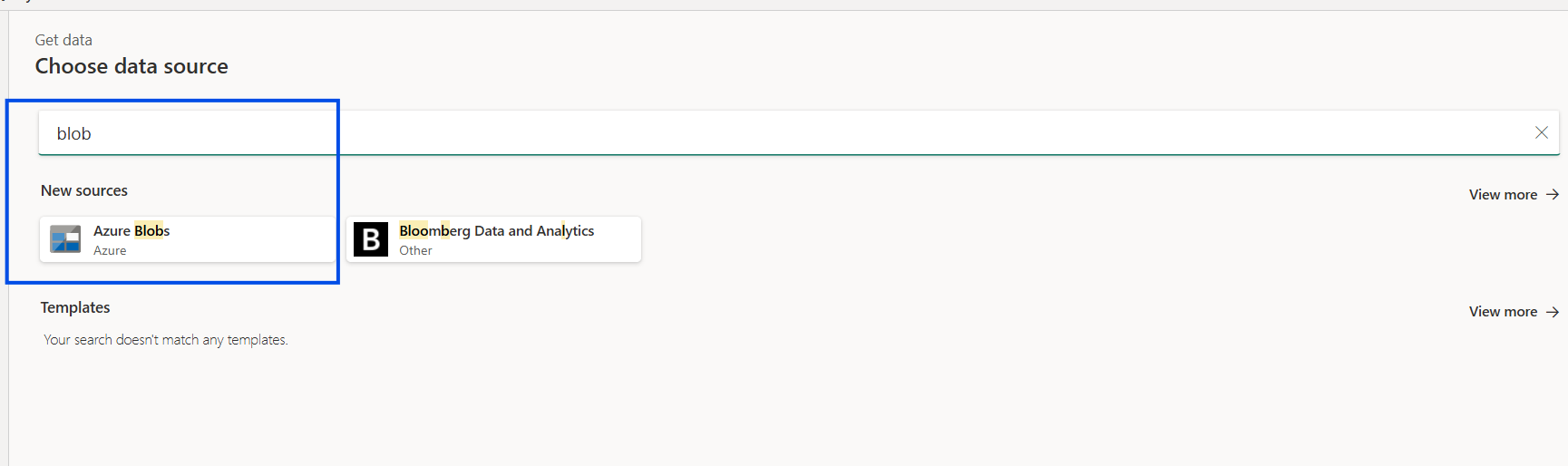
**Using Dataflows in Power BI Premium or Microsoft Fabric**

**Introduction**

Dataflows are a powerful feature in Power BI and Microsoft Fabric that allow you to create, manage, and reuse data transformation logic outside of traditional Power BI Desktop models. This article demonstrates how to create Dataflows using Azure Blob Storage as a data source and how to set them up in both Power BI Premium (Gen1) and Microsoft Fabric (Gen2). We’ll discuss licensing considerations, workspace settings, authentication methods, and best practices around storage and gateways.

**1. Understanding Dataflow Generations and Licensing**

Power BI offers two generations of Dataflows:

* **Dataflow Gen1** (Power BI Premium): Requires Premium capacities such as P1, P2, or P3 (or embedded capacities).
* **Dataflow Gen2** (Microsoft Fabric): Part of Microsoft Fabric’s suite of features, offering more scalability and integration with other Fabric components.

When you create a **New item** in a Fabric workspace, you will see both **Dataflow Gen1** (red box in [Image 3]) and **Dataflow Gen2** (blue box in [Image 3]).

* **Dataflow Gen1** is available to Premium (P SKUs) or Premium Per User (PPU) licenses.
* **Dataflow Gen2** is more tightly integrated with the broader Fabric ecosystem.

For this walkthrough, we’ll look at how to leverage either version of dataflows in your environment.

**2. Setting Up Your Workspace**

Before you can create dataflows, ensure you have a workspace that’s backed by either a **Premium capacity** or a **Fabric capacity**:

1. Go to **Settings** in Power BI Service.
2. Under **Admin Portal**, configure your **Premium capacity** or enable your **Fabric capacity**.
3. In **Workspace Settings**, as shown in [Image 6], you can connect an **Azure Data Lake Gen2** storage account or continue using the default storage. If you opt for Azure Data Lake Gen2, you may need to configure appropriate RBAC permissions to ensure that the Power BI service (or Fabric service) can access it.

**3. Creating an Azure Blob Storage and Uploading Files**

1. In the Azure Portal, create a **Storage Account**.
2. Within the Storage Account, create a **Blob container** (e.g., excelcontainer) as shown in [Image 4].
3. Upload your files (e.g., an Excel file) to that container (see [Image 4] and [Image 1]).

A **private storage** (such as an Azure Blob container with restricted access) is often safer than using a publicly accessible SharePoint folder. It can also be more scalable and manageable for large or sensitive datasets.

**4. Connecting Your Dataflow to Azure Blob Storage**

1. In Power BI (or Fabric), go to **Dataflows** and choose **Add new Dataflow**.
2. You will see the **Get data** screen. Search for **“blob”** (as in [Image 1]) and select **Azure Blobs**.
3. Enter your **Account name or URL** and choose the **Authentication kind** (as shown in [Image 7]). Options include:
   * **Anonymous** (public blob access)
   * **Account key**
   * **Organizational account**
   * **Shared Access Signature (SAS)**
   * **Service Principal**

For production scenarios, **Account key** or **SAS tokens** are common approaches. Using **Organizational account** is also an option if your blob is protected by Azure Active Directory.

1. After connecting, you’ll see your containers (e.g., excelcontainer in [Image 2]). Select the files to be included in the dataflow.

**5. Privacy Levels and Gateways**

When your storage is private (not publicly accessible), you may need to configure a **gateway**:

* **On-premises data gateway** if your storage is on-premises or requires a specialized network configuration.
* **Virtual Network (VNet) gateway** if you are using Azure VNets to protect the storage.

Also, consider **organizational privacy** levels when dealing with sensitive data. Properly classifying data in Power BI’s privacy settings helps maintain compliance and control over how data is combined or moved within your organization.

**6. Transforming and Saving the Dataflow**

1. Once your data source is connected, you’ll be taken to the **Power Query** interface to shape and transform your data.
2. After applying transformations, click **Save**. Provide a name and optional description (as in [Image 5]).

By creating dataflows, you keep your transformation logic in one place, making it reusable across multiple datasets and reports—much more scalable than embedding queries in individual dashboards or reports.

**7. Dataflow Refresh and Management**

After saving, you can:

* **Schedule refresh** of your dataflow.
* Reuse the dataflow in multiple **Power BI datasets** or **Fabric notebooks**.
* Monitor refreshes in the workspace **Settings** or in the **Admin Portal** (for capacity-level management).

**8. Best Practices**

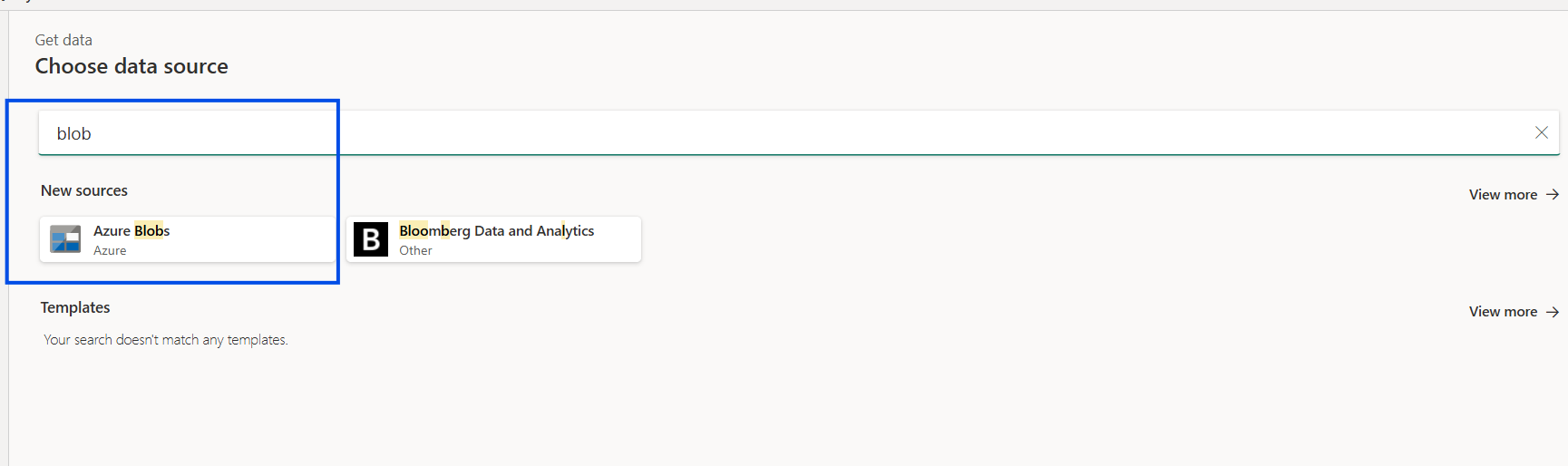
1. **Use Premium or Fabric Capacity**: Ensures you have enough resources and features (e.g., incremental refresh, AI enrichments, large model sizes, Dataflow Gen2).
2. **Leverage Azure Storage**: Storing files in a private Azure Blob container is often more secure and scalable than SharePoint for large data scenarios. Remember to create a private endpoint to make sure the Storage is accessible privately instead over the public internet.
3. **Optimize Authentication**: Use SAS tokens or service principals to manage secure access at scale or use the access token as an last option.
4. **Configure Privacy and Gateways**: If your data is sensitive, properly configure privacy levels and use gateways for private or on-prem data sources.
5. **Reuse and Modularize**: Dataflows can be referenced by multiple datasets, promoting consistent data transformations across your organization.

**Conclusion**

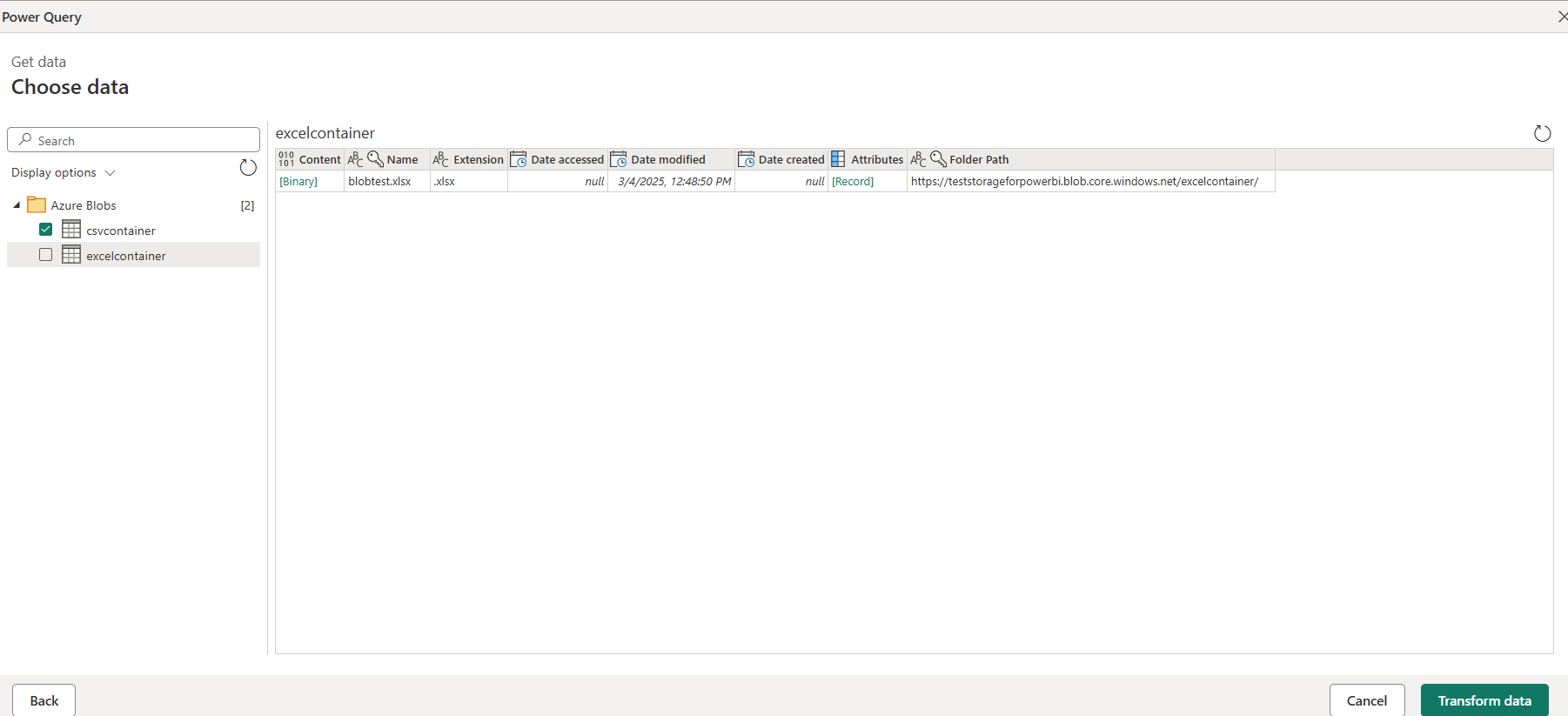
Dataflows in Power BI Premium or Microsoft Fabric provide a powerful, scalable way to manage and reuse data transformations. By connecting to Azure Blob Storage, you can securely store and process large datasets. Leveraging the right authentication method, privacy level, and gateway setup ensures your data remains protected while enabling high-performance data refreshes.

**References to Images**

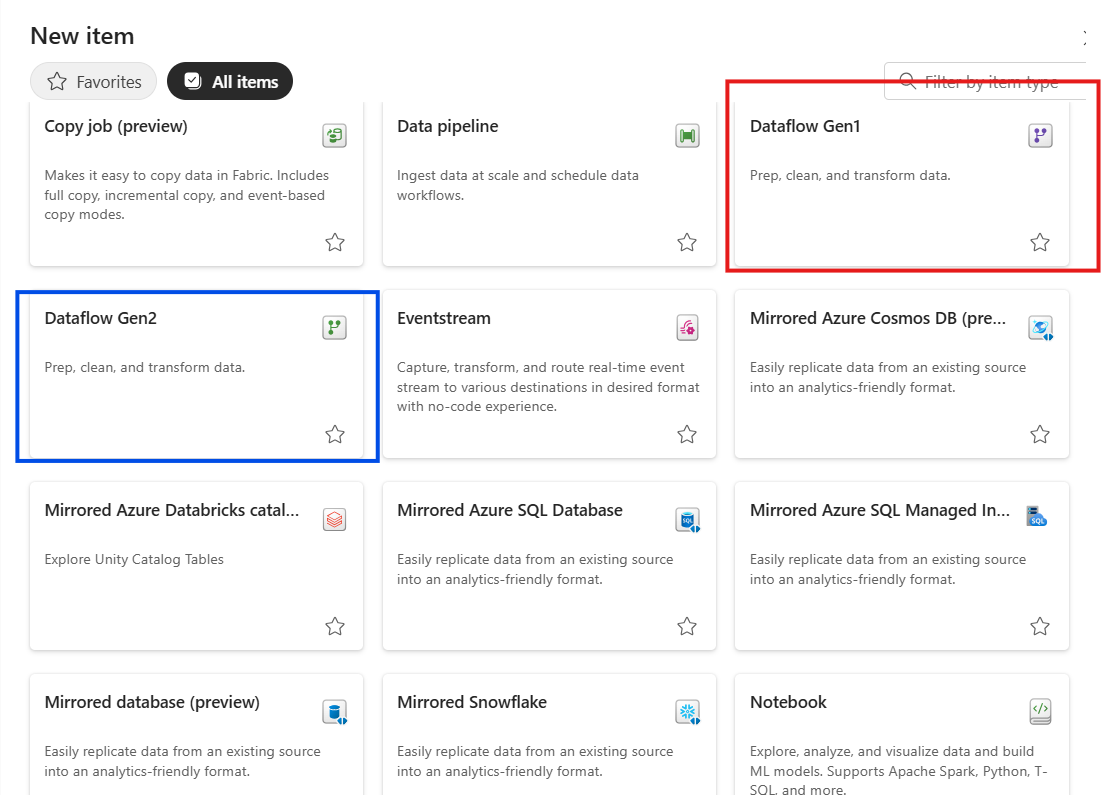
* **[Image 1]**: Searching for “blob” in the **Get data** dialog.

****

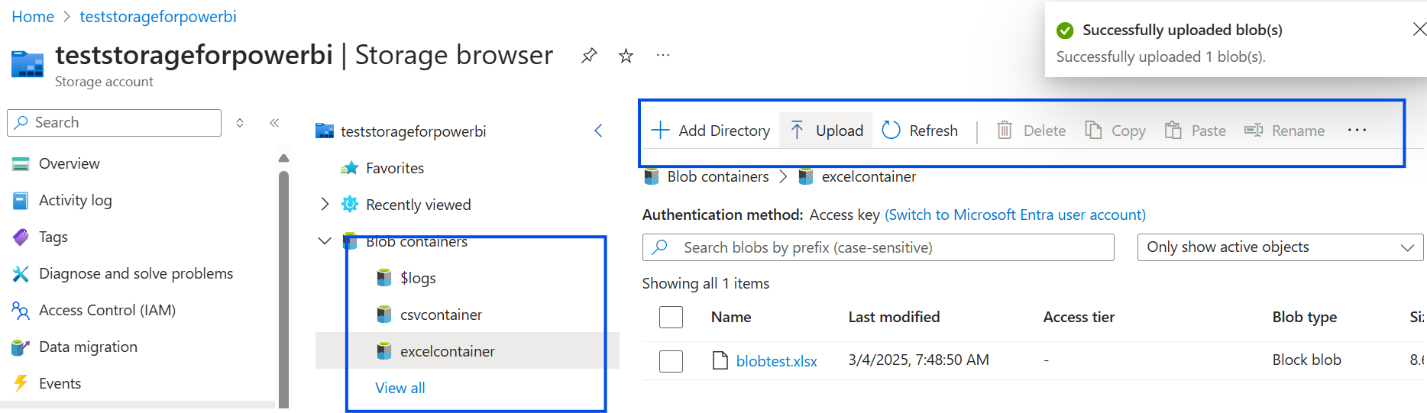
* **[Image 2]**: Viewing the blob container (excelcontainer) in Power Query.



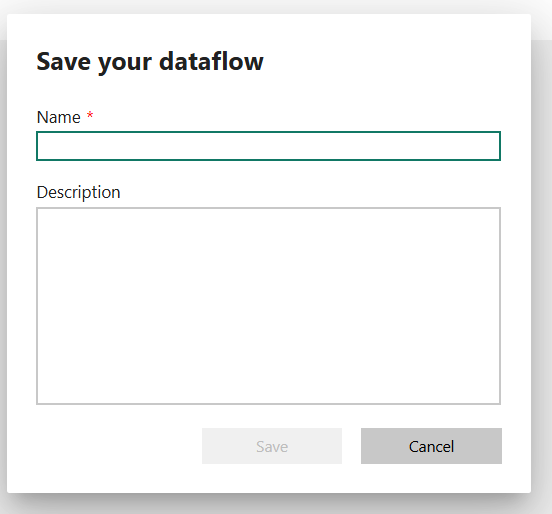
* **[Image 3]**: Differences between **Dataflow Gen1** and **Dataflow Gen2** in the Fabric “New item” menu.



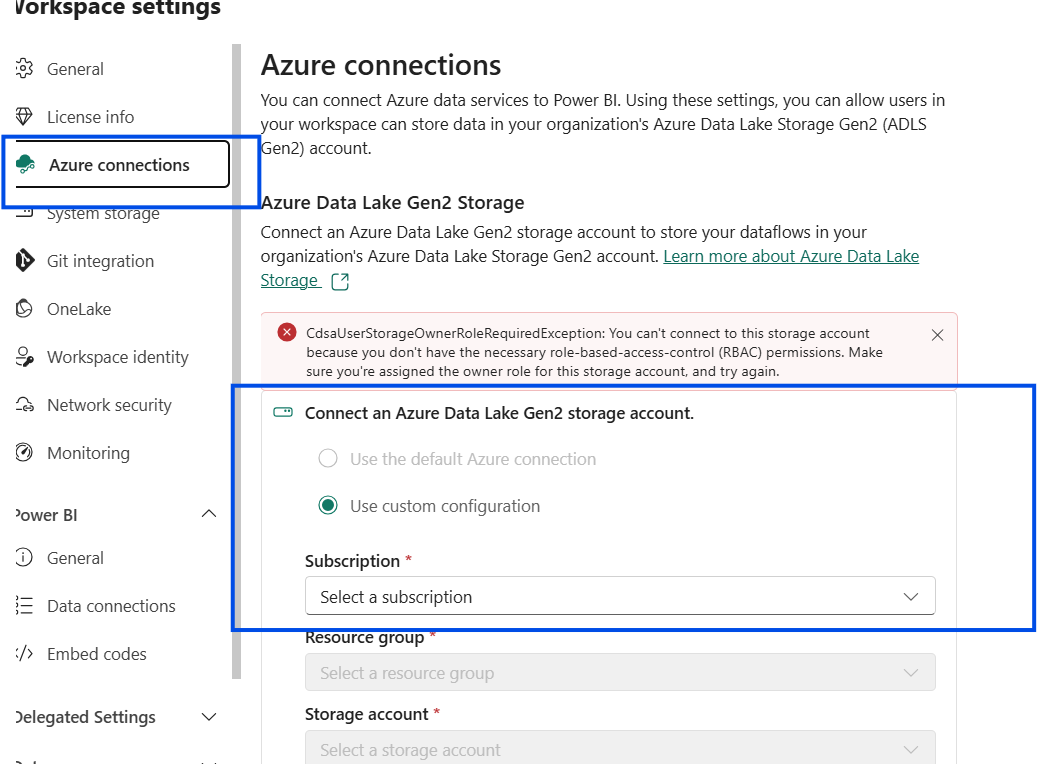
* **[Image 4]**: Azure Storage browser showing containers.



* **[Image 5]**: Dialog to save your dataflow.



* **[Image 6]**: Azure connections in Power BI workspace settings.



* **[Image 7]**: Selecting the authentication kind for Azure Blobs.

