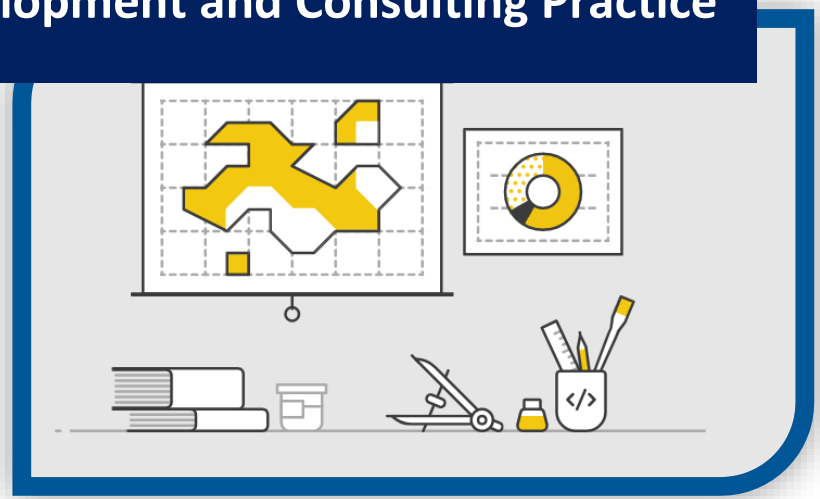
AppSource Power BI Visuals - Customization, Development and Consulting Practice



Cognitive Convergence
http://www.cognitiveconvergence.com
+1 4242530744
shahzad@cognitiveconvergence.com

Cognitive Convergence is Subject Matter Expert in Office 365, Dynamics 365, SharePoint, Project Server, Power Platform: Power Apps-Power BI-Power Automate-Power Virtual Agents.
We offer Power BI consulting services covering solution architecture refinement, customization, integration, transformation, visualization and analytics to uncover insights hidden within data and enhance data exploration.

Contents

OBJECTIVES

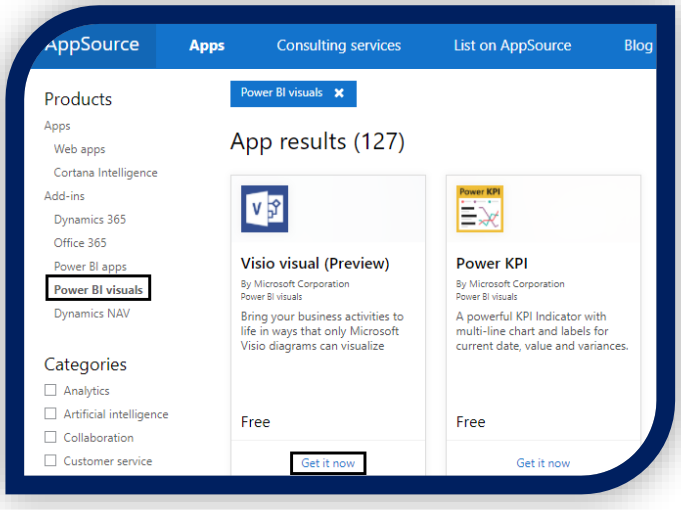
This paper will discuss different visuals of Power BI, available on AppSource. A brief introduction along with their usage will discuss here.

BACKGROUND

Power BI comes with many out-of-the box Power BI visuals. These visuals are available in the visualization pane of both Power BI Desktop and Power BI service, and can be downloaded from AppSource, these visuals can be used for creating and editing Power BI content. These visuals are needed to meet the advance and complex requirements of data representation. Custom visuals available in AppSource and within the integrated custom visuals store for Power BI Desktop are all approved for running in browsers and on mobile devices via the Power BI mobile apps. A subset of these visuals have been certified by Microsoft and support additional Power BI features such as email subscriptions and export to PowerPoint. Additionally, certified custom visuals have met a set of code requirements and have passed strict security tests.

APPSOURCE POWER BI VISUALS

Microsoft and community members contribute Power BI visuals for public benefit, and publish them to the AppSource. You can download these visuals and add them to your Power BI reports. Microsoft has tested and approved these Power BI visuals for functionality and quality.



What is AppSource?

AppSource is the place for apps, add-ins, and extensions for your Microsoft software. AppSource connects millions of users of products such as Microsoft 365, Azure, Dynamics 365, Cortana, and Power BI, to solutions that help them get work done more efficiently and insightfully than before.

Certified Power BI visuals

Certified Power BI visuals are visuals in AppSource that meet certain specified code requirements that Microsoft Power BI team has tested and approved. The tests are designed to check that the visual doesn't access external services or resources.

ADDING A CUSTOM VISUAL

Custom visual can be added in Power BI either by direct import from store, or download .pbiviz file from AppSource and import in Power BI as file.

AppSource

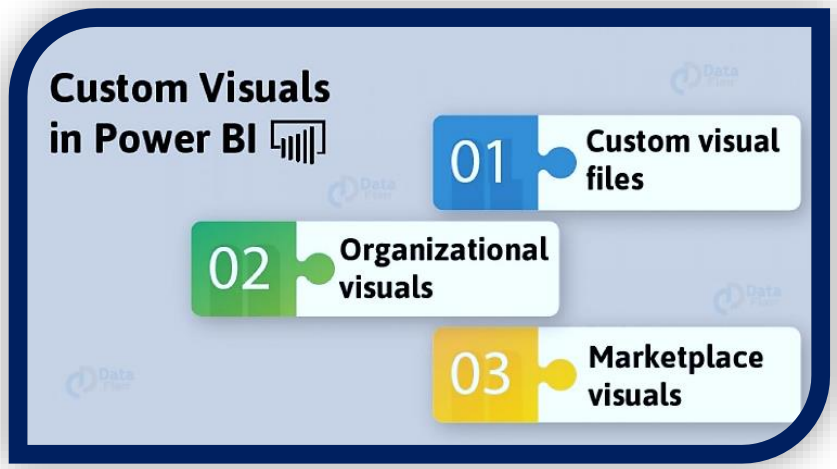
Custom visuals can be added to Power BI reports by either downloading .pbiviz files from Microsoft AppSource or via the integrated Office Store of custom visuals in Power BI Desktop. Utilizing AppSource requires the additional step of downloading the file; however, it can be more difficult to find the appropriate visual as the visuals are not categorized. However, AppSource provides a link to download a sample Power BI report (.pbix file) to learn how the visual is used, such as how it uses field inputs and formatting options. Additionally, AppSource includes a short video tutorial on building report visualizations with the custom visual.

Import from File

To add custom visuals directly to Power BI reports, click the Import from store option via the ellipsis of the Visualizations pane. If a custom visual (.pbiviz file) has been downloaded from AppSource, the Import from file option can be used to import this custom visual to the report. Additionally, both the Import from store and Import from file options are available as icons on the Home tab of the Report view in Power BI Desktop.

Import from Store

Selecting Import from store launches an MS Office Store window of Power BI Custom Visuals. Unlike AppSource, the visuals are assigned to categories such as KPIs, Maps, and Advanced Analytics, making it easy to browse and compare related visuals. More importantly, utilizing the integrated Custom Visuals store avoids the need to manage .pbiviz files and allows report authors to remain focused on report development.



Import from My organization

Organizations can also upload custom visuals to the Power BI service via the organization visuals page of the Power BI Admin portal. Once uploaded, these visuals are exposed to report authors in the MY ORGANIZATION tab of the custom visuals MARKETPLACE. This feature can help both organizations and report authors simplify their use of custom visuals by defining and exposing a particular set of approved custom visuals. The list of organizational custom visuals could potentially only include a subset of the visuals which have been certified by Microsoft. Alternatively, an approval process could be implemented so that the use case for a custom visual would have to be proven or validated prior to adding this visual to the list of organizational custom visuals.

DOWNLOADING AND IMPORTING CUSTOM VISUALS FROM MICROSOFT APPSOURCE

To download and import custom visual from AppSource in Power BI following steps are required.

- Open Microsoft AppSource website.
- Select the Apps tab from the bar on the top.
- Select Power BI Visuals options.
- The page will show more than 200 results of Power BI visuals. Browse the list of available custom visuals.
- Select the visual needed for report. Click on Get it now. Click on Continue in the next window.
- Click on Download for Power BI.

BASIC IDEAS OF DEVELOPMENT

Developers can create custom visual for Power BI in order to meet their complex requirements and data needs. They can import these visual in Power BI. And can also upload that in AppSource after getting certificate. These visuals are needed to meet the advance and complex requirements of data representation. It also allows one to embed the visualizations in any web application. By embedding the visualizations in a web page, developers can offer a seamless experience to the end users where the users can drill-down through the visualized data without knowing anything about Power BI. Power BI visuals are packages that include code for rendering the data served to them. Anyone can create a custom visual and package it as a single .pbiviz file that can then be imported into a Power BI report.

DEVELOPMENT STEPS

Solution architecture for these AppSource visuals includes TypeScript and SDK implementation. It includes

- Installing TypeScript
- Downloading pbiviz for creating visualization and also pbiviz certificates.
- Adding visual elements to custom visual like SDK D3.js, core-js.
- Installing Power BI Custom Visuals API, "powerbi-visual-api".
- Writing code for developing custom visual.
- Open Power BI Service with admin panel.
- Select the Developer Visual from the Visualizations pane in Power BI service.
- The developed visual will import automatically.

SOLUTION ARCHITECTURE

Power BI custom visual have following solution architecture which is created by creating a project in Visual studio by typing the below command in Power shell.

```
pbiviz new <visual project name>
```

Project created by this command have following structure

- Project
  - .vscode
    - launch.json
    - settings.json
  - assets
    - icon.png
  - node_modules
  - src
    - settings.ts
    - visual.ts
  - style
    - visual.less
  - capabilities.json
  - package-lock.json
  - package.json
  - pbiviz.json
  - tsconfig.json
  - tslint.json

- **.vscode:** This folder contains the VS code project settings.
- **Assets:** This folder contains the icon.png file. The Power BI visuals tool uses this file as the new Power BI visual icon in the Power BI visualization pane.
- **Src:** This folder contains the visual's source code. In this folder, the Power BI visuals tool creates the following files:
  - **visual.ts:** The visual's main source code.
  - **settings.ts:** The code of the visual's settings. The classes in the file provide an interface for defining finance matrix's properties
  - **Style:** This folder contains the visual.less file, which holds the matrix styles.
- **capabilities.json:** This file contains the main properties and settings for the visual. It allows the visual to declare supported features, objects, properties, and data view mapping.
- **package-lock.json:** This file is automatically generated for any operations where npm modifies either the node_modules tree, or the package.json file.
- **package.json*:*** This file describes the project package. It contains information about the project such as authors, description, and project dependencies.
- **pbiviz.json:** *This file contains the visual metadata.*
- **tsconfig.json:** A configuration file for Typescript. This file must contain the path to *.ts file where the main class of the visual is located, as specified in the visualClassName property in the pbiviz.json file.
- **tslint.json:** This file contains the TSLint configuration.

IVisual

Represents a visualization displayed within an application (PowerBI dashboards, ad-hoc reporting, etc.). This interface does not make assumptions about the underlying JS/HTML constructs the visual uses to render itself. Update() and enumerateObjectInstance() are its methods. The visual class name matches to the define class name in the pbiviz.json file. The following methods implement in it

- "constructor", a standard constructor to initialize the visual's state
- "update", to update the visual's data
- "enumerateObjectInstances", to return objects to populate the property pane (formatting options) where you can modify them as needed.

```
class MyVisual implements IVisual {
        Constructor(options: VisualConstructorOptions) {
         }
         public update(options: VisualUpdateOptions): void {
          }
          public enumerateObjectInstances(options: EnumerateVisualObjectInstancesOptions):
    VisualObjectInstanceEnumeration {
     }
}
```

DataView

The "DataViewUtils" is a set of functions and classes to simplify parsing of the DataView object for Power BI visuals. The DataView is the structure that Power BI provides to your visual, which contains the queried data to be visualized. However, DataView can provide data in different forms, such as categorical and tabular. To build a categorical visual like a bar chart, you only need to use the categorical property on the DataView. Defining visualTransform lets you convert DataView into a view model your visual will use. A "dataViewMappings" describes how the data roles relate to each other and allows you to specify conditional requirements for them. There is a section for each of the "dataMappings". Power BI creates hierarchical data structure. The root of tree includes the data from the first column of Category data role with children from the second column of data role**.**

```
"dataViewMappings": [
          {
              "conditions": [ ... ],
           "categorical": { ... },
           "single": { ... },
          "table": { ... },
            "matrix": { ... }
        }
       ]
```

Static objects

You can add objects to the Property pane to further customize the visual. These customizations can be user interface changes, or changes related to the data that was queried. The sample uses static objects to render the X-axis for the bar chart. These objects can toggle on or off in the Property pane of any visual. Objects property s defined inside capabilities.json file for objects to display in the Property pane.

- **enableAxis** is the internal name that the dataView references.
- **displayName** is the name shown on the Property pane.
- **bool** is a primitive value that is typically used with static objects, such as text boxes or switches.
- **show** is a special property on properties that enables the show switch on the object. Since show is a switch, it is typed as a bool.

```
"objects": {
    "enableAxis": {
        "displayName": "Enable Axis",
        "properties": {
            "show": {
                "displayName": "Enable Axis",
                "type": { "bool": true }
            }
        }
    }
}
```

ObjectEnumerationUtility

Object property values are available as metadata on the dataView, but there's no service to help retrieve these properties. ObjectEnumerationUtility is a set of static functions you can use to retrieve object values from the dataView, and for other visual projects. The ObjectEnumerationUtility is optional, but is great for iterating through the dataView to retrieve object properties.

```
export function getValue<T>(objects: DataViewObjects, objectName: string, propertyName: string, defaultValue: T): T {
    if (objects) {
        let object = objects[objectName];
        if (object) {
            let property: T = <T>object[propertyName];
            if (property !== undefined) {
                return property;
            }
        }
    }
    return defaultValue;
}
```

RISK ANALYSIS

Developing own custom visuals in Power BI is good approach to get all the requirements fulfilled with even minor compromising, but building visual entirely could be a challenging task. Following could be the risks and there purposed solution.

Risk	Purposed Solution
limitation of formatting API support	Adavance level CSS skills will be used to avoid issues with UI formating
Performance of visual in millions of records	Special customization will be done in matrix to handle performance issue in matrix for millions of records
Inter visual communication can be challenging	Curently, visual SDK does not support inter visual communication but special configuration and customization will be done to make sure minimal possible support is provided.

CONCLUSION

In this paper different visuals of Power BI, available on AppSource is discussed. A brief introduction along with their usage and solution architecture and development steps is discussed.

Cognitive Convergence will provide consulting services that help in designing, deploying, managing, enhancing or troubleshooting on-premises, cloud-based or hybrid Power BI environment. Cognitive Convergence will provide help to start fresh with Power BI to modernize current business analytics solution or revamp existing Power BI deployment by incorporating new data sources or adding new services.

Contact Us
Cognitive Convergence
http://www.cognitiveconvergence.com
+1 4242530744
shahzad@cognitiveconvergence.com