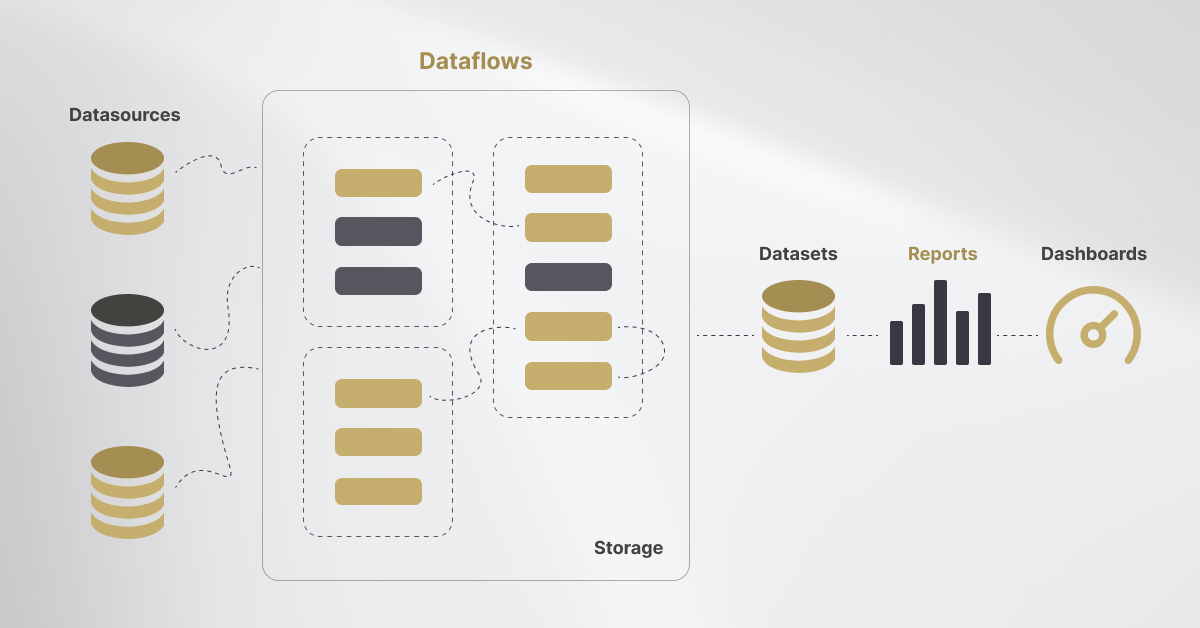
Power BI Dataflow

Power BI Data Flows are independent data transformation engines that can significantly enhance the development and maintenance of your Power BI solution. It is same like power query, but it is done online and run independent from the power bi dataset. Dataflow is a service only object, means it can only create on power bi service and power bi desktop we can only create dashboard and visualization.

It basically a self-service data preparation tool that enables users to easily extract, transform, and load (ETL) data from various sources into the Power BI service. They centralize and standardize data preparation tasks and also allow you to easily create reusable data transformation processes.



Advantages of Dataflow:

1. Decoupling Data Transformation from Modeling and Visualization:

The advantage of decoupling the data transformation layer from the modeling and visualization layer in a Power BI is that it provides a more modular and flexible architecture. Changes to data transformation can be made independently without affecting the modeling and visualization, leading to easier updates and maintenance.

2. Centralized Data Transformation Code in a Dataflow:

A single, centralized dataflow ensures uniform transformations and calculations are applied across multiple reports and dashboards, avoiding discrepancies. Updates to the transformation logic are done in one place, making it easier to manage and apply changes without needing to modify multiple Power BI reports.

3. Promoting Reusability:

By creating a dataflow once, the transformed data can be reused across multiple reports, dashboards, and applications. This reduces the need to recreate the same data transformations, saving time and effort and ensures that all reports and visualizations are based on identical data transformations, eliminating discrepancies and ensuring consistency across different projects.

4. **Scheduled Refreshes:**

Scheduling a refresh ensures that data is updated at regular intervals without manual intervention, keeping reports and dashboards always up-to-date with the latest information and by scheduling refreshes during off-peak hours, you can reduce the load on the system during high-usage times, improving performance and ensuring faster data processing.

5. Optimized for Large Datasets:

Dataflows are optimized to handle large datasets efficiently. By offloading data transformation and storage to Azure, it helps in managing and analyzing larger datasets without overloading individual Power BI reports or dashboards.

6. AI and Machine Learning Integration (Power BI Premium):

For Power BI Premium users, dataflows offer access to advanced AI capabilities, including automated machine learning and entity extraction. This enables users to apply AI-driven transformations to their data without needing to write custom code.

Limitations of Dataflow:

1. No Direct Support for Row-Level Security (RLS):

**It doesn’t support Row-Level Security (RLS)** directly within a dataflow. RLS can only be applied in **Power BI Datasets** or **Power BI Reports**. This means that security filtering must be handled after the data is loaded into a dataset.

2. Incremental Refresh (Premium Only):

**Incremental Refresh** is available only to **Power BI Premium** or **Premium Per User (PPU)** subscribers. In Power BI dataflow, Incremental Refresh is available only in workspaces that are backed by Power BI Premium or Premium Per User (PPU) capacity.

3. Limited Transformation Capabilities Compared to Power BI Desktop:

While **Power Query** in dataflows provides extensive transformation options, some advanced M functions are either not supported or require additional workarounds. It may not offer the same level of transformation capabilities as **Power BI Desktop**. For example, some advanced transformations functionality may not be fully supported in Power BI Dataflows.

4. **Slower performance with** Complex Transformations and Large Data Volumes:

When you apply complex transformations (such as multiple joins, merges, grouping, or pivoting) on large datasets, Dataflows can become slower, especially during the refresh process. Dataflows use **Power Query Online**, and the more transformations or steps applied to the data, the more time it takes to complete.

## Use-case scenarios for dataflows:

* Building a data warehouse using dataflows:

Using dataflows in Power BI allows you to create a robust data warehouse by enabling the extraction, transformation, and loading (ETL) of data from various sources into a centralized location. Dataflows facilitate data integration and cleansing, making it easier to manage and refresh data across multiple reports and dashboards

* Building a dimensional model using dataflows:

Building a dimensional model using dataflows in Power BI involves creating structured data pipelines to organize data into fact and dimension tables. You start by designing the schema, with fact tables representing measurable data and dimension tables providing context.

* Centralize Data preparation:

Centralizing data preparation in Power BI using dataflows streamlines the process by consolidating data extraction, transformation, and loading into a single, reusable pipeline. This approach allows you to manage and maintain data consistency across multiple reports and dashboards, reducing redundancy and errors

* Reusing semantic models across multiple reports:

Reusing semantic models across multiple Power BI solutions involves creating a central data model that can be shared and leveraged in various reports and dashboards. This approach allows you to define consistent metrics, calculations, and business rules in one place, ensuring uniformity and reducing the need for redundant work.

Creating a dataflow in Power BI from SQL Server:

### 1. **Open Power BI Service**

* Go to Power BI Service and log in with your credentials.

### 2. **Navigate to Dataflows**

* In the Power BI Service, select the workspace where you want to create the dataflow.
* In the left navigation pane, select **"Dataflows"**.

### 3. **Create a New Dataflow**

* Click on the **"Create"** button at the top right.
* Select **"Dataflow"** from the dropdown menu.

### 4. **Choose Data Source**

* In the **"Choose data source"** page, select **"SQL Server"**.

### 5. **Enter Connection Details**

* Enter the **Server name** and **Database name**.
* Select Data Gateway
* Choose your authentication method and provide the necessary credentials.
* **Click "Next".**

### 6. **Select Tables or Views**

### 7. **Transform Data** and Save & close

### 8. **Use Dataflow in Power BI Desktop**

* Open Power BI Desktop.
* Go to **"Home"** and select **"Get Data"**.
* Choose **"Power BI dataflows"** and connect to the workspace where you created the dataflow.
* Select the dataflow and the entities you need.