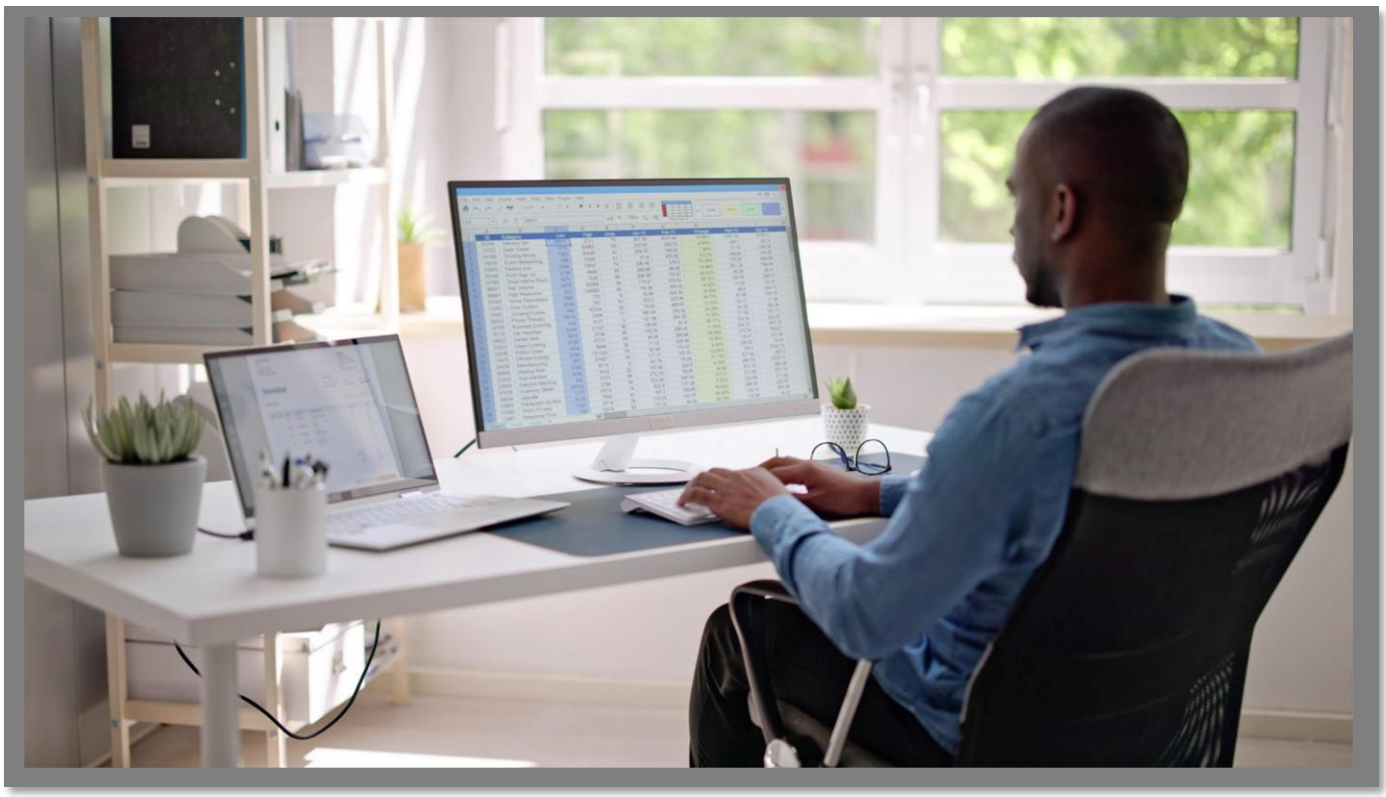
Arbutus Connectors

Azure Active Directory
CONFIGURATION GUIDE



ARBUTUS
Powerful Analytics Simplified

Arbutus Connectors

Contents

Arbutus Connector – Azure Active Directory

A. Introduction

The purpose of this Guide is to provide assistance with configuring the Arbutus Azure Active Directory 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 Azure Active Directory

Azure Active Directory (Azure AD) is Microsoft's cloud-based identity and access management service. It provides secure authentication, single sign-on (SSO), and role-based access control (RBAC) for users, applications, and devices. It is used to manage access to Microsoft's 365, Azure services, third-party SaaS apps, and on-premises resources. It supports Multi-Factor Authentication (MFA) and integration with AD for hybrid identity solutions. Azure AD is a hierarchical directory that stores user accounts, groups, roles, application registrations, and policies.

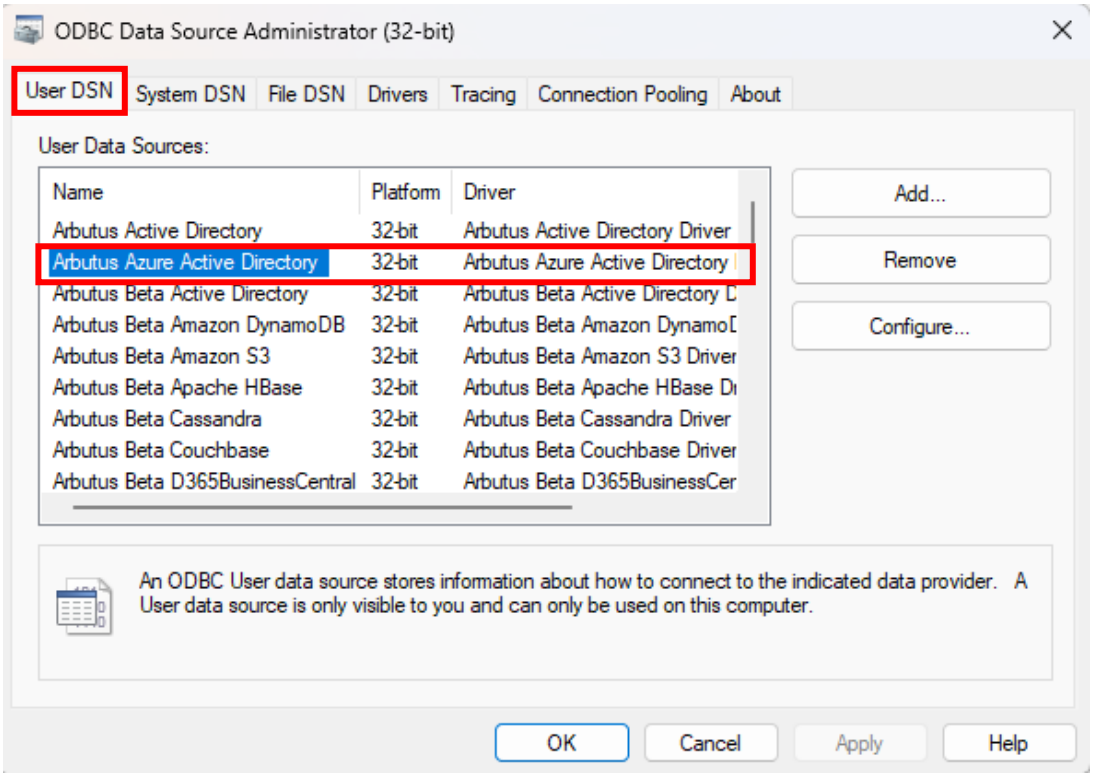
Data is structured in a way similar to traditional AD but optimized for cloud-scale performance and availability. Azure AD replicates identity data across multiple geographically distributed Azure data centres to ensure high availability and prevent data loss.

C. Determining if the Connector exists prior to configuring

Installation of the Arbutus Azure Active Directory 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 Azure Active Directory 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.

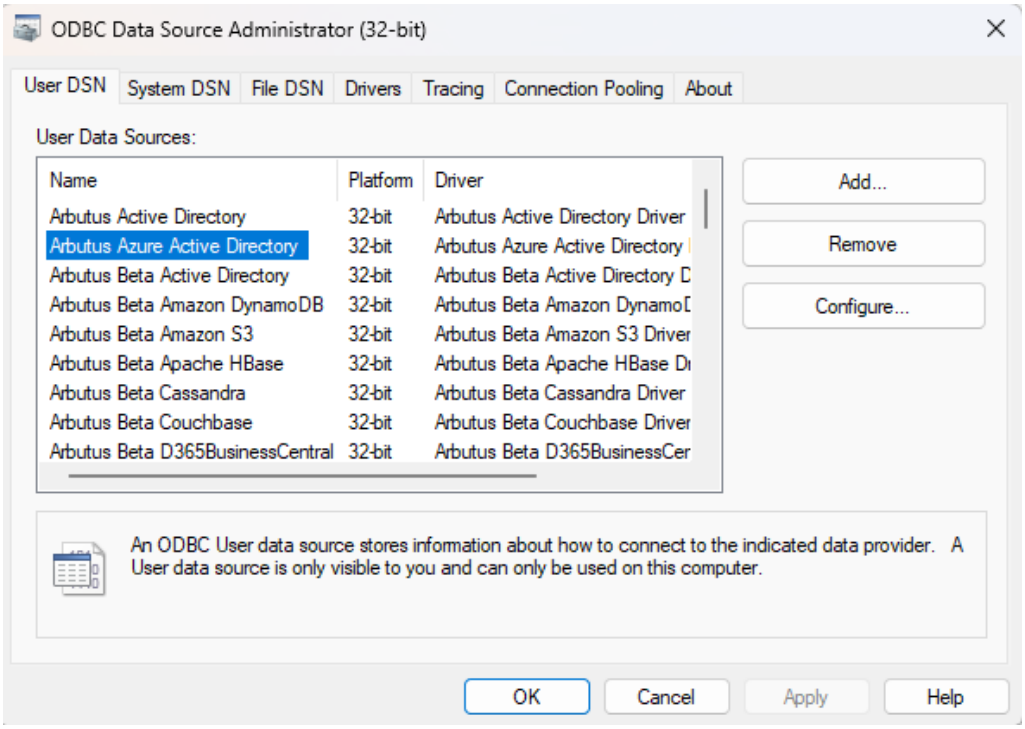
Each of these DSNs has a tab in the **ODBC Data Source Administrator** dialog.

The Arbutus ODBC Driver for Azure Active Directory enables real-time access to Azure Active Directory 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**.



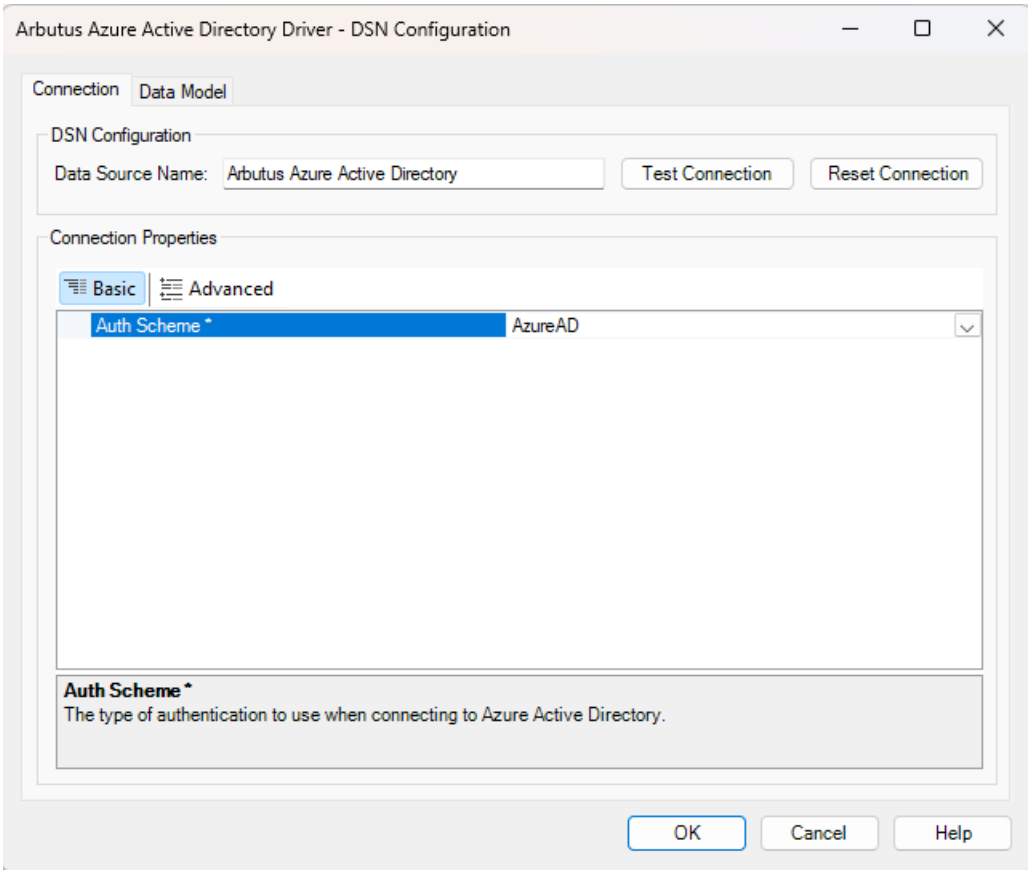
2. Click the **User DSN** tab.

   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 Azure Active Directory.**

3. Select the Arbutus Connector, in this case it is **Arbutus Azure Active Directory**.
4. Click **Configure**.

This opens the **Arbutus Azure Active Directory Driver – DSN Configuration** dialog.



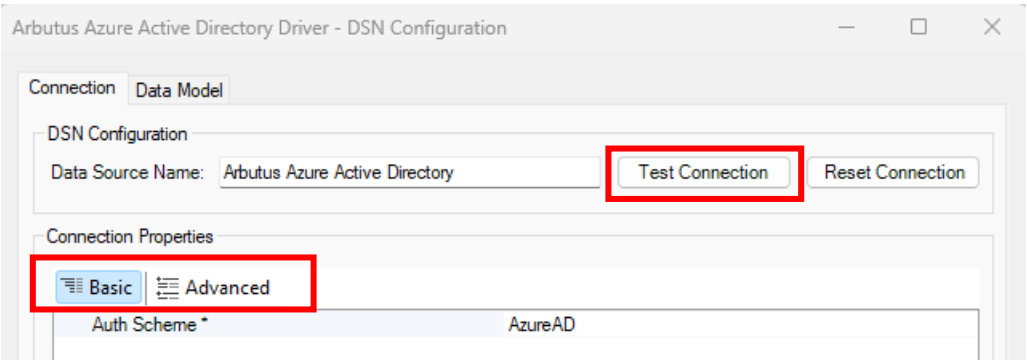
E. Editing the DSN properties – the Basic and Advanced tabs

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 Azure Active Directory system by clicking the **Test Connection** button, as highlighted in the screenshot below.



In the **Basic** tab, there is just **one** main property to review:

1. **Auth Scheme** – click the dropdown to select from the list the appropriate scheme used for authentication when connecting to Azure Active Directory. The options available for selection are as follows:
   - AzureAD – select this when you need to authenticate users via Azure Active Directory, leveraging Azure AD's identity management capabilities. Therefore, you would select this when your database is integrated with Azure AD for identity management.

     Additionally, when you want to enable SSO for users, allowing them to access multiple Azure services with a single set of credentials.

     Some organizations require Azure AD authentication with MFA (Multi-Factor Authentication) for security compliance. Selecting Azure AD ensures the driver can handle this type of authentication.

If you are connecting from an Azure VM, App Service, or Function, using Managed Identity, Azure AD authentication allows secure, password-less access.

- Azure MSI – select this when you want to use Azure's Managed Service Identity (MSI) feature to authenticate without managing credentials manually. This is particularly useful for applications running on Azure services like VMs, App Services, or Azure Functions.

  MSI provides a secure way to authenticate by automatically handling the credentials, reducing the risk of credential exposure – eliminates the need for hardcoded credentials in configuration files, environment variables, or code, thus reducing the risk of credentials risk.

  As well, with Managed Identity, the authentication process is handled by Azure, meaning you don't need to sore or manage credentials like usernames, passwords, or client secrets.

- AzureServicePrincipal – select this when you need to authenticate applications programmatically using a client ID and secret, without user interaction. Selecting this is appropriate when you need non-interactive, application-based authentication to Azure SQL Database or Azure Synapse Analytics using an Azure AD Service Principal.

  Additionally, when you want to assign specific roles and permissions to the service principal, ensuring precise access control. A Service Principal allows fine-grained role-based access control (RBAC) in Azure AD, ensuring that the application only has the necessary permissions to access specific databases.

Selecting **Azure Service Principal** requires you to specify the following:

- Azure Tenant - this identifies the Azure Active Directory tenant being used to access data, either by name (for example, contoso.omnicrosoft.com) or ID.

  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 specifies the client Id that was assigned the custom OAuth application was created. (Also known as the consumer key.) This ID registers the custom application with the 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 Client Secret – this specifies the client secret that was assigned when the custom OAuth application was created. (Also known as the consumer secret). This secret registers the custom application with the OAuth authorization server.

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

- AzureServicePrincipalCert – select this when you need strong security for authenticating applications using certificates, which are more secure than passwords. Since Client secrets expire frequently (typically every 6-12 months), Certificates can have longer lifespans and are more secure when properly managed.

  Additionally, when you want to authenticate applications programmatically without user interaction, using a certificate for seamless access – allows authentication without manual login.

  Selecting **Azure Service Principal Cert** requires you to specify the following:

  - OAuth JWT (JSON Web Token) Cert - this is 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 (see below) 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.

- OAuth JWT Cert Type – select from the dropdown list the type of key store containing the JWT Certificate.

  The options available for selection are:

  - 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

  The default value is **USER**.

- OAuth JWT Cert Password – this is the password for the OAuth JWT certificate used to access a certificate store that requires a password. If the certificate store does not require a password, leave this property blank.

This property specifies the password needed to open the certificate store, but only if the store type requires one. To determine if a password is necessary, refer to the documentation or configuration for your specific certificate store.

- OAuth JWT Cert Subject – this is the subject of the OAuth JWT certificate used to locate a matching certificate in the store. 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@abcd.com

  The default value is *****.

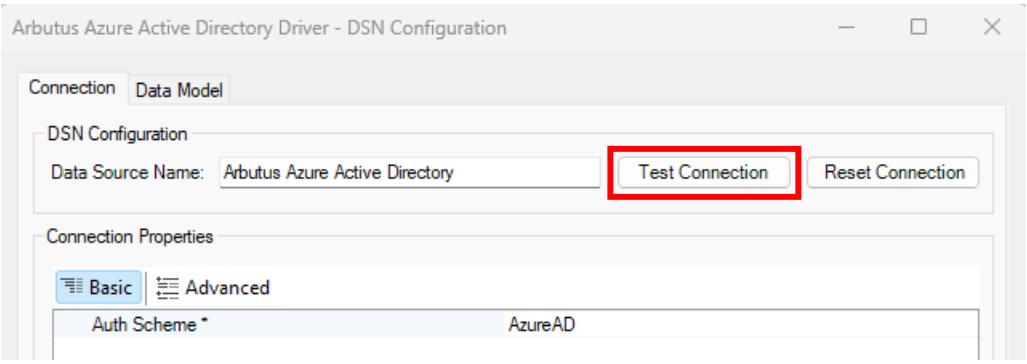
  Other common fields and the meanings include:
  O = Organization, L = Loyalty, S = State

  If a field value contains a comma, enclose it in quotes. For example: "O=ACME, Inc."

E2. Editing the DSN properties in the Advanced tab

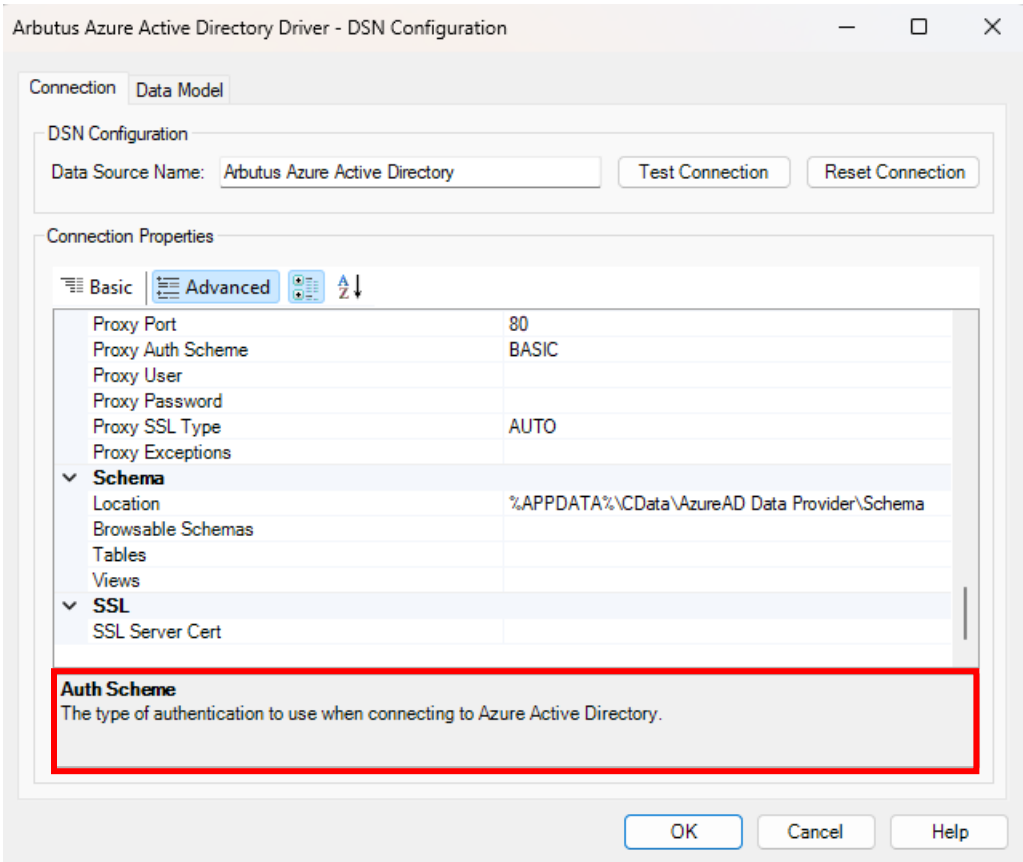
This tab includes more detailed and technical properties. It is intended for those users who need more control over the configuration and are comfortable with more complex options. The **Advanced** tab often includes properties that can fine-tune the behaviour of the system feature.

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 Azure Active Directory 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.