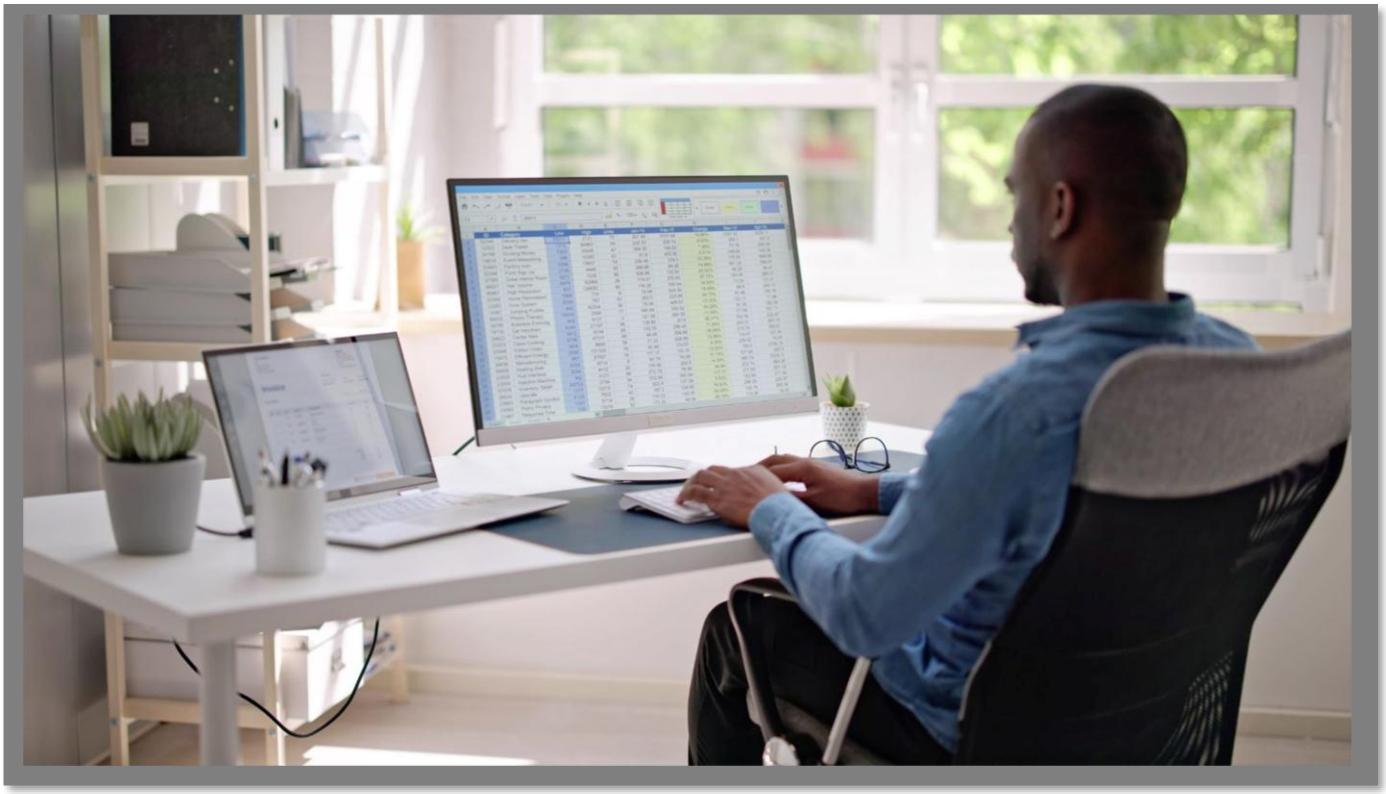
Arbutus Connectors

Apache Spark

CONFIGURATION GUIDE



ARBUTUS

Powerful Analytics Simplified

Contents

Arbutus Connector – Apache Spark

A. Introduction

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

Apache Spark is an open-source, distributed computing system designed for big data processing and analytics. It provides a fast, in-memory data processing engine that can handle large-scale data sets across clusters of computers. Spark supports various workloads, including batch processing, real-time streaming, machine learning, and graph processing. It is known for its speed and scalability compared to other big data frameworks like Hadoop, and it integrates well with tools like Hadoop, SQL, and data science libraries.

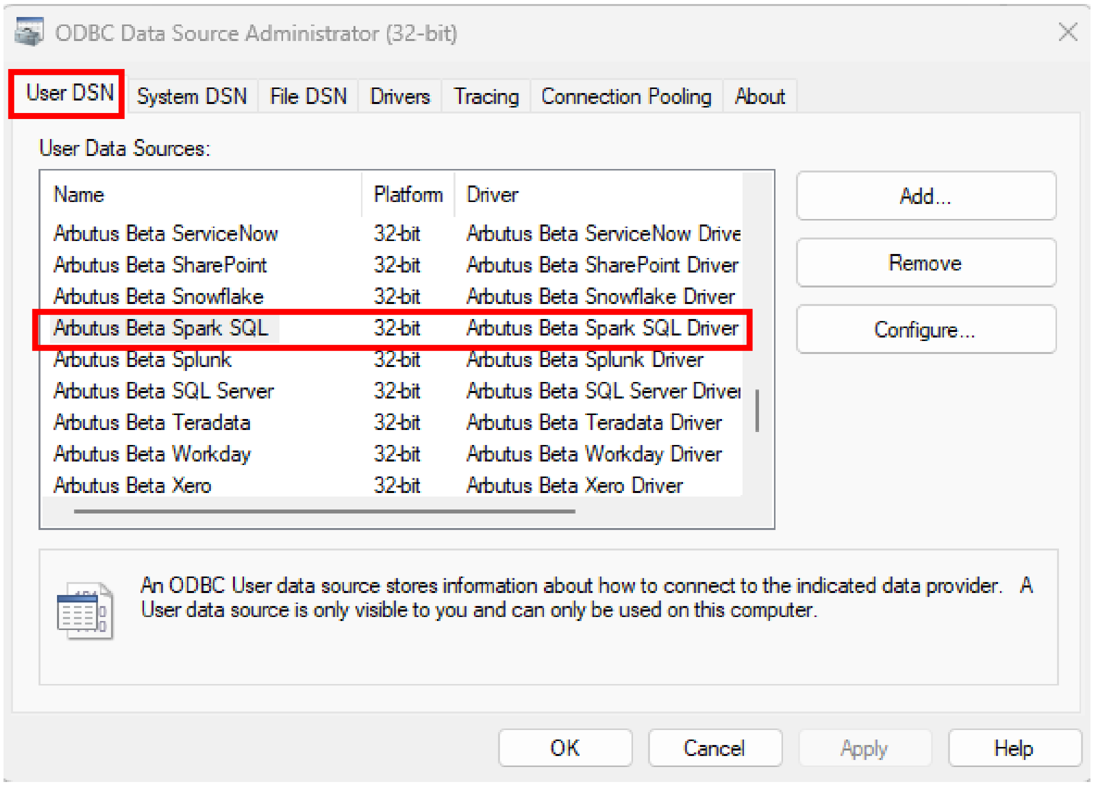
Spark doesn't store data itself; instead, it reads data from distributed storage systems. Common storage options include Hadoop Distributed File System (HDFS), Amazon S3, Azure Blob Storage, local file systems, and databases.

C. Determining if the Connector exists prior to configuring

Installation of the Arbutus Apache Spark 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**.



User Data Sources:

Name	Platform	Driver
Arbutus Beta ServiceNow	32-bit	Arbutus Beta ServiceNow Drive
Arbutus Beta SharePoint	32-bit	Arbutus Beta SharePoint Driver
Arbutus Beta Snowflake	32-bit	Arbutus Beta Snowflake Driver
Arbutus Beta Spark SQL	32-bit	Arbutus Beta Spark SQL Driver
Arbutus Beta Splunk	32-bit	Arbutus Beta Splunk Driver
Arbutus Beta SQL Server	32-bit	Arbutus Beta SQL Server Driver
Arbutus Beta Teradata	32-bit	Arbutus Beta Teradata Driver
Arbutus Beta Workday	32-bit	Arbutus Beta Workday Driver
Arbutus Beta Xero	32-bit	Arbutus Beta Xero Driver

An ODBC User data source stores information about how to connect to the indicated data provider. A User data source is only visible to you and can only be used on this computer.

- If the Arbutus Apache Spark 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.

- ***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.

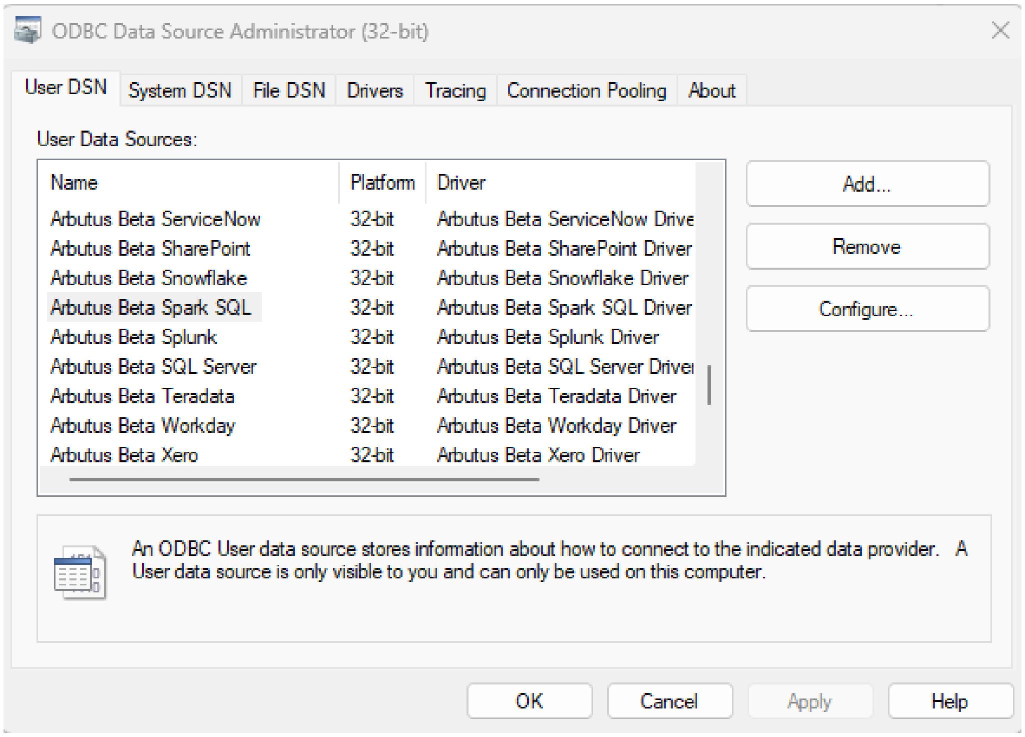
Each of these DSNs has a tab in the **ODBC Data Source Administrator** dialog.

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



Name	Platform	Driver
Arbutus Beta ServiceNow	32-bit	Arbutus Beta ServiceNow Drive
Arbutus Beta SharePoint	32-bit	Arbutus Beta SharePoint Driver
Arbutus Beta Snowflake	32-bit	Arbutus Beta Snowflake Driver
Arbutus Beta Spark SQL	32-bit	Arbutus Beta Spark SQL Driver
Arbutus Beta Splunk	32-bit	Arbutus Beta Splunk Driver
Arbutus Beta SQL Server	32-bit	Arbutus Beta SQL Server Drive
Arbutus Beta Teradata	32-bit	Arbutus Beta Teradata Driver
Arbutus Beta Workday	32-bit	Arbutus Beta Workday Driver
Arbutus Beta Xero	32-bit	Arbutus Beta Xero Driver

An ODBC User data source stores information about how to connect to the indicated data provider. A User data source is only visible to you and can only be used on this computer.

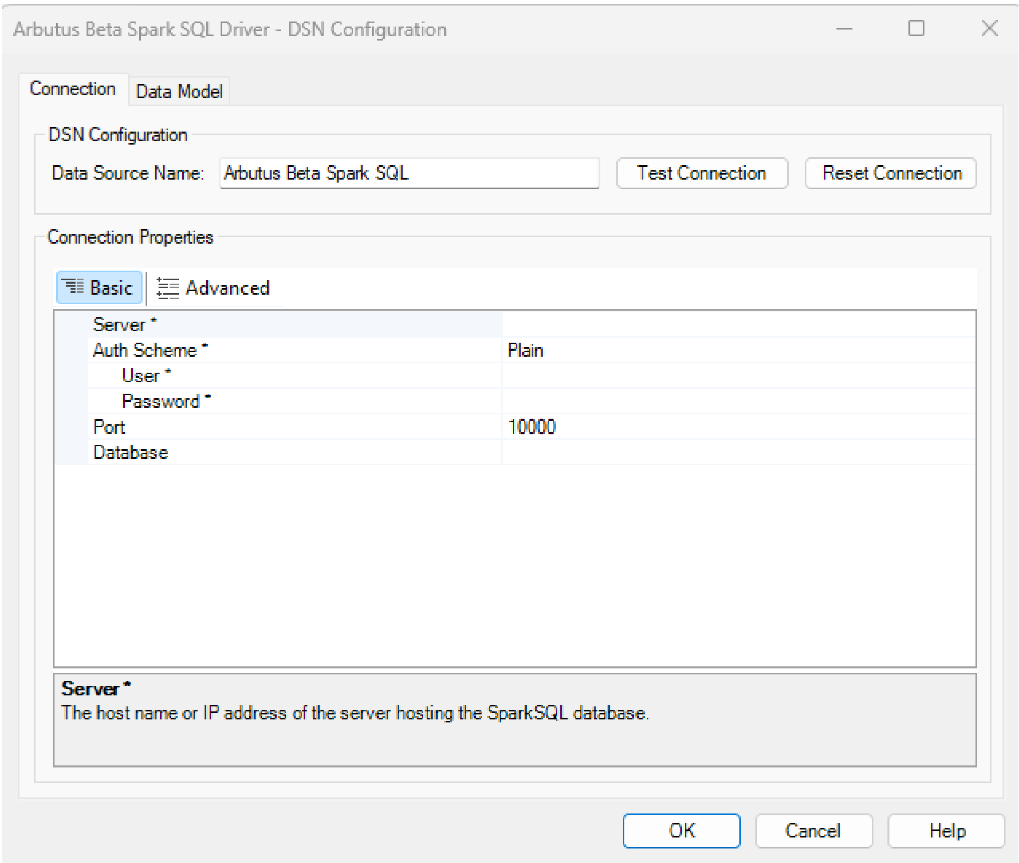
2. Click the **User DSN** tab.

   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 Apache Spark.**

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

This opens the **Arbutus Apache Spark Driver – DSN Configuration** dialog.



Data Source Name: Arbutus Beta Spark SQL

Server *	
Auth Scheme *	Plain
User *	
Password *	
Port	10000
Database	

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

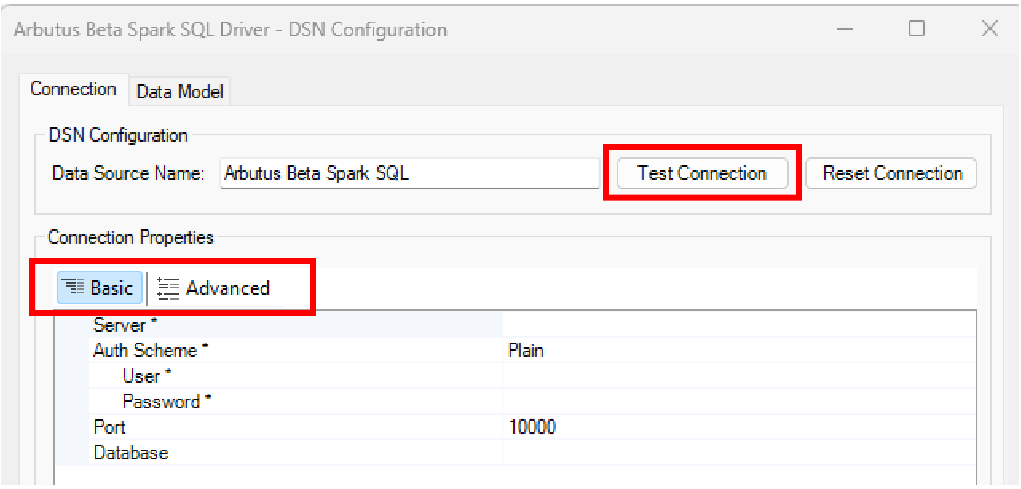
E. Editing the DSN properties – the Basic and Advanced tabs

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.

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



Data Source Name: Arbutus Beta Spark SQL

Server *	
Auth Scheme *	Plain
User *	
Password *	
Port	10000
Database	

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

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

2. **Auth Scheme** – click the dropdown to select from the list the appropriate scheme used for authentication. The options available for selection are as follows:
   - Plain – select this when you need a simple, straightforward method to authenticate users without additional security layers.

     It is important to note that Plain authentication sends credentials in an unencrypted form, which can be a security risk. For production environments or scenarios where security is a concern, consider using more secure authentication methods like Kerberos or SSL/TLS.

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

- User – this is the username used to authenticate with SparkSQL.

- Password – this is the password used to authenticate with SparkSQL.

The default value is **Plain**.

- LDAP – select this when you need centralized user management, enhanced security through encrypted communication, integration with existing LDAP infrastructure, and enforced access control policies. It is ideal for production environments where security and centralized user management are critical.

  LDAP allows you to manage user credentials and permissions centrally, simplifying user management across multiple systems. LDAP provides a more secure authentication method compared to Plain authentication.

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

  - User - this is the username used to authenticate with SparkSQL.

  - Password – this is the password used to authenticate with SparkSQL.

- NoSasl – select this when you need a straightforward configuration without additional security layers.

  It is important to note that **NoSasl** does not provide encryption or secure authentication, so it's not recommended for environments where security is a concern.

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

- User - this is the username used to authenticate with SparkSQL.
- Password – this is the password used to authenticate with SparkSQL.

○ Kerberos – select this when you need robust security for your Spark environment, as Kerberos provides strong encryption and secure authentication. Kerberos is ideal for production environments where security and compliance are critical.

Kerberos supports SSO, allowing users to authenticate once and gain access to multiple services without needing to re-enter credentials.

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

- User - this is the username used to authenticate with SparkSQL.
- Password – this is the password used to authenticate with SparkSQL.
- Kerberos KDC – this is the Kerberos Key Distribution Center (KDC) service used to authenticate the user.

  The Kerberos properties are used when using SPNEGO or Windows Authentication. The driver will request session tickets and temporary session keys from the Kerberos KDC service. The Kerberos KDC service is conventionally collocated with the domain controller.

  If Kerberos KDC is not specified, the driver will attempt to detect these properties automatically from the following locations:

a. KRB5 Config File (lrb5.ini/krb5.conf)
b. Domain Name and Host

- Kerberos Realm – this is the Kerberos Realm used to authenticate the user.

  The Kerberos properties are used when using SPNEGO or Windows Authentication. The Kerberos Realm is used to authenticate the user with the Kerberos Key Distribution Service (KDC). The Kerberos Realm can be configured by an administrator to be any string, but conventionally it is based on the domain name.

  If Kerberos Realm is not specified, the driver will attempt to detect these properties automatically from the following locations:

  a. KRB5 Config File (lrb5.ini/krb5.conf)
  b. Domain Name and Host

- Kerberos SPN – this is the service principal name (SPN) for the Kerberos Domain Controller. If the SPN on the Kerberos Domain Controller is not the same as the URL that you are authenticating to, use this property to set the SPN.

- Sasl Qop – this is a dropdown selection to specify the Quality of protection (Qop) for the SASL framework. The level of quality is negotiated between the client and server during authentication. Used by Kerberos authentication with TCP transport.

  This property should be set to the 'hive.server2.thrift.sasl.qop' value specified in your Hive configuration file (hive-site.xml).

The options available for selection are as follows:

- auth – authentication only
- auth-int – authentication plus integrity protection
- auth-conf – authentication plus integrity and confidentiality protection

The default value is **auth**.

- Kerberos Keytab File – this is the Keytab file containing your pairs of Kerberos principals and encrypted keys.

- Kerberos Ticket Cache – this is the full file path to an MIT Kerberos credential cache file. This property can be set if you wish to use a credential cache file that was created using the MIT Kerberos Ticket Manager or kinit command.

3. **Port** – this is the port for the SparkSQL database.

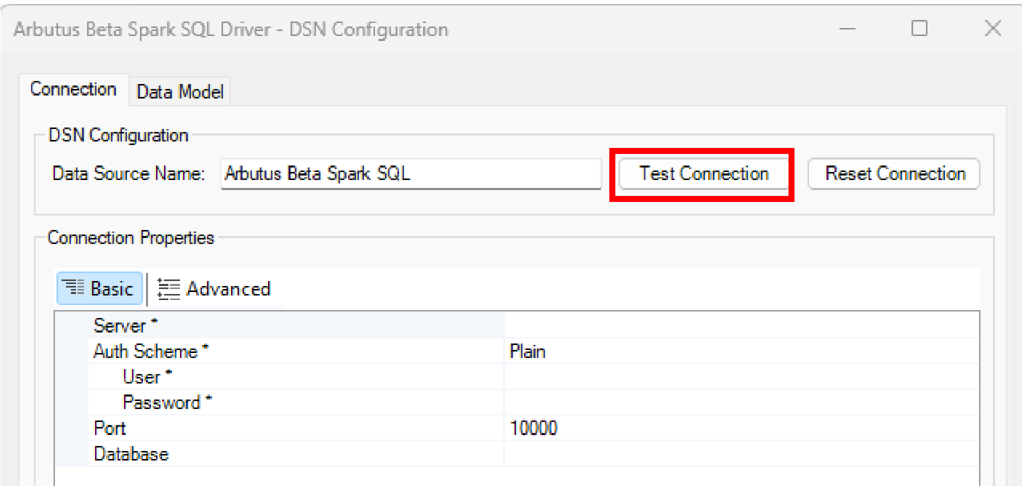
   The default value is **10000**.

4. **Database** – this is the name of the SparkSQL database.

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 Apache Spark system by clicking the **Test Connection** button.

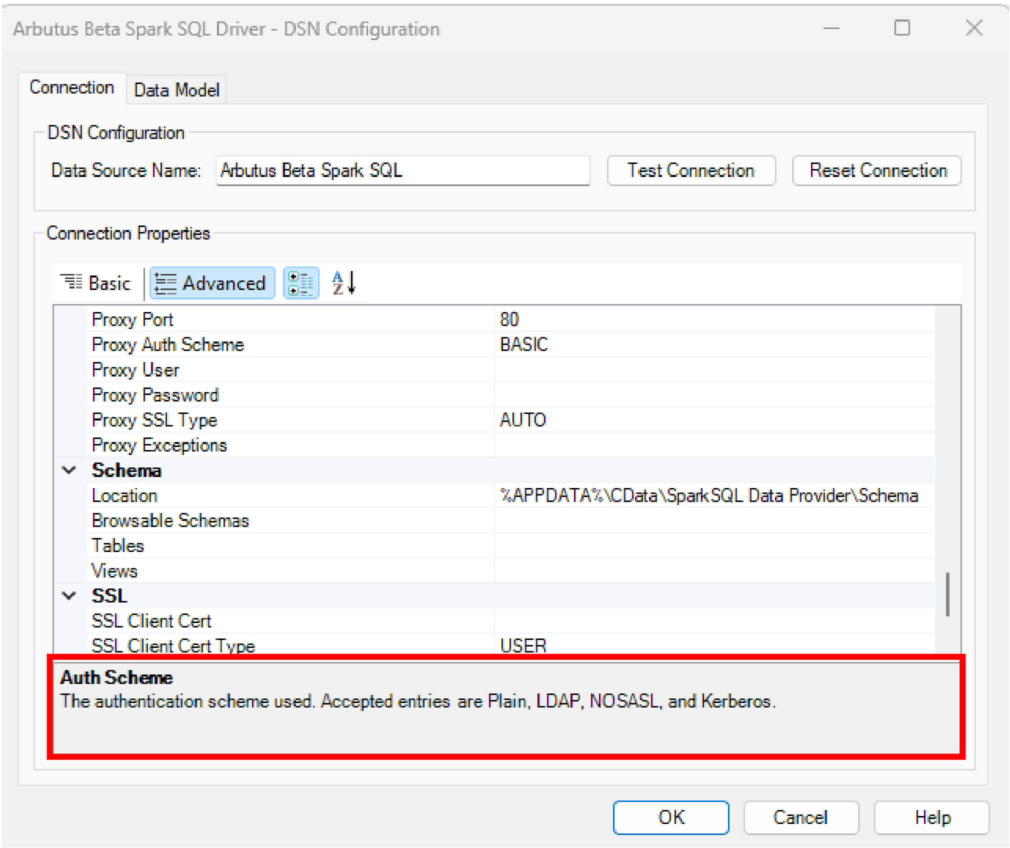


Data Source Name: Arbutus Beta Spark SQL

Server *	
Auth Scheme *	Plain
User *	
Password *	
Port	10000
Database	

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.



Data Source Name: Arbutus Beta Spark SQL

Proxy Port	80
Proxy Auth Scheme	BASIC
Proxy User	
Proxy Password	
Proxy SSL Type	AUTO
Proxy Exceptions	
Schema	
Location	%APPDATA%\CData\SparkSQL Data Provider\Schema
Browsable Schemas	
Tables	
Views	
SSL	
SSL Client Cert	
SSL Client Cert Type	USER

Auth Scheme
The authentication scheme used. Accepted entries are Plain, LDAP, NOSASL, and Kerberos.

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.