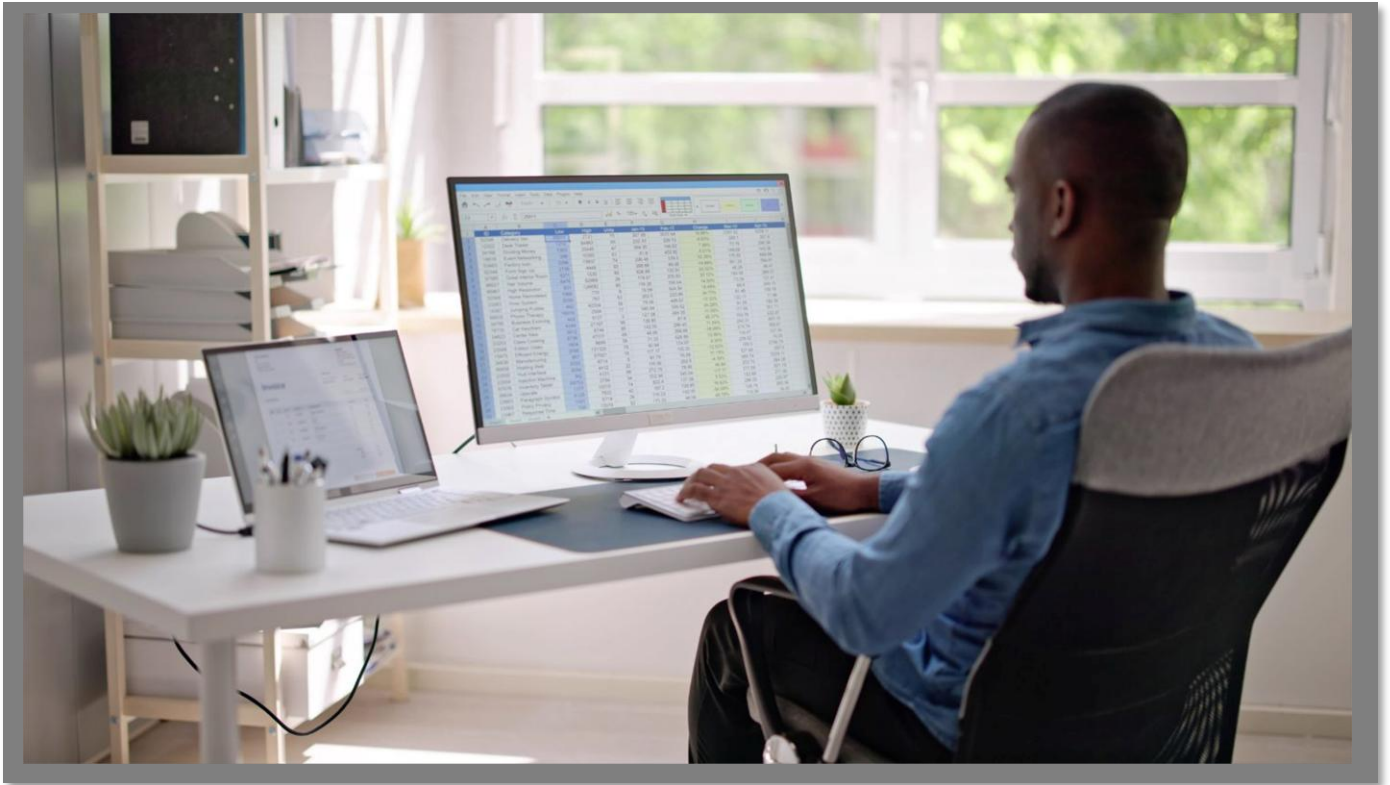


Arbutus Connectors

MongoDB CONFIGURATION GUIDE



 **ARBUTUS**
Powerful Analytics Simplified

Arbutus Connectors

Contents

A. Introduction	1
B. About MongoDB	1
C. Determining if the Connector exists prior to configuring	2
D. Configuring the Connector after it has been installed	3
E. Editing the DSN properties – the Basic and Advanced tabs .	6
E1. Editing the DSN properties in the Basic tab	6
E2. Editing the DSN properties in the Advanced tab	10
F. Other questions and/or request for assistance	12

Arbutus Connectors

Arbutus Connector – MongoDB

A. Introduction

The purpose of this Guide is to provide assistance with configuring the Arbutus MongoDB 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 MongoDB

MongoDB is a NoSQL database known for its flexibility, scalability, and ease of use. Unlike traditional relational databases, MongoDB stores data in JSON-like documents, which allows for a more dynamic and schema-less data model. This makes it ideal for handling large volumes of unstructured or semi-structured data. MongoDB is widely used for modern applications due to its high performance, horizontal scalability, and ability to handle diverse data types.

Data is stored in a document-oriented format within a NoSQL database. A MongoDB instance can have multiple databases, where each database contains collections (similar to tables in SQL).

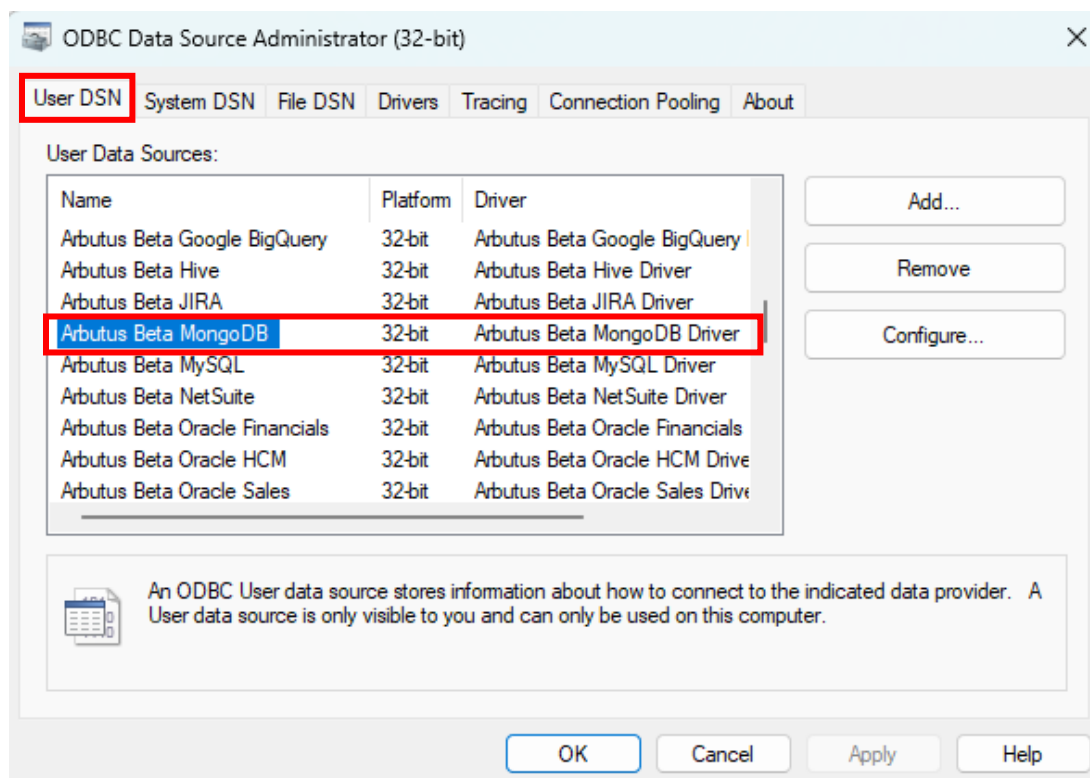
Arbutus Connectors

C. Determining if the Connector exists prior to configuring

Installation of the Arbutus Mongo DB 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**.



Arbutus Connectors

- If the Arbutus Mongo DB 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.

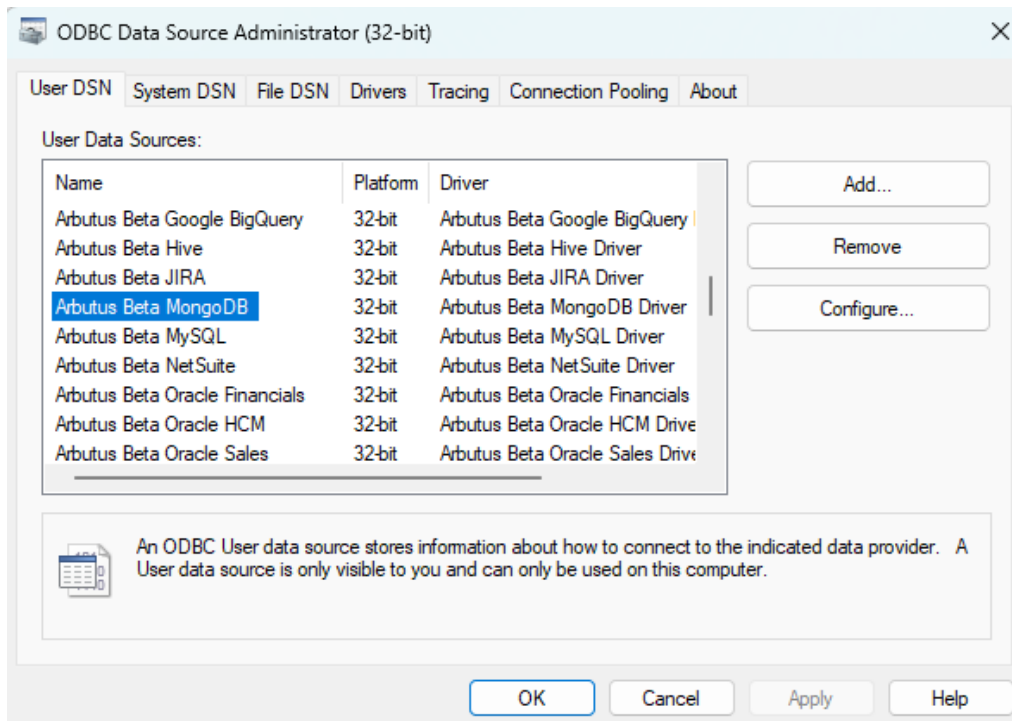
Each of these DSNs has a tab in the **ODBC Data Source Administrator** dialog.

The Arbutus ODBC Driver for MongoDB enables real-time access to MongoDB 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 MongoDB**.

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

Arbutus Connectors

This opens the **Arbutus MongoDB Driver – DSN Configuration** dialog.

Arbutus Beta MongoDB Driver - DSN Configuration

Connection Data Model

DSN Configuration

Data Source Name:

Connection Properties

Basic Advanced

Server *	localhost
Port *	27017
Auth Scheme *	NONE
Database	
Auth Database	
Replica Set	
DNS Server	
Use SSL	False

Server *
The host name or IP address of the server hosting the MongoDB database.

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

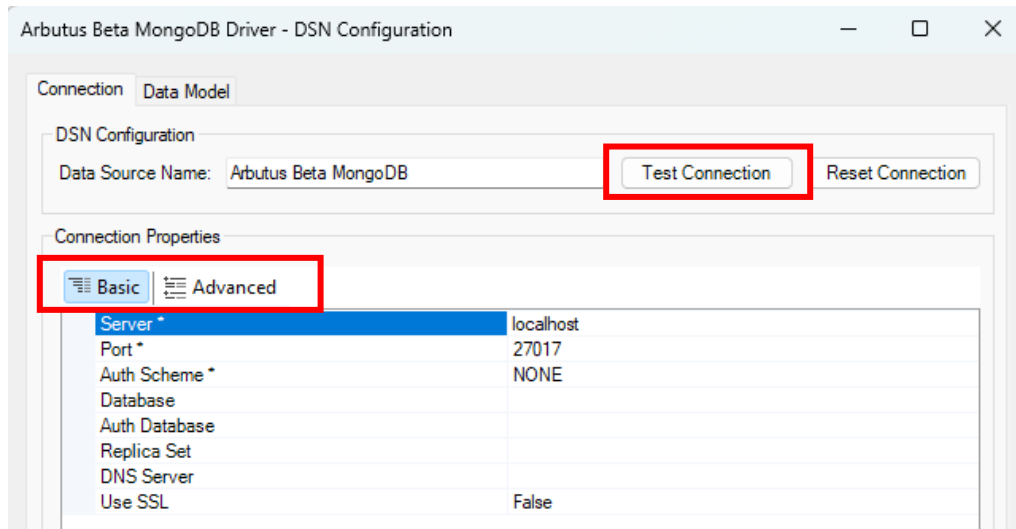
With the DSN Configuration dialog open, the next step is to edit the DSN properties, where necessary, in the **Basic** and **Advanced** tabs. For example, editing the **Auth Scheme properties** (per screenshot below) to ensure correct authentication to the server is applied.

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.

Arbutus Connectors

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



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

1. **Server** – this is the host name or IP address of the server hosting the MongoDB database.

If you choose to connect using DNS seed lists, set this option to "mongodb+srv://" + the name of the server your MongoDB instance is running on.

If connecting through MongoDB Atlas, set the **Server** connection property to the shard value of the primary cluster (ex: cluster0-shard-00-00-test.mongodb.net)

The default value is **localhost**.

2. **Port** – this is the port for the MongoDB database.

The default value is **27017**.

Arbutus Connectors

3. **Auth Scheme** – click the dropdown to select from the list the appropriate authentication mechanism that MongoDB will use to authenticate the connection. The options available for selection are as follows:

- **NONE** – select this if you are connecting to a MongoDB instance that does not require authentication. For example, when working in a development or testing environment where security is not a concern, and you want to simplify the connection process.
- **MONGODB-CR** – MongoDB 2 uses **MONGODB-CR** to authenticate. MONGODB-CR is a challenge-response authentication mechanism that was the default in MongoDB versions prior to 3.0. If you are working with older MongoDB deployments that still use this mechanism, you would select MONGODB-CR to maintain compatibility.
- **SCRAM-SHA-1** – MongoDB 3 uses **SCRAM-SHA-1** by default; new users you create in MongoDB 3 use this authentication method. However, MongoDB 3 servers will continue to use MONGODB-CR to authenticate users created in MongoDB 2.6.
- **SCRAM-SHA-256** – MongoDB 4 supports both **SCRAM-SHA-1** and **SCRAM-SHA-256**. However, **SCRAM-SHA-256** is the **default authentication method** starting from MongoDB version 4.0.

Select this if you want to use a more secure authentication mechanism. **SCRAM-SHA-256** is a salted challenge-response authentication mechanism that uses the SHA-256 hashing algorithm, providing enhanced security compared to SCRAM-SHA-1.

- **PLAIN** – set Auth Scheme to **PLAIN** to use LDAP authentication. This value specifies the SASL PLAIN mechanism; note that this mechanism transmits credentials over plain-text, so it is not suitable for use without TLS/SSL on untrusted networks.

Arbutus Connectors

- **GSSAPI** – set Auth Scheme to **GSSAPI** to use Kerberos authentication. Additionally, configure the following properties (see the Advanced tab of the DSN Configuration dialog) as configured for the MongoDB environment:
 - **KerberosKDC** – this is the FQDN of the domain controller.
 - **KerberosRealm** – this is the Kerberos Realm (for Windows this will be the AD domain)
 - **KerberosSPN** – this is the assigned service principal name for the user
 - **AuthDatabase** – this value should be set to \$external database
 - **User** – this is the user created in the \$external database
 - **Password** – this is the corresponding User's password
 - **X509** – set Auth Scheme to X509 to use X.509 certificate authentication.
4. **Database** – this is the name of the MongoDB database.
 5. **Auth Database** – this is the name of the MongoDB database for authentication. This is only needed if the authentication database is different from the database to retrieve data from.
 6. **Replica Set** – this property allows you to specify multiple servers in addition to the one configured in Server (see above) and Port (see above). Specify both a server name and port; separate server names with a comma. For example:

Server=localhost;Port=27017;ReplicaSet=localhost:27018,localhost:27019;

Arbutus Connectors

This property allows you to specify the other servers in the replica set in addition to the one configured in **Server** (see above) and **Port** (see above). You must specify all servers in the replica set using **Replica Set**, **Server**, and **Port**.

To find the primary server, the driver queries the servers in **Replica Set** and the server specified by **Server** and **Port**.

7. **DNS Server** – this is the DNS server when resolving MongoDB seed list.
8. **Use SSL** – this is a True/False dropdown selection to specify whether SSL is enabled.

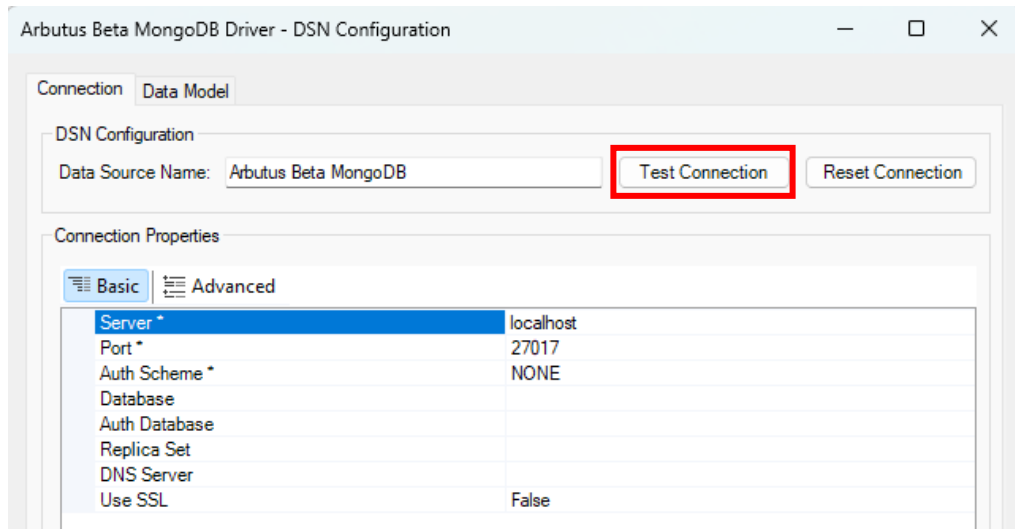
This field sets whether the driver will attempt to negotiate TLS/SSL connections to the server. By default, the driver checks the server's certificate against the system's trusted certificate store.

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 MongoDB system by clicking the **Test Connection** button.

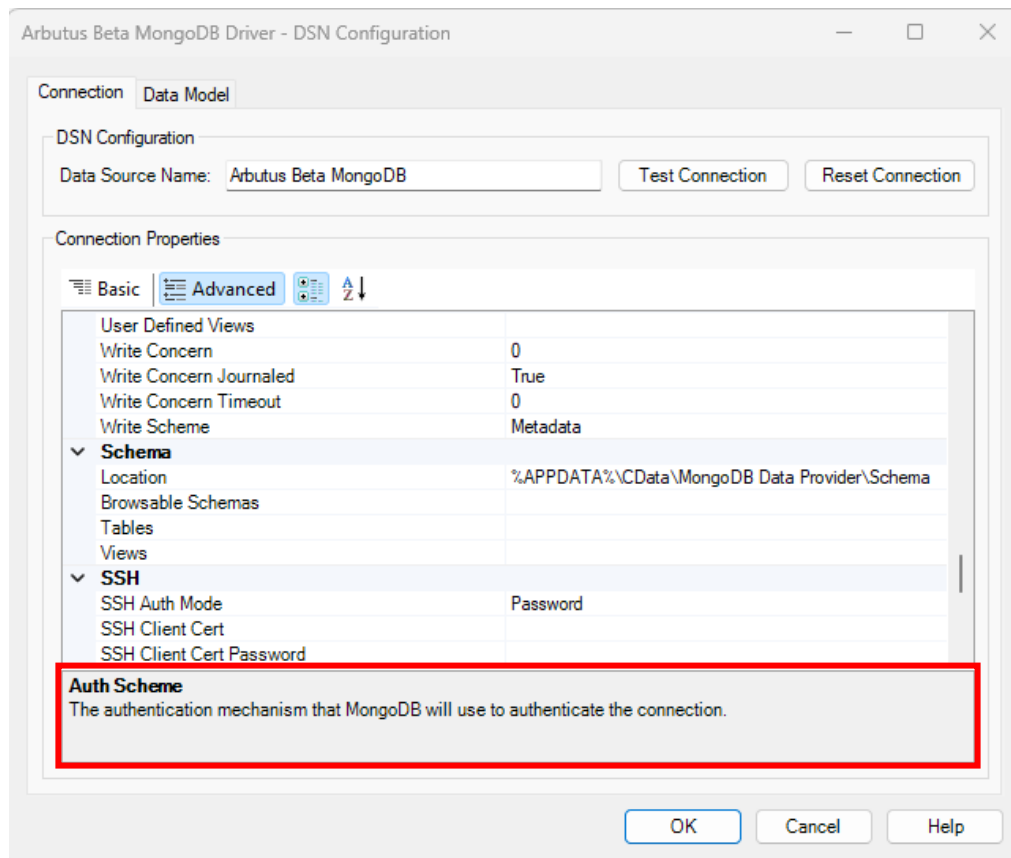
Arbutus Connectors



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.

Arbutus Connectors



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.