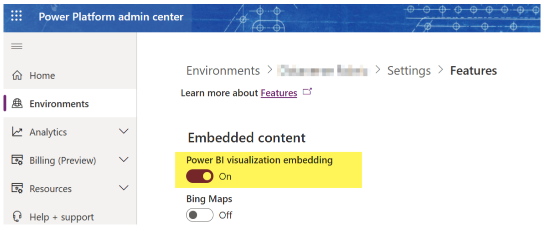
This approach is appropriate for relatively small datasets – returning less than 50,00 opportunities + customers. Larger datasets or more complex reporting is best handled through Fabric.

NOTE: There is a parallel version of this document for the Excel-based version of these reports. – You may still see 'excel' in some screenshots, but whenever a screenshot in this document shows a report title containing "Excel", assume it should say "TDS" instead.

Getting your environment ready

Task 1: Enable the Environment to allow Power BI visualizations

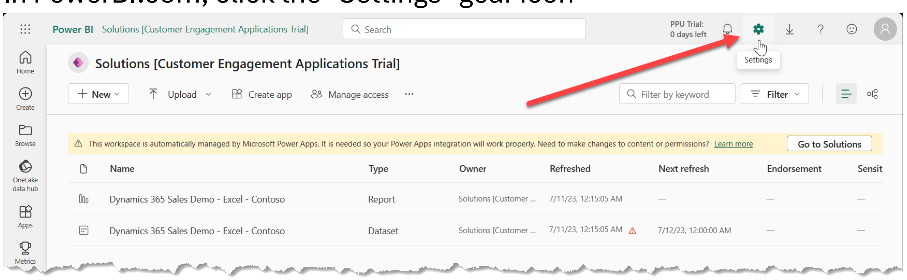
In Power Platform Admin Center navigate to “Settings/Product/Features.” Under “Embedded content”, ensure that the “Power BI visualization embedding” option is set to “On.” (Click Save)



Task 2: Enable Azure Maps and Filled Visual Maps in Power BI

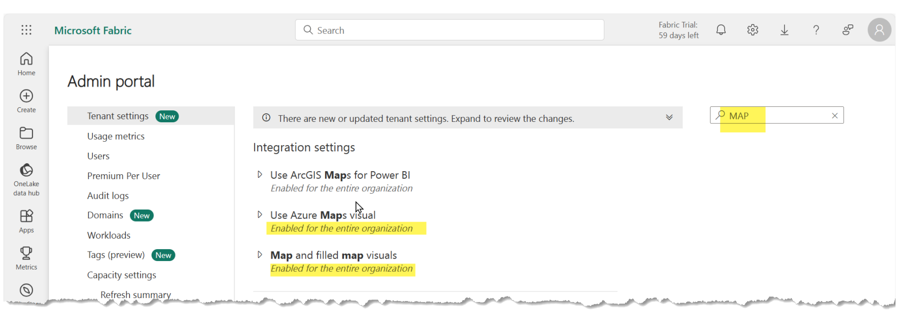
For any new environment you may need to enable the Azure Maps and “Maps and Filled Visuals” toggle switch. This is a tenant-wide setting, so you only need to set it once per organization.

a. In PowerBI.com, click the “Settings” gear icon

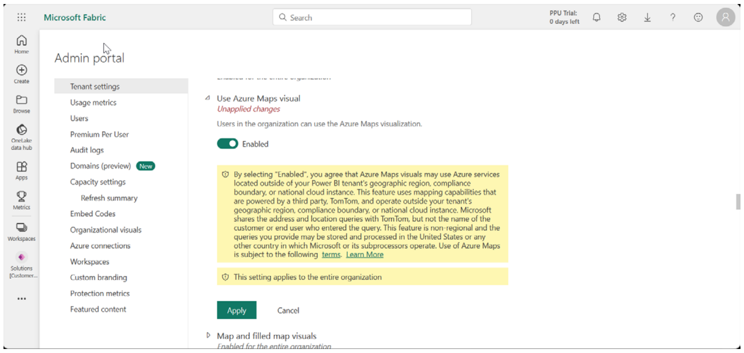


b. Under “Governance and Insights” Select the “Admin portal”

c. On the right, search for “Map”
Scroll down the list of Tenant settings until you find the “Use Azure Maps visual” setting (in the “Integration settings” section.) – You’ll want to ensure it’s ‘Enabled’ for both.

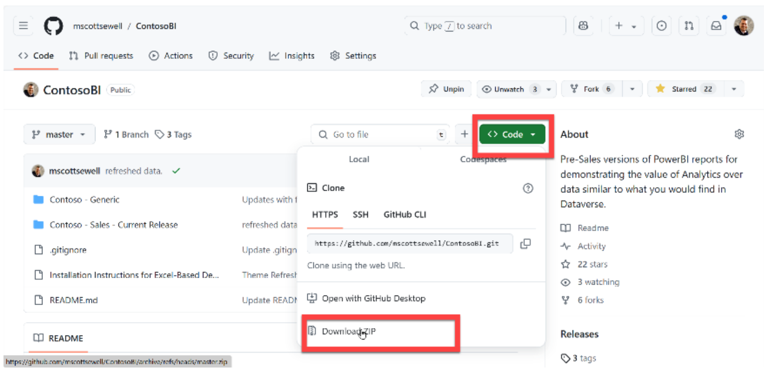


d. If needed, set expand an option and set the value to "Enabled" and click "Apply"

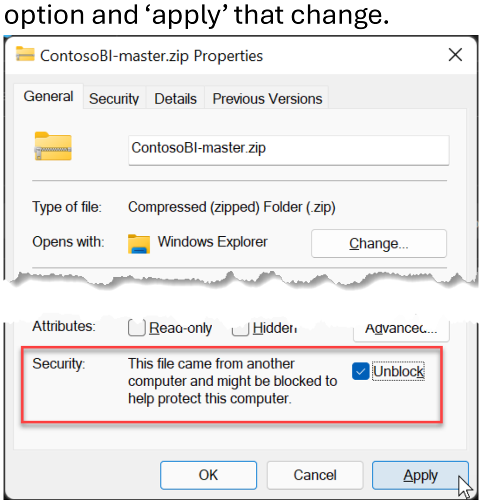


Task 3: Create Local Copies of the Reports and Excel File

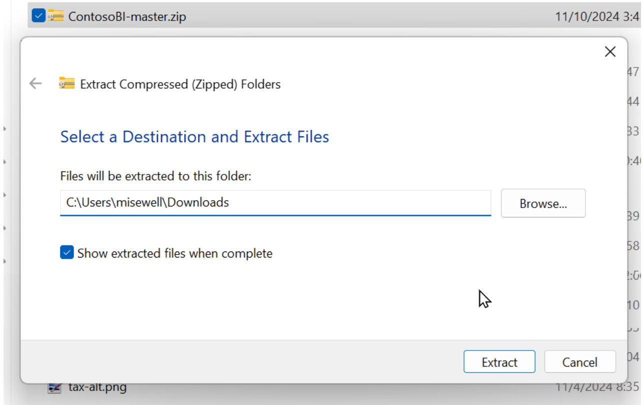
1. Download the example files from the source: https://github.com/mscottsewell/ContosoBI (From the "Code" menu choose "Download Zip")



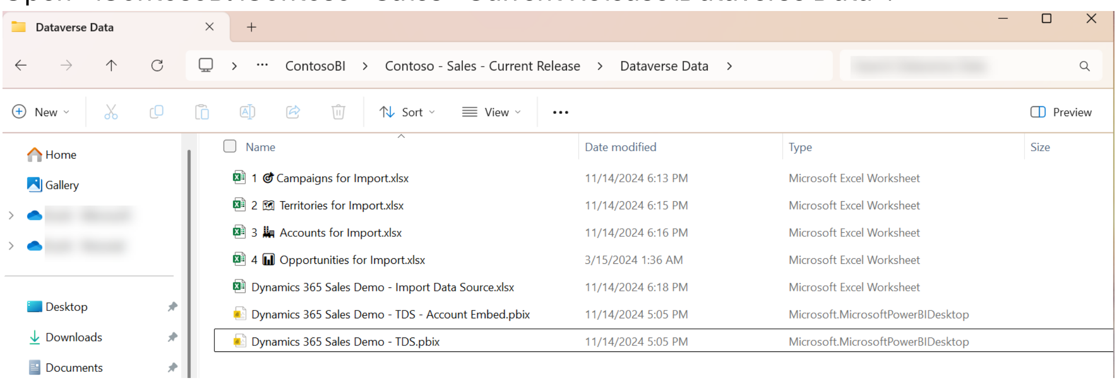
2. Once Downloaded - right click on the file, select 'Properties' and then check the 'unblock' option and 'apply' that change.



3. Right click on the zip file and "extract" it to a temporary folder on your pc. – We'll make a few changes before publishing it to your instance.



4. Open "\ContosoBI\Contoso - Sales - Current Release\Dataverse Data":



5. We'll use the files from this folder in the remaining steps.

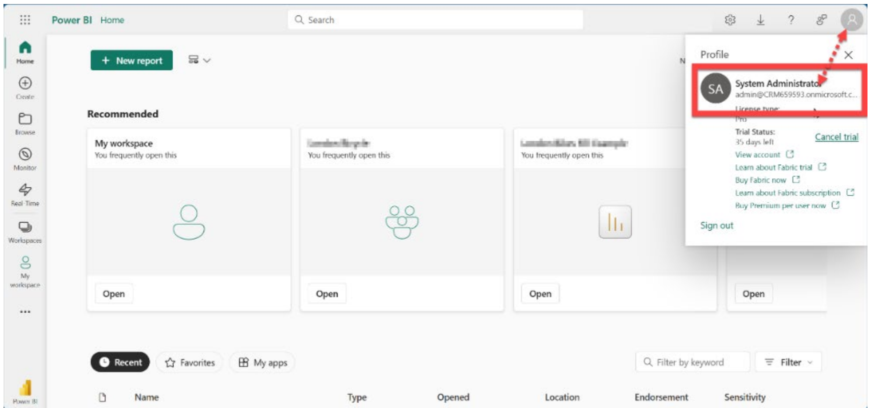
Task 4: Optional : Import Sample Data from the Four Excel Files

There are 4 source files with **Campaign**, **Territory**, **Account** and **Opportunity** sample data. These are intended to give you a faster way to get a fully loaded demo to 'light up' the reports with a rich and varied dataset. See [Import or export data from Microsoft Dataverse - Power Apps | Microsoft Learn]() for instructions on loading data from Excel into a demo environment. –

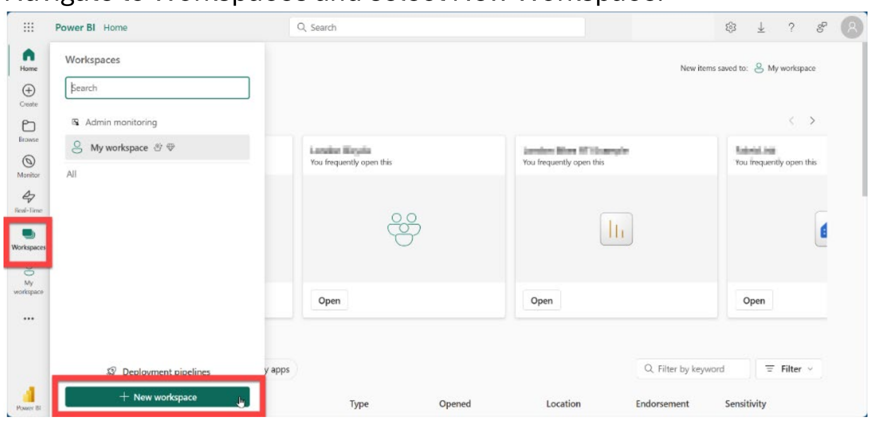
Note that the Account and Opportunity "Owners" in the files are based on one of the standard demo datasets, but that list may not match yours. – If the list of Owners in your environment is different from the ones used in the Excel spreadsheets, you'll need to bulk update the excel files with the users in your environment.

Task 5: Set up Sales Enablement workspace in Power BI online

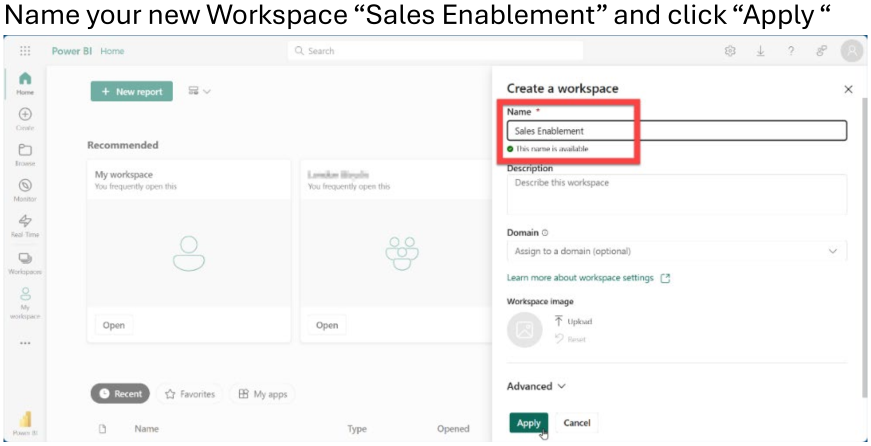
1. Open Power BI Online https://app.powerbi.com/
   (Ensure that you are signed in to your Online Demo Instance as an Admin)



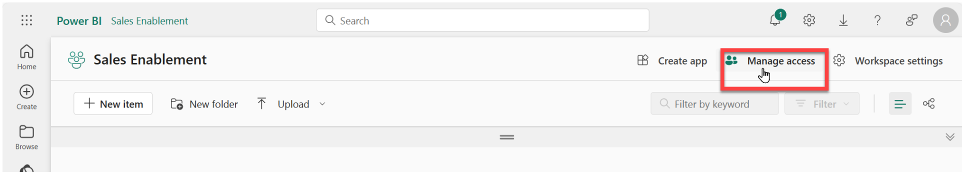
2. Navigate to Workspaces and select New Workspace.



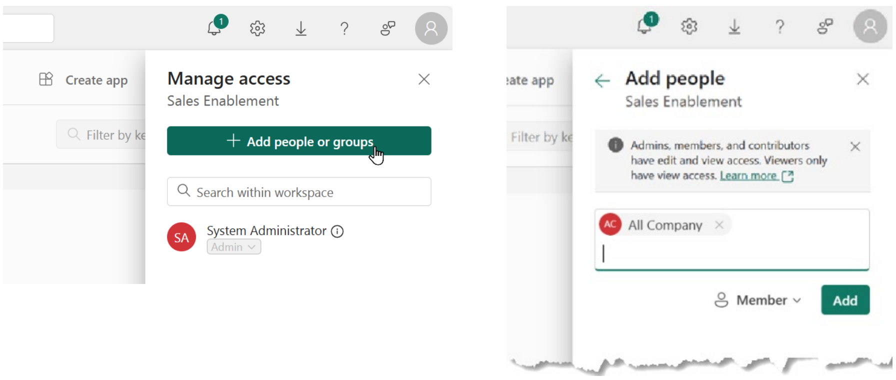
3. Name your new Workspace "Sales Enablement" and click "Apply "



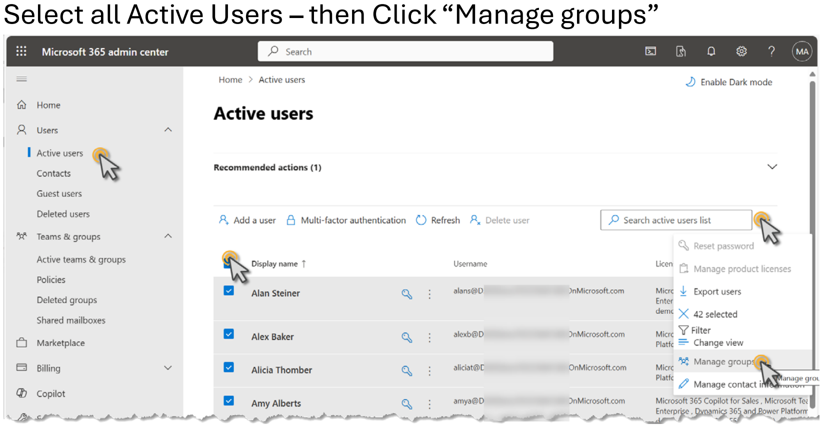
4. Once created, Power BI will open to that new workspace.
   Click on “Manage Access”



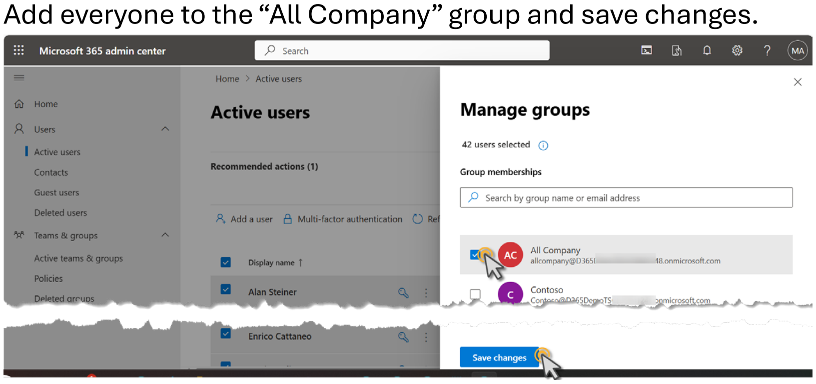
5. Then select “Add people or groups” and Add “All Company” as members.



6. In Portal.Office.Com – you may need to add your users to that group.
   Select all Active Users – then Click “Manage groups”



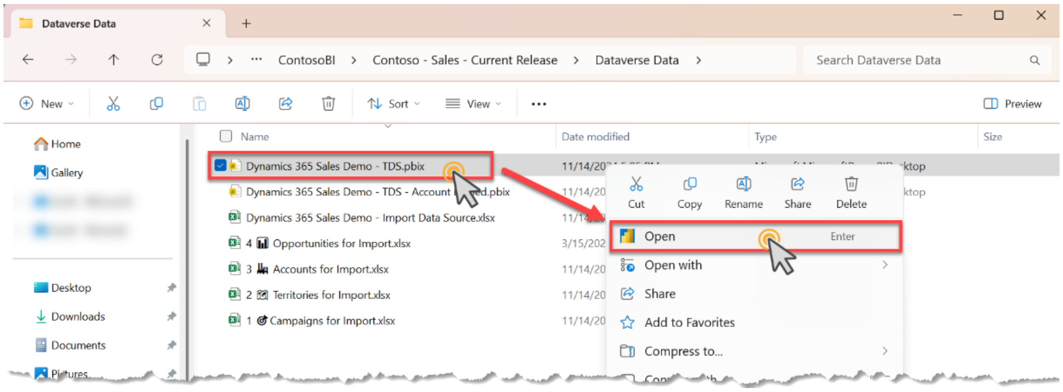
Add everyone to the “All Company” group and save changes.



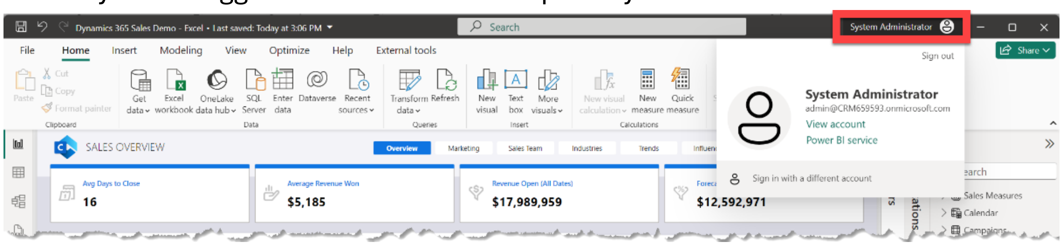
Configure the Report in the Power BI service

Task 1: Connect the Power BI report to the Excel Source

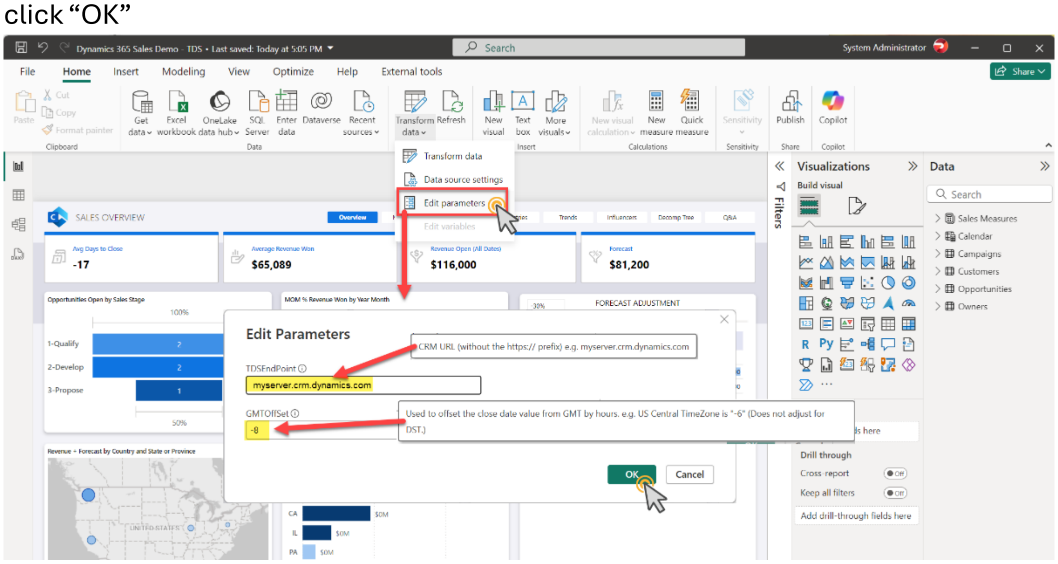
1. From the folder you downloaded the files to earlier, open the "**`Dynamics 365 Sales Demo - TDS.pbix`**" file in the Power BI Desktop App.



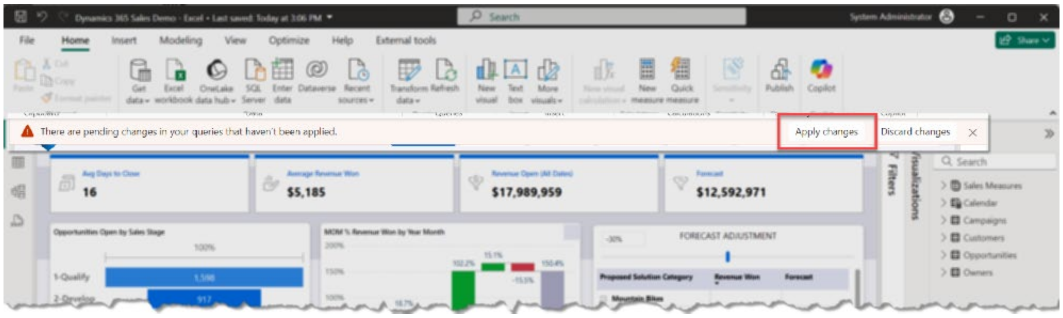
2. Ensure you are logged into Power BI desktop with your Demo Tenant's admin credentials.



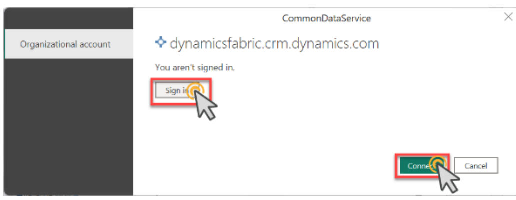
3. Update the data source, by selecting "Transform Data → Edit Parameters"
Then paste the CRM URL (Without the https:// prefix into the TDSEndPoint parameter and set the GMT offset to reflect the difference between GMT and your local timezone. Then click "OK"



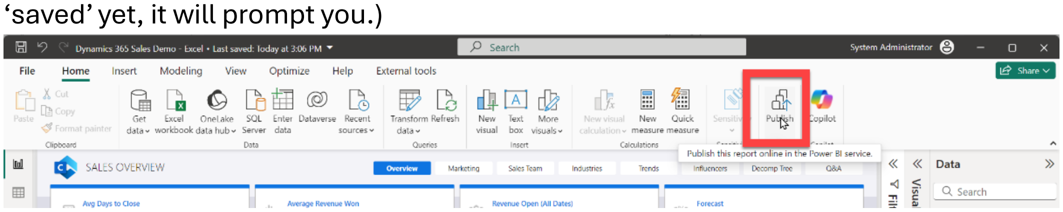
4. After Changing the parameters, you will see a banner pop on the top of the screen to Apply Changes. Select Apply Changes.



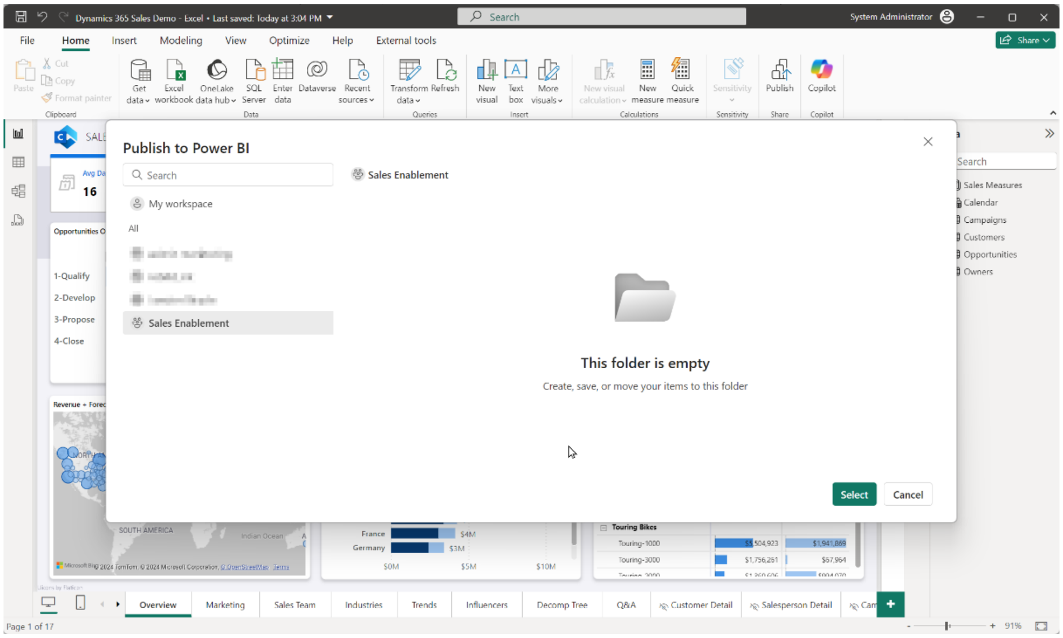
5. For the connection to Dataverse's TDS endpoint, the report will need credentials
The "**Common Data Service**" dialog should appear. – possibly behind the Load dialog
Click on it to bring it to the front.-
Click on "**Sign in**", then provide it with your Demo Tenant's Admin account. (You may also be prompted to authenticate via your phone.)
Then click "**Connect**"



6. Give the report a moment to load the data from Dataverse, then click "Publish" in the ribbon to publish the Power BI report to your new Workspace in Power BI Online (If you haven't 'saved' yet, it will prompt you.)

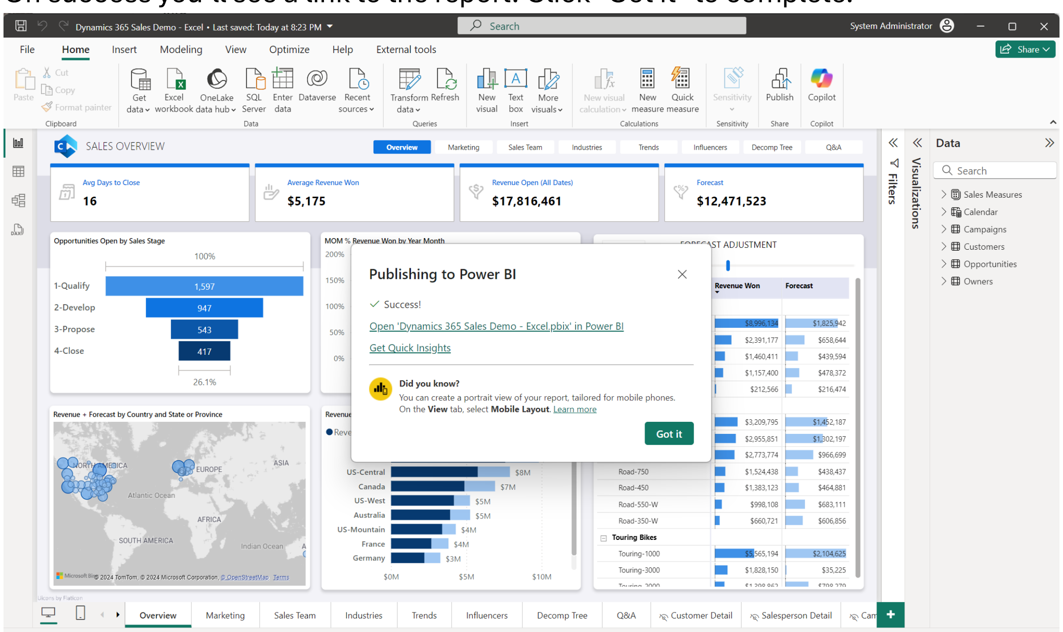


7. Select Power BI workspace that was created in Section 3 Step 3 above.



Click "Select"

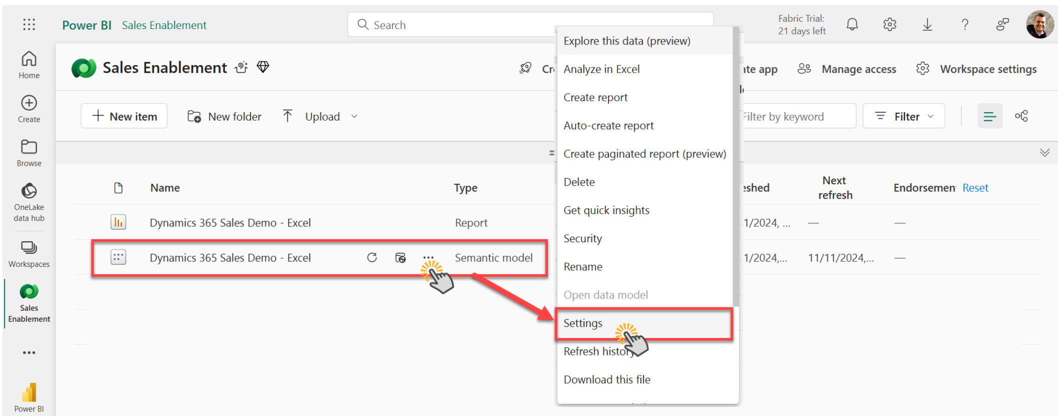
8. On success you'll see a link to the report. Click "Got it" to complete.



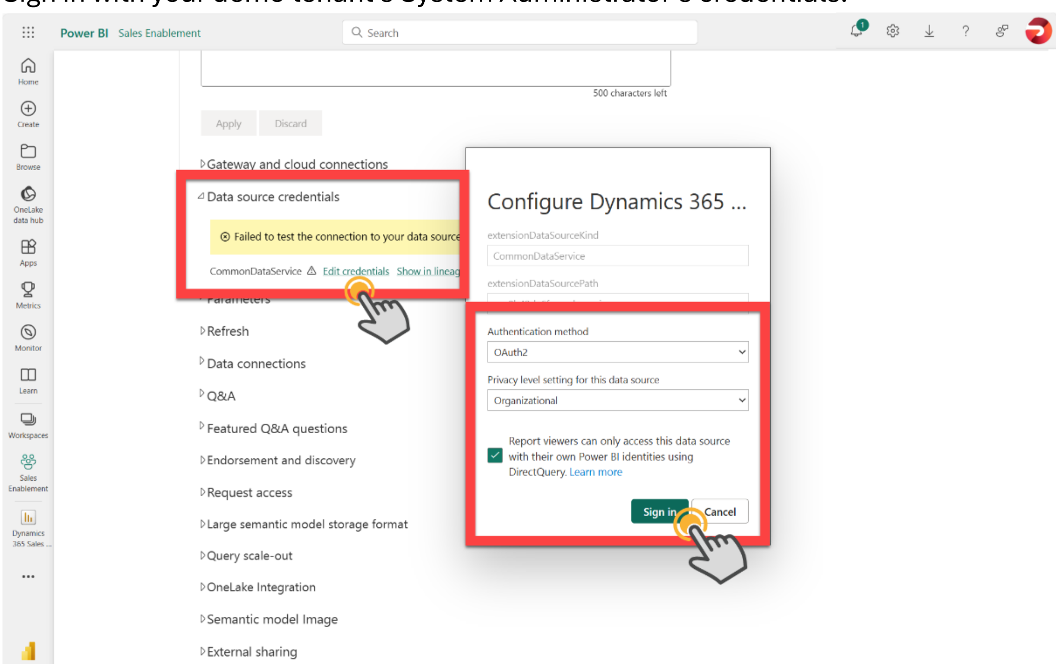
Task 2: Set up the automatic refresh of the report

Once you have published the report into the Power BI service, you'll need to set up the credentials for it to connect to the Excel file as well as set a refresh schedule for it in the service.

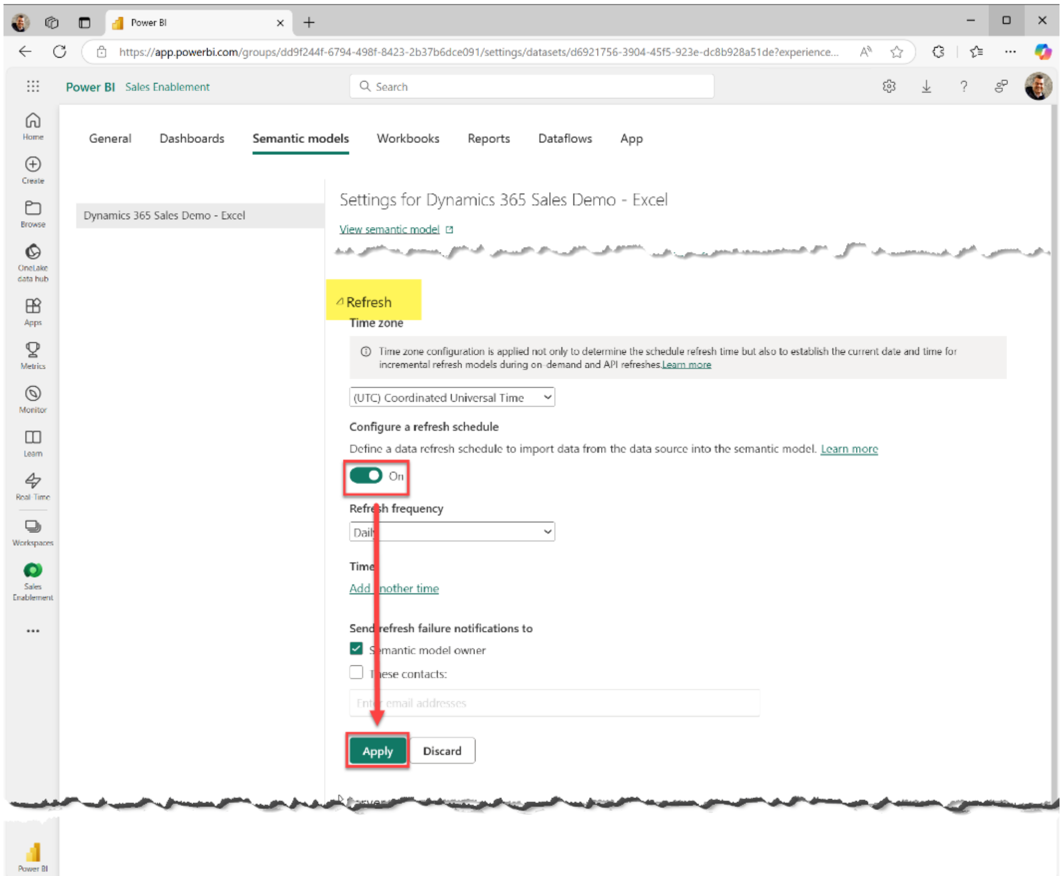
1. Open the Power BI service to the workspace where the report was published.
   Highlight the Semantic model for the report. (The "Semantic model" contains both the data and the data model definitions for the report.)
   Click the ellipse "…" and choose "Settings"



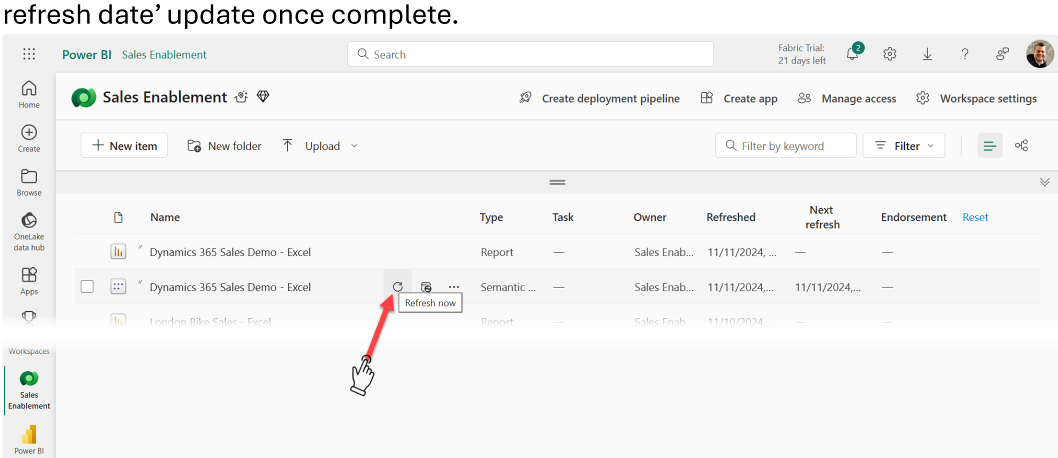
2. In the "Data source credentials" section in the Settings, Click on "Edit Credentials" and then Sign in with your demo tenant's System Administrator's credentials.



3. Open the “Refresh” section to set up a daily refresh of the data. (In this demo environment, we refresh the data daily not because the data itself is changing, but because we want the dates in the report to be current to today.)



4. The report will now refresh daily – if you want to ‘Force’ a refresh immediately, such as after you’ve updated the product list in the dashboard, Open the Workspace and click on the ‘Refresh’ button beside the Semantic model – you’ll see the refresh take place and the ‘last refresh date’ update once complete.



Add the report to Dynamics 365 as a Dashboard

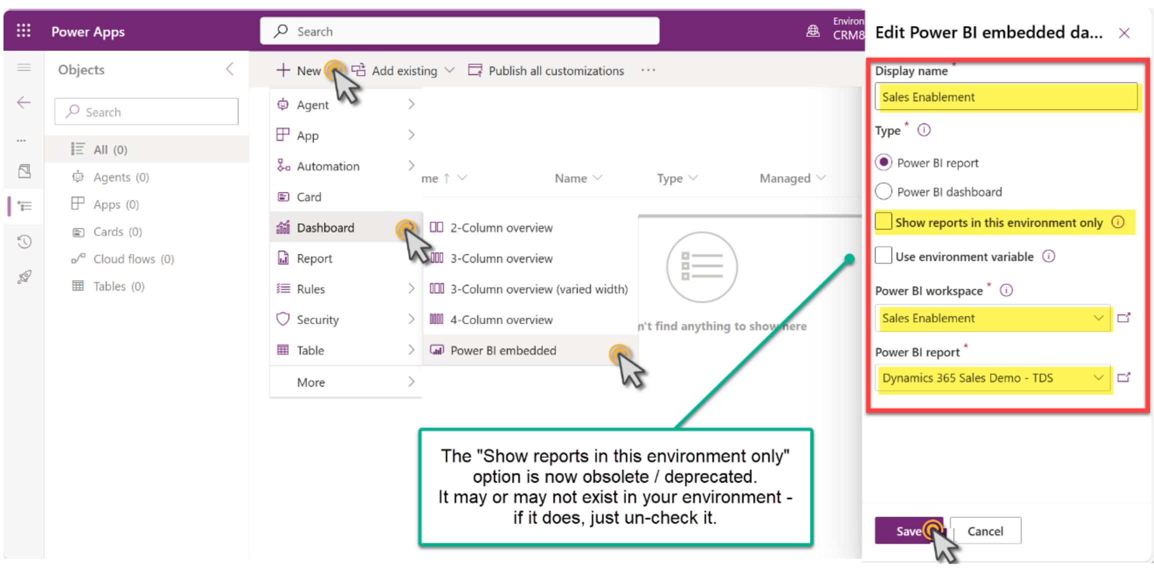
Task 1: Create the Power BI Embedded pointer in the Solution

1. Open the PowerApps maker portal https://make.powerapps.com/ (Ensure that you are working in the correct environment where you want to make this modification)

2. Select a solution that you want to use to make these changes. (You can also create a new solution to host these reports and for this dashboard modification.)

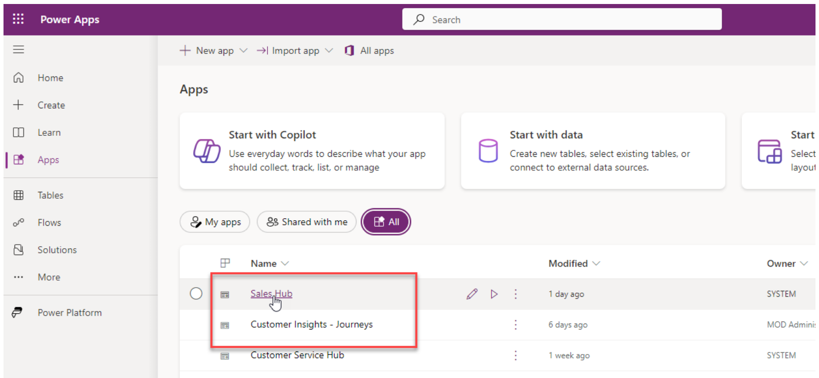
3. Once within a solution, select New → Dashboard → Power BI Embedded

4. Name your D365 Dashboard something like “Sales Enablement” or whatever you want it named in the UI visible to the user.
   - Select Power BI report. (Uncheck the “Use environment variable” IF it’s shown/checked.)
   - Select the Power BI workspace (where you deployed it in above step)
   - The select the Power BI report
   - Click ‘Save’ once you have everything ready.

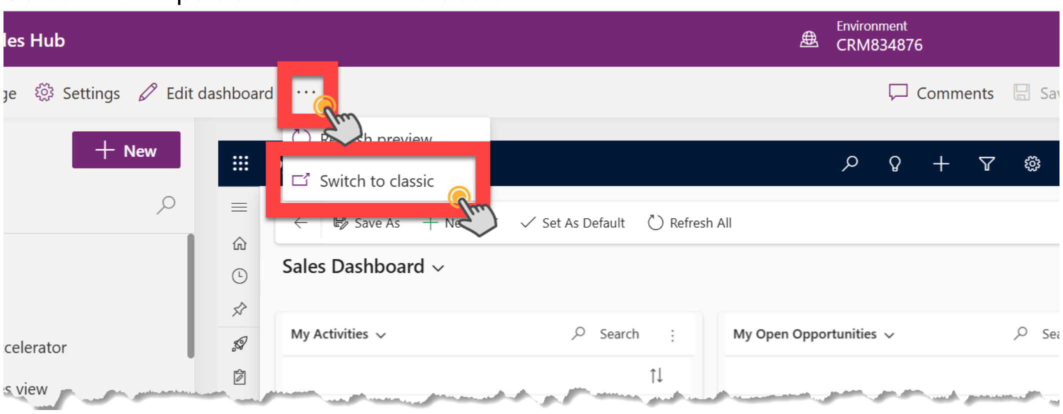


Task 2: Add the report to the Application's Dashboard

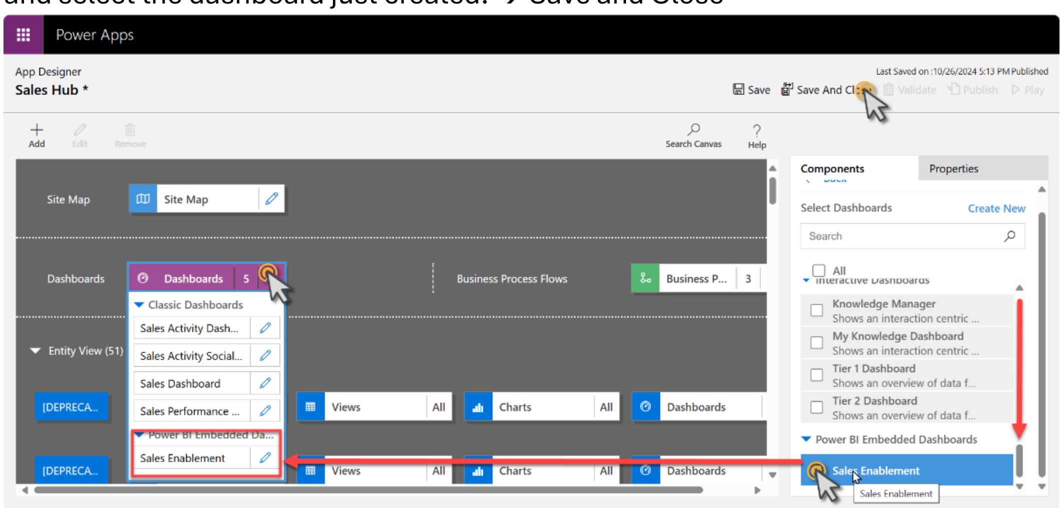
1. Go to the PowerApps Home Page and select the D365 App where you want to publish this Dashboard. (We'll start with Sales Hub – but you can repeat this task for any app)



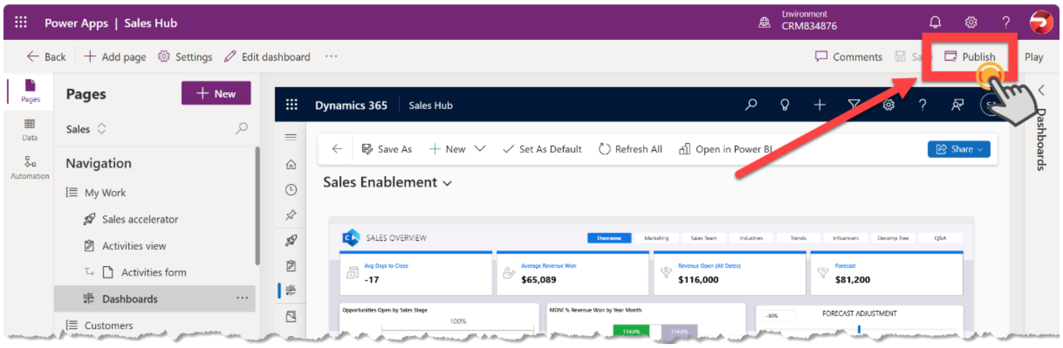
2. Select the Ellipsis and Switch to Classic



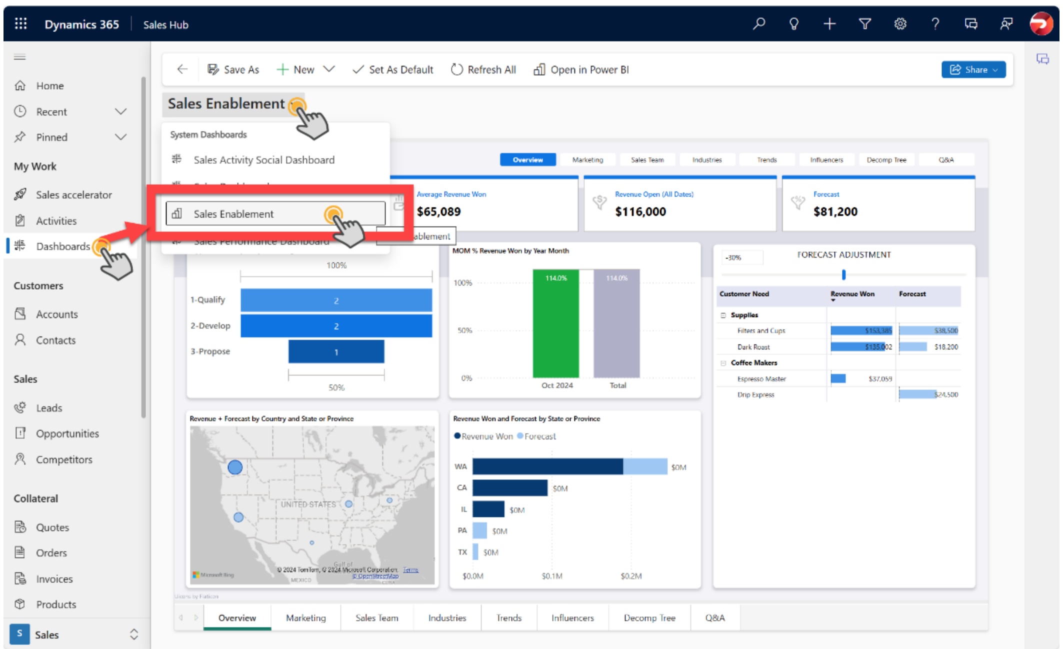
3. From The App Designer Select Dashboards → Scroll to the bottom on the right-hand side and select the dashboard just created. → Save and Close



4. Publish your Changes



5. Validate the Dashboard now shows in the dashboard dropdown.

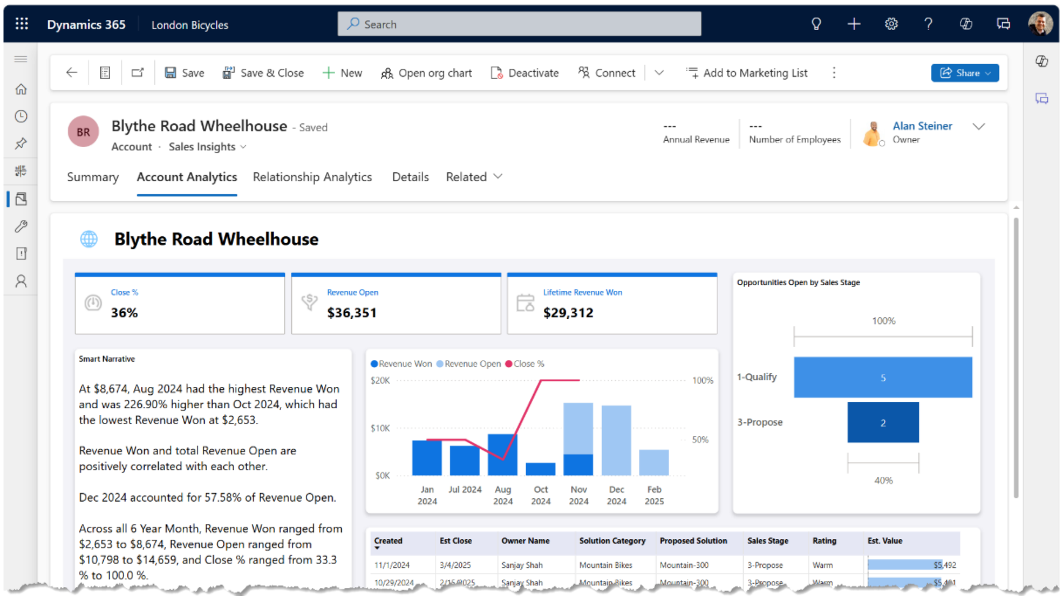


Add the In-Context/Filtered Report to the Account Form

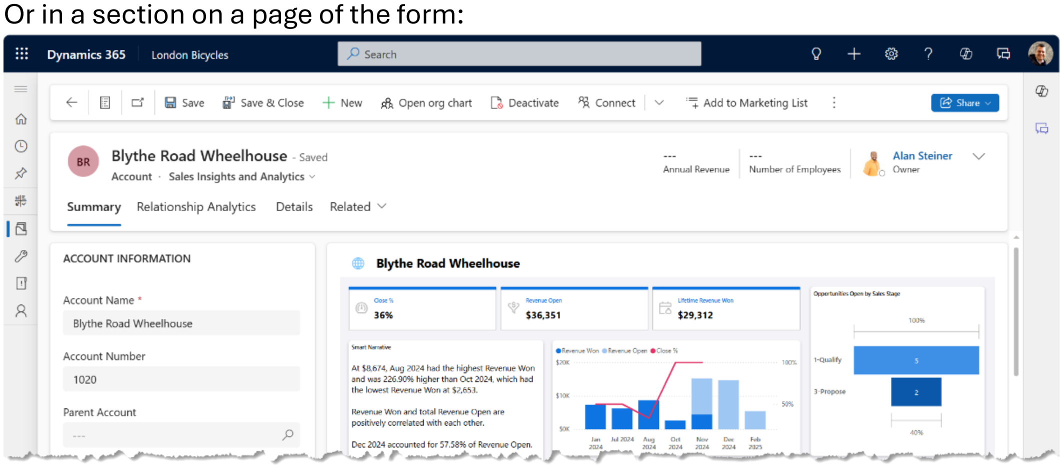
We can add a Power BI report to the Account form to give the user an expanded view of both the data inside Dynamics 365 related to that Account, but also data from outside of Dynamics 365 related to the current record.

It's an incredibly productive feature, as well as a proven driver for user adoption when Dynamics 365 has such rich data pre-filtered and presented within the context of the current record.

The report can be presented as a full tab on the form:



Or in a section on a page of the form:

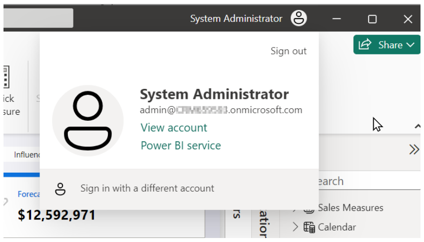


In both cases, notice how the report is dynamically filtered to the current record.

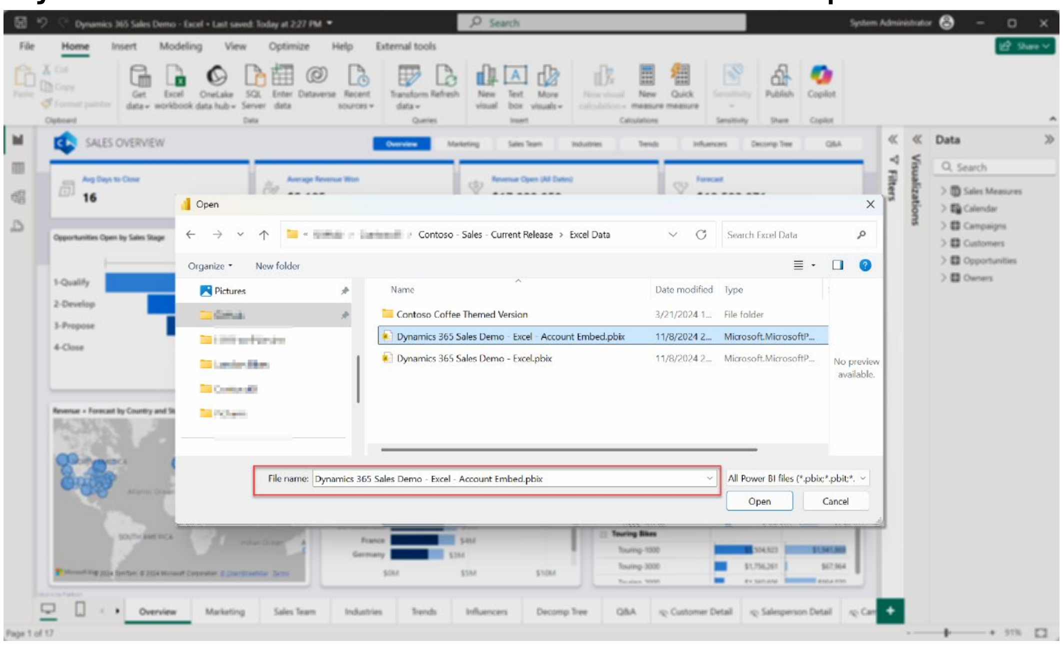
In this section, we'll import a new report that references the existing report's dataset. It ensures that all the queries and measures come from a single source of truth between the overall report at the dashboard and the individual, contextual report in the account form.

Task 1: Publish the Account-specific Report to Power BI

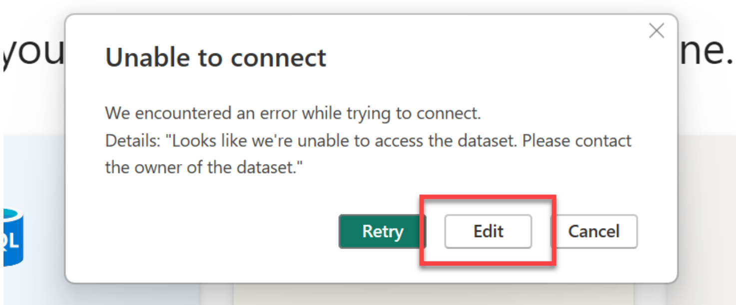
1. Open Power BI Desktop. Be sure that you are logged into the Power BI desktop app using the administrator account for the destination environments (both Fabric and Dataverse)



2. From within Power BI Desktop, Choose File, Open to open the Account report, named: "**`Dynamics 365 Sales Demo - TDS - Account Embed.pbix`**".

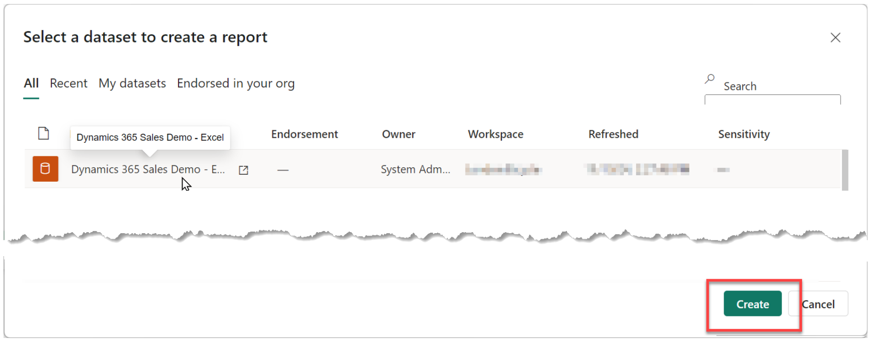


3. You'll be presented with an error message like this:

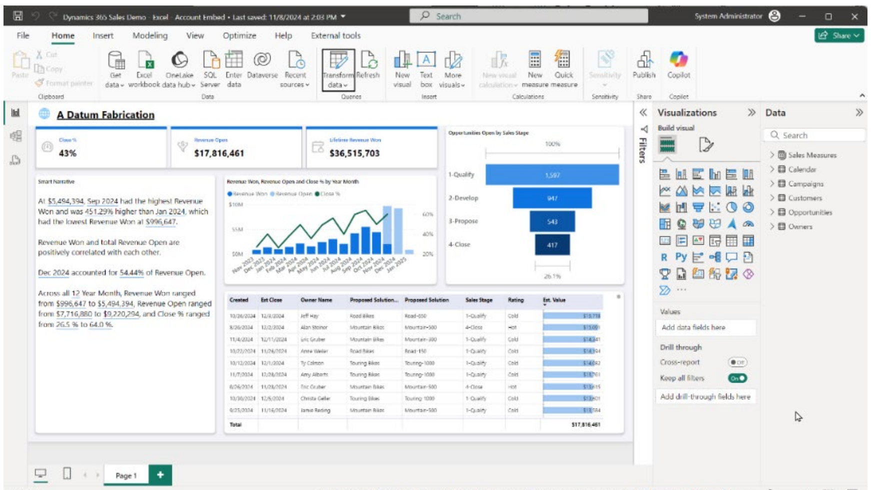


Click "**Edit**"

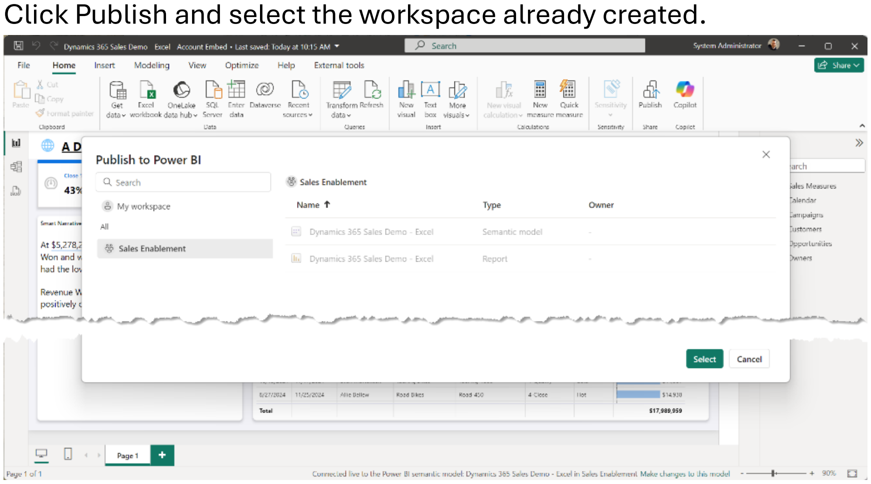
4. Choose the dataset (or semantic model) of the "**Dynamics 365 Sales Demo - TDS**" report we imported earlier.) Then click "**Create**"



5. The report should open, connected to the dataset online in your workspace in Power BI.

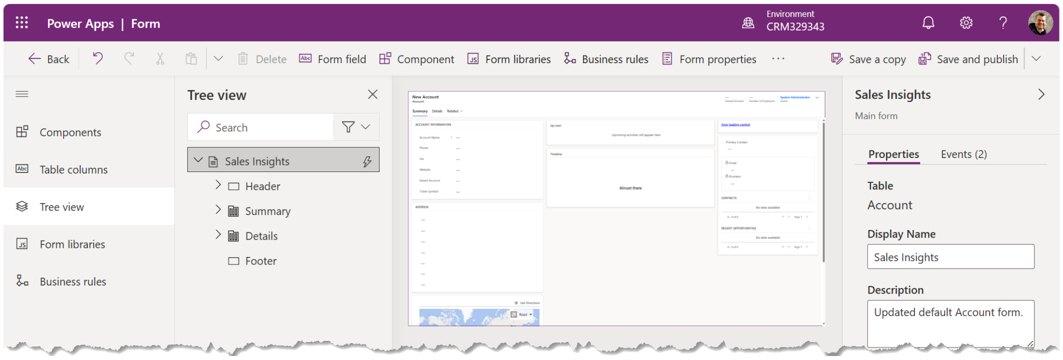


6. Click Publish and select the workspace already created.

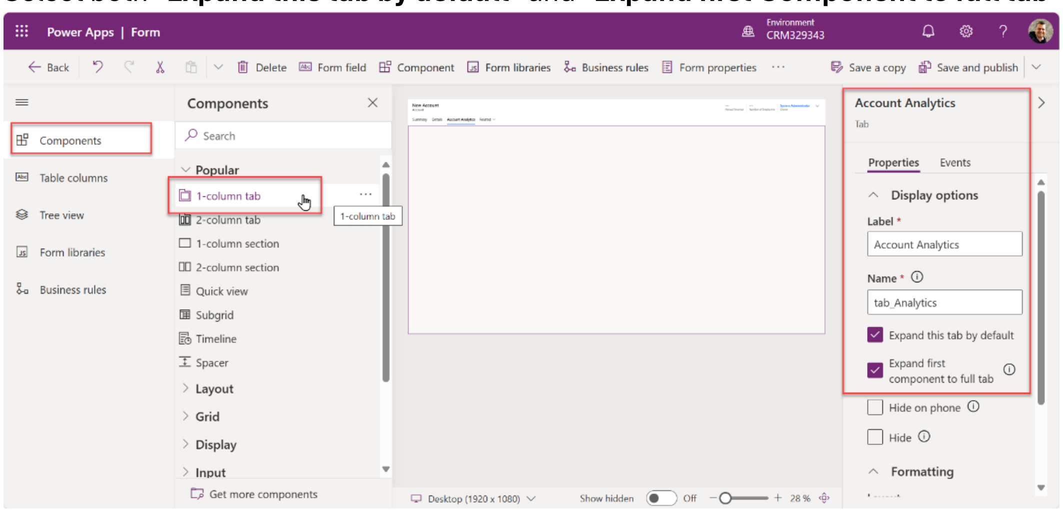


Task 2: Add a Power BI placeholder section to an Account Form

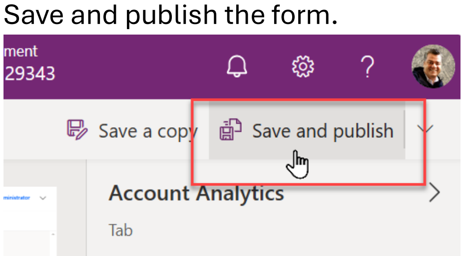
1. In https://make.powerapps.com, open an account form to add a Power BI report placeholder.



2. From “Components” add a new 1-column tab.
   Label: “Account Analytics”
   Name: “tab_Analytics”
   Select both “**Expand this tab by default**” and “**Expand first Component to full tab**”



3. Save and publish the form.

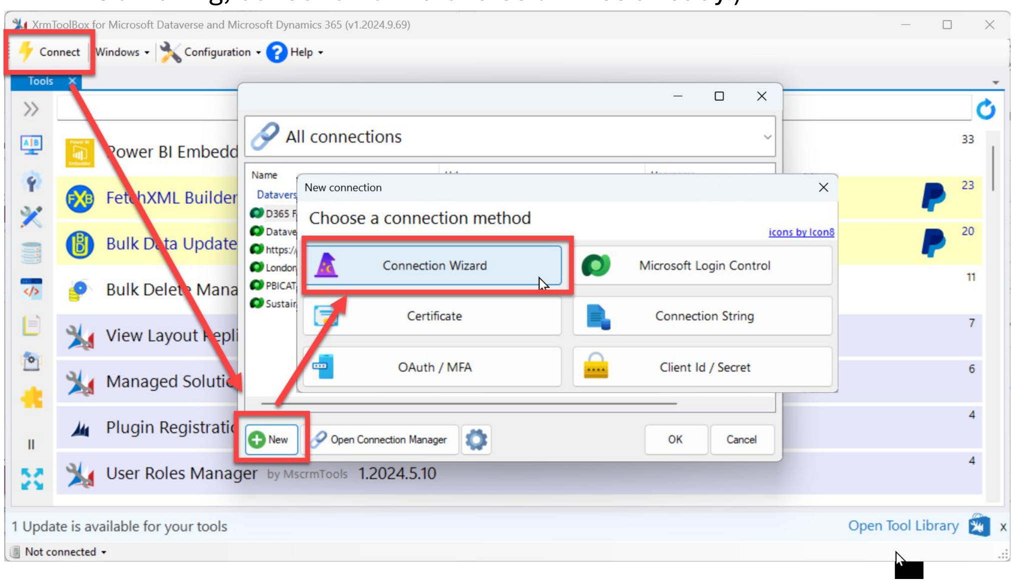


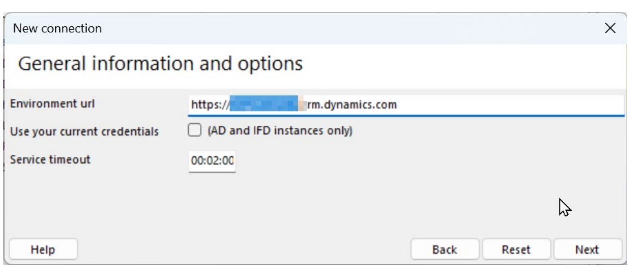
4. Close the form.

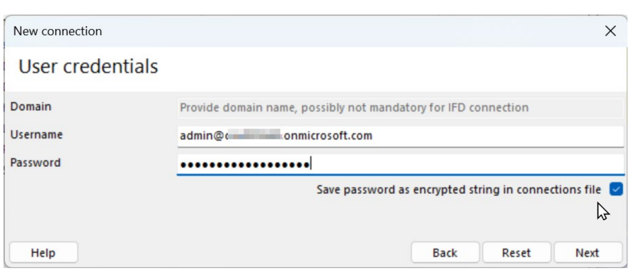
Task 2: Use an XrmToolBox plugin to add the report to the Account form

This task adds the report to the form using a free, community-supported tool created by Ivan Ficko, the “Power BI Embedder” (Power BI Embedder · XrmToolBox) hosted in the larger XrmToolbox - https://www.xrmtoolbox.com/. This utility is simply an easier way to implement the method documented here: Embed a Power BI report in a model-driven app main form - Power Apps | Microsoft Learn

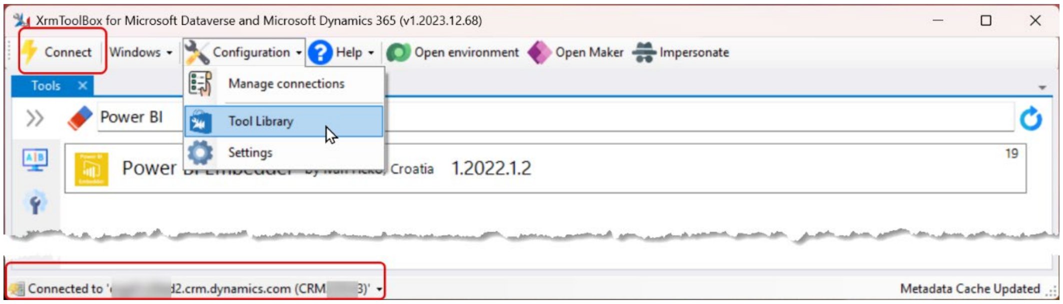
1. Open the XrmToolBox and connect to your Dataverse environment. (Instructions for the setup and use of the XrmToolBox is available at the XrmToolBox homepage https://www.xrmtoolbox.com/documentation - review this if you’re not familiar with this amazing, collection of Dataverse utilities already.)



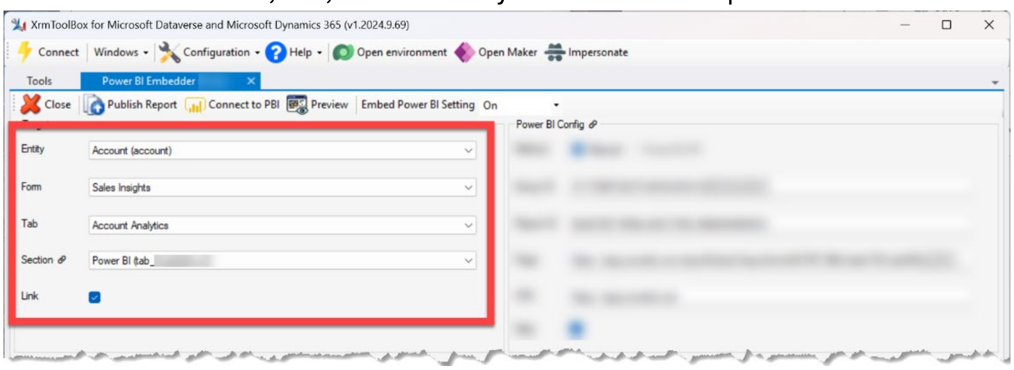




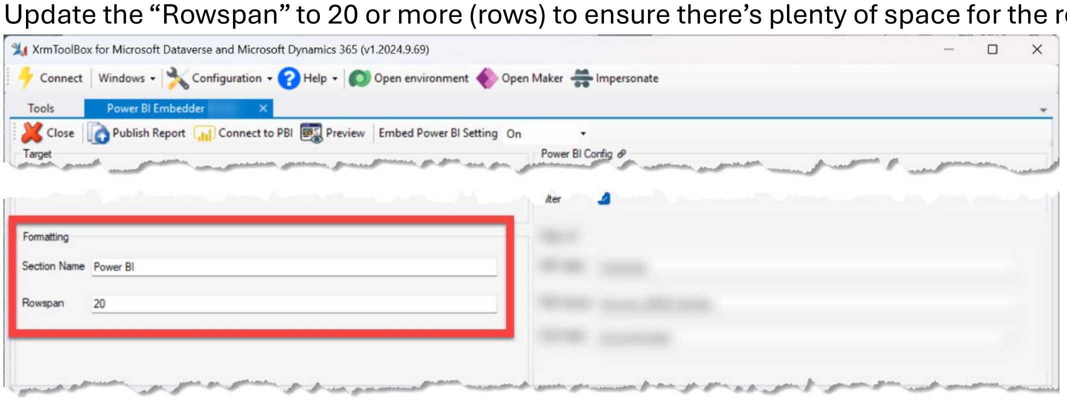
2. Once you've connected, search for the "Power BI Embedder" by Ivan Fiko – If it's not already installed, you'll need to add it from the Tool Library.



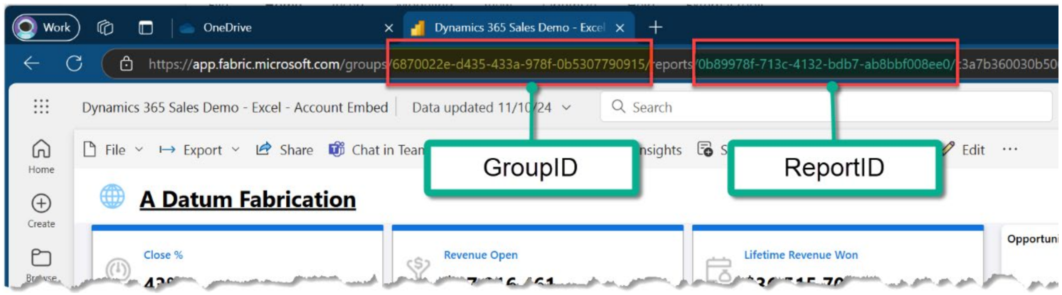
3. Open the Power BI Embedder tool and select the "Account" Entity.
   Then select the Form, Tab, and Section you created in the previous Task



4. Update the "Rowspan" to 20 or more (rows) to ensure there's plenty of space for the report.



5. Open the report in Fabric / Power BI and note the URL
   The URL will include both a group and report ID – along with other information.
   I'm showing the values from my demo environment, your values will be different.

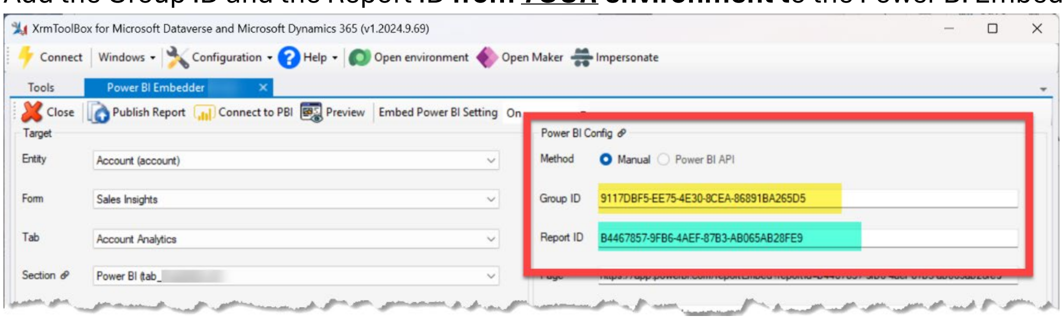


In this example environment, the full URL is: (I've formatted it into rows for clarity)

```
https://app.fabric.microsoft.com
  /groups/6870022e-d435-433a-978f-0b5307790915
  /reports/0b89978f-713c-4132-bdb7-ab8bbf008ee0
  /c3a7b360030b506cda6c?experience=power-bi
```

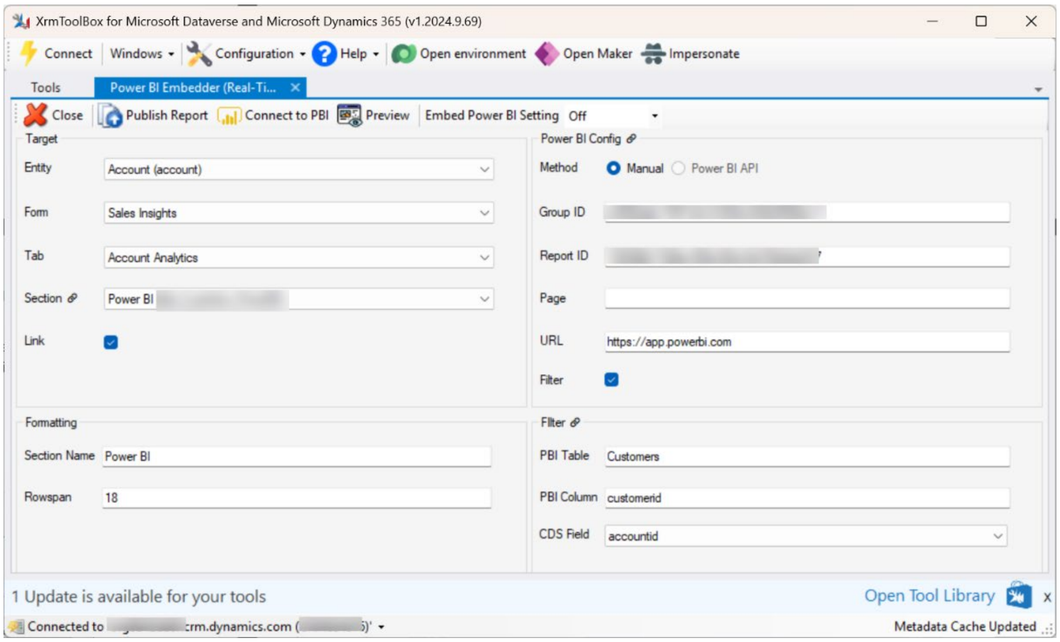
We'll ONLY use the groups ID and the reports ID – (without the "/" separator)
The values in your environment will be different – use the ones from your environment.
Group ID = `6870022e-d435-433a-978f-0b5307790915`
Report ID = `0b89978f-713c-4132-bdb7-ab8bbf008ee0`

6. Add the Group ID and the Report ID **from *YOUR* environment t**o the Power BI Embedder



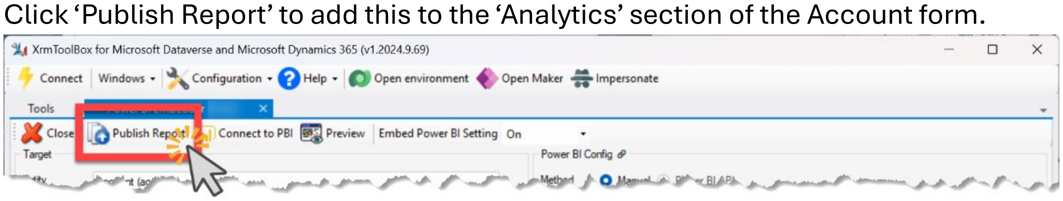
7. Click the "Filter" checkbox and fill in the filter details – (Note: these values are **Case Sensitive**) and the data in the Account tab in Excel must match the data in your Dataverse account table.

   PBI Table: **Customers**
   PBI Column: **customerid**
   CDS Field: **accountid**



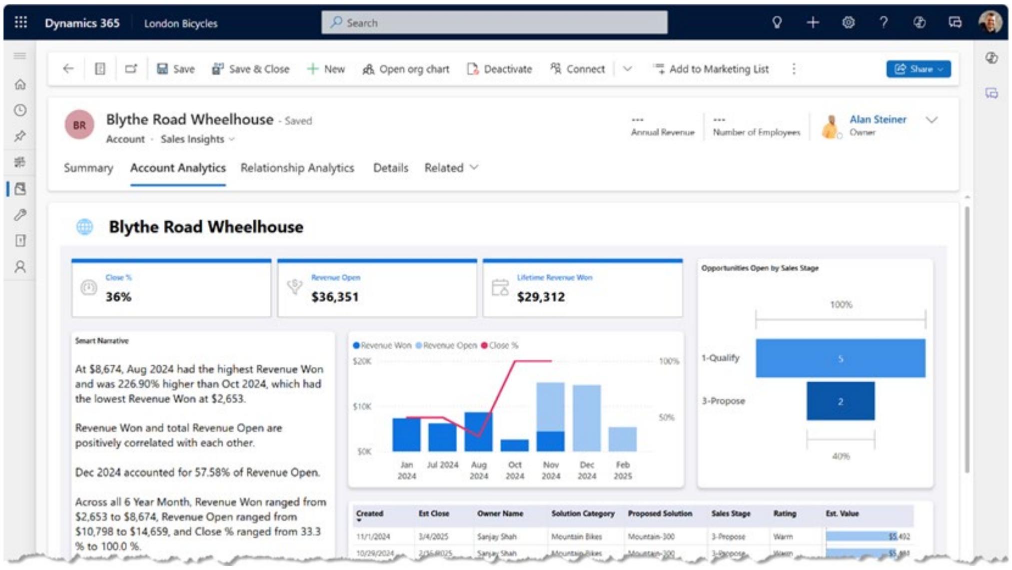
Note: The "Page" value is automatically populated later, you can just ignore it.

8. Click 'Publish Report' to add this to the 'Analytics' section of the Account form.



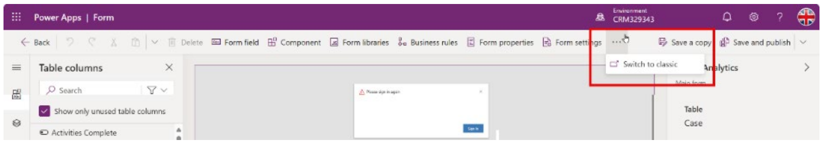
(It may take a few minutes to complete – just be patient. Also, frequently it will time out during the first attempt to publish with a message about "the underlying connection" being "closed", if it does, just click the Publish Report again.)

9. When the publish completes, open the Account form in Dynamics 365.
   *Be sure you've added it to the app you're using, and switch to the form if necessary.

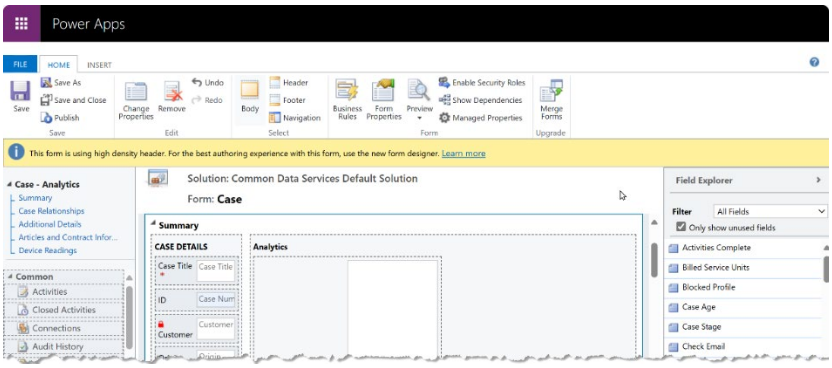


Also, a "Publish All" and a full refresh of the browser window might be needed to ensure that the configuration updates are fully available.

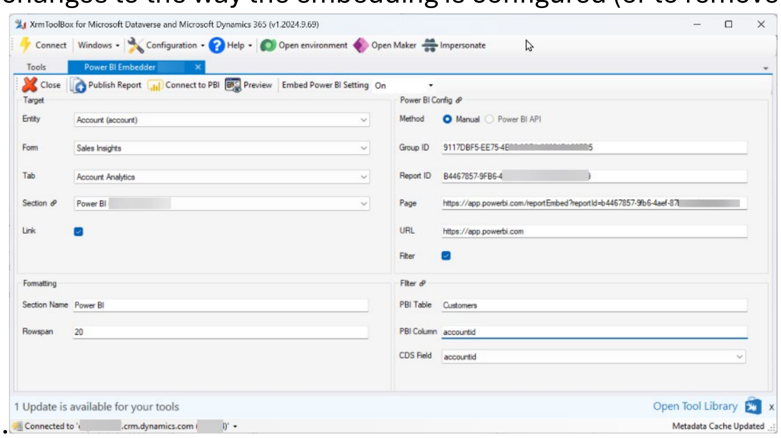
10. Once the report is e
    mbedded, you may have trouble editing the form in the current form editor in Power Apps - If you do, just switch to the classic editor.



You'll be able to edit in the classic editor –



For any changes to the embedded report configuration, use the XrmToolBox again to update changes to the way the embedding is configured (or to remove it.)



Hints

1. No more than an hour before a demo it's a good idea to load the report once and click through everything just to warm up the cache.
2. If the report is open in a tab and sits idle for a period, it may time out – just click the browser's page refresh button. The 'Refresh All' button inside CRM does nothing to reload the report.