

POWER BI CONNECTORS FOR ENHANCED DATA INTEGRATION

Kremena Stefanova, Veselina Naneva

Abstract. *Microsoft Power BI is one of the most recognizable platforms for processing unrelated sources of information into meaningful data structures and visual insights analyzed by the business. It maintains three separate methods for data connections and the choice of an appropriate one is crucial to ensure the balance between data size and report performance. Although a single data source can support multiple connection types, there are predefined specifics for each method which the developer should consider about data access approaches. The paper will illustrate applications of the corresponding data connectors based on different case studies and examples. The formed inference will provide principals for using the correct connector for enhanced data integration, taking into account the database complexity and the report dependencies.*

Key words: Data Connectors, Power BI, Enhanced Data Integration.

1. Introduction

Business Intelligence (BI) is a complex sphere for analyzation and an informative data visualization which satisfies certain enterprise requirements. It indicates all activities which help to collect, store and summarize data produced by a company. As part of the BI process, organizations collect data from internal IT tools and external sources, prepare it for investigation, run queries on the data and create data visualizations into complete reports to make the ready results available to corporate users for operational decision-making and strategic planning [1].

BI primarily improves the timeliness and quality of information and enables managers to better understand the position of their company [2]. Its applications and technologies assist businesses to track changing trends in market share; switches in customer behavior and spending patterns, capabilities; and market conditions. There are plenty of BI tools such as Power BI, Tableau, Qlick Sence and others, providing almost the same opportunities to the end user. The most popular is Power BI, which is a

data visualization and reporting platform, used by interested ones every day. In addition to the regular functionalities, in the perspective of the access, Power BI allows the corresponding developer to select the connecting mode of the source system and this has impact on data model that is going to be created. Due to the differences of their behavior, it is necessary to follow certain rules for each data connection approach and it is crucial the specifics and application of the data connectors to be considered.

2. Power BI data connectors and their application

Power BI maintains diverse data sources but the connection type determines how the relation with them will be established and offer varied methods of connectivity. A single one can support multiple connection types. For instance, linking to a SQL Server database can be done through Import Data or DirectQuery modes, thus Power BI interacts with data and generates responses to visualization queries since every informative graphic element within a report relies on a dataset. Although other means also existing, we will compare these two due to the fact they are the commonly preferred in Power BI.

DirectQuery Connector

DirectQuery Connector allows us to query the underlying data source in real time instead of importing it into Power BI's local storage [3]. Thereby reports always reflect the most recent state of the data as the actual snapshot is read. When an appropriate visualization is chosen, the corresponding source must be inquired and taking into account the performance, the time for refresh differs. Also, DirectQuery observes the centralized data principles, i.e. ensures data remains in its original repository. As a result, it reduces the risk of duplications and improves compliance with organizational security protocols because sensitive information does not leave the root source system [4]. As queries are sent directly to the server or other source type, each time a report or visualization is requested, any changes made in the database will be immediately visible in Power BI reports without demanding further actions from users. Another benefit of DirectQuery is that it can handle large datasets more efficiently than other modes since all data should not be loaded into memory before analysis. In this case when dealing with very huge collection of resources, performance is faster and scalability is better.

However, a limitation can be found for calculated columns and mea-

asures because they cannot be applicable when working with direct queries [5, 6]. By dint of these, operations require access to the imported data, stored within Power BI's internal memory. Even in consideration with such mode, the ability to retrieve table views has been removed. This triggers an extra time to process the calculation and has impact on the speed of data determination. Furthermore, DirectQuery mode is not suitable for complex models with multiple joins and aggregations as it can provoke high resource usage on the source system's server. Theoretically it can output an ambiguity in the data model.

Import Connector

Import mode is a connector which refers to adding data into the Power BI storage and is equivalent to a copy and snapshot, so when interacting with reports, users can communicate directly with Power BI data cache, as the software copies the data. This causes a must to refresh the dataset to fetch changes from the underlying database. With the Import method, data is embedded into Power BI Desktop and Service tools. It is proper when dealing with smaller data sets or infrequently updating one [7]. Users can create complex models and advanced calculations with Data Analysis Expressions (DAX) language, and build reports about the imported data.

The core feature of Import mode is that it significantly enhances the performance of queries and interactions, as the system does not depend on external databases for real-time execution [8]. By storing data in memory, Power BI can execute complex calculations, aggregations, and transformations with minimal latency, even for slightly larger datasets. Another notable benefit of the considered mode is the ability to handle data modeling and transformation within Power BI itself. Users can generate calculated columns, measures and relationships between tables and in addition they can be administered from the table view in the tool.

While Import mode offers numerous advantages, there are several negative limitations. A primary disadvantage is the dependency on memory resources [9]. The size of the added dataset is constrained by the available memory on the system hosting Power BI. Most often it is up to 1 GB. This makes the chosen mode less suitable for extremely large datasets that exceed the capacity of the local hardware. The problem can be reduced with a larger cloud storage but even then, the report's performance can be corrupted. Also, working with relevant data demands manual or scheduled

refreshes, which can introduce latency in highly dynamic environments.

3. Comparison application examples

In order to compare DirectQuery and Import connectivity modes and to derive some conclusions about their impacts, we create a server and database in the Azure cloud service provider [10]. The connection string refers to:

```
Server=tcp:shopsample.database.windows.net,1433;
InitialCatalog=shopsample;
Encrypt=True;
TrustServerCertificate=False;
ConnectionTimeout=30;
Authentication="Active DirectoryDefault";
```

Furthermore, since we are accessing internal service resource, we should manage the required Firewall rules based on the corresponding IPv4 client address. The sample database is AdventureWorks with size of 13.6 MB and has 876 rows in the biggest table. It includes information for a store, split by products, customer and order data.

Each connecting mode can be set before the actual access to the database concerning a data schema (see Figure 1). For the purpose of the research, we will compose two dashboards with the separate comparable modes.

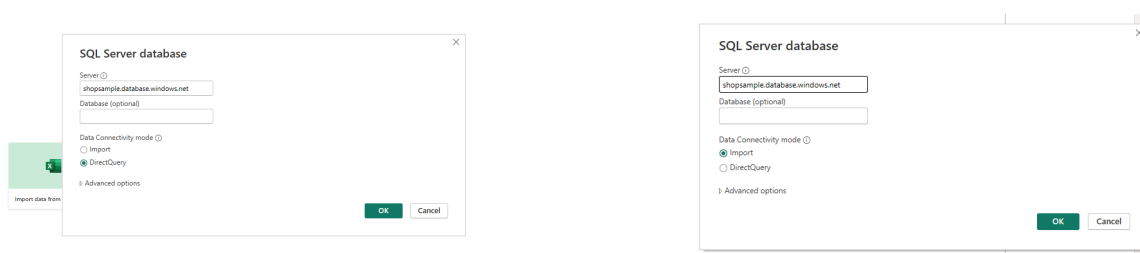


Figure 1. Set up of connectivity modes

After model has been determined, the first contrast can be found in auto detected and established relationships. Both has 10 tables and 11 relationships in a star schema, but from the perspective of Import Connector, there is an inactive relationship (see Figure 2). Although in such case data can be extracted through the active ones, in situations where all tables should be actively related, the mode of DirectQuery is more applicable.

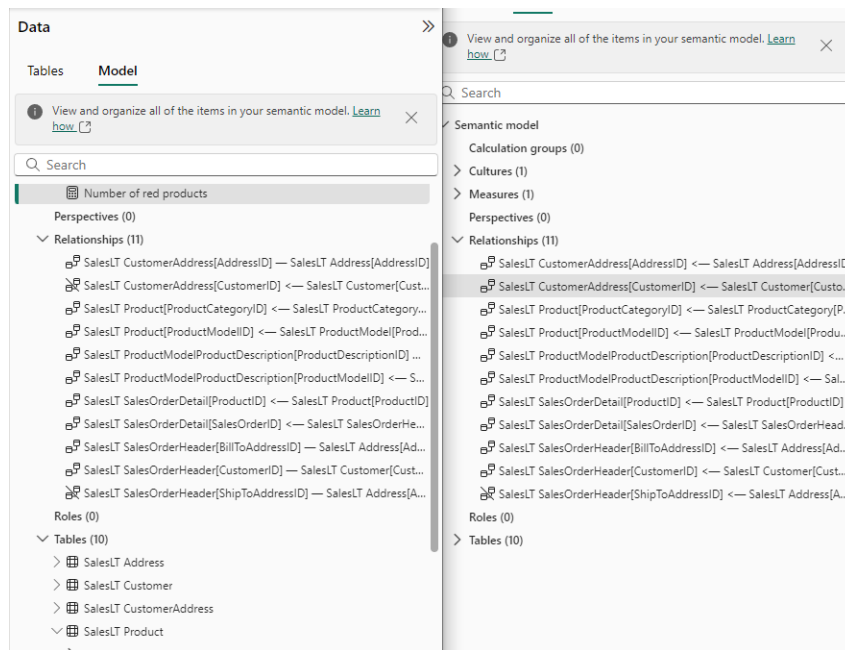


Figure 2. Difference in data model

The visualizations in Power BI can accept as values a mixture of regular fields from the database and new calculated with DAX formulas. We consider using equal data fields to create stacked column charts in the samples' dashboards. The visuals' presence speed is evaluated by a performance analyzer as internal tool in Power BI (see Figure 3 and Figure 4). It outputs the result that if we select the DirectQuery mode, we will need double *ms* time for the same data representation. Moreover, due to the difference in the relationship mentioned above, it can be slightly misleading in data provided. A discrepancy in the aggregation is illustrated in X-axes of the chart, thus an additional action in the slower mode is required.

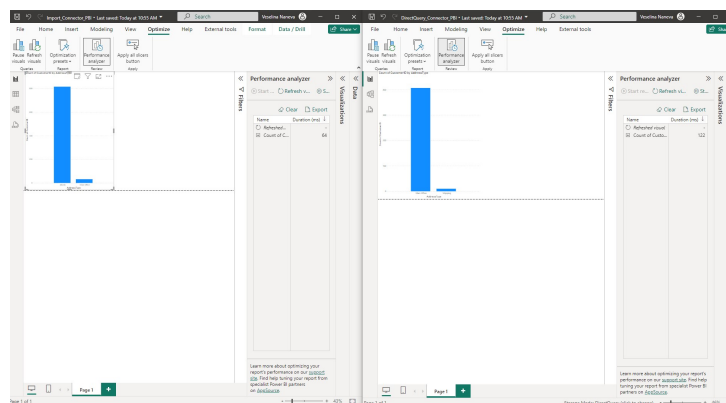


Figure 3. Difference in the speed of data loading and values

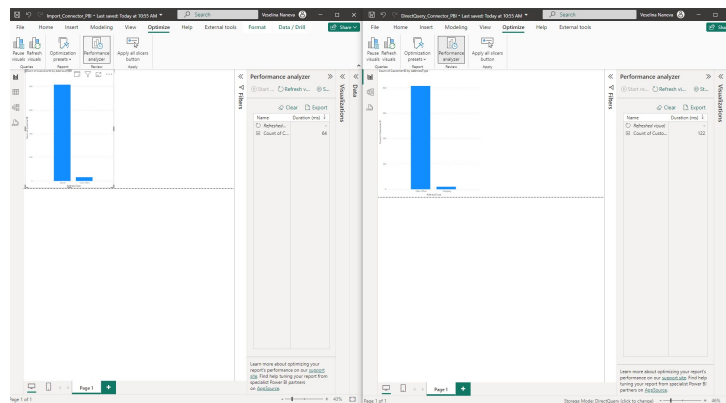


Figure 4. Difference in the speed of data loading and values

As a third example for dissimilarity is taken a measure's state by the following simple DAX calculation:

```
Number of red products =
CALCULATE(COUNT('SalesLT Product'[ProductID]),
FILTER('SalesLT Product', 'SalesLT Product'[Color] = "Red")).
```

It counts the number of products with a certain criterion and it is set as a field for a card visual. On Figure 5 can easily be seen the performance analyzer for compiling it in the two versions. We can deduce that when we rely on DirectQuery mode, an extra step appears for processing the relationship with the server which doubles the other operations time. Since this is a sample example, in more complex reports such characteristic may significantly impact the visuals' loading in a page.

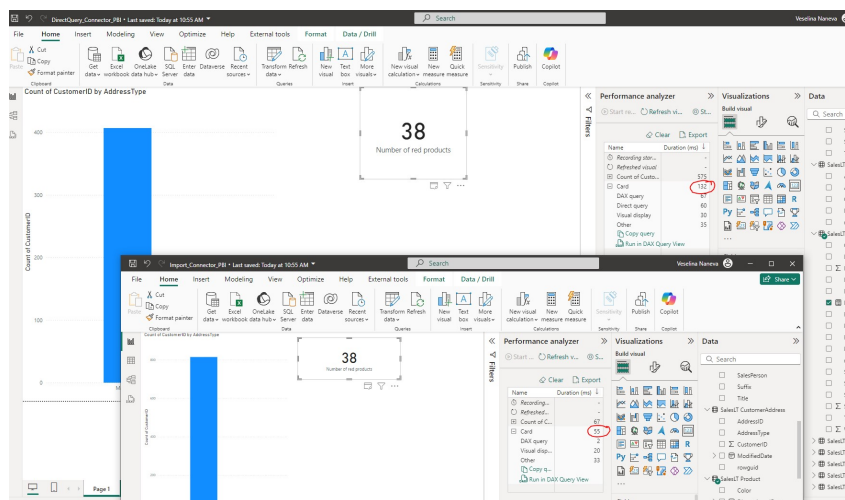


Figure 5. Difference in the speed of DAX measure calculation

4. Conclusion

Depending on the database used and the connector mode difference reviewed, several conclusions can be derived which may be applicable in case of similar information produced. If there is no need for real-time tracking and scheduled data refresh can be implemented, the appropriate connector mode is Import from performance perspective. Furthermore, the same inference can be made if there is a necessity of data manipulation, because choosing DirectQuery, the developer has no access to the stored dataset and Table view of Power BI. Despite that, if a business requires multiple, even hourly, information updates, DirectQuery is the best option due to limitless update queries. Moreover, if the database is with a bigger size, Import cannot cope with such data volume.

Power BI developers should always scale the pros and cons of a certain connector mode taking into consideration the root obligation. The main approach for selecting an appropriate one is data itself to be firstly analyzed as the priority is the end result to be a well-structured and optimized data report.

Acknowledgments

This paper is partially supported by project MUPD23-FMI-009 of the Scientific Research Fund of Paisii Hilendarski University of Plovdiv, Bulgaria.

References

- [1] <https://www.techtarget.com/searchbusinessanalytics/definition/business-intelligence-BI>, (last visited on 12.10.2024).
- [2] R. Khan, S. Quadri, Business Intelligence: an integrated approach, *Business Intelligence Journal*, Vol. 5, No. 1, 2012, pp. 64–70, <https://shorturl.at/AttSs>.
- [3] <https://www.linkedin.com/pulse/day-20-import-vs-directquery-live-connection-power-bi-jaywant-thorat/>, (last visited on 12.10.2024).
- [4] <https://learn.microsoft.com/en-us/power-bi/connect-data/desktop-use-directquery>, (last visited on 12.10.2024).
- [5] <https://radacad.com/connection-types-in-power-bi-import-data-or-scheduled-refresh-a-detailed-look-inside-and-out>,

- (last visited on 12.10.2024).
- [6] <https://learn.microsoft.com/en-us/power-bi/guidance/directquery-model-guidance>, (last visited on 12.10.2024).
 - [7] <https://community.fabric.microsoft.com/t5/Fabric-platform/Best-Practice-for-Import-vs-DirectQuery-Mode-in-Power-BI-Reports/m-p/3648714>, (last visited on 12.10.2024).
 - [8] <https://powerbiconsulting.com/direct-query-vs-import-mode-in-power-bi/>, (last visited on 12.10.2024).
 - [9] M. Allington, *Supercharge Power BI: Power BI is better when you learn to write DAX*, Packt Publishing Ltd, 2024, ISBN-13: 9781836643029.
 - [10] R. Gohil, H. Patel, Comparative Analysis of Cloud Platform: Amazon Web Service, Microsoft Azure, And Google Cloud Provider: A Review, *2024 15th International Conference on Computing Communication and Networking Technologies (ICCCNT)*, Kamand, India, 2024, pp. 1–5, doi: 10.1109/ICCCNT61001.2024.10725914.

Kremena Stefanova¹, Veselina Naneva²,
^{1,2} Paisii Hilendarski University of Plovdiv,
Faculty of Mathematics and Informatics,
236 Bulgaria Blvd., 4027 Plovdiv, Bulgaria
Corresponding author: kstefanova@uni-plovdiv.bg