POWER BI INTEGRATION

Monitor your most critical reporting data in Web or Mobile app.

EXTENSION FOR MAGENTO 2

USER GUIDE

June 2017

Version 1.0.0

Kyrylo Kostiukov

kostiukov@bimproject.net

Date	Version Number	Document Changes
14.06.2017	1.0.0	Initial Draft
28.06.2017	1.0.1	First version
16.07.2017	1.0.2	Adaptation for Magento 2

PREFACE

This document intended to be used to support end users in usage of the Power BI Integration extension for Magento developed by Kyrylo Kostiukov.

Introduction part provides information of features of the extension and general concepts that has to be understood in order to achieve best results from the Power BI Integration and to organize usage of it in the most efficient way.

Configuration part provides information on how to setup initial configuration and where to find additional information on how to enable integration between Power BI Online service and Magento Store. It provides details on configuring regular data updates between Magento 2 and Power BI and some tips on how it could be done efficiently.

Next part of the User Guide provides some usage examples that may help users to get valuable results from the provided integration between Magento 2 store and Power BI Online service.

Final notes give users an overview of what could be the future of the extension and how can users achieve extra functionality and grow their BI infrastructure based on the provided integration.

This document is not covering details related to installation of the extension on Magento 2 platform or development information needed to extend the functionality of the provided extension. Such information might be found in the following documents:

- Power BI Integration M2 Installation Guide
- Power BI Integration M2 Reference Guide

Please, contact the extension provider if you want to receive the listed documents.

CONTENT

INTRODUCTION

GENERAL CONCEPTS

It is important to understand how Power BI API is working in order to know how to organize proper development of reports and dashboards with possibility of making changes in the future and to be able to keep data delivery to dashboards and reports stable.

The provided extension was developed with intention to simplify usage of the API as much as possible. From one side, the goal is to protect users from making mistakes that may lead to broken reports or dashboards. From another side, we want user to be able to deliver data to Power BI as easy and fast as possible. This was not a trivial issue due to limitations of Power BI API.

For these reasons, we developed a compromised solution based on *Temporary* and *Permanent* datasets.

In this chapter we first go through basic terminology, part of which is taken from Power BI. Next we describe our concepts of temporary and permanent datasets.

BASIC TERMINOLOGY

In order to explain the concepts, we start from covering some basic terminology used in the extension and in the user guides.

POWER BI REPORT

A Power BI Report is a visual representation of data from a Power BI Dataset.

More information on Power BI Reports could be found in Power BI documentation:
https://powerbi.microsoft.com/en-us/documentation/powerbi-service-reports/

POWER BI DASHBOARD

Power BI Dashboard is a set of Power BI Dashboard Tiles. Unlike Power BI Report, Dashboard combines visual representation of data that may come from different datasets. For example, you may display KPIs calculated based on a dataset coming from Magneto 2 store with KPIs based on a dataset coming from Salesforce, Google Analytics or other sources.

More information on Power BI Dashboards could be found in Power BI documentation:

https://powerbi.microsoft.com/en-us/documentation/powerbi-service-dashboards/

POWER BI DASHBOARD TILE

Power BI Dashboard Tiles are single data visualizations created using a single dataset and placed on a dashboard.

More information on Power BI Dashboard Tiles could be found in Power BI documentation:

https://powerbi.microsoft.com/en-us/documentation/powerbi-service-dashboard-tiles/

POWER BI DATASET

More information on Power BI Dataset concept could be found in Power BI documentation:

https://powerbi.microsoft.com/en-us/documentation/powerbi-service-basic-concepts/#datasets

POWER BI TABLE

Datasets in Power BI consist of tables. Tables have set of columns with specific data types and data rows. In general, each Magento Report (without summary rows) or Magento grid represents a data that might be considered as a table once imported into Power BI.

LOAD PROFILE

Load Profile is a term specific for the Power BI Integration extension for Magneto 2. Load profile associates portion of data in Magento with a Power BI Table. Load profiles the same as Power BI Tables form a Dataset.

In particular, once Magento backend user creates a Load profile based on a Magento Report, he creates an association of Magento Report with Power BI Table. Such Load Profile stores selected filters and controls specific for the report. It also stores a name for the Power BI Table. Under this name it will appear as a Power BI Table in the Power BI Dataset.

LOAD DATASET

Load Dataset is another term specific for the Power BI Integration extension for Magento 2. Load Dataset associates set of Load Profiles with a Power BI Dataset. Load Dataset might be Temporary or Permanent.

DATASET PUBLISHING

Publishing of the dataset can be done. Publishing for permanent dataset happens only once.

DATASET RE-PUBLISHING

Re-publishing of the dataset happens when some new tables have to be added to the Power BI Dataset or table names have to be modified. A modification of columns does not require re-publishing. Current implementation supports re-publishing only through delete/create operations. It means that old dataset should be deleted from Power BI and new dataset has to be published again. Permanent Load Datasets cannot be re-published.

LOADING

Loading of Load Profile updates the data or Power BI Table columns in a dataset. Loading of Load Dataset is loading of all Load Profiles included in the dataset. It does not change the amount of tables in dataset or their names. Any Load Profile or Load Dataset can be loaded multiple times. If Load Profile is assigned to a Permanent Dataset, it can be loaded only after publishing of the dataset. If Load Profile is assigned to a Temporary Dataset, dataset will be published automatically if needed.

TEMPORARY LOAD DATASET

Permanent Load Dataset is a Load Dataset that can be published or reloaded without any restrictions. The word temporary means that it users should not expect this dataset to persist on Power BI side. Power BI API does not allow you to change the structure (amount of tables or their names) of already created dataset. Every time, dataset changes its structure, it will be deleted from Power BI and new dataset instance will be published.

Once datasets is being deleted in Power BI, all reports and dashboard tiles based on this dataset will be deleted. For that reason temporary datasets should never be used to develop reports and dashboards that intended to be used in the future.

By default, Temporary Load Dataset has name PBI_TEMP. Under this name users may see the dataset in Power BI Service.

Permanent Load Dataset is a Load Dataset that can be published only once. After that it should not be possible to make modifications to the Load Dataset or its structure (amount of tables and their names).

Permanent Load Dataset helps to make sure that published Power BI Dataset will stay stable and can be used for Reports and Dashboards creation.

If you want to republish it, you will need to delete dataset manually in Power BI Service. But once you do so, you may delete reports and tiles based on that dataset.

To add more tables into Permanent Dataset, you need to create new Permanent Dataset with another name and newly create all related reports and dashboards. For that reason, it is better to plan your dashboards and reports development in advance and start with dataset that contains all data that will be needed.
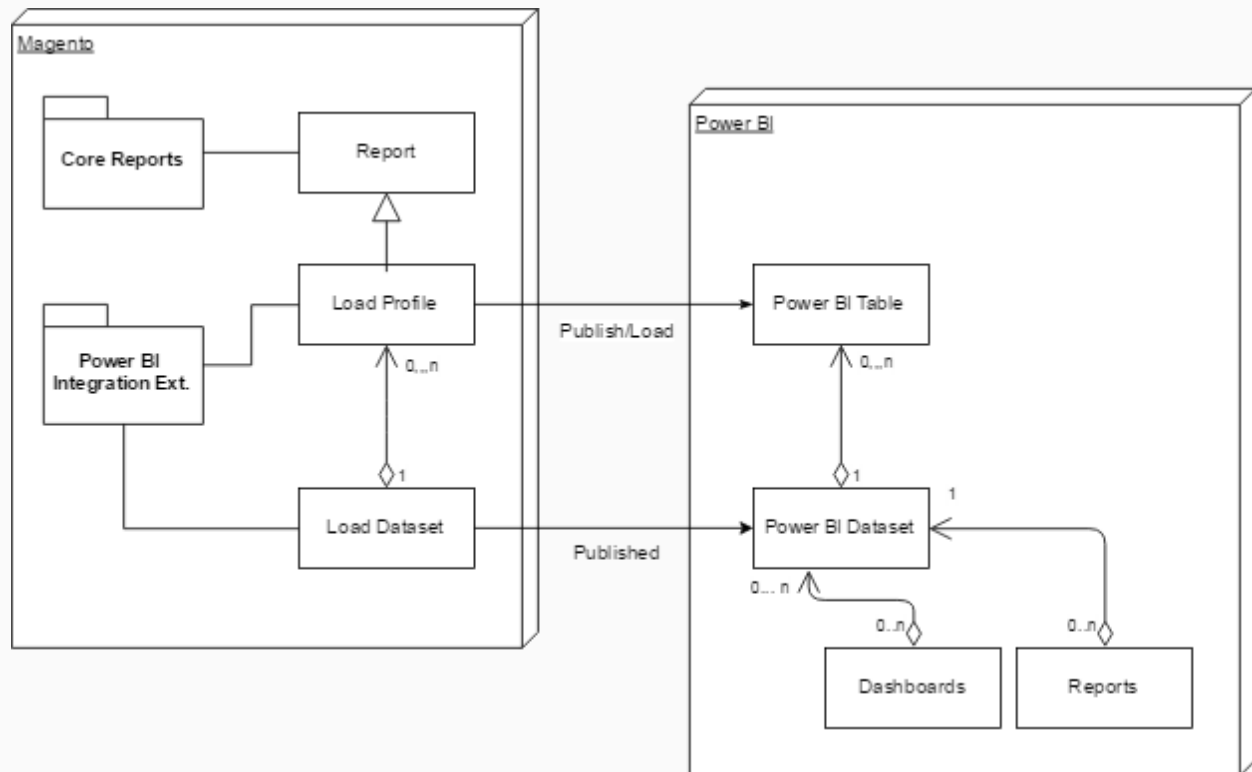
BRINGING IT ALL TOGETHER

Image 1. Components diagram and main relations.

On the components diagram displayed on Image 1 we brought together items described above. Load Profile can be created based on a Magento Report and keeps the reference to that report to be

able to retrieve the data at any time. Load Profile is assigned to Load Dataset. Load Profiles and Load Datasets are then being translated into Power BI Tables and Power BI Datasets. Based on Power BI Datasets users then can create Dashboard and Reports. Reports can be associated only with one dataset. On the other side, Dashboards can have associations with multiple datasets and display data from multiple data sources.

LOADING PROCESS

Loading process describes the process of transferring data from Magento store to Power BI with usage of Power BI Integration extension from user's perspective.

STEP 1: CREATION OF LOAD PROFILE

To create a Load Profile, user navigates to a Magento Report. Example of a report you may see on the Image 2. As you may see, there is an option of exporting to Power BI available on the page. Triggering the export will create a Load Profile that will keep association with selected filter values.

Image 2. Export option "Export to Power BI" on Total Refunded Report

As the next step the user may transfer the report to Power BI Service and see if expected results might be achieved. To do so, user may navigate to created Load Profile and press "Load To Power BI" button. You may see this button on an Image 3 below.

Since created Load Profile is assigned to a Temporary Load Dataset, the dataset will be re-published automatically to add new table and the data will be transferred to populate the table with data.

Note that temporary dataset should not be used for reports or dashboards development if you plan to use them in the future. Every time dataset gets republished, related data visualizations will be deleted.
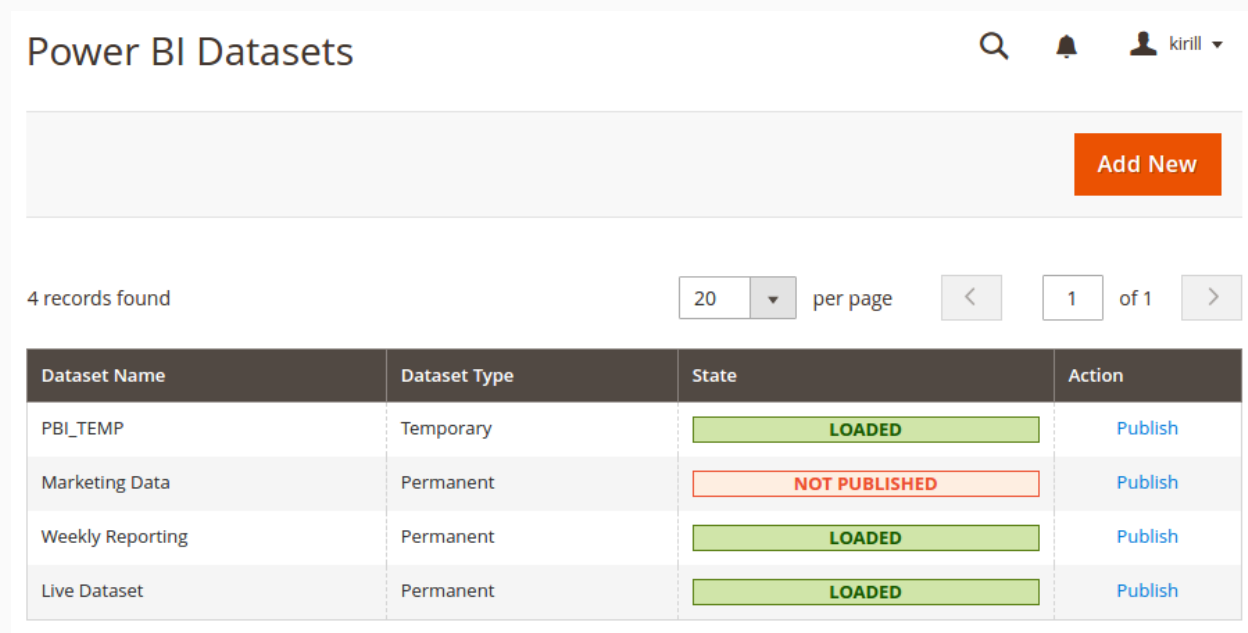

Image 3. Load Profile Editor

STEP 3: CREATION OF LOAD DATASET

In order to load data to Permanent Dataset, such dataset should be created. To create new Load Dataset, navigate to Reports -> Power BI -> Datasets and press "Add New" button. New dataset just requires a unique name.

Example list of Load Datasets you may see on the Image 4. Backend user can create only Permanent Datasets. New Dataset will get the status "Not Published".

Do not publish the dataset before adding Load Profiles to it. Normally you would want to publish the dataset only when you assigned all Load Profiles, you wish to have in the dataset since you can publish it only once.

The name that you assign to the dataset will be displayed in Power BI Service. If you will create new dataset with a name that conflicts with other dataset in Power BI, you may get unpredicted behavior of the integration since Magento system will expect structure of the Dataset to be equal between Power BI Dataset and Load Dataset in the integration extension.

Image 4. List of Load Datasets

STEP 4: CONFIGURING LOAD PROFILE

You may want to configure Load Profile before publishing it. You may do so in Load Profile Editor, example of which you was on Image 3.

Important configuration parameter here is the Dataset Name. Here you may select name of newly created on Step 3 dataset.

In addition, you may want to set different Table Name. This would be the table name that you will see inside Power BI Service.

See "Configuration" section of this user guide for more detail regarding the configuration of the Load Profile where you will find description of other options.

STEP 5: DATASET PUBLISHING

If you are ready with your Load Dataset and you have assigned all Load Profiles you want to use with this dataset, you may publish it.

Publishing of the dataset will create Power BI Dataset in Power BI Service with Tables associated with it.

Loading will be triggered automatically during publishing.

Power BI Data Load   🔍 🔔 👤 kirill ▾

Disable Auto Load   Disconnect from Power BI API   Reload all loaded Profiles

Connection with Power BI: Enabled
Automatic data actualization for current user: Enabled

Search   Reset Filter

Load ▾ Submit   13 records found   20 ▾ per page   ‹   1   of 1   ›

	Name	Dataset	Table	File Size	Preprocessed At	Loaded At	Load Mode	State	Action
☐	Orders2	Weekly Reporting	Orders2	279B	7/15/17, 5:06 PM	7/15/17, 5:07 PM	Manual	LOADED	Load
☐	Orders	Weekly Reporting	Orders	11B	7/15/17, 5:07 PM	7/15/17, 5:07 PM	Regular	LOADED	Load
☐	Invoiced	PBI_TEMP	Invoiced	134B	7/16/17, 4:14 AM	7/16/17, 4:14 AM	Live	TEMPORARY	Load
☐	Products in Cart	Live Dataset	ProductsinCart	351B	7/15/17, 5:05 PM	7/15/17, 5:05 PM	Manual	LOADED	Load
☐	Products in Cart (1)	Weekly Reporting	ProductsinCart_1	351B	7/15/17, 5:07 PM	7/15/17, 5:07 PM	Manual	LOADED	Load
☐	Abandoned Carts	Marketing Data	AbandonedCarts	225B	7/15/17, 4:58 PM	7/15/17, 4:58 PM	Live	NOT PUBLISHED	Load
☐	Abandoned Carts (2)	Live Dataset	AbandonedCarts_2	225B	7/15/17, 5:05 PM	7/15/17, 5:05 PM	Manual	LOADED	Load
☐	Tax	Weekly Reporting	Tax	111B	7/15/17, 5:07 PM	7/15/17, 5:07 PM	Manual	LOADED	Load
☐	Products Views	Marketing Data	ProductsViews	11B	7/13/17, 3:40 PM	7/13/17, 3:40 PM	Manual	NOT PUBLISHED	Load

Image 5. Load Profiles. Activation of automatic data actualization.

STEP 6: LOADING

If the assigned dataset was published before or if you want to update the data of a table in Power BI Service, you may trigger Loading of a particular Load Profile. This could be done from Load Profile Editor as it was described on Step 2, "Data Preview in Temporary Dataset".

STEP 7: ACTIVATION OF AUTO LOAD

Finally user may want their reports and dashboards in Power BI Service to be updated automatically with recent data.

In order to have scheduled updates working it has to be activated and configured in general configuration System -> Configuration -> Reports -> BIM Power BI Scheduler. In system configuration you setup how often the data updates should be triggered. More about configuration of scheduled updates you may read in "Configuration" section of this User Guide.

Loading of Load Profile will be triggered during regular data updates if all of the following conditions hold true:

1. User has activated Automatic data actualization. Activation is possible on the page with list of Load Profiles, example of this page you may see on Image 5 above. Button "Enable Auto Load" will enable Automatic data actualization.
2. Load Mode of the Load Profile is set to "Regular" or "Live". You may see this status in the column Load Mode on Image 5. You may change the mode in Load Profile editor. Depending on the selected mode, the dataset will be updated by one of the schedules. Live schedule is supposed to be triggered more often than the Regular one, but its frequency depends on configuration.
3. Load Profile should be Active. This status flag might be changed in Load Profile Editor.
4. If Load Profile is included in Permanent Dataset, this dataset has to be published. Publishing of the dataset is possible from Reports -> Power BI -> Datasets page.

SUPPORTED REPORTS

This section describes what data can be sent to Power BI Service with current version of the Power BI Integration extension.

As mentioned earlier, Load Profiles are created based on tables or grids in Magneto backend. Most of them are taken from Magento Reporting pages. The Power BI Integration is supported for the grid if there is an option of exporting to Power BI available in the list of possible exports as it is shown on Image 2.

The Magento Reports that can be exported to Power BI Service at the moment are the following:

1. Total Ordered Report, *Reports -> Sales -> Orders*
2. Order Taxes Report Grouped by Tax Rate, *Reports -> Sales -> Tax*
3. Total Invoiced vs. Paid Report, *Reports -> Sales -> Invoiced*
4. Total Shipped Report, *Reports -> Sales -> Shipping*
5. Total Refunded Report, *Reports -> Sales -> Refund*
6. Coupons Usage Report, *Reports -> Sales -> Coupons*
7. Products in carts, *Reports -> Shopping Cart -> Products in carts*
8. Abandoned carts, *Reports -> Shopping Cart -> Abandoned carts*
9. Products Bestsellers Report, *Reports -> Products -> Bestsellers*
10. Products Ordered, *Reports -> Products -> Products Ordered*
11. Most Viewed (Products) , *Reports -> Products -> Most Viewed*
12. Low stock (Products) , *Reports -> Products -> Low stock*
13. Downloads, *Reports -> Products -> Downloads*
14. New Accounts, *Reports -> Customers -> New Accounts*
15. Customers by Orders Total, *Reports -> Customers -> Customers by orders total*
16. Customers by number of orders, *Reports -> Customers -> Customers by number of orders*
17. Customers Reviews, *Reports -> Reviews -> Customers Reviews*
18. Products Reviews, *Reports -> Reviews -> Products Reviews*
19. Search Terms, *Reports -> Search Terms*

Additionally creation of Power BI Load Profile is possible based on non-reporting lists (only one list is connected in current version):

Orders List, *Sales -> Orders*

NOTES ABOUT ADDING MORE DATA

The Power BI Integration extension might be used to transfer more data than it is supported at the moment. Current architecture supports creation of modules dependent on Power BI Integration that will allow connecting more data.

Data may come not only from backend lists, but it could also be an SQL query. From our perspective, it seems to be more reasonable to create first new custom report page with a grid that will render the required data and then connect this grid to Power BI Integration extension.

Please, contact the extension provider if you are interested in extending the functionality.
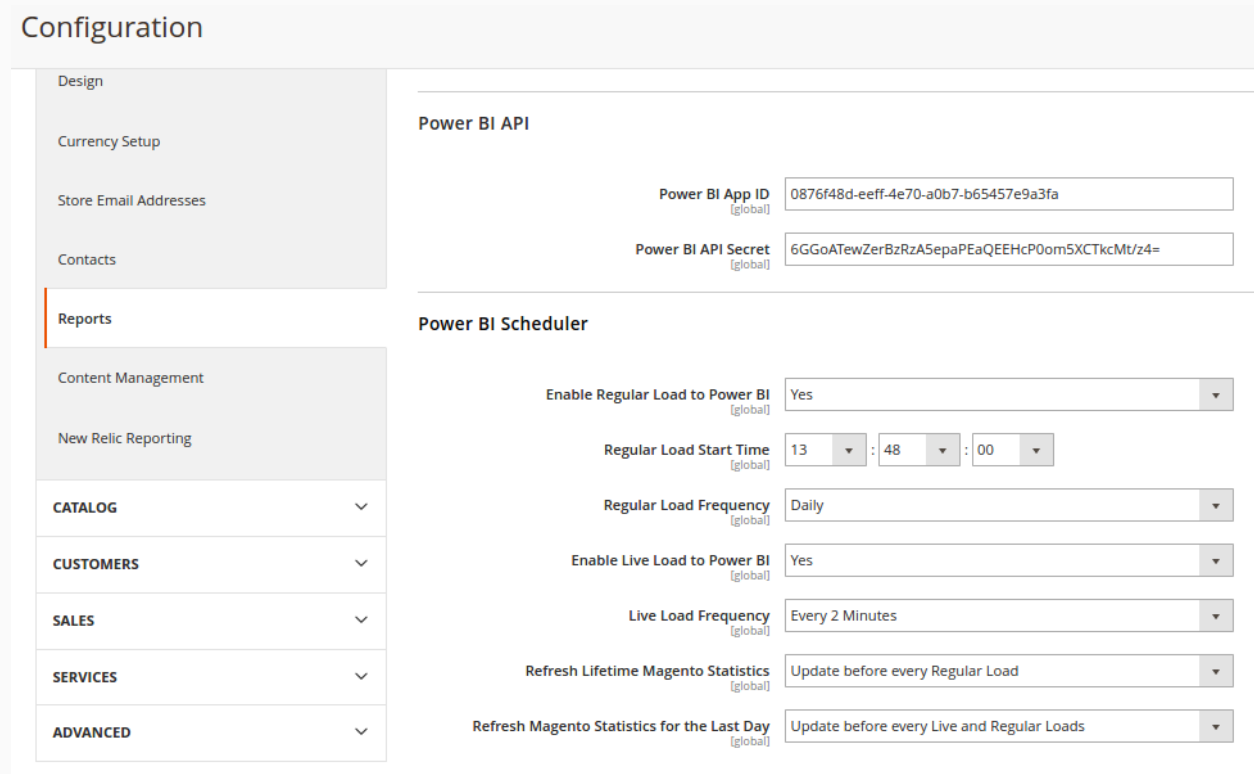
CONFIGURATION

This section covers System Configuration parameters related to Power BI Integration extension and also touches configuration of some parameters of Load Profiles that allow more dynamic control on subset of retrieved data.

After being installed the Power BI Integration extension needs to be configured. First and the most important configuration is access to Power BI Rest API.

Configuration section for Power BI Integration could be found by navigating to *System -> Configuration* menu.

Under *General -> Reports* tab you may find "BIM Power BI" and "BIM Power BI Scheduler" sections. Example of the configuration you may see on Image 6 below.

Image 6. Power BI Integration configuration.

POWER BI APP REGISTRATION

Power BI extension is integrated with Azure Active Directory (Azure AD) to provide secure sign in and authorization for your Magento store. Before you start using Power BI extension you need an Azure Active Directory tenant, an organizational user, and a Power BI service account. All users, who will be using Power BI from your Magento store will have to be added to your Azure Active Directory tenant. The extension uses a Client ID to identify itself to Azure AD. You also need a client secret key. See following resources to learn how to register and authenticate a Power BI extension: Register a client app, Authenticate a Power BI client app, Register a web app and Authenticate a Power BI web app. Obtained Client ID and client secret key might be provided to the Power BI extension on Magento configuration page.

Setting up Power BI API might take some time, especially if your organization were not using Active Directory before. From our side the only information you may need to know is the Redirect URI.

Please, use the following value as Redirect URI: *https://your-website-domain.com/**

For the rest of information, please, refer to the Microsoft documentation as main source of the information.

As additional support, for a quick reference, we list the steps you may need to perform in order to obtain a Client ID and a client secret key for Power BI API.

STEP 1: CREATE AN AZURE ACCOUNT

You should be able to create an Azure account for free. You pay only for Services you are using or for an optional support subscription.

Here is the URL where you may start with registration of an account:

https://azure.microsoft.com/en-in/free/

STEP 2: CREATE AN AZURE ACTIVE DIRECTORY TENANT

Please, follow the instructions from Power BI documentation in order to create an Azure AD tenant:

https://powerbi.microsoft.com/en-us/documentation/powerbi-developer-create-an-azure-active-directory-tenant/

Please, note that you do not need to activate Power BI Pro account since you do not have to use Power BI embedded in order to setup the API.

STEP 3: CREATE SOME USERS IN YOUR AZURE ACTIVE DIRECTORY TENANT

You may want to create some users if you did not do that before. Each Magento backend user who will use Power BI may need to have separate Power BI user.

STEP 4. REGISTER POWER BI APPLICATION

You may use new Power BI App Registration Tool for that or using Azure portal.

https://powerbi.microsoft.com/en-us/documentation/powerbi-developer-register-app/

Follow instructions related to *Server-side Web app* or *Web Applications* instead of registering a Native app.

Use Redirect URL as your domain name (of the backend system) with asterisks in the end.

Redirect URI: *https://your-website-domain.com/*

STEP 5: GET CLIENT ID AND SECRET KEY

Please, refer to the "How to get the client id" section of the documentation to get the Client ID:

https://powerbi.microsoft.com/en-us/documentation/powerbi-developer-register-app/#how-to-get-the-client-id

To get the client secret key, you may need just few additional steps to be performed on the Azure portal: after you have selected the application, click on *Keys* menu under API ACCESS section and add new key there.

STEP 6: ACTIVATE POWER BI (FREE) SUBSCRIPTION

This step is optional.

You may need a subscription added within Office 365. In such case, you may use the flowing documentation page:

 https://powerbi.microsoft.com/en-us/documentation/powerbi-admin-powerbi-free-in-your-organization/

As you may see on the Image 6, there is a separate section called "*BIM Power BI Scheduler*" in System Configuration.

This section allows users to configure:

1. Regular Load to Power BI.
2. Live Load to Power BI.
3. Lifetime Magento Statistics Refreshing
4. Refreshing of Magento Statistics for the Last Day

Below you will find details about each of the listed items.

REGULAR LOAD TO POWER BI

Regular load is a scheduled data update performed on a daily, weekly or monthly basis. To enable this kind of update, set "*Enable Regular Load to Power BI*" to "*Yes*".

Once the Regular Load option is enabled, you can set Regular Load Frequency and when to start the loading of the data. Usually users may want to set this time to a period with lowest activity, at night.

With Regular Load you may update Load Profiles what contain statistics for long periods of time such as months or years.

LIVE LOAD TO POWER BI

Live load is a scheduled data update performed several times within the day. To enable this kind of update, set "*Enable Live Load to Power BI*" to "*Yes*".

Once the Live Load option is enabled, you can set Live Load Frequency. Allowed options are periods of 1 minute, 2, 3, 5, 10, 15, 30, 60 or 120 minutes.

LIFETIME MAGENTO STATISTICS REFRESHING

By setting the option "*Refresh Lifetime Magento Statistics*" you may activate automated actualization of the Magento Statistics. It is especially important if your Load Profiles relay data from on Magento Reports.

Available options are:

- *No scheduled statistics updates.*
  Magento statistics will not be refreshed.

- *Update before every Live and Regular Loads.*
  Refresh of Lifetime Magento will be triggered before every Live and Regular Loads.

- *Update before every Regular Load.*
  This is the recommended option.

- *Update by separate schedule.*
  This option allows to enter additional parameter to setup Cron Expression for updating Lifetime statistics independently from loading of the data into Power BI Service.

REFRESHING OF MAGENTO STATISTICS FOR THE LAST DAY

This option "*Refresh Magento Statistics for the Last Day*" is similar to previous one, but it will trigger only refresh of the statistics for the last day. This option is more preferable to be used in connection with Live data updates.

The options are the same as for "*Refresh Lifetime Magento Statistics*". But in this case the recommended option would be

- *Update before every Live and Regular Loads.*

LIMITATIONS

There are certain limitations associated with Power BI API that put some boundaries on the usage of the extension.

- Maximum of 75 columns in a table

- Maximum of 75 tables (Load Profiles) in a dataset

- Maximum of 10 000 rows per table (this limit may increase in future releases of the extension)

- 10 000 rows added per hour per dataset (with pro service plan: 1 000 000 rows added per hour per dataset)

- Maximum of 5 pending POST rows requests per dataset

- 120 POST rows requests per minute per dataset

- 4 000 characters per value for string column in POST rows operation

See the Microsoft documentation for the recent list of limitations:

https://msdn.microsoft.com/en-us/library/dn950053.aspx

Additionally users may face limited possibilities to share reports and dashboards between colleagues. More sharing options may be available for Power BI Pro users.

USAGE EXAMPLES

This section is intended to help users to easier gain most value of the integration with Power BI Service.

Below you will find some useful references and some notes on how to use features of Power BI Service that will be available to Magneto users with the integration: reports creation, dashboards, access through the mobile App and usage of notifications.

This section may grow with future versions of the User Guide.

POWER BI REPORTS

Please, refer to the Power BI documentation for example of reports creation process

https://powerbi.microsoft.com/en-us/documentation/powerbi-service-create-a-new-report/#add-a-radial-gauge-to-the-report

POWER BI DASHBOARDS

Please, refer to the Power BI documentation for the information on how you can create dashboards:

https://powerbi.microsoft.com/en-us/documentation/powerbi-service-create-a-dashboard/

POWER BI MOBILE APP

Please, use the following page to find the information about installing a Mobile App on your device:

https://powerbi.microsoft.com/en-us/mobile/

After logging in with your Power BI user's credentials you will be able to access Reports and Dashboards you have created in Power BI Service.

Note, that you may also control look of the reports in mobile app by switching the editor view in Power BI Service.

NOTIFICATIONS

Setting up notifications may appear a bit tricky since this functionality does not seem to be very intuitive at the beginning. To setup notifications, you need to have a dashboard with a tile that supports notifications. Simple example is a KPI. So, before setting up notifications, you need to create a dashboard. Please, refer to the documentation of Power BI for more information:

https://powerbi.microsoft.com/en-us/documentation/powerbi-service-set-data-alerts/

FINAL NOTES

ENTERPRISE FEATURES

In 2017 Microsoft announced some changes in their subscription policies. Power BI Embedded functionality was moved to Premium package.

Users of Power BI Premium package may activate an extra module called Bim_PbiPremium to which will add three additional pages in the backend:

- Reports -> Power BI -> Dashboards
- Reports -> Power BI -> Reports
- Reports -> Power BI -> Tiles

On the listed pages users may open their dashboards and reports that were created in Power BI Online service. Such Dashboards and Reports may also contain data from other sources such as Google Analytics, Salesforce, CRM platforms and lot of other sources that may be connected to Power BI.

Please, note that this part of functionality is at Beta version since the REST API that support this functionality is at Beta level and may change in the future.

To activate the package, simply edit the file *app/etc/modules/Bim_PbiPremium.xml* and set the module *active* status to "true".

ROADMAP

Power BI product prom Microsoft is relatively young product but very promising. Almost every month they publish their release notes.

The extension integrates Magento store with Power BI Online offering and the more functionality Microsoft provides, the more Magento store owners can benefit from the integration.

One of the main areas that will allow enhancing the extension in the future is the API. Some of the great improvements such as specifying relations between datasets are already announced and we are working on extending the functionality based on those improvements.

EXTENDING THE FUNCTIONALITY

The license of the extension allows store owners to develop additional functionality by extending the base functionality provided with the module.

One of such uses is adding of datasets. Each store owner may have different strategy and may want to concentrate on specific data, not covered by current set of data delivered to Power BI Online. This would require creation of custom data grid or just a php class that will play role of a data provider. Such provider might be connected to the extension to deliver the data to Power BI. Please, contact extension provider to request development documentation and IT support that may also include professional services to help you deliver the data you want.

Another use case might be embedding of the reports, dashboards or just separate tiles into other areas of Magento. For example, one might want to add some KPIs or charts on main dashboard of Magento backend. This will be possible by using part of the functionality of Bim_PbiPremium module. Please note, that this part of functionality requires premium subscription to Power BI service. Also, each Magento backend user should use its own user account to setup a connection with Power BI API.

Please note also that redistribution of the extension or its modifications are not allowed without agreement with the extension provider.

ABOUT THE AUTHOR

After getting education degrees first in IT, later in Management, I concentrated on statistical analysis with accent on financial data.

I got my education in Ukraine and was working in Kiev as IT Consultant first on local market and later for European multinational companies. After relocation to Germany, I was working for several years as Developer specialized on Magento platform within a Gold Solution Magento Partner company located in Karlsruhe. During this time I got certifications "Magento Certified Developer Plus" and "Magento Certified Frontend Developer".

Currently I am working as Business Intelligence Developer in Aviation industry and continue to work on data analytics extensions for Magento platform.