DW-SFTIES

Power BI Custom Connector

# Table of Contents

1. Prerequisites
2. Capabilities Supported
3. Adding the Connector to Power BI
4. Using the Connector
5. Modifying the Connector
6. Troubleshooting

# Prerequisites

Before you start, ensure you have the following installed:

1. Using the Connector (in Power BI Desktop)

* Power BI Desktop (latest version)

1. Modifying the Connector’s Code

* Visual Studio Code
* Power Query SDK for Visual Studio and Extensions for Visual Studio Code

# Adding the Connector to Power BI

Custom Connectors are deployed using a .mez file.

1. Obtain the .mez file, which can be found in the “bin\AnyCPU\Debug” project folder
2. Copy the .mez file to the Custom Connectors folder:

*For Power BI Desktop:*

c:\users\<<username>>\Documents\Power BI Desktop Custom Connectors

1. Paste the .mez file in the following folder:

C:\users\<<username>>\Document\Microsoft Power BI Connector

1. Open Power BI Desktop
2. Enable Custom Connectors in Power BI

* In Power BI Desktop, navigate to ‘File > Options and settings > Options > Security’
* Enable "Allow any extension to load without validation or warning" under ‘Data’

1. Click ‘OK’ and Restart Power BI Desktop

# Using the Custom Connector

Using the Custom Connector is a similar process to using any other Power BI Connector, such as SQL Server or Excel file.

1. Once the Custom Connector has been added and enabled in Power BI according to Section B, open Power BI Desktop.
2. Create a Blank Report.
3. Select ‘Get data from another source’ or ‘Get Data 🡪 More’ from the Menu.
4. Search for and select the ‘DWSFTIES\_API (Custom)’ Connector from the list of connectors and click ‘Connect’.

\**You may receive a warning regarding connections to a third party-service. If you don’t want to receive this warning, select ‘Don’t warn me again for this connector’.*

1. Enter your 2-Digit primacy agency code (e.g., AL, IA, etc.)
2. From the dropdown, select an available API and click ‘OK’
3. If prompted for credentials, enter your CDX Username and Password.

*\*If Power BI has cached your credentials from previous connections. To view permissions, you can …*

1. A Navigator showing all API endpoints that **do not require a parameter** will be displayed.
2. In the Navigator, select any endpoint to preview the data. To load data, select one or more endpoints and click ‘Load’.
3. The endpoint(s) selected will appear in the Data tab and can be used just like any other data source to create visuals, reports, etc.

# Modifying the Connector

1. Open Visual Studio
2. Ensure the Power Query SDK extension has been installed

<https://learn.microsoft.com/en-us/power-query/install-sdk#installing-the-power-query-sdk>

1. Clone the repository using Git or download the ZIP file and extract it.

git clone <https://github.com/adchavers/Power-BI-Custom-Connector-SFTIES.git>

1. Open the Solution in Visual Studio Code by either right-clicking on the folder and selecting ‘Open in Visual Studio Code’ or by selecting File 🡪 Open Folder within Visual Studio Code and selecting the folder.
2. Edit the Code

All functional code for the Custom Connector is contained in the ‘.pq’ file. This file defines the data connector’s logic, including authentication, navigation setting, data source queries, and transformation.

Although the DW-SFTIES Power BI Custom Connector is custom-coded by the author, Microsoft’s Trip Pin Tutorial was a useful resource in structuring the connector and may help clarify the high-level logic being followed.

<https://learn.microsoft.com/en-us/power-query/samples/trippin/readme>

1. Test Changes by evaluating the ‘. query.pq’ file included in the connector. The strings for primacy agency code, API, and endpoint can be modified in this file without harm; however, writing other tests against the connector would require knowledge of the Powery Query M formula language.
2. Build the solution:

* Press Ctrl+Shift+P or select ‘Show and Run Commands’ from the search menu
* Press Ctrl+Shift+B or select ‘Tasks: Run Build Task’
* Select ‘build: Build connector project using MakePQX’

This will generate a .mez file in the ‘bin\AnyCPU\Debug’ folder.

# Troubleshooting-

Below are some common issues with Custom Connectors and how to resolve them:

1. Connector Not Appearing in Power BI

* Ensure that the .mez file is placed in the correct directory.
* Verify that "Allow any extension to load without validation or warning" is enabled in Power BI options.
* Restart Power BI Desktop after copying the .mez file.

1. Build Errors:

* Check the output window in Visual Studio for error details.
* Ensure all necessary dependencies are installed and referenced correctly in the project.

1. Data Loading Issues:

* Verify that your queries and transformations in the `.pq` file are correct.
* Clear permissions from any existing data sources to the API to ensure incorrect cached credentials are not being applied.
* Check the API or data source for any connectivity issues.