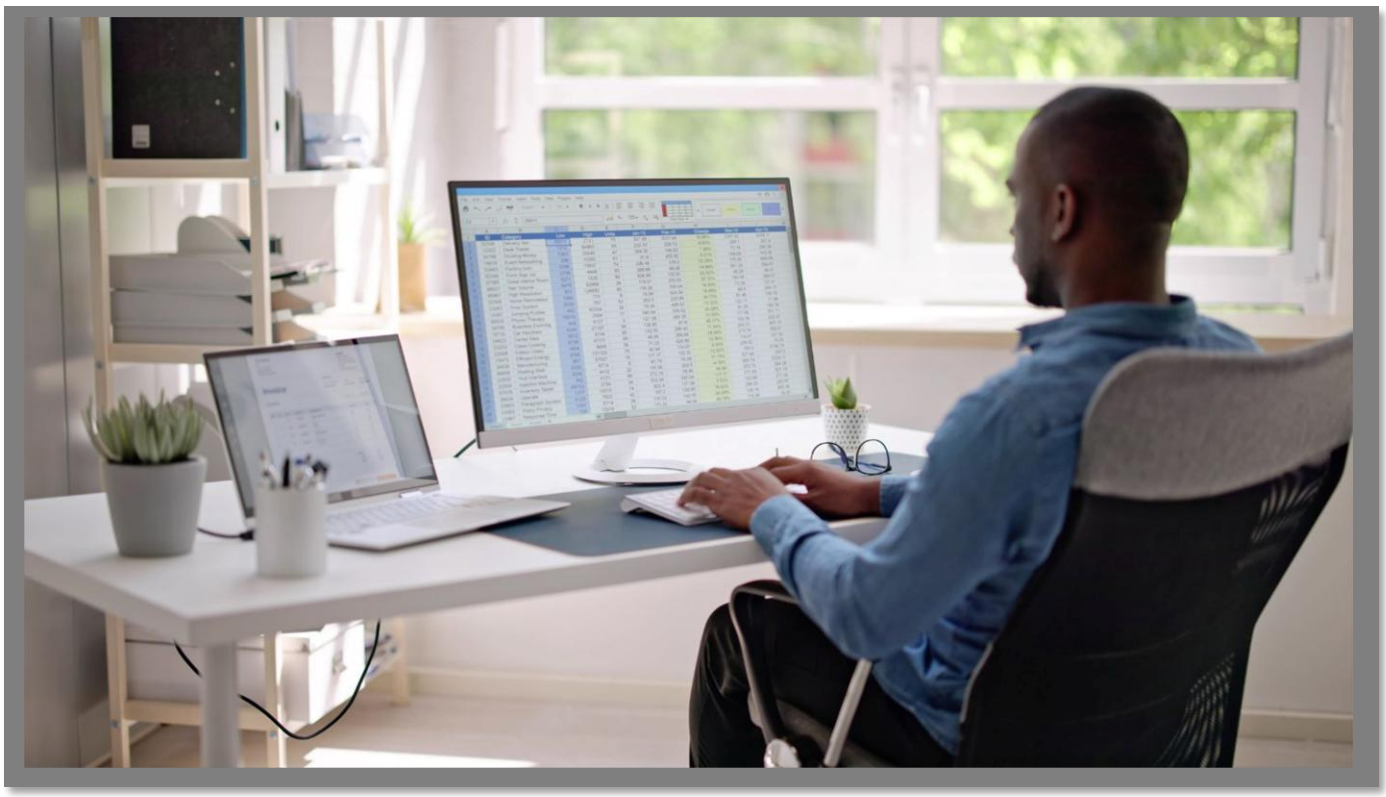
Arbutus Connectors

Excel Online
CONFIGURATION GUIDE



ARBUTUS
Powerful Analytics Simplified

Contents

Arbutus Connector – Excel Online

A. Introduction

The purpose of this Guide is to provide assistance with configuring the Arbutus Excel Online Connector using the ODBC Data Source Administrator. The configuration process can involve several technical steps that require a good understanding of IT systems and database management.

To make the most of this guide, it's advisable to have a good understanding of database connectivity, driver installation, and system settings. The ODBC Data Source Administrator, which is used as part of the configuration process, allows for the setup and management of data sources, enabling applications to access data from various database systems.

Due to the complexity and potential impact of these configurations, it is recommended that only those individuals with IT or database expertise undertake this task. In addition, it should also be understood that each client's network environment is different. A one-size-fits-all approach is rarely effective, as what works well in one environment may not be suitable in another.

B. About Excel Online

Excel Online is a cloud-based version of Microsoft Excel that allows users to create, edit, and share Excel spreadsheets through a web browser. It is part of the Microsoft 365 suite and offers real-time collaboration, automatic saving, and access to files from any device with internet access, without needing to install Excel locally. It provides many of the core features of the desktop version, though with some limitations in advanced functionalities.

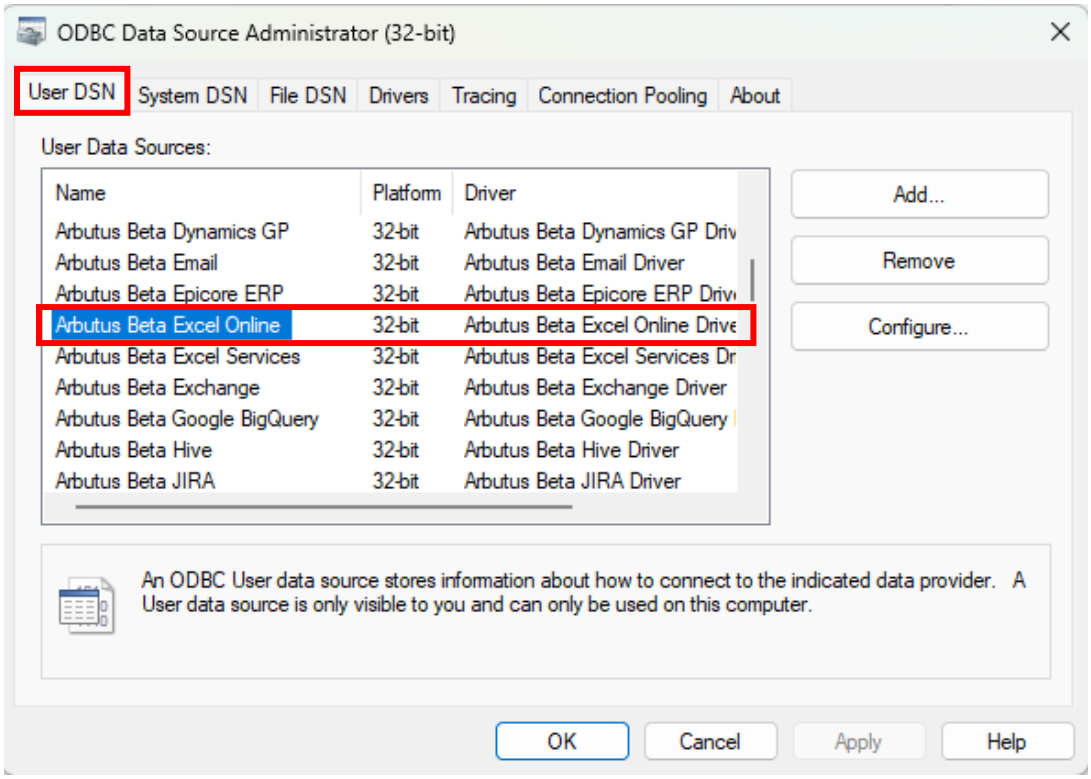
Excel Online stores data in **OneDrive for Business** (personal cloud storage for each user in a Microsoft 365 account) or **SharePoint Online** if the file is stored within a team or organization's shared space), both of which are part of the **Microsoft 365 cloud ecosystem**. Both storage locations are cloud-based, meaning your Excel files are stored on Microsoft's secure servers.

C. Determining if the Connector exists prior to configuring

Installation of the Arbutus Excel Online Connector is done at the time of installing the Arbutus software. For more information on this, please see the **Overview Guide Document**.

Once the Connector has been installed, the next step is to configure it.

Prior to configuring it, you can check to see if the Connector has been installed by opening the **32-bit ODBC Data Source Administrator**, pictured below, and clicking the **User DSN** tab. Included below is information on how you can access the **ODBC Data Source Administrator**.



- If the Arbutus Excel Online Connector appears in the list, it can be considered as installed.
- If it is not listed, it is likely that you did not select it during the installation or modification of the Arbutus software. In this case, it is recommended to reinstall the Arbutus software and choose the **Modify** option when prompted. For more details, please refer to the **Overview Guide Document**.

Below is the file path to access and run the **ODBC Data Source Administrator** application:

C:\Windows\SysWOW64\odbcad32.exe

Alternative, you can also try locating and opening the **ODBC Data Source Administrator** application by doing a search on your desktop application.

D. Configuring the Connector after it has been installed

Once you have verified that the Arbutus Connector has been installed, it is time to configure it.

This process is done using the **ODBC Data Source Administrator**. It can be described as "**editing the DSN configuration"**.

DSN, Drivers, and Data Sources
What is a DSN? DSN stands for Data Source Name, and is a unique name used to create a data connection to a database using open database connectivity (ODBC). A DSN is a data structure that contains the information required to connect to a database. It is essentially a string that identifies the source database, including the driver details, the database name, and often authentication credentials and other necessary connection parameters. DSNs facilitate a standardized method for applications to access databases without needing hard-coded connection details, enhancing flexibility and scalability in database management.

Arbutus Connectors

- ***Drivers*** are the components that process ODBC requests and return data to the application. If necessary, drivers modify an application's request into a form that is understood by the data source. The **Drivers** tab in the **ODBC Data Source Administrator** dialog box lists all drivers installed on your computer, including the name, version, company, file name, and file creation date of each driver.

- ***Data sources*** are the databases of files accessed by a driver and are identified by a data source name (DSN). You use the ODBC Data Source Administrator to add, configure, and delete data sources from your system.

All ODBC connections require that a DSN be configured to support the connection. When a client application wants to access an ODBC-compliant database, it references the database using the DSN.

The types of DSNs are:

- **User DSN** – User DSNs are local to a computer and can be used only by the current user. They are registered in the HKEY_Current_USER registry subtree.

- **System DSN** – System DSNs are local to a computer rather than dedicated to a user. The system or any user with privileges can use a data source set up with a system DSN. System DSNs are registered in the HKEY_LOCAL_MACHINE registry subtree.

- **File DSN** – File DSNs are file-based sources that can be shared among all users who have the same drivers installed and therefore have access to the database. These data sources need not be dedicated to a user nor be local to a computer. File data source names are identified by a file name with a .dsn extension.

User and system data sources are collectively known as *machine* data sources because they are local to a computer.

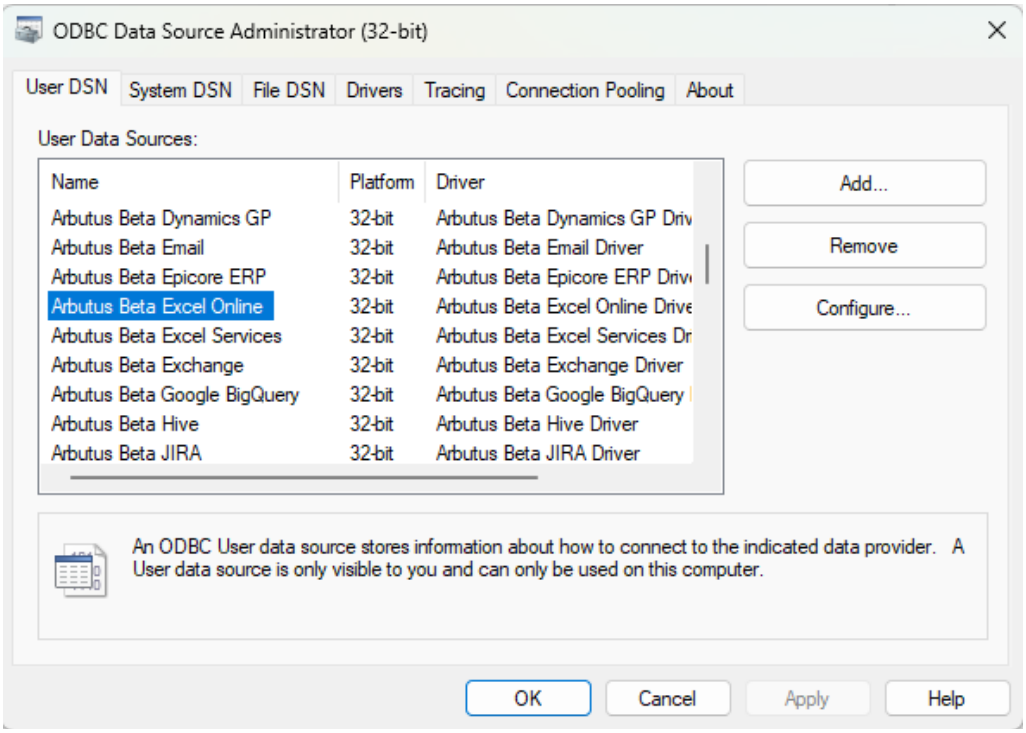
Each of these DSNs has a tab in the **ODBC Data Source Administrator** dialog.

The Arbutus ODBC Driver for Excel Online enables real-time access to Excel Online data, directly from any applications that support ODBC connectivity, the most widely supported interface for connecting applications with data.

Arbutus Connectors

Follow these steps to edit the DSN configuration and configure the Connector.

1. First open the **ODBC Data Source Administrator**.



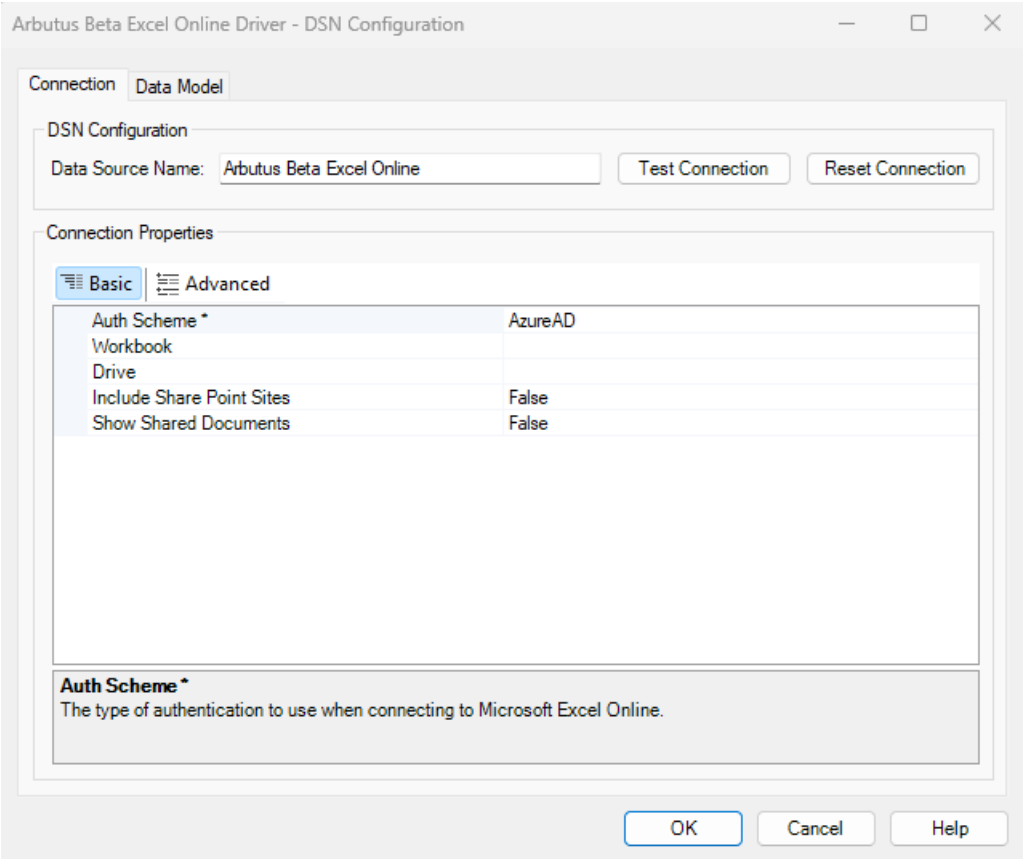
2. Click the **User DSN** tab.

   Selected data connectors are installed as **User DSN's** in Window's 32 Bit **ODBC Data Source Administrator**.

   Also, each of the data connector's names is prefaced with Arbutus, for example, **Arbutus Excel Online.**

3. Select the Arbutus Connector, in this case it is **Arbutus Excel Online**.
4. Click **Configure**.

This opens the **Arbutus Excel Online Driver – DSN Configuration** dialog.



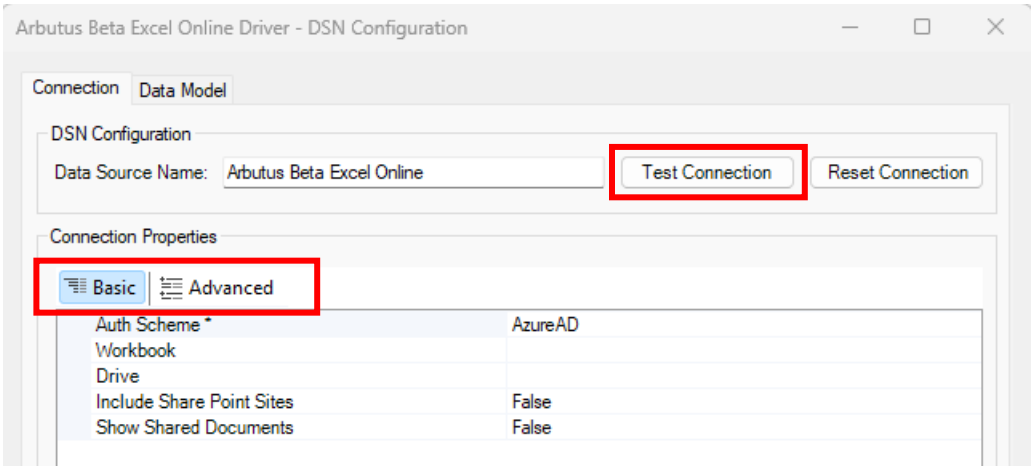
E. Editing the DSN properties – the Basic and Advanced tabs

With the DSN Configuration dialog open, the next step is to edit the DSN properties, where necessary, in the **Basic** and **Advanced** tabs. For example, editing the **Auth Scheme properties** (per screenshot below) to ensure correct authentication to the server is applied.

E1. Editing the DSN properties in the Basic tab

The properties listed in the **Basic** tab are typically the ones that are most commonly used, and as such are designed to be more user-friendly and straightforward, allowing you to quickly make changes without needing in-depth technical knowledge.

Once you have completed editing the properties in the **Basic** tab, you can go ahead and try testing the connection to the Excel Online system by clicking the **Test Connection** button, as highlighted in the screenshot below.



In the **Basic** tab, there are **five** main properties to review:

1. **Auth Scheme** – click the dropdown to select from the list the appropriate type of authentication to use when connecting to Microsoft Excel Online. The options available for selection are as follows:
   - AzureAD – select this when you want to authenticate using a user's credentials directly, typically in scenarios where the user is logging in with a username and password (e.g., interactive logins).

     This is the most common authentication method. It uses your Azure AD credentials (username/password) to authenticate. It is commonly used when you're connecting as an individual user and you want to authenticate with their Azure Active Directory account.

You should select **Azure AD** if your application needs to authenticate with user-specific credentials, or if you're dealing with direct user sign-ins to Azure AD.

The default value is **AzureAD**.

- AzureMSI – select this if your application is running inside Azure and you want to authenticate using the Managed Service Identity (MSI) associated with the service.

  This removes the need to manage any credentials, as Azure automatically handles the identity for you.

  This option is designed for applications that run in **Azure services** like Azure VMs, Azure App Service, or Azure Functions. Managed Identity is a special identity automatically managed by Azure AD.

  You should select **AzureMSI** if you want to authenticate automatically using the Azure environment, and your application is hosted inside Azure (e.g., a web app, VM, or function).

- AzureServicePrincipal – select this when you want to authenticate using a **service principal** (a special identity for an Azure application) and have **explicit credentials** (App ID and Client Secret). This is useful for server-to-server communication or automation scenarios where user-specific logins aren't needed.

  A **Service Principal** is essentially an identity for an application or service to authenticate and access Azure resources securely without requiring user credentials.

  You should select **AzureServicePrincipal** if you're working with a **service or application** that needs to authenticate without a user context, and you want to use application-level credentials (such as in automated tasks or background services).

Service Principals are typically used for **non-interactive** authentication, such as when applications or services authenticate to Azure resources. Therefore, they do not use Single-Sign-On (SSO) because they are meant for app-level authentication and not user-level authentication.

Selecting **AzureServicePrincipal** requires you to specify the following:

- Azure Tenant ID – this is the Microsoft Online tenant being used to access data. If not specified, your default tenant is used.

  A tenant is a digital representation of your organization, Primarily associated with a domain, for example, Microsoft.com. The tenant is managed through a Tenant ID (also known as the directory ID), which is specified whenever you assign users permissions to access or manage Azure resources.

  To locate the directory ID in the Azure Portal, navigate to **Azure Active Directory > Properties**.

  Specifying **AzureTenant** is required when **AuthScheme** = either **AzureServicePrincipal** or **AzureServicePrincipalCert**, or if **AuthScheme** = **AzureAD** and the user belongs to more than one tenant.

- Azure Client Id – this is the client Id assigned when you register your application with an OAuth authorization server.

- Azure Client Secret – this is the client secret assigned when you register your application with an OAuth authorization server.

- AzureServicePrincipalCert – select this when you need higher security for authentication and want to use a **certificate** instead of a client secret. This method is often used when managing services in a more secure, certificate-based manner.

  It is similar to AzureServicePrincipal, but instead of a **client secret**, this method uses a **certificate** to authenticate the service principal.

  You should select **AzureServicePrincipalCert** if you need **certificate-based authentication** for higher security or are working in an environment where secrets are avoided for security reasons.

  Selecting **AzureServicePrincipalCert** requires you to specify the following:

  - Azure Tenant – this is the Microsoft Online tenant being used to access data. If not specified, your default tenant is used.

    A tenant is a digital representation of your organization, Primarily associated with a domain, for example, Microsoft.com. The tenant is managed through a Tenant ID (also known as the directory ID), which is specified whenever you assign users permissions to access or manage Azure resources.

    To locate the directory ID in the Azure Portal, navigate to **Azure Active Directory > Properties**.

    Specifying **AzureTenant** is required when **AuthScheme** = either **AzureServicePrincipal** or **AzureServicePrincipalCert**, or if **AuthScheme** = **AzureAD** and the user belongs to more than one tenant.

- OAuth Client ID – this is client Id assigned when you register your application with an OAuth authorization server.

  OAuth Client Id is one of a handful of connection parameters that need to be set before users can authenticate via OAuth.

- OAuth JWT Cert – this is the name of the JWT Certificate store.

  The **OAuth JWT Cert Type** field (see below) specifies the type of the certificate store specified by **OAuth JWT Cert**. If the store is password protected, specify the password in **OAuth JWT Cert Password** (see below)

  OAuth JWT Cert is used in conjunction with the **OAuth JWT Cert Subject** field in order to specify client certificates. If **OAuth JWT Cert** has a value, and **OAuth JWT Cert Subject** is set, a search for a certificate is initiated. Please refer to the **OAuth JWT Cert Subject** field for details.

  Designations of certificate stores are platform-dependent.

  The following are designations of the most common User and Machine certificate stores in Windows:

MY	A certificate store holding personal certificates with their associated private keys
CA	Certifying authority certificates
ROOT	Root certificates
SPC	Software publisher certificates

In Java, the certificate store normally is a file containing certificates and optional private keys.

When the certificate store type is PFXFile, this property must be set to the name of the file. When the type is PFXBlob, the property must be set to the binary contents of a PFX file (i.e. PKCS12 certificate store).

- OAuth JWT Cert Type – this is dropdown selection to specify the type of key store containing the JWT Certificate.

  - USER
  - MACHINE
  - PFXFILE
  - PFXBLOB
  - JKSFILE
  - JKSBLOB
  - PEMKEY_FILE
  - PEMKEY_BLOB
  - PUBLIC_KEY_FILE
  - PUBLIC_KEY_BLOB
  - SSH PUBLIC_KEY_FILE
  - SSH PUBLIC_KEY_BLOB
  - P7BFILE
  - PPKFILE
  - XMLFILE
  - XMLBLOB
  - BFKSFILE
  - BCFKSBLOB

  The default value is **USER**.

- OAuth JWT Cert Password – this is the password for the OAuth JWT certificate. This property specifies the password needed to open the certificate store, but only if the store type requires one.

- OAuth JWT Cert Subject – this is the subject of the OAuth JWT certificate. Supports partial matches and the wildcard **'*'** to select the first certificate.

  The value of this property is used to locate a matching certificate in the store. The search process works as follows:

  - If an exact match for the subject is found, the corresponding certificate is selected
  - If no exact match is found, the store is searched for certificates whose subjects contain the property value
  - If no match is found, no certificate is selected

  You can set the value to '*' to automatically select the first certificate in the store. The certificate subject is a comma-separated list of distinguished name fields and values. For example: CN=www.server.com, OU=test, C=US, E=support@cdata.com.

  Other common fields and their meanings include: O = Organization ; L = Locality ; S = State

  The default value is **'*'**

2. **Workbook** – this is the name or Id of the workbook. Specifies the name or Id of the workbook. A list of all workbooks is available from the Workbooks view.

3. **Drive** – this is the Id of the drive. Specifies the Id of the drive. A list of all drives is available from the Drives view. This property takes precedence over **SharepointURL**. This means that if **SharepointURL** and **Drive** are specified, a schema will only be identified for the drive specified by **Drive**, and tables will only be identified from the worksheets in workbooks in this drive.

4. **Include Share Point Sites** – this is a **True/False** selection to specify whether to retrieve drives for all SharePoint sites when querying Drives view. If **True** is specified, the provider will retrieve all Site IDs recursively and for each of them issue a separate call to get their drives.

   Therefore, be aware that setting this property to **True** may decrease performance for the **Drives** view. Note that the **SharePoint Access Token** or **OAuth JWT Cert** connection property must be specified to query the **Share Point Sites** view and other views if **Include Share Point Sites** is set to **True** when using the CLIENT **OAuth Grant Type** or the **AzureServicePrincipal AuthScheme**.

   This property affects only Drives views.

   The default value is **False**.

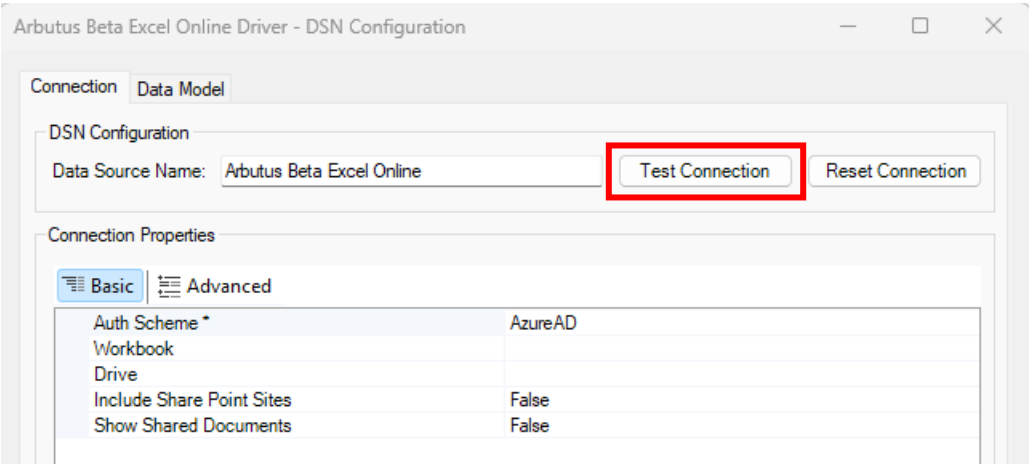
5. **Show Shared Documents** – this is a True/False selection to specify whether or not to show shared documents. If set to **True**, shared documents will be listed along-side user owned workbooks as database tables. Ultimately, the specific files should have been granted direct access or explicitly shared with the authenticated user.

   The default value is **False**.

E2. Editing the DSN properties in the Advanced tab

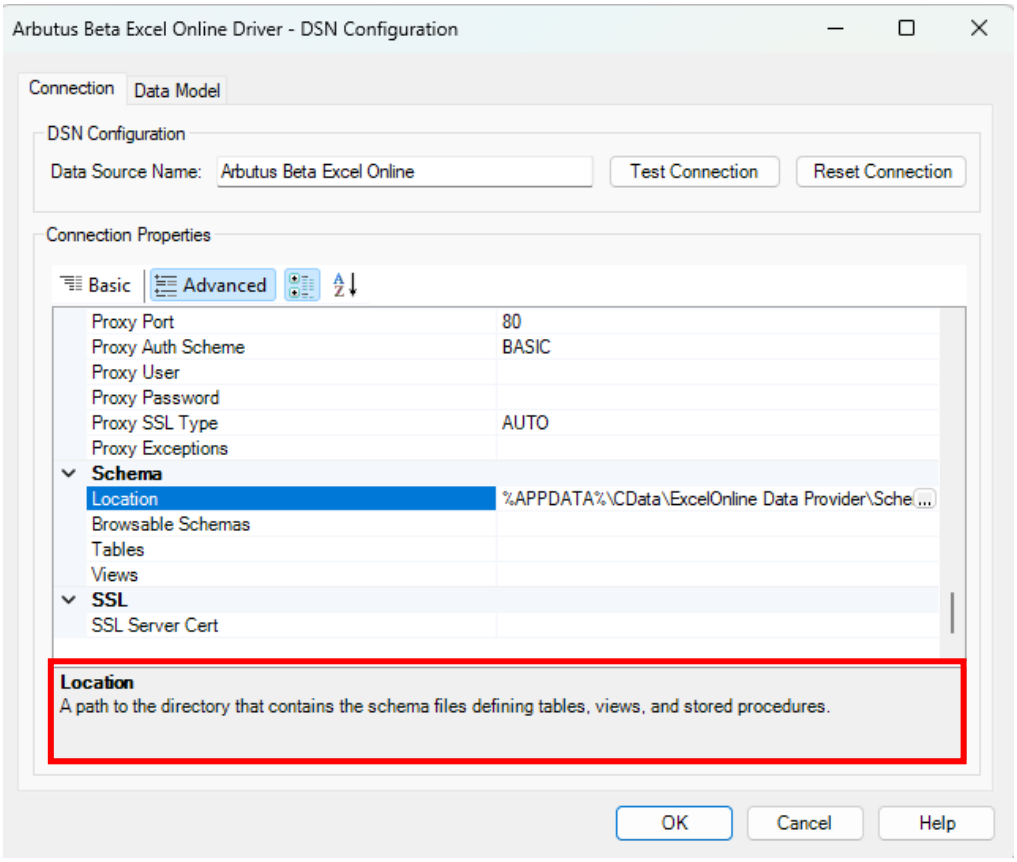
This tab includes more detailed and technical properties. It is intended for those users who need more control over the configuration and are comfortable with more complex options. The **Advanced** tab often includes properties that can fine-tune the behaviour of the system feature.

If you have already completed editing the properties in the **Basic** tab, as required, you do not necessarily need to also complete editing the properties in the **Advanced** tab. Instead, once you have completed editing the properties in the **Basic** tab, you may opt to proceed to testing the connection to the Excel Online system by clicking the **Test Connection** button.



There are a lot more properties included for editing in the **Advanced** tab.

However, it is useful to know that each property does provide a short description of it and as such serves as a guide in terms of what to edit and/or enter. This short description can be seen at the bottom of the **DSN Configuration** dialog box, as seen in the screenshot below.



If it is deemed necessary to complete some/all the properties in the **Advanced** tab, it is recommended that you refer to the description shown for any of the properties being edited and/or entered.

If required, more information on the properties listed in the **Advanced** tab can also be provided.

F. Other questions and/or request for assistance

There may be times when you need to consult with the technical support team at Arbutus Software. If so, please send an email request to support@ArbutusSoftware.com.

For more information, please refer to the CONTACT US page on our website.