

Hortonworks ODBC Driver with SQL Connector for Apache Spark

User Guide

Revised: December 21, 2015



© 2012-2015 Hortonworks Inc. All Rights Reserved.

Parts of this Program and Documentation include proprietary software and content that is copyrighted and licensed by Simba Technologies Incorporated. This proprietary software and content may include one or more feature, functionality or methodology within the ODBC, JDBC, ADO.NET, OLE DB, ODBO, XMLA, SQL and/or MDX component(s).

For information about Simba's products and services, visit: www.simba.com

Table of Contents

Introduction	4
Contact Us	5
Windows Driver	6
Windows System Requirements	6
Installing the Driver	6
Creating a Data Source Name	7
Configuring a DSN-less Connection	9
Configuring Authentication	10
Configuring Advanced Options	15
Configuring Server-Side Properties	17
Configuring HTTP Options	17
Configuring SSL Verification	18
Configuring Logging Options	19
Configuring Kerberos Authentication for Windows	21
Verifying the Version Number	25
Linux Driver	26
Linux System Requirements	26
Installing the Driver	26
Setting the LD_LIBRARY_PATH Environment Variable	27
Verifying the Version Number	28
Mac OS X Driver	29
Mac OS X System Requirements	29
Installing the Driver	29
Setting the DYLD_LIBRARY_PATH Environment Variable	30
Verifying the Version Number	30
Configuring ODBC Connections for Non-Windows Platforms	31
Configuration Files	31
Sample Configuration Files	32
Configuring the Environment	32

Defining DSNs in odbc.ini	33
Specifying ODBC drivers in odbcinst.ini	34
Configuring Driver Settings in hortonworks.sparkodbc.ini	35
Configuring Authentication	36
Configuring SSL Verification	39
Testing the Connection	40
Configuring Logging Options	42
Authentication Options	44
Using a Connection String	46
DSN Connection String Example	46
DSN-less Connection String Examples	46
Features	51
SQL Connector for HiveQL	51
Data Types	51
Catalog and Schema Support	52
spark_system Table	53
Server-Side Properties	53
Get Tables With Query	53
Active Directory	54
Write-back	54
Driver Configuration Options	55
Configuration Options Appearing in the User Interface	55
Configuration Options Having Only Key Names	70
Contact Us	73
Third Party Trademarks and Licenses	74

Introduction

The Hortonworks ODBC Driver with SQL Connector for Apache Spark is used for direct SQL and HiveQL access to Apache Hadoop / Spark distributions, enabling Business Intelligence (BI), analytics, and reporting on Hadoop-based data. The driver efficiently transforms an application's SQL query into the equivalent form in HiveQL, which is a subset of SQL-92. If an application is Spark-aware, then the driver is configurable to pass the query through to the database for processing. The driver interrogates Spark to obtain schema information to present to a SQL-based application. Queries, including joins, are translated from SQL to HiveQL. For more information about the differences between HiveQL and SQL, see "Features" on page 51.

The Hortonworks ODBC Driver with SQL Connector for Apache Spark complies with the ODBC 3.80 data standard and adds important functionality such as Unicode and 32- and 64-bit support for high-performance computing environments.

ODBC is one of the most established and widely supported APIs for connecting to and working with databases. At the heart of the technology is the ODBC driver, which connects an application to the database. For more information about ODBC, see <http://www.simba.com/resources/data-access-standards-library>. For complete information about the ODBC specification, see the *ODBC API Reference* at [http://msdn.microsoft.com/en-us/library/windows/desktop/ms714562\(v=vs.85\).aspx](http://msdn.microsoft.com/en-us/library/windows/desktop/ms714562(v=vs.85).aspx).

The *User Guide* is suitable for users who are looking to access data residing within Hadoop from their desktop environment. Application developers may also find the information helpful. Refer to your application for details on connecting via ODBC.



Architecting the Future of Big Data

Contact Us

If you have difficulty using the Hortonworks ODBC Driver with SQL Connector for Apache Spark, please contact our support staff. We welcome your questions, comments, and feature requests.

Please have a detailed summary of the client and server environment (OS version, patch level, Hadoop distribution version, Spark version, configuration, etc.) ready, before you call or write us. Supplying this information accelerates support.

By telephone:

USA: (855) 8-HORTON

International: (408) 916-4121

On the Internet:

Visit us at www.hortonworks.com

Windows Driver

Windows System Requirements

You install the Hortonworks ODBC Driver with SQL Connector for Apache Spark on client machines that access data stored in a Hadoop cluster with the Spark service installed and running. Each machine that you install the driver on must meet the following minimum system requirements:

- One of the following operating systems:
 - Windows 7 SP1, 8, or 8.1
 - Windows Server 2008 R2 SP1, 2012, or 2012 R2
- 100 MB of available disk space

Important: To install the driver, you must have Administrator privileges on the computer.

The driver supports Apache Spark versions 0.8 through 1.5.

Installing the Driver

On 64-bit Windows operating systems, you can execute both 32- and 64-bit applications. However, 64-bit applications must use 64-bit drivers and 32-bit applications must use 32-bit drivers. Make sure that you use the version of the driver matching the bitness of the client application accessing data in Hadoop / Spark:

- **HortonworksSparkODBC32.msi** for 32-bit applications
- **HortonworksSparkODBC64.msi** for 64-bit applications

You can install both versions of the driver on the same computer.

To install the Hortonworks ODBC Driver with SQL Connector for Apache Spark:

1. Depending on the bitness of your client application, double-click to run **HortonworksSparkODBC32.msi** or **HortonworksSparkODBC64.msi**.
2. Click **Next**.
3. Select the check box to accept the terms of the License Agreement if you agree, and then click **Next**.
4. To change the installation location, click **Change**, then browse to the desired folder, and then click **OK**. To accept the installation location, click **Next**.
5. Click **Install**.
6. When the installation completes, click **Finish**.

Creating a Data Source Name

Typically, after installing the Hortonworks ODBC Driver with SQL Connector for Apache Spark, you need to create a Data Source Name (DSN).

Alternatively, for information about DSN-less connections, see "Configuring a DSN-less Connection" on page 9.

To create a Data Source Name:

1. Open the ODBC Administrator:
 - If you are using Windows 7 or earlier, click the **Start** button , then click **All Programs**, then click the **Hortonworks Spark ODBC Driver 1.1** program group corresponding to the bitness of the client application accessing data in Hadoop / Spark, and then click **ODBC Administrator**.
 - Or, if you are using Windows 8 or later, on the Start screen, type **ODBC administrator**, and then click the **ODBC Administrator** search result corresponding to the bitness of the client application accessing data in Hadoop / Spark.
2. In the ODBC Data Source Administrator, click the **Drivers** tab, and then scroll down as needed to confirm that the Hortonworks Spark ODBC Driver appears in the alphabetical list of ODBC drivers that are installed on your system.
3. Choose one:
 - To create a DSN that only the user currently logged into Windows can use, click the **User DSN** tab.
 - Or, to create a DSN that all users who log into Windows can use, click the **System DSN** tab.
4. Click **Add**.
5. In the Create New Data Source dialog box, select **Hortonworks Spark ODBC Driver** and then click **Finish**.
6. In the **Data Source Name** field, type a name for your DSN.
7. Optionally, in the **Description** field, type relevant details about the DSN.
8. In the **Host** field, type the IP address or host name of the Spark server.
9. In the **Port** field, type the number of the TCP port that the Spark server uses to listen for client connections.
10. In the **Database** field, type the name of the database schema to use when a schema is not explicitly specified in a query.

Note: You can still issue queries on other schemas by explicitly specifying the schema in the query. To inspect your databases and determine the appropriate schema to use, type the **show databases** command at the Spark command prompt.
11. In the **Spark Server Type** list, select the appropriate server type for the version of Spark that you are running:

- If you are running Shark 0.8.1 or earlier, then select **SharkServer**
 - If you are running Shark 0.9.*, then select **SharkServer2**
 - If you are running Spark 1.1 or later, then select **SparkThriftServer**
12. In the Authentication area, configure authentication as needed. For more information, see "Configuring Authentication" on page 10.
Note: Shark Server does not support authentication. Most default configurations of Shark Server 2 or Spark Thrift Server require User Name authentication. To verify the authentication mechanism that you need to use for your connection, check the configuration of your Hadoop / Spark distribution. For more information, see "Authentication Options" on page 44.
 13. Optionally, if the operations against Spark are to be done on behalf of a user that is different than the authenticated user for the connection, type the name of the user to be delegated in the **Delegation UID** field.
Note: This option is applicable only when connecting to a Shark Server 2 or Spark Thrift Server instance that supports this feature.
 14. In the **Thrift Transport** list, select the transport protocol to use in the Thrift layer.
 15. If the Thrift Transport option is set to HTTP, then to configure HTTP options such as custom headers, click **HTTP Options**. For more information, see "Configuring HTTP Options" on page 17.
 16. To configure client-server verification over SSL, click **SSL Options**. For more information, see "Configuring SSL Verification" on page 18.
 17. To configure advanced driver options, click **Advanced Options**. For more information, see "Configuring Advanced Options" on page 15.
 18. To configure server-side properties, click **Advanced Options** and then click **Server Side Properties**. For more information, see "Configuring Server-Side Properties" on page 17.
Important: When connecting to Spark 0.14 or later, the Temporary Tables feature is always enabled and you do not need to configure it in the driver.
 19. To configure logging behavior for the driver, click **Logging Options**. For more information, see "Configuring Logging Options" on page 19.
 20. To test the connection, click **Test**. Review the results as needed, and then click **OK**.
Note: If the connection fails, then confirm that the settings in the Hortonworks Spark ODBC Driver DSN Setup dialog box are correct. Contact your Spark server administrator as needed.
 21. To save your settings and close the Hortonworks Spark ODBC Driver DSN Setup dialog box, click **OK**.
 22. To close the ODBC Data Source Administrator, click **OK**.

Configuring a DSN-less Connection

Some client applications provide support for connecting to a data source using a driver without a Data Source Name (DSN). To configure a DSN-less connection, you can use a connection string or the Hortonworks Spark ODBC Driver Configuration tool that is installed with the Hortonworks ODBC Driver with SQL Connector for Apache Spark. The following section explains how to use the driver configuration tool. For information about using connection strings, see "DSN-less Connection String Examples" on page 46.

To configure a DSN-less connection using the driver configuration tool:

1. Choose one:
 - If you are using Windows 7 or earlier, click the **Start** button , then click **All Programs**, and then click the **Hortonworks Spark ODBC Driver 1.1** program group corresponding to the bitness of the client application accessing data in Hadoop / Spark.
 - Or, if you are using Windows 8 or later, click the arrow button at the bottom of the Start screen, and then find the **Hortonworks Spark ODBC Driver 1.1** program group corresponding to the bitness of the client application accessing data in Hadoop / Spark.
2. Click **Driver Configuration**, and then click **OK** if prompted for administrator permission to make modifications to the computer.
Note: You must have administrator access to the computer to run this application because it makes changes to the registry.
3. In the Spark Server Type list, select the appropriate server type for the version of Spark that you are running:
 - If you are running Shark 0.8.1 or earlier, then select **SharkServer**.
 - If you are running Shark 0.9.*, then select **SharkServer2**.
 - If you are running Spark 1.1 or later, then select **SparkThriftServer**.
4. In the Authentication area, configure authentication as needed. For more information, see "Configuring Authentication" on page 10.
Note: Spark Server 1 does not support authentication. Most default configurations of Spark Server 2 require User Name authentication. To verify the authentication mechanism that you need to use for your connection, check the configuration of your Hadoop / Spark distribution. For more information, see "Authentication Options" on page 44.
5. Optionally, if the operations against Spark are to be done on behalf of a user that is different than the authenticated user for the connection, then in the **Delegation UID** field, type the name of the user to be delegated.
Note: This option is applicable only when connecting to a Shark Server 2 or Spark Thrift Server instance that supports this feature.
6. In the **Thrift Transport** list, select the transport protocol to use in the Thrift layer.

7. If the Thrift Transport option is set to HTTP, then to configure HTTP options such as custom headers, click **HTTP Options**. For more information, see "Configuring HTTP Options" on page 17.
8. To configure client-server verification over SSL, click **SSL Options**. For more information, see "Configuring SSL Verification" on page 18.
9. To configure advanced options, click **Advanced Options**. For more information, see "Configuring Advanced Options" on page 15.
10. To configure server-side properties, click **Advanced Options** and then click **Server Side Properties**. For more information, see "Configuring Server-Side Properties" on page 17.
11. To save your settings and close the Hortonworks Spark ODBC Driver Configuration tool, click **OK**.

Configuring Authentication

Some Spark servers are configured to require authentication for access. To connect to a Spark server, you must configure the Hortonworks ODBC Driver with SQL Connector for Apache Spark to use the authentication mechanism that matches the access requirements of the server and provides the necessary credentials.

For information about how to determine the type of authentication your Spark server requires, see "Authentication Options" on page 44.

ODBC applications that connect to Shark Server 2 or Spark Thrift Server using a DSN can pass in authentication credentials by defining them in the DSN. To configure authentication for a connection that uses a DSN, use the ODBC Data Source Administrator.

Normally, applications that are not Shark Server 2 or Spark Thrift Server aware and that connect using a DSN-less connection do not have a facility for passing authentication credentials to the Hortonworks ODBC Driver with SQL Connector for Apache Spark for a connection. However, the Hortonworks Spark ODBC Driver Configuration tool enables you to configure authentication without using a DSN.

Important: Credentials defined in a DSN take precedence over credentials configured using the driver configuration tool. Credentials configured using the driver configuration tool apply for all connections that are made using a DSN-less connection unless the client application is Shark Server 2 or Spark Thrift Server aware and requests credentials from the user.

Using No Authentication

When connecting to a Spark server of type Shark Server, you must use No Authentication. When you use No Authentication, Binary is the only Thrift transport protocol that is supported.

To configure a connection without authentication:

1. Choose one:
 - To access authentication options for a DSN, open the ODBC Data Source Administrator where you created the DSN, then select the DSN, and then click **Configure**.
 - Or, to access authentication options for a DSN-less connection, open the Hortonworks Spark ODBC Driver Configuration tool.
2. In the Mechanism list, select **No Authentication**.
3. If the Spark server is configured to use SSL, then click **SSL Options** to configure SSL for the connection. For more information, see "Configuring SSL Verification" on page 18.
4. To save your settings and close the dialog box, click **OK**

Example connection string for Shark Server:

```
Driver=Hortonworks Spark ODBC Driver;Host=server_  
host;Port=10000;SparkServerType=1;AuthMech=0;Schema=Spark_database
```

Example connection string for Shark Server 2:

```
Driver=Hortonworks Spark ODBC Driver;Host=server_  
host;Port=10000;SparkServerType=2;AuthMech=0;Schema=Spark_database
```

Example connection string for Spark Thrift Server:

```
Driver=Hortonworks Spark ODBC Driver;Host=server_  
host;Port=10000;SparkServerType=3;AuthMech=0;Schema=Spark_database
```

Using Kerberos

Kerberos must be installed and configured before you can use this authentication mechanism. For more information, see "Configuring Kerberos Authentication for Windows" on page 21.

Note: This authentication mechanism is available only for Shark Server 2 or Spark Thrift Server on non-HDInsight distributions.

To configure Kerberos authentication:

1. Choose one:
 - To access authentication options for a DSN, open the ODBC Data Source Administrator where you created the DSN, then select the DSN, and then click **Configure**.
 - Or, to access authentication options for a DSN-less connection, open the Hortonworks Spark ODBC Driver Configuration tool.
2. In the **Mechanism** list, select **Kerberos**.
3. Choose one:

- To use the default realm defined in your Kerberos setup, leave the **Realm** field empty.
 - Or, if your Kerberos setup does not define a default realm or if the realm of your Shark Server 2 or Spark Thrift Server host is not the default, then, in the **Realm** field, type the Kerberos realm of the Shark Server 2 or Spark Thrift Server.
4. In the **Host FQDN** field, type the fully qualified domain name of the Shark Server 2 or Spark Thrift Server host.
Note: To use the Spark server host name as the fully qualified domain name for Kerberos authentication, in the **Host FQDN** field, type `_HOST`.
 5. In the **Service Name** field, type the service name of the Spark server.
 6. In the **Thrift Transport** list, select the transport protocol to use in the Thrift layer.
Important: When using this authentication mechanism, the Binary transport protocol is not supported.
 7. If the Spark server is configured to use SSL, then click **SSL Options** to configure SSL for the connection. For more information, see "Configuring SSL Verification" on page 18.
 8. To save your settings and close the dialog box, click **OK**.

Example connection string:

```
Driver=Hortonworks Spark ODBC Driver;Host=server_  
host;Port=10000;SparkServerType=2;AuthMech=1;Schema=Spark_  
database;KrbRealm=Kerberos_realm;KrbHostFQDN=domain_  
name;KrbServiceName=service_name
```

Using User Name

This authentication mechanism requires a user name but not a password. The user name labels the session, facilitating database tracking.

Note: This authentication mechanism is available only for Shark Server 2 or Spark Thrift Server on non-HDInsight distributions. Most default configurations of Shark Server 2 or Spark Thrift Server require User Name authentication.

To configure User Name authentication:

1. Choose one:
 - To access authentication options for a DSN, open the ODBC Data Source Administrator where you created the DSN, then select the DSN, and then click **Configure**.
 - Or, to access authentication options for a DSN-less connection, open the Hortonworks Spark ODBC Driver Configuration tool.
2. In the **Mechanism** list, select **User Name**
3. In the **User Name** field, type an appropriate user name for accessing the Spark server.

4. To save your settings and close the dialog box, click **OK**.

Example connection string:

```
Driver=Hortonworks Spark ODBC Driver;Host=server_  
host;Port=10000;SparkServerType=2;AuthMech=2;Schema=Spark_  
database;UID=user_name
```

Using User Name and Password

This authentication mechanism requires a user name and a password.

To configure User Name and Password authentication:

1. Choose one:
 - To access authentication options for a DSN, open the ODBC Data Source Administrator where you created the DSN, then select the DSN, and then click **Configure**.
 - Or, to access authentication options for a DSN-less connection, open the Hortonworks Spark ODBC Driver Configuration tool.
2. In the **Mechanism** list, select **User Name and Password**
3. In the **User Name** field, type an appropriate user name for accessing the Spark server.
4. In the **Password** field, type the password corresponding to the user name you typed in step 3.
5. To save the password, select the **Save Password (Encrypted)** check box.
Important: The password is obscured, that is, not saved in plain text. However, it is still possible for the encrypted password to be copied and used.
6. In the **Thrift Transport** list, select the transport protocol to use in the Thrift layer.
7. If the Spark server is configured to use SSL, then click **SSL Options** to configure SSL for the connection. For more information, see "Configuring SSL Verification" on page 18.
8. To save your settings and close the dialog box, click **OK**.

Example connection string:

```
Driver=Hortonworks Spark ODBC Driver;Host=server_  
host;Port=10000;SparkServerType=2;AuthMech=3;Schema=Spark_  
database;UID=user_name;PWD=password
```

Using Windows Azure HDInsight Emulator

This authentication mechanism is available only for Shark Server 2 or Spark Thrift Server instances running on Windows Azure HDInsight Emulator.

To configure a connection to a Spark server on Windows Azure HDInsight Emulator:

1. Choose one:
 - To access authentication options for a DSN, open the ODBC Data Source Administrator where you created the DSN, then select the DSN, and then click **Configure**.
 - Or, to access authentication options for a DSN-less connection, open the Hortonworks Spark ODBC Driver Configuration tool.
2. In the **Mechanism** list, select **Windows Azure HDInsight Emulator**.
3. In the **HTTP Path** field, type the partial URL corresponding to the Spark server.
4. In the **User Name** field, type an appropriate user name for accessing the Spark server.
5. In the **Password** field, type the password corresponding to the user name you typed in step 4.
6. To save your settings and close the dialog box, click **OK**.

Example connection string:

```
Driver=Hortonworks Spark ODBC Driver;Host=server_  
host;Port=10000;SparkServerType=2;AuthMech=5;Schema=Spark_  
database;UID=user_name;PWD=password;HTTPath=Spark_HTTP_path
```

Using Windows Azure HDInsight Service

This authentication mechanism is available only for Shark Server 2 or Spark Thrift Server on HDInsight distributions.

To configure a connection to a Spark server on Windows Azure HDInsight Service:

1. Choose one:
 - To access authentication options for a DSN, open the ODBC Data Source Administrator where you created the DSN, then select the DSN, and then click **Configure**.
 - Or, to access authentication options for a DSN-less connection, open the Hortonworks Spark ODBC Driver Configuration tool.
2. In the **Mechanism** list, select **Windows Azure HDInsight Service**.
3. In the **HTTP Path** field, type the partial URL corresponding to the Spark server.
4. In the **User Name** field, type an appropriate user name for accessing the Spark server.
5. In the **Password** field, type the password corresponding to the user name you typed in step 4.
6. Click **HTTP Options**, and in the **HTTP Path** field, type the partial URL corresponding to the Spark server. Click **OK** to save your HTTP settings and close the dialog box.

Note: If necessary, you can create custom HTTP headers. For more information, see "Configuring HTTP Options" on page 17.

7. Choose one:
 - To configure the driver to load SSL certificates from a specific PEM file, click **Advanced Options** and type the path to the file in the **Trusted Certificates** field.
 - Or, to use the trusted CA certificates PEM file that is installed with the driver, leave the **Trusted Certificates** field empty.
8. Click **SSL Options** and configure SSL settings as needed. For more information, see "Configuring SSL Verification" on page 18.
9. Click **OK** to save your SSL configuration and close the dialog box, and then click **OK** to save your authentication settings and close the dialog box.

Example connection string:

```
Driver=Hortonworks Spark ODBC Driver;Host=server_  
host;Port=10000;SparkServerType=2;AuthMech=6;Schema=Spark_  
database;UID=user_name;PWD=password;HTTPath=Spark_HTTP_path
```

Configuring Advanced Options

You can configure advanced options to modify the behavior of the driver.

To configure advanced options:

1. Choose one:
 - To access advanced options for a DSN, open the ODBC Data Source Administrator where you created the DSN, then select the DSN, then click **Configure**, and then click **Advanced Options**.
 - Or, to access advanced options for a DSN-less connection, open the Hortonworks Spark ODBC Driver Configuration tool, and then click **Advanced Options**.
2. To disable the SQL Connector feature, select the **Use Native Query** check box.
3. To defer query execution to SQLExecute, select the **Fast SQLPrepare** check box.
4. To allow driver-wide configurations to take precedence over connection and DSN settings, select the **Driver Config Take Precedence** check box.
5. To use the asynchronous version of the API call against Spark for executing a query, select the **Use Async Exec** check box.
6. To retrieve the names of tables in a database by using the SHOW TABLES query, select the **Get Tables With Query** check box.

Note: This option is applicable only when connecting to Shark Server 2 or Spark Thrift Server.

7. To enable the driver to return SQL_WVARCHAR instead of SQL_VARCHAR for STRING and VARCHAR columns, and SQL_WCHAR instead of SQL_CHAR for CHAR columns, select the **Unicode SQL character types** check box.
8. To enable the driver to return the spark_system table for catalog function calls such as SQLTables and SQLColumns, select the **Show System Table** check box.
9. To handle Kerberos authentication using the SSPI plugin instead of MIT Kerberos by default, select the **Use Only SSPI** check box.
10. To enable the driver to automatically open a new session when the existing session is no longer valid, select the **Invalid Session Auto Recover** check box.
Note: This option is applicable only when connecting to Shark Server 2 or Spark Thrift Server.
11. In the **Rows Fetched Per Block** field, type the number of rows to be fetched per block.
12. In the **Default String Column Length** field, type the maximum data length for STRING columns.
13. In the **Binary Column Length** field, type the maximum data length for BINARY columns.
14. In the **Decimal Column Scale** field, type the maximum number of digits to the right of the decimal point for numeric data types.
15. In the **Async Exec Poll Interval (ms)** field, type the time in milliseconds between each poll for the query execution status.
Note: This option is applicable only to HDInsight clusters.
16. To allow the common name of a CA-issued SSL certificate to not match the host name of the Spark server, select the **Allow Common Name Host Name Mismatch** check box.
Note: This option is applicable only to the User Name and Password (SSL) and HTTPS authentication mechanisms.
17. Choose one:
 - To configure the driver to load SSL certificates from a specific PEM file, type the path to the file in the **Trusted Certificates** field.
 - Or, to use the trusted CA certificates PEM file that is installed with the driver, leave the **Trusted Certificates** field empty.
Note: This option is applicable only to the User Name and Password (SSL), Windows Azure HDInsight Service, and HTTPS authentication mechanisms.
18. In the **Socket Timeout** field, type the number of seconds that an operation can remain idle before it is closed.
Note: This option is applicable only when asynchronous query execution is being used against Shark Server 2 or Spark Thrift Server instances.
19. To save your settings and close the Advanced Options dialog box, click **OK**.

Configuring Server-Side Properties

You can use the driver to apply configuration properties to the Spark server.

To configure server-side properties:

1. Choose one:
 - To configure server-side properties for a DSN, open the ODBC Data Source Administrator where you created the DSN, then select the DSN, then click **Configure**, then click **Advanced Options**, and then click **Server Side Properties**.
 - Or, to configure server-side properties for a DSN-less connection, open the Hortonworks Spark ODBC Driver Configuration tool, then click **Advanced Options**, and then click **Server Side Properties**.
2. To create a server-side property, click **Add**, then type appropriate values in the **Key** and **Value** fields, and then click **OK**.
3. To edit a server-side property, select the property from the list, then click **Edit**, then update the Key and Value fields as needed, and then click **OK**.
4. To delete a server-side property, select the property from the list, and then click **Remove**. In the confirmation dialog box, click **Yes**.
5. To configure the driver to apply each server-side property by executing a query when opening a session to the Spark server, select the **Apply Server Side Properties with Queries** check box.

Or, to configure the driver to use a more efficient method for applying server-side properties that does not involve additional network round-tripping, clear the **Apply Server Side Properties with Queries** check box.

Note: The more efficient method is not available for Shark Server, and it might not be compatible with some Shark Server 2 or Spark Thrift Server builds. If the server-side properties do not take effect when the check box is clear, then select the check box.

6. To force the driver to convert server-side property key names to all lower case characters, select the **Convert Key Name to Lower Case** check box.
7. To save your settings and close the Server Side Properties dialog box, click **OK**.

Configuring HTTP Options

You can configure options such as custom headers when using the HTTP transport protocol in the Thrift layer.

To configure HTTP options:

1. Chose one:
 - If you are configuring HTTP for a DSN, open the ODBC Data Source Administrator where you created the DSN, then select the DSN, then click **Configure**, and then ensure that the Thrift Transport option is set to **HTTP**.
 - Or, if you are configuring HTTP for a DSN-less connection, open the Hortonworks Spark ODBC Driver Configuration tool and then ensure that the Thrift Transport option is set to **HTTP**.
2. To access HTTP options, click **HTTP Options**.
Note: The HTTP options are available only when the Thrift Transport option is set to HTTP.
3. In the **HTTP Path** field, type the partial URL corresponding to the Spark server.
4. To create a custom HTTP header, click **Add**, then type appropriate values in the **Key** and **Value** fields, and then click **OK**.
5. To edit a custom HTTP header, select the header from the list, then click **Edit**, then update the **Key** and **Value** fields as needed, and then click **OK**.
6. To delete a custom HTTP header, select the header from the list, and then click **Remove**. In the confirmation dialog box, click **Yes**.
7. To save your settings and close the HTTP Options dialog box, click **OK**.

Configuring SSL Verification

You can configure verification between the client and the Spark server over SSL.

To configure SSL verification:

1. Choose one:
 - To access SSL options for a DSN, open the ODBC Data Source Administrator where you created the DSN, then select the DSN, then click **Configure**, and then click **SSL Options**.
 - Or, to access advanced options for a DSN-less connection, open the Hortonworks Spark ODBC Driver Configuration tool, and then click **SSL Options**.
2. Select the **Enable SSL** check box.
3. To allow self-signed certificates from the server, select the **Allow Self-signed Server Certificate** check box.
4. To allow the common name of a CA-issued SSL certificate to not match the host name of the Spark server, select the **Allow Common Name Host Name Mismatch** check box.
5. Choose one:
 - To configure the driver to load SSL certificates from a specific PEM file when

verifying the server, specify the full path to the file in the **Trusted Certificates** field.

- Or, to use the trusted CA certificates PEM file that is installed with the driver, leave the **Trusted Certificates** field empty.
6. To configure two-way SSL verification, select the **Two Way SSL** check box and then do the following:
 - a) In the **Client Certificate File** field, specify the full path of the PEM file containing the client's certificate.
 - b) In the **Client Private Key File** field, specify the full path of the file containing the client's private key.
 - c) If the private key file is protected with a password, type the password in the **Client Private Key Password** field. To save the password, select the **Save Password (Encrypted)** check box.
Important: The password is obscured, that is, not saved in plain text. However, it is still possible for the encrypted password to be copied and used.
 7. To save your settings and close the SSL Options dialog box, click **OK**.

Configuring Logging Options

To help troubleshoot issues, you can enable logging. In addition to functionality provided in the Hortonworks ODBC Driver with SQL Connector for Apache Spark, the ODBC Data Source Administrator provides tracing functionality.

Important: Only enable logging or tracing long enough to capture an issue. Logging or tracing decreases performance and can consume a large quantity of disk space.

The driver allows you to set the amount of detail included in log files. Table 1 lists the logging levels provided by the Hortonworks ODBC Driver with SQL Connector for Apache Spark, in order from least verbose to most verbose.

Table 1. Hortonworks ODBC Driver with SQL Connector for Apache Spark Logging Levels

Logging Level	Description
OFF	Disables all logging.
FATAL	Logs very severe error events that will lead the driver to abort.
ERROR	Logs error events that might still allow the driver to continue running.

Logging Level	Description
WARNING	Logs potentially harmful situations.
INFO	Logs general information that describes the progress of the driver.
DEBUG	Logs detailed information that is useful for debugging the driver.
TRACE	Logs more detailed information than the DEBUG level.

To enable driver logging:

1. To access logging options, open the ODBC Data Source Administrator where you created the DSN, then select the DSN, then click **Configure**, and then click **Logging Options**.
2. From the **Log Level** drop-down list, select the desired level of information to include in log files.
3. In the **Log Path** field, specify the full path to the folder where you want to save log files.
4. In the **Max Number Files** field, type the maximum number of log files to keep.
Note: After the maximum number of log files is reached, each time an additional file is created, the driver deletes the oldest log file.
5. In the **Max File Size** field, type the maximum size of each log file in megabytes (MB).
Note: After the maximum file size is reached, the driver creates a new file and continues logging.
6. Click **OK**.
7. Restart your ODBC application to ensure that the new settings take effect.

The Hortonworks ODBC Driver with SQL Connector for Apache Spark produces a log file named SparkODBC_driver.log at the location that you specify in the Log Path field.

To disable driver logging:

1. Open the ODBC Data Source Administrator where you created the DSN, then select the DSN, then click **Configure**, and then click **Logging Options**.
2. From the **Log Level** drop-down list, select **LOG_OFF**.
3. Click **OK**.

To start tracing using the ODBC Data Source Administrator:

1. In the ODBC Data Source Administrator, click the **Tracing** tab.

2. In the **Log File Path** area, click **Browse**. In the Select ODBC Log File dialog box, browse to the location where you want to save the log file, then type a descriptive file name in the **File name** field, and then click **Save**.
3. On the Tracing tab, click **Start Tracing Now**.

To stop ODBC Data Source Administrator tracing:

- On the Tracing tab in the ODBC Data Source Administrator, click **Stop Tracing Now**.

For more information about tracing using the ODBC Data Source Administrator, see the article *How to Generate an ODBC Trace with ODBC Data Source Administrator* at <http://support.microsoft.com/kb/274551>.

Configuring Kerberos Authentication for Windows

Active Directory

The Hortonworks ODBC Driver with SQL Connector for Apache Spark supports Active Directory Kerberos on Windows. There are two prerequisites for using Active Directory Kerberos on Windows:

- MIT Kerberos is not installed on the client Windows machine.
- The MIT Kerberos Hadoop realm has been configured to trust the Active Directory realm so that users in the Active Directory realm can access services in the MIT Kerberos Hadoop realm. For more information, see *Setting up One-Way Trust with Active Directory* in the Hortonworks documentation at http://docs.hortonworks.com/HDPDocuments/HDP2/HDP-2.1.7/bk_installing_manually_book/content/ch23s05.html.

MIT Kerberos

Downloading and Installing MIT Kerberos for Windows 4.0.1

For information about Kerberos and download links for the installer, see the MIT Kerberos website at <http://web.mit.edu/kerberos/>.

To download and install MIT Kerberos for Windows 4.0.1:

1. Download the appropriate Kerberos installer:
 - For a 64-bit computer, use the following download link from the MIT Kerberos website: <http://web.mit.edu/kerberos/dist/kfw/4.0/kfw-4.0.1-amd64.msi>.
 - For a 32-bit computer, use the following download link from the MIT Kerberos website: <http://web.mit.edu/kerberos/dist/kfw/4.0/kfw-4.0.1-i386.msi>.

Note: The 64-bit installer includes both 32-bit and 64-bit libraries. The 32-bit installer includes 32-bit libraries only.

2. To run the installer, double-click the .msi file that you downloaded in step 1.

3. Follow the instructions in the installer to complete the installation process.
4. When the installation completes, click **Finish**.

Setting Up the Kerberos Configuration File

Settings for Kerberos are specified through a configuration file. You can set up the configuration file as a .INI file in the default location, which is the C:\ProgramData\MIT\Kerberos5 directory, or as a .CONF file in a custom location.

Normally, the C:\ProgramData\MIT\Kerberos5 directory is hidden. For information about viewing and using this hidden directory, refer to Microsoft Windows documentation.

Note: For more information on configuring Kerberos, refer to the MIT Kerberos documentation.

To set up the Kerberos configuration file in the default location:

1. Obtain a krb5.conf configuration file . You can obtain this file from your Kerberos administrator, or from the /etc/krb5.conf folder on the computer that is hosting the Shark Server 2 or Spark Thrift Server instance.
2. Rename the configuration file from krb5.conf to krb5.ini.
3. Copy the krb5.ini file to the C:\ProgramData\MIT\Kerberos5 directory and overwrite the empty sample file.

To set up the Kerberos configuration file in a custom location:

1. Obtain a krb5.conf configuration file . You can obtain this file from your Kerberos administrator, or from the /etc/krb5.conf folder on the computer that is hosting the Shark Server 2 or Spark Thrift Server instance.
2. Place the krb5.conf file in an accessible directory and make note of the full path name.
3. Open your computer Properties window:
 - If you are using Windows 7 or earlier, click the **Start** button , then right-click **Computer**, and then click **Properties**.
 - Or, if you are using Windows 8 or later, right-click **This PC** on the Start screen, and then click **Properties**.
4. Click **Advanced System Settings**.
5. In the System Properties dialog box, click the **Advanced** tab and then click **Environment Variables**.
6. In the Environment Variables dialog box, under the System variables list, click **New**.
7. In the New System Variable dialog box, in the Variable Name field, type **KRB5_CONFIG**.
8. In the **Variable Value** field, type the absolute path to the krb5.conf file from step 2.
9. Click **OK** to save the new variable.
10. Make sure that the variable is listed in the System variables list..

11. Click **OK** to close the Environment Variables dialog box, and then click **OK** to close the System Properties dialog box.

Setting Up the Kerberos Credential Cache File

Kerberos uses a credential cache to store and manage credentials.

To set up the Kerberos credential cache file:

1. Create a directory where you want to save the Kerberos credential cache file. For example, create a directory named C:\temp.
2. Open your computer Properties window:
 - If you are using Windows 7 or earlier, click the **Start** button , then right-click **Computer**, and then click **Properties**.
 - Or, if you are using Windows 8 or later, right-click **This PC** on the Start screen, and then click **Properties**.
3. Click **Advanced System Settings**
4. In the System Properties dialog box, click the **Advanced** tab and then click **Environment Variables**.
5. In the Environment Variables dialog box, under the System Variables list, click **New**.
6. In the New System Variable dialog box, in the **Variable Name** field, type **KRB5CCNAME**.
7. In the **Variable Value** field, type the path to the folder you created in step 1, and then append the file name krb5cache. For example, if you created the folder C:\temp in step 1, then type C:\temp\krb5cache.
Note: krb5cache is a file (not a directory) that is managed by the Kerberos software, and it should not be created by the user. If you receive a permission error when you first use Kerberos, ensure that the krb5cache file does not already exist as a file or a directory.
8. Click **OK** to save the new variable.
9. Make sure that the variable appears in the System Variables list.
10. Click **OK** to close the Environment Variables dialog box, and then click **OK** to close the System Properties dialog box.
11. To ensure that Kerberos uses the new settings, restart your computer.

Obtaining a Ticket for a Kerberos Principal

A principal refers to a user or service that can authenticate to Kerberos. To authenticate to Kerberos, a principal must obtain a ticket by using a password or a keytab file. You can specify a keytab file to use, or use the default keytab file of your Kerberos configuration.

To obtain a ticket for a Kerberos principal using a password:

1. Open **MIT Kerberos Ticket Manager**.

2. In the MIT Kerberos Ticket Manager, click **Get Ticket**.
3. In the Get Ticket dialog box, type your principal name and password, and then click **OK**.

If the authentication succeeds, then your ticket information appears in the MIT Kerberos Ticket Manager.

To obtain a ticket for a Kerberos principal using a keytab file:

1. Open a command prompt:
 - If you are using Windows 7 or earlier, click the **Start** button , then click **All Programs**, then click **Accessories**, and then click **Command Prompt**.
 - If you are using Windows 8 or later, click the arrow button at the bottom of the Start screen, then find the Windows System program group, and then click **Command Prompt**.

2. In the Command Prompt, type a command using the following syntax:

```
kinit -k -t keytab_path principal
```

keytab_path is the full path to the keytab file. For example:

```
C:\mykeytabs\myUser.keytab
```

principal is the Kerberos user principal to use for authentication. For example:
myUser@EXAMPLE.COM

3. If the cache location KRB5CCNAME is not set or used, then use the -c option of the kinit command to specify the location of the credential cache. In the command, the -c argument must appear last. For example:

```
kinit -k -t C:\mykeytabs\myUser.keytab myUser@EXAMPLE.COM -c  
C:\ProgramData\MIT\krbcache
```

Krbcache is the Kerberos cache file, not a directory.

To obtain a ticket for a Kerberos principal using the default keytab file:

Note: For information about configuring a default keytab file for your Kerberos configuration, refer to the MIT Kerberos documentation.

1. Open a command prompt:
 - If you are using Windows 7 or earlier, click the **Start** button , then click **All Programs**, then click **Accessories**, and then click **Command Prompt**.
 - If you are using Windows 8 or later, click the arrow button at the bottom of the Start screen, then find the Windows System program group, and then click **Command Prompt**.
2. In the Command Prompt, type a command using the following syntax:

```
kinit -k principal
```

principal is the Kerberos user principal to use for authentication, for example,
MyUser@EXAMPLE.COM.

3. If the cache location KRB5CCNAME is not set or used, then use the -c option of the kinit command to specify the location of the credential cache. In the command, the -c argument must appear last. For example:

```
kinit -k -t C:\mykeytabs\myUser.keytab myUser@EXAMPLE.COM -c  
C:\ProgramData\MIT\krbcache
```

Krbcache is the Kerberos cache file, not a directory.

Verifying the Version Number

If you need to verify the version of the Hortonworks ODBC Driver with SQL Connector for Apache Spark that is installed on your Windows machine, you can find the version number in the ODBC Data Source Administrator.

To verify the version number:

1. Open the ODBC Administrator:
 - If you are using Windows 7 or earlier, click the **Start** button , then click **All Programs**, then click the **Hortonworks Spark ODBC Driver 1.1** program group corresponding to the bitness of the client application accessing data in Spark, and then click **ODBC Administrator**
 - Or, if you are using Windows 8 or later, on the Start screen, type **ODBC administrator**, and then click the **ODBC Administrator** search result corresponding to the bitness of the client application accessing data in Spark.
2. Click the **Drivers** tab and then find the Hortonworks Spark ODBC Driver in the list of ODBC drivers that are installed on your system. The version number is displayed in the **Version** column.

Linux Driver

Linux System Requirements

You install the Hortonworks ODBC Driver with SQL Connector for Apache Spark on client machines that access data stored in a Hadoop cluster with the Spark service installed and running. Each machine that you install the driver on must meet the following minimum system requirements:

- One of the following distributions:
 - Red Hat® Enterprise Linux® (RHEL) 5 or 6
 - CentOS 5, 6, or 7
 - SUSE Linux Enterprise Server (SLES) 11 or 12
- 150 MB of available disk space
- One of the following ODBC driver managers installed:
 - iODBC 3.52.7 or later
 - unixODBC 2.2.14 or later

The driver supports Apache Spark versions 0.8 through 1.5.

Installing the Driver

There are two versions of the driver for Linux:

- **spark-odbc-native-32bit-Version-Release.LinuxDistro.i686.rpm** for the 32-bit driver
- **spark-odbc-native-Version-Release.LinuxDistro.x86_64.rpm** for the 64-bit driver

Version is the version number of the driver, and *Release* is the release number for this version of the driver. *LinuxDistro* is either el5 or el6. For SUSE, the *LinuxDistro* placeholder is empty.

The bitness of the driver that you select should match the bitness of the client application accessing your Hadoop-based data. For example, if the client application is 64-bit, then you should install the 64-bit driver. Note that 64-bit editions of Linux support both 32- and 64-bit applications. Verify the bitness of your intended application and install the appropriate version of the driver.

Important: Ensure that you install the driver using the RPM corresponding to your Linux distribution.

The Hortonworks ODBC Driver with SQL Connector for Apache Spark driver files are installed in the following directories:

- **/usr/lib/spark/lib/native/sparkodbc** contains release notes, the *Hortonworks ODBC Driver with SQL Connector for Apache Spark User Guide* in PDF format, and a `Readme.txt` file that provides plain text installation and configuration instructions.
- **/usr/lib/spark/lib/native/sparkodbc/ErrorMessage**s contains error message files required by the driver.
- **/usr/lib/spark/lib/native/Linux-i386-32** contains the 32-bit driver and the `hortonworks.sparkodbc.ini` configuration file.
- **/usr/lib/spark/lib/native/Linux-amd64-64** contains the 64-bit driver and the `hortonworks.sparkodbc.ini` configuration file.

To install the Hortonworks ODBC Driver with SQL Connector for Apache Spark:

1. Choose one:

- In Red Hat Enterprise Linux or CentOS, log in as the root user, then navigate to the folder containing the driver RPM packages to install, and then type the following at the command line, where *RPMFileName* is the file name of the RPM package containing the version of the driver that you want to install:

```
yum --nogpgcheck localinstall RPMFileName
```

- In SUSE Linux Enterprise Server, log in as the root user, then navigate to the folder containing the driver RPM packages to install, and then type the following at the command line, where *RPMFileName* is the file name of the RPM package containing the version of the driver that you want to install:

```
zypper install RPMFileName
```

The Hortonworks ODBC Driver with SQL Connector for Apache Spark depends on the following resources:

- `cyrus-sasl-2.1.22-7` or above
- `cyrus-sasl-gssapi-2.1.22-7` or above
- `cyrus-sasl-plain-2.1.22-7` or above

If the package manager in your Linux distribution cannot resolve the dependencies automatically when installing the driver, then download and manually install the packages required by the version of the driver that you want to install.

Setting the LD_LIBRARY_PATH Environment Variable

The `LD_LIBRARY_PATH` environment variable must include the paths to the installed ODBC driver manager libraries.

For example, if ODBC driver manager libraries are installed in `/usr/local/lib`, then set `LD_LIBRARY_PATH` as follows:

```
export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:/usr/local/lib
```

For information about how to set environment variables permanently, refer to your Linux shell documentation.

For information about creating ODBC connections using the Hortonworks ODBC Driver with SQL Connector for Apache Spark, see "Configuring ODBC Connections for Non-Windows Platforms" on page 31.

Verifying the Version Number

If you need to verify the version of the Hortonworks ODBC Driver with SQL Connector for Apache Spark that is installed on your Linux machine, you can query the version number through the command-line interface.

To verify the version number:

- Depending on your version of Linux, at the command prompt, run one of the following commands:

```
yum list | grep spark-odbc-native
```

```
rpm -qa | grep spark-odbc-native
```

The command returns information about the Hortonworks ODBC Driver with SQL Connector for Apache Spark that is installed on your machine, including the version number.

Mac OS X Driver

Mac OS X System Requirements

You install the Hortonworks ODBC Driver with SQL Connector for Apache Spark on client machines that access data stored in a Hadoop cluster with the Spark service installed and running. Each machine that you install the driver on must meet the following minimum system requirements:

- Mac OS X version 10.9 or 10.10
- 100 MB of available disk space
- iODBC 3.52.7 or later

The driver supports both 32- and 64-bit client applications.

The driver supports Apache Spark versions 0.8 through 1.5.

Installing the Driver

The Hortonworks ODBC Driver with SQL Connector for Apache Spark driver files are installed in the following directories:

- `/opt/hortonworks/sparkodbc` contains release notes and the *Hortonworks ODBC Driver with SQL Connector for Apache Spark User Guide* in PDF format.
- `/opt/hortonworks/sparkodbc/ErrorMessages` contains error message files required by the driver.
- `/opt/hortonworks/sparkodbc/Setup` contains sample configuration files named `odbc.ini` and `odbcinst.ini`.
- `/opt/hortonworks/sparkodbc/lib/universal` contains the driver binaries and the `hortonworks.sparkodbc.ini` configuration file.

To install the Hortonworks ODBC Driver with SQL Connector for Apache Spark:

1. Double-click **spark-odbc-native.dmg** to mount the disk image.
2. Double-click **spark-odbc-native.pkg** to run the installer.
3. In the installer, click **Continue**.
4. On the Software License Agreement screen, click **Continue**, and when the prompt appears, click **Agree** if you agree to the terms of the License Agreement.
5. Optionally, to change the installation location, click **Change Install Location**, then select the desired location, and then click **Continue**.
6. To accept the installation location and begin the installation, click **Install**.
7. When the installation completes, click **Close**.

Setting the DYLD_LIBRARY_PATH Environment Variable

The DYLD_LIBRARY_PATH environment variable must include the paths to the installed ODBC driver manager libraries.

For example, if ODBC driver manager libraries are installed in `/usr/local/lib`, then set DYLD_LIBRARY_PATH as follows:

```
export DYLD_LIBRARY_PATH=$DYLD_LIBRARY_PATH:/usr/local/lib
```

For information about how to set environment variables permanently, refer to your Mac OS X shell documentation.

For information about creating ODBC connections using the Hortonworks ODBC Driver with SQL Connector for Apache Spark, see "Configuring ODBC Connections for Non-Windows Platforms" on page 31.

Verifying the Version Number

If you need to verify the version of the Hortonworks ODBC Driver with SQL Connector for Apache Spark that is installed on your Mac OS X machine, you can query the version number through the Terminal.

To verify the version number:

- At the Terminal, run the following command:

```
pkgutil --info com.hortonworks.sparkodbc
```

The command returns information about the Hortonworks ODBC Driver with SQL Connector for Apache Spark that is installed on your machine, including the version number.

Configuring ODBC Connections for Non-Windows Platforms

The following sections describe how to configure ODBC connections when using the Hortonworks ODBC Driver with SQL Connector for Apache Spark with non-Windows platforms:

- "Configuration Files" on page 31
- "Sample Configuration Files" on page 32
- "Configuring the Environment" on page 32
- "Defining DSNs in odbc.ini" on page 33
- "Specifying ODBC drivers in odbcinst.ini" on page 34
- "Configuring Driver Settings in hortonworks.sparkodbc.ini" on page 35
- "Configuring Authentication" on page 36
- "Configuring SSL Verification" on page 39
- "Configuring Logging Options" on page 42

Configuration Files

ODBC driver managers use configuration files to define and configure ODBC data sources and drivers. By default, the following configuration files residing in the user's home directory are used:

- **.odbc.ini** is used to define ODBC data sources, and it is required for DSNs.
- **.odbcinst.ini** is used to define ODBC drivers, and it is optional.

Also, by default the Hortonworks ODBC Driver with SQL Connector for Apache Spark is configured using the **hortonworks.sparkodbc.ini** file, which is located in one of the following directories depending on the version of the driver that you are using:

- **/usr/lib/spark/lib/native/Linux-i386-32** for the 32-bit driver on Linux
- **/usr/lib/spark/lib/native/Linux-amd64-64** for the 64-bit driver on Linux
- **/opt/hortonworks/sparkodbc/lib/universal** for the driver on Mac OS X

The `hortonworks.sparkodbc.ini` file is required.

Note: The `hortonworks.sparkodbc.ini` file in the `/lib` subfolder provides default settings for most configuration options available in the Hortonworks ODBC Driver with SQL Connector for Apache Spark.

You can set driver configuration options in your `odbc.ini` and `hortonworks.sparkodbc.ini` files. Configuration options set in a `hortonworks.sparkodbc.ini` file apply to all connections, whereas configuration options set in an `odbc.ini` file are specific to a connection. Configuration options set in `odbc.ini` take precedence over configuration options set in

hortonworks.sparkodbc.ini. For information about the configuration options available for controlling the behavior of DSNs that are using the Hortonworks ODBC Driver with SQL Connector for Apache Spark, see "Driver Configuration Options" on page 55.

Sample Configuration Files

The driver installation contains the following sample configuration files in the Setup directory:

- **odbc.ini**
- **odbcinst.ini**

These sample configuration files provide preset values for settings related to the Hortonworks ODBC Driver with SQL Connector for Apache Spark.

The names of the sample configuration files do not begin with a period (`.`) so that they appear in directory listings by default. A file name beginning with a period (`.`) is hidden. For `odbc.ini` and `odbcinst.ini`, if the default location is used, then the file names must begin with a period (`.`).

If the configuration files do not exist in the home directory, then you can copy the sample configuration files to the home directory, and then rename the files. If the configuration files already exist in the home directory, then use the sample configuration files as a guide to modify the existing configuration files.

Configuring the Environment

Optionally, you can use three environment variables, `ODBCINI`, `ODBCSYSINI`, and `HORTONWORKSSPARKINI`, to specify different locations for the `odbc.ini`, `odbcinst.ini`, and `hortonworks.sparkodbc.ini` configuration files by doing the following:

- Set `ODBCINI` to point to your `odbc.ini` file.
- Set `ODBCSYSINI` to point to the directory containing the `odbcinst.ini` file.
- Set `HORTONWORKSSPARKINI` to point to your `hortonworks.sparkodbc.ini` file.

For example, if your `odbc.ini` and `hortonworks.sparkodbc.ini` files are located in `/etc` and your `odbcinst.ini` file is located in `/usr/local/odbc`, then set the environment variables as follows:

```
export ODBCINI=/etc/odbc.ini
export ODBCSYSINI=/usr/local/odbc
export HORTONWORKSSPARKINI=/etc/hortonworks.sparkodbc.ini
```

The following search order is used to locate the `hortonworks.sparkodbc.ini` file:

1. If the HORTONWORKSSPARKINI environment variable is defined, then the driver searches for the file specified by the environment variable.
Important: HORTONWORKSSPARKINI must specify the full path, including the file name.
2. The directory containing the driver's binary is searched for a file named `hortonworks.sparkodbc.ini` (not beginning with a period).
3. The current working directory of the application is searched for a file named `hortonworks.sparkodbc.ini` (not beginning with a period).
4. The directory `~/` (that is, `$HOME`) is searched for a hidden file named `.hortonworks.sparkodbc.ini`.
5. The directory `/etc` is searched for a file named `hortonworks.sparkodbc.ini` (not beginning with a period).

Defining DSNs in `odbc.ini`

ODBC Data Source Names (DSNs) are defined in the `odbc.ini` configuration file. This file is divided into several sections:

- `[ODBC]` is optional. This section is used to control global ODBC configuration, such as ODBC tracing.
- `[ODBC Data Sources]` is required. This section lists the DSNs and associates them with a driver.
- A section having the same name as the data source specified in the `[ODBC Data Sources]` section is required to configure the data source.

The following is an example of an `odbc.ini` configuration file for Linux:

```
[ODBC Data Sources]
Sample Hortonworks Spark ODBC DSN 32=Hortonworks Spark ODBC Driver
32-bit
[Sample Hortonworks Spark DSN 32]
Driver=/usr/lib/spark/lib/native/Linux-i386-
32/libhortonworkssparkodbc32.so
HOST=MySparkServer
PORT=10000
```

MySparkServer is the IP address or host name of the Spark server.

The following is an example of an `odbc.ini` configuration file for Mac OS X:

```
[ODBC Data Sources]
Sample Hortonworks Spark DSN=Hortonworks Spark ODBC Driver
[Sample Hortonworks Spark DSN]
```

```
Driver=/opt/hortonworks/sparkodbc/lib/universal/libhortonworksspark  
odbc.dylib
```

```
HOST=MySparkServer
```

```
PORT=10000
```

MySparkServer is the IP address or host name of the Spark server.

To create a Data Source Name:

1. In a text editor, open the `odbc.ini` configuration file.
2. In the `[ODBC Data Sources]` section, add a new entry by typing the Data Source Name (DSN), then an equal sign (=), and then the driver name.
3. Add a new section to the file, with a section name that matches the DSN you specified in step 2, and then add configuration options to the section. Specify the configuration options as key-value pairs.

Note: Shark Server does not support authentication. Most default configurations of Shark Server 2 or Spark Thrift Server require User Name authentication, which you configure by setting the `AuthMech` key to 2. To verify the authentication mechanism that you need to use for your connection, check the configuration of your Hadoop / Spark distribution. For more information, see "Authentication Options" on page 44.

4. Save the `odbc.ini` configuration file.

For information about the configuration options available for controlling the behavior of DSNs that are using the Hortonworks ODBC Driver with SQL Connector for Apache Spark, see "Driver Configuration Options" on page 55.

Specifying ODBC drivers in `odbcinst.ini`

ODBC drivers are defined in the `odbcinst.ini` configuration file. This configuration file is optional because drivers can be specified directly in the `odbc.ini` configuration file, as described in "Defining DSNs in `odbc.ini`" on page 33.

The `odbcinst.ini` file is divided into the following sections:

- `[ODBC Drivers]` lists the names of all the installed ODBC drivers.
- For each driver, a section having the same name as the driver name specified in the `[ODBC Drivers]` section lists the driver attributes and values.

The following is an example of an `odbcinst.ini` configuration file for Linux:

```
[ODBC Drivers]  
Hortonworks Spark ODBC Driver 32-bit=Installed  
Hortonworks Spark ODBC Driver 64-bit=Installed  
[Hortonworks Spark ODBC Driver 32-bit]  
Description=Hortonworks Spark ODBC Driver (32-bit)
```

```
Driver=/usr/lib/spark/lib/native/Linux-i386-32/libhortonworkssparkodbc32.so
[Hortonworks Spark ODBC Driver 64-bit]
Description=Hortonworks Spark ODBC Driver (64-bit)
Driver=/usr/lib/spark/lib/native/Linux-amd64-64/libhortonworkssparkodbc64.so
```

The following is an example of an `odbcinst.ini` configuration file for Mac OS X:

```
[ODBC Drivers]
Hortonworks Spark ODBC Driver=Installed
[Hortonworks Spark ODBC Driver]
Description=Hortonworks Spark ODBC Driver
Driver=/opt/hortonworks/sparkodbc/lib/universal/libhortonworkssparkodbc.dylib
```

To define a driver:

1. In a text editor, open the `odbcinst.ini` configuration file.
2. In the `[ODBC Drivers]` section, add a new entry by typing the driver name and then typing `=Installed`.
Note: Give the driver a symbolic name that you want to use to refer to the driver in connection strings or DSNs.
3. Add a new section to the file with a name that matches the driver name you typed in step 2, and then add configuration options to the section based on the sample `odbcinst.ini` file provided in the Setup directory. Specify the configuration options as key-value pairs.
4. Save the `odbcinst.ini` configuration file.

Configuring Driver Settings in `hortonworks.sparkodbc.ini`

The `hortonworks.sparkodbc.ini` file contains configuration settings for the Hortonworks ODBC Driver with SQL Connector for Apache Spark. Settings that you define in the `hortonworks.sparkodbc.ini` file apply to all connections that use the driver.

You do not need to modify the settings in the `hortonworks.sparkodbc.ini` file to use the driver and connect to your data source.

However, to help troubleshoot issues, you can configure the `hortonworks.sparkodbc.ini` file to enable logging in the driver. For information about configuring logging, see "Configuring Logging Options" on page 42.

Configuring Authentication

Some Spark servers are configured to require authentication for access. To connect to a Spark server, you must configure the Hortonworks ODBC Driver with SQL Connector for Apache Spark to use the authentication mechanism that matches the access requirements of the server and provides the necessary credentials.

For information about how to determine the type of authentication your Spark server requires, see "Authentication Options" on page 44.

You can select the type of authentication to use for a connection by defining the AuthMech connection attribute in a connection string or in a DSN (in the odbc.ini file). Depending on the authentication mechanism you use, there may be additional connection attributes that you must define. For more information about the attributes involved in configuring authentication, see "Driver Configuration Options" on page 55.

Using No Authentication

When connecting to a Spark server of type Shark Server, you must use No Authentication. When you use No Authentication, Binary is the only Thrift transport protocol that is supported.

To configure a connection without authentication:

1. Set the AuthMech connection attribute to 0.
2. If the Spark server is configured to use SSL, then configure SSL for the connection. For more information, see "Configuring SSL Verification" on page 39.

Example connection string for Shark Server:

```
Driver=Hortonworks Spark ODBC Driver;Host=server_  
host;Port=10000;SparkServerType=1;AuthMech=0;Schema=Spark_database
```

Example connection string for Shark Server 2:

```
Driver=Hortonworks Spark ODBC Driver;Host=server_  
host;Port=10000;SparkServerType=2;AuthMech=0;Schema=Spark_database
```

Example connection string for Spark Thrift Server:

```
Driver=Hortonworks Spark ODBC Driver;Host=server_  
host;Port=10000;SparkServerType=3;AuthMech=0;Schema=Spark_database
```

Kerberos

Kerberos must be installed and configured before you can use this authentication mechanism. For more information, refer to the MIT Kerberos documentation.

To configure Kerberos authentication:

1. Set the AuthMech connection attribute to 1.

2. To use the default realm defined in your Kerberos setup, do not set the KrbRealm attribute.
If your Kerberos setup does not define a default realm or if the realm of your Spark server is not the default, then set the appropriate realm using the KrbRealm attribute.
3. Set the KrbHostFQDN attribute to the fully qualified domain name of the Shark Server 2 or Spark Thrift Server host.
Note: To use the Spark server host name as the fully qualified domain name for Kerberos authentication, set KrbHostFQDN to **_HOST**
4. Set the KrbServiceName attribute to the service name of the Shark Server 2 or Spark Thrift Server.
5. Set the ThriftTransport connection attribute to the transport protocol to use in the Thrift layer.
Important: When using this authentication mechanism, Binary (**ThriftTransport=0**) is not supported.
6. If the Spark server is configured to use SSL, then configure SSL for the connection. For more information, see "Configuring SSL Verification" on page 39.

Example connection string:

```
Driver=Hortonworks Spark ODBC Driver;Host=server_  
host;Port=10000;SparkServerType=2;AuthMech=1;Schema=Spark_  
database;KrbRealm=Kerberos_realm;KrbHostFQDN=domain_  
name;KrbServiceName=service_name
```

Using User Name

This authentication mechanism requires a user name but does not require a password. The user name labels the session, facilitating database tracking.

This authentication mechanism is available only for Shark Server 2 or Spark Thrift Server on non-HDInsight distributions. Most default configurations of require User Name authentication. When you use User Name authentication, SSL is not supported and SASL is the only Thrift transport protocol available.

To configure User Name authentication:

1. Set the AuthMech connection attribute to 2.
2. Set the UID attribute to an appropriate user name for accessing the Spark server.

Example connection string:

```
Driver=Hortonworks Spark ODBC Driver;Host=server_  
host;Port=10000;SparkServerType=2;AuthMech=2;Schema=Spark_  
database;UID=user_name
```

Using User Name and Password

This authentication mechanism requires a user name and a password.

This authentication mechanism is available only for Shark Server 2 or Spark Thrift Server on non-HDInsight distributions.

To configure User Name and Password authentication:

1. Set the AuthMech connection attribute to 3.
2. Set the UID attribute to an appropriate user name for accessing the Spark server.
3. Set the PWD attribute to the password corresponding to the user name you provided in step 2.
4. Set the ThriftTransport connection attribute to the transport protocol to use in the Thrift layer.
5. If the Spark server is configured to use SSL, then configure SSL for the connection. For more information, see "Configuring SSL Verification" on page 39.

Example connection string:

```
Driver=Hortonworks Spark ODBC Driver;Host=server_  
host;Port=10000;SparkServerType=2;AuthMech=3;Schema=Spark_  
database;UID=user_name;PWD=password
```

Using Windows Azure HDInsight Emulator

This authentication mechanism is available only for Shark Server 2 or Spark Thrift Server instances running on Windows Azure HDInsight Emulator. When you use this authentication mechanism, SSL is not supported and HTTP is the only Thrift transport protocol available.

To configure a connection to a Spark server on Windows Azure HDInsight Emulator:

1. Set the AuthMech connection attribute to 5.
2. Set the HTTPPath attribute to the partial URL corresponding to the Spark server.
3. Set the UID attribute to an appropriate user name for accessing the Spark server.
4. Set the PWD attribute to the password corresponding to the user name you provided in step 3.
5. If necessary, you can create custom HTTP headers. For more information, see "http.header." on page 72.

Example connection string:

```
Driver=Hortonworks Spark ODBC Driver;Host=server_  
host;Port=10000;SparkServerType=2;AuthMech=5;Schema=Spark_  
database;UID=user_name;PWD=password;HTTPPath=Spark_HTTP_path
```

Using Windows Azure HDInsight Service

This authentication mechanism is available only for Shark Server 2 or Spark Thrift Server on HDInsight distributions. When you use this authentication mechanism, you must enable SSL and HTTP is the only Thrift transport protocol available.

To configure a connection to a Spark server on Windows Azure HDInsight Service:

1. Set the AuthMech connection attribute to 6.
2. Set the HTTPPath attribute to the partial URL corresponding to the Spark server.
3. Set the UID attribute to an appropriate user name for accessing the Spark server.
4. Set the PWD attribute to the password corresponding to the user name you typed in step 3.
5. If necessary, you can create custom HTTP headers. For more information, see "http.header." on page 72.
6. Configure SSL settings as needed. For more information, see "Configuring SSL Verification" on page 39.
7. Choose one:
 - To configure the driver to load SSL certificates from a specific file, set the TrustedCerts attribute to the path of the file.
 - Or, to use the trusted CA certificates PEM file that is installed with the driver, do not specify a value for the TrustedCerts attribute.

Example connection string:

```
Driver=Hortonworks Spark ODBC Driver;Host=server_  
host;Port=10000;SparkServerType=2;AuthMech=6;Schema=Spark_  
database;UID=user_name;PWD=password;HTTPPath=Spark_HTTP_path
```

Configuring SSL Verification

You can configure verification between the client and the Spark server over SSL.

To configure SSL verification:

1. Open the `odbc.ini` configuration file in a text editor.
2. To enable SSL, set the SSL attribute to 1.
3. To allow self-signed certificates from the server, set the AllowSelfSignedServerCert attribute to 1.
4. To allow the common name of a CA-issued SSL certificate to not match the host name of the Spark server, set the CAIssuedCertNamesMismatch attribute to 1.
5. Choose one:
 - To configure the driver to load SSL certificates from a specific PEM file when

- verifying the server, set the TrustedCerts attribute to the full path of the PEM file.
- To use the trusted CA certificates PEM file that is installed with the driver, do not specify a value for the TrustedCerts attribute.
6. To configure two-way SSL verification, set the TwoWaySSL attribute to 1 and then do the following:
 - a) Set the ClientCert attribute to the full path of the PEM file containing the client's certificate.
 - b) Set the ClientPrivateKey attribute to the full path of the file containing the client's private key.
 - c) If the private key file is protected with a password, set the ClientPrivateKeyPassword attribute to the password.
 7. Save the `odbc.ini` configuration file.

Testing the Connection

To test the connection, you can use an ODBC-enabled client application. For a basic connection test, you can also use the test utilities that are packaged with your driver manager installation. For example, the iODBC driver manager includes simple utilities called `iodbctest` and `iodbctestw`. Similarly, the unixODBC driver manager includes simple utilities called `isql` and `iusql`.

Using the iODBC Driver Manager

You can use the `iodbctest` and `iodbctestw` utilities to establish a test connection with your driver. Use `iodbctest` to test how your driver works with an ANSI application, or use `iodbctestw` to test how your driver works with a Unicode application.

Note: There are 32-bit and 64-bit installations of the iODBC driver manager available. If you have only one or the other installed, then the appropriate version of `iodbctest` (or `iodbctestw`) is available. However, if you have both 32- and 64-bit versions installed, then you need to ensure that you are running the version from the correct installation directory.

For more information about using the iODBC driver manager, see <http://www.iodbc.org>.

To test your connection using the iODBC driver manager:

1. Run `iodbctest` or `iodbctestw`.
2. Optionally, if you do not remember the DSN, then type a question mark (?) to see a list of available DSNs.
3. Type an ODBC connection string using the following format, specifying additional connection attributes as needed:

```
DSN=DataSourceName;Key=Value
```

DataSourceName is the DSN that you are using for the connection. *Key* is any connection attribute that is not already specified as a configuration key in the DSN, and *Value* is the value for the attribute. Add key-value pairs to the connection string as needed, separating each pair with a semicolon (;).

Or, if you are using a DSN-less connection, then type an ODBC connection string using the following format, specifying additional connection attributes as needed:

```
Driver=DriverNameOrFile;HOST=MySparkServer;  
PORT=PortNumber;Schema=DefaultSchema;  
SparkServerType=ServerType
```

The placeholders in the connection string are defined as follows:

- *DriverNameOrFile* is either the symbolic name of the installed driver defined in the `odbcinst.ini` file or the absolute path of the shared object file for the driver. If you use the symbolic name, then you must ensure that the `odbcinst.ini` file is configured to point the symbolic name to the shared object file. For more information, see "Specifying ODBC drivers in `odbcinst.ini`" on page 34.
- *MySparkServer* is the IP address or hostname of the Spark Server.
- *PortNumber* is the number of the TCP port that the Spark server uses to listen for client connections.
- *DefaultSchema* is the database schema to use when a schema is not explicitly specified in a query.
- *ServerType* is 1 (for Shark Server), 2 (for Shark Server 2) or 3 (for Spark Thrift Server).

If the connection is successful, then the `SQL>` prompt appears.

Note: For information about the connection attributes that are available, see "Driver Configuration Options" on page 55.

Using the unixODBC Driver Manager

You can use the `isql` and `iusql` utilities to establish a test connection with your driver and your DSN. `isql` and `iusql` can only be used to test connections that use a DSN. Use `isql` to test how your driver works with an ANSI application, or use `iusql` to test how your driver works with a Unicode application.

Note: There are 32-bit and 64-bit installations of the unixODBC driver manager available. If you have only one or the other installed, then the appropriate version of `isql` (or `iusql`) is available. However, if you have both 32- and 64-bit versions installed, then you need to ensure that you are running the version from the correct installation directory.

For more information about using the unixODBC driver manager, see <http://www.unixodbc.org>.

To test your connection using the unixODBC driver manager:

- Run either `isql` or `iusql` using the following syntax:

```
isql DataSourceName
```

```
iusql DataSourceName
```

DataSourceName is the DSN that you are using for the connection.

If the connection is successful, then the `SQL>` prompt appears.

Note: For information about the available options, run `isql` or `iusql` without providing a DSN.

Configuring Logging Options

To help troubleshoot issues, you can enable logging in the driver.

Important: Only enable logging long enough to capture an issue. Logging decreases performance and can consume a large quantity of disk space.

Use the `LogLevel` key to set the amount of detail included in log files. Table 2 lists the logging levels provided by the Hortonworks ODBC Driver with SQL Connector for Apache Spark, in order from least verbose to most verbose.

Table 2. Hortonworks ODBC Driver with SQL Connector for Apache Spark Logging Levels

LogLevel Value	Description
0	Disables all logging.
1	Logs very severe error events that will lead the driver to abort.
2	Logs error events that might still allow the driver to continue running.
3	Logs potentially harmful situations.
4	Logs general information that describes the progress of the driver.
5	Logs detailed information that is useful for debugging the driver.
6	Logs more detailed information than LogLevel=5

To enable logging:

1. Open the `hortonworks.sparkodbc.ini` configuration file in a text editor.

2. Set the `LogLevel` key to the desired level of information to include in log files. For example:
`LogLevel=2`
3. Set the `LogPath` key to the full path to the folder where you want to save log files. For example:
`LogPath=/localhome/employee/Documents`
4. Set the `LogFileCount` key to the maximum number of log files to keep.
Note: After the maximum number of log files is reached, each time an additional file is created, the driver deletes the oldest log file.
5. Set the `LogFileSize` key to the maximum size of each log file in megabytes (MB).
Note: After the maximum file size is reached, the driver creates a new file and continues logging.
6. Save the `hortonworks.sparkodbc.ini` configuration file.
7. Restart your ODBC application to ensure that the new settings take effect.

The Hortonworks ODBC Driver with SQL Connector for Apache Spark produces a log file named `SparkODBC_driver.log` at the location you specify using the `LogPath` key.

To disable logging:

1. Open the `hortonworks.sparkodbc.ini` configuration file in a text editor.
2. Set the `LogLevel` key to 0.
3. Save the `hortonworks.sparkodbc.ini` configuration file.

Authentication Options

To connect to a Spark server, you must configure the Hortonworks ODBC Driver with SQL Connector for Apache Spark to use the authentication mechanism that matches the access requirements of the server and provides the necessary credentials. To determine the authentication settings that your Spark server requires, check the server type and then refer to the corresponding section below.

Shark Server

You must use No Authentication as the authentication mechanism. Shark Server instances do not support authentication.

Shark Server 2 or Spark Thrift Server on an HDInsight Distribution

If you are connecting to HDInsight Emulator running on Windows Azure, then you must use the Windows Azure HDInsight Emulator mechanism.

If you are connecting to HDInsight Service running on Windows Azure, then you must use the Windows Azure HDInsight Service mechanism.

Shark Server 2 or Spark Thrift Server on a non-HDInsight Distribution

Note: Most default configurations of Shark Server 2 or Spark Thrift Server on non-HDInsight Distributions require User Name authentication.

Configuring authentication for a connection to a Shark Server 2 or Spark Thrift Server instance on a non-HDInsight Distribution involves setting the authentication mechanism, the Thrift transport protocol, and SSL support. To determine the settings that you need to use, check the following three properties in the `hive-site.xml` file in the Spark server that you are connecting to:

- `hive.server2.authentication`
- `hive.server2.transport.mode`
- `hive.server2.use.SSL`

Use Table 3 to determine the authentication mechanism that you need to configure, based on the `hive.server2.authentication` value in the `hive-site.xml` file:

Table 3. Authentication Mechanism to Use

<code>hive.server2.authentication</code>	Authentication Mechanism
NOSASL	No Authentication
KERBEROS	Kerberos
NONE	User Name
LDAP	User Name and Password

Use Table 4 to determine the Thrift transport protocol that you need to configure, based on the `hive.server2.authentication` and `hive.server2.transport.mode` values in the `hive-site.xml` file:

Table 4. Thrift Transport Protocol Setting to Use

<code>hive.server2.authentication</code>	<code>hive.server2.transport.mode</code>	Thrift Transport Protocol
NOSASL	binary	Binary
KERBEROS	binary or http	SASL or HTTP
NONE	binary or http	SASL or HTTP
LDAP	binary or http	SASL or HTTP

To determine whether SSL should be enabled or disabled for your connection, check the `hive.server2.use.SSL` value in the `hive-site.xml` file. If the value is true, then you must enable and configure SSL in your connection. If the value is false, then you must disable SSL in your connection.

For detailed instructions on how to configure authentication when using the Windows driver, see "Configuring Authentication" on page 10.

For detailed instructions on how to configure authentication when using a non-Windows driver, see "Configuring Authentication" on page 36.

Using a Connection String

For some applications, you may need to use a connection string to connect to your data source. For detailed information about how to use a connection string in an ODBC application, refer to the documentation for the application that you are using.

The connection strings in the following topics are examples showing the minimum set of connection attributes that you must specify to successfully connect to the data source. Depending on the configuration of the data source and the type of connection you are working with, you may need to specify additional connection attributes. For detailed information about all the attributes that you can use in the connection string, see "Driver Configuration Options" on page 55.

- "DSN Connection String Example" on page 46
- "DSN-less Connection String Examples" on page 46

DSN Connection String Example

The following is an example of a connection string for a connection that uses a DSN:

```
DSN=DataSourceName;
```

DataSourceName is the DSN that you are using for the connection.

You can set additional configuration options by appending key-value pairs to the connection string. Configuration options that are passed in using a connection string take precedence over configuration options that are set in the DSN.

For information about creating a DSN on a Windows computer, see "Creating a Data Source Name" on page 7. For information about creating a DSN on a non-Windows computer, see "Defining DSNs in `odbc.ini`" on page 33.

DSN-less Connection String Examples

Some applications provide support for connecting to a data source using a driver without a DSN. To connect to a data source without using a DSN, use a connection string instead.

The placeholders in the examples are defined as follows, in alphabetical order:

- *DomainName* is the fully qualified domain name of the Spark server host.
- *PortNumber* is the number of the TCP port that the Spark server uses to listen for client connections.
- *Realm* is the Kerberos realm of the Spark server host.
- *Server* is the IP address or host name of the Spark server to which you are connecting.
- *ServerURL* is the partial URL corresponding to the Spark server.

- *ServiceName* is the Kerberos service principal name of the Spark server.
- *YourPassword* is the password corresponding to your user name.
- *YourUserName* is the user name that you use to access the Spark server.

Connecting to a Shark Server Instance

The following is the format of a DSN-less connection string that connects to a Shark Server instance:

```
Driver=Hortonworks Spark ODBC Driver;SparkServerType=1;  
Host=Server;Port=PortNumber;
```

For example:

```
Driver=Hortonworks Spark ODBC Driver;SparkServerType=1;  
Host=192.168.222.160;Port=10000;
```

Connecting to a Standard Shark Server 2 Instance

The following is the format of a DSN-less connection string for a standard connection to a Shark Server 2 instance. Most default configurations of Shark Server 2 require **User Name** authentication. When configured to provide **User Name** authentication, the driver uses **anonymous** as the user name by default.

```
Driver=Hortonworks Spark ODBC Driver;SparkServerType=2;  
Host=Server;Port=PortNumber;AuthMech=2;
```

For example:

```
Driver=Hortonworks Spark ODBC Driver;SparkServerType=2;  
Host=192.168.222.160;Port=10000;AuthMech=2;
```

Connecting to a Standard Spark Thrift Server Instance

The following is the format of a DSN-less connection string for a standard connection to a Spark Thrift Server instance. By default, the driver is configured to connect to a Spark Thrift Server instance. Most default configurations of Spark Thrift Server require **User Name** authentication. When configured to provide **User Name** authentication, the driver uses **anonymous** as the user name by default.

```
Driver=Hortonworks Spark ODBC Driver;Host=Server;Port=PortNumber;  
AuthMech=2;
```

For example:

```
Driver=Hortonworks Spark ODBC  
Driver;Host=192.168.222.160;Port=10000;  
AuthMech=2;
```

Connecting to a Spark Thrift Server Instance Without Authentication

The following is the format of a DSN-less connection string that for a Spark Thrift Server instance that does not require authentication.

```
Driver=Hortonworks Spark ODBC  
Driver;Host=Server;Port=PortNumber;AuthMech=0;
```

For example:

```
Driver=Hortonworks Spark ODBC  
Driver;Host=192.168.222.160;Port=10000;AuthMech=0;
```

Connecting to a Spark Server that Requires Kerberos Authentication

The following is the format of a DSN-less connection string that connects to a Shark Server 2 instance requiring **Kerberos** authentication:

```
Driver=Hortonworks Spark ODBC Driver;SparkServerType=2;  
Host=Server;Port=PortNumber;AuthMech=1;KrbRealm=Realm;  
KrbHostFQDN=DomainName;KrbServiceName=ServiceName;
```

For example:

```
Driver=Hortonworks Spark ODBC Driver;SparkServerType=2;  
Host=192.168.222.160;Port=10000;AuthMech=1;KrbRealm=HORTONWORKS;  
KrbHostFQDN=localhost.localdomain;KrbServiceName=spark;
```

The following is the format of a DSN-less connection string that connects to a Spark Thrift Server instance requiring **Kerberos** authentication. By default, the driver is configured to connect to a Spark Thrift Server instance.

```
Driver=Hortonworks Spark ODBC Driver;Host=Server;Port=PortNumber;  
AuthMech=1;KrbRealm=Realm;KrbHostFQDN=DomainName;  
KrbServiceName=ServiceName;
```

For example:

```
Driver=Hortonworks Spark ODBC  
Driver;Host=192.168.222.160;Port=10000;  
AuthMech=1;KrbRealm=HORTONWORKS;KrbHostFQDN=localhost.localdomain;  
KrbServiceName=spark;
```

Connecting to a Spark Server that Requires User Name and Password Authentication

The following is the format of a DSN-less connection string that connects to a Shark Server 2 instance requiring **User Name and Password** authentication:

```
Driver=Hortonworks Spark ODBC Driver;SparkServerType=2;  
Host=Server;Port=PortNumber;AuthMech=3;UID=YourUserName;  
PWD=YourPassword;
```

For example:

```
Driver=Hortonworks Spark ODBC Driver;SparkServerType=2;  
Host=192.168.222.160;Port=10000;AuthMech=3;UID=hortonworks;  
PWD=hortonworks;
```

The following is the format of a DSN-less connection string that connects to a Spark Thrift Server instance requiring **User Name and Password** authentication. By default, the driver is configured to connect to a Spark Thrift Server instance.

```
Driver=Hortonworks Spark ODBC Driver;Host=Server;Port=PortNumber;  
AuthMech=3;UID=YourUserName;PWD=YourPassword;
```

For example:

```
Driver=Hortonworks Spark ODBC  
Driver;Host=192.168.222.160;Port=10000;  
AuthMech=3;UID=hortonworks;PWD=hortonworks;
```

Connecting to a Spark Server on Windows Azure HDInsight Emulator

The following is the format of a DSN-less connection string that connects to a Shark Server 2 instance running on Windows Azure HDInsight Emulator:

```
Driver=Hortonworks Spark ODBC Driver;SparkServerType=2;  
Host=Server;Port=PortNumber;AuthMech=5;UID=YourUserName;  
PWD=YourPassword;HTTPPath=ServerURL;
```

For example:

```
Driver=Hortonworks Spark ODBC Driver;SparkServerType=2;  
Host=192.168.222.160;Port=10000;AuthMech=5;UID=hortonworks;  
PWD=hortonworks;HTTPPath=gateway/sandbox/spark;
```

The following is the format of a DSN-less connection string that connects to a Spark Thrift Server instance running on Windows Azure HDInsight Emulator. By default, the driver is configured to connect to a Spark Thrift Server instance.

```
Driver=Hortonworks Spark ODBC Driver;Host=Server;  
Port=PortNumber;AuthMech=5;UID=YourUserName;  
PWD=YourPassword;HTTPPath=ServerURL;
```

For example:

```
Driver=Hortonworks Spark ODBC Driver;Host=192.168.222.160;  
Port=10000;AuthMech=5;UID=hortonworks;  
PWD=hortonworks;HTTPPath=gateway/sandbox/spark;
```

Connecting to a Spark Server on Windows Azure HDInsight Service

The following is the format of a DSN-less connection string that connects to a Shark Server 2 instance running on Windows Azure HDInsight Service:

```
Driver=Hortonworks Spark ODBC Driver;SparkServerType=2;  
Host=Server;Port=PortNumber;AuthMech=6;UID=YourUserName;  
PWD=YourPassword;HTTPPath=ServerURL;
```



For example:

```
Driver=Hortonworks Spark ODBC Driver;SparkServerType=2;  
Host=192.168.222.160;Port=10000;AuthMech=6;UID=hortonworks;  
PWD=hortonworks;HTTPPath=gateway/sandbox/spark;
```

The following is the format of a DSN-less connection string that connects to a Spark Thrift Server instance running on Windows Azure HDInsight Service. By default, the driver is configured to connect to a Spark Thrift Server instance.

```
Driver=Hortonworks Spark ODBC Driver;Host=Server;  
Port=PortNumber;AuthMech=6;UID=YourUserName;  
PWD=YourPassword;HTTPPath=ServerURL;
```

For example:

```
Driver=Hortonworks Spark ODBC Driver;Host=192.168.222.160;  
Port=10000;AuthMech=6;UID=hortonworks;  
PWD=hortonworks;HTTPPath=gateway/sandbox/spark;
```

Features

More information is provided on the following features of the Hortonworks ODBC Driver with SQL Connector for Apache Spark:

- "SQL Connector for HiveQL" on page 51
- "Data Types" on page 51
- "Catalog and Schema Support" on page 52
- "spark_system Table" on page 53
- "Server-Side Properties" on page 53
- "Get Tables With Query" on page 53
- "Active Directory" on page 54
- "Write-back" on page 54

SQL Connector for HiveQL

The native query language supported by Spark is HiveQL. For simple queries, HiveQL is a subset of SQL-92. However, the syntax is different enough that most applications do not work with native HiveQL.

To bridge the difference between SQL and HiveQL, the SQL Connector feature translates standard SQL-92 queries into equivalent HiveQL queries. The SQL Connector performs syntactical translations and structural transformations. For example:

- **Quoted Identifiers:** The double quotes (") that SQL uses to quote identifiers are translated into back quotes (`) to match HiveQL syntax. The SQL Connector needs to handle this translation because even when a driver reports the back quote as the quote character, some applications still generate double-quoted identifiers.
- **Table Aliases:** Support is provided for the AS keyword between a table reference and its alias, which HiveQL normally does not support.
- **JOIN, INNER JOIN, and CROSS JOIN:** SQL JOIN, INNER JOIN, and CROSS JOIN syntax is translated to HiveQL JOIN syntax.
- **TOP N/LIMIT:** SQL TOP N queries are transformed to HiveQL LIMIT queries.

Data Types

The Hortonworks ODBC Driver with SQL Connector for Apache Spark supports many common data formats, converting between Spark data types and SQL data types.

Table 5 lists the supported data type mappings.

Table 5. Supported Data Types

Spark Type	SQL Type
BIGINT	SQL_BIGINT
BINARY	SQL_VARBINARY
BOOLEAN	SQL_BIT
CHAR(n)	SQL_CHAR
DATE	SQL_TYPE_DATE
DECIMAL	SQL_DECIMAL
DECIMAL(p,s)	SQL_DECIMAL
DOUBLE	SQL_DOUBLE
FLOAT	SQL_REAL
INT	SQL_INTEGER
SMALLINT	SQL_SMALLINT
STRING	SQL_VARCHAR
TIMESTAMP	SQL_TYPE_TIMESTAMP
TINYINT	SQL_TINYINT
VARCHAR(n)	SQL_VARCHAR

Note:

The aggregate types (ARRAY, MAP, and STRUCT) are not yet supported. Columns of aggregate types are treated as STRING columns.

Catalog and Schema Support

The Hortonworks ODBC Driver with SQL Connector for Apache Spark supports both catalogs and schemas to make it easy for the driver to work with various ODBC applications. Since Spark only organizes tables into schemas/databases, the driver provides a synthetic catalog called "SPARK" under which all of the schemas/databases are organized. The driver also maps the ODBC schema to the Spark schema/database.

The Hortonworks ODBC Driver with SQL Connector for Apache Spark supports querying against Hive schemas.

spark_system Table

A pseudo-table called `spark_system` can be used to query for Spark cluster system environment information. The pseudo-table is under the pseudo-schema called `spark_system`. The table has two `STRING` type columns, `envkey` and `envvalue`. Standard SQL can be executed against the `spark_system` table. For example:

```
SELECT * FROM SPARK.spark_system.spark_system WHERE envkey LIKE '%spark%'
```

The above query returns all of the Spark system environment entries whose key contains the word “spark.” A special query, `set -v`, is executed to fetch system environment information. Some versions of Spark do not support this query. For versions of Spark that do not support querying system environment information, the driver returns an empty result set.

Server-Side Properties

The Hortonworks ODBC Driver with SQL Connector for Apache Spark allows you to set server-side properties via a DSN. Server-side properties specified in a DSN affect only the connection that is established using the DSN.

You can also specify server-side properties for connections that do not use a DSN. To do this, use the Hortonworks Spark ODBC Driver Configuration tool that is installed with the Windows version of the driver, or set the appropriate configuration options in your connection string or the `hortonworks.sparkodbc.ini` file. Properties specified in the driver configuration tool or the `hortonworks.sparkodbc.ini` file apply to all connections that use the Hortonworks ODBC Driver with SQL Connector for Apache Spark.

For more information about setting server-side properties when using the Windows driver, see “Configuring Server-Side Properties” on page 17. For information about setting server-side properties when using the driver on a non-Windows platform, see “Driver Configuration Options” on page 55.

Get Tables With Query

The Get Tables With Query configuration option allows you to choose whether to use the `SHOW TABLES` query or the `GetTables` API call to retrieve table names from a database.

Shark Server 2 and Spark Thrift Server both have a limit on the number of tables that can be in a database when handling the `GetTables` API call. When the number of tables in a database is above the limit, the API call will return a stack overflow error or a timeout error. The exact limit and the error that appears depends on the JVM settings.

As a workaround for this issue, enable the Get Tables with Query configuration option (or `GetTablesWithQuery` key) to use the query instead of the API call.

Active Directory

The Hortonworks ODBC Driver with SQL Connector for Apache Spark supports Active Directory Kerberos on Windows. There are two prerequisites for using Active Directory Kerberos on Windows:

- MIT Kerberos is **not** installed on the client Windows machine.
- The MIT Kerberos Hadoop realm has been configured to trust the Active Directory realm so that users in the Active Directory realm can access services in the MIT Kerberos Hadoop realm. For more information, see *Setting up One-Way Trust with Active Directory* in the Hortonworks documentation at http://docs.hortonworks.com/HDPDocuments/HDP2/HDP-2.1.7/bk_installing_manually_book/content/ch23s05.html.

Write-back

The Hortonworks ODBC Driver with SQL Connector for Apache Spark supports translation for INSERT, UPDATE, and DELETE syntax when connecting to a Shark Server 2 or Spark Thrift Server instance.

Driver Configuration Options

Driver Configuration Options lists the configuration options available in the Hortonworks ODBC Driver with SQL Connector for Apache Spark alphabetically by field or button label. Options having only key names, that is, not appearing in the user interface of the driver, are listed alphabetically by key name.

When creating or configuring a connection from a Windows computer, the fields and buttons are available in the Hortonworks Spark ODBC Driver Configuration tool and the following dialog boxes:

- Hortonworks Spark ODBC Driver DSN Setup
- HTTP Properties
- SSL Options
- Advanced Options
- Server Side Properties

When using a connection string or configuring a connection from a Linux/Mac OS X computer, use the key names provided.

Note: You can pass in configuration options in your connection string, or set them in your `odbc.ini` and `hortonworks.sparkodbc.ini` files if you are using a non-Windows version of the driver. Configuration options set in a `hortonworks.sparkodbc.ini` file apply to all connections, whereas configuration options passed in in the connection string or set in an `odbc.ini` file are specific to a connection. Configuration options passed in using the connection string take precedence over configuration options set in `odbc.ini`. Configuration options set in `odbc.ini` take precedence over configuration options set in `hortonworks.sparkodbc.ini`

Configuration Options Appearing in the User Interface

The following configuration options are accessible via the Windows user interface for the Hortonworks ODBC Driver with SQL Connector for Apache Spark, or via the key name when using a connection string or configuring a connection from a Linux/Mac OS X computer:

- "Allow Common Name Host Name Mismatch" on page 56
- "Allow Self-signed Server Certificate" on page 57
- "Apply Properties with Queries" on page 57
- "Async Exec Poll Interval" on page 57
- "Host FQDN" on page 62
- "HTTP Path" on page 63
- Invalid Session Auto Recover on page 1
- "Mechanism" on page 63
- "Password" on page 64
- "Port" on page 64

- "Binary Column Length" on page 58
- "Client Certificate File" on page 58
- "Client Private Key File" on page 58
- "Client Private Key Password" on page 59
- "Convert Key Name to Lower Case" on page 59
- "Database" on page 59
- "Decimal Column Scale" on page 60
- "Default String Column Length" on page 60
- "Delegation UID" on page 60
- "Driver Config Take Precedence" on page 60
- "Enable SSL" on page 61
- "Fast SQLPrepare" on page 61
- "Get Tables With Query" on page 61
- "Host" on page 62
- "Realm" on page 64
- "Rows Fetched Per Block" on page 65
- "Save Password (Encrypted)" on page 65
- "Service Name" on page 65
- "Show System Table" on page 66
- "Socket Timeout" on page 66
- "Spark Server Type" on page 66
- "Thrift Transport" on page 67
- "Trusted Certificates" on page 67
- "Two Way SSL" on page 68
- "Unicode SQL Character Types" on page 68
- "Use Async Exec" on page 69
- "Use Native Query" on page 69
- "Use Only SSPI Plugin" on page 69
- "User Name" on page 70

Allow Common Name Host Name Mismatch

Key Name	Default Value	Required
CAIssuedCertNamesMismatch	Clear (0)	No

Description

When this option is enabled (1), the driver allows a CA-issued SSL certificate name to not match the host name of the Spark server.

When this option is disabled (0), the CA-issued SSL certificate name must match the host name of the Spark server.

Note: This setting is applicable only when SSL is enabled.

Note: This option is applicable only to the User Name and Password (SSL) and HTTPS authentication mechanisms.

Allow Self-signed Server Certificate

Key Name	Default Value	Required
AllowSelfSignedServerCert	Clear (0)	No

Description

When this option is enabled (1), the driver authenticates the Spark server even if the server is using a self-signed certificate.

When this option is disabled (0), the driver does not allow self-signed certificates from the server.

Note: This setting is applicable only when SSL is enabled.

Apply Properties with Queries

Key Name	Default Value	Required
ApplySSPWithQueries	Selected (1)	No

Description

When this option is enabled (1), the driver applies each server-side property by executing a `set SSPKey=SSPValue` query when opening a session to the Spark server.

When this option is disabled (0), the driver uses a more efficient method for applying server-side properties that does not involve additional network round-tripping. However, some Shark Server 2 or Spark Thrift Server builds are not compatible with the more efficient method.

Note: When connecting to a Shark Server instance, `ApplySSPWithQueries` is always enabled.

Async Exec Poll Interval

Key Name	Default Value	Required
AsyncExecPollInterval	100	No

Description

The time in milliseconds between each poll for the query execution status.

“Asynchronous execution” refers to the fact that the RPC call used to execute a query against Spark is asynchronous. It does not mean that ODBC asynchronous operations are supported.

Note: This option is applicable only to HDInsight clusters.

Binary Column Length

Key Name	Default Value	Required
BinaryColumnLength	32767	No

Description

The maximum data length for BINARY columns.

By default, the columns metadata for Spark does not specify a maximum data length for BINARY columns.

Client Certificate File

Key Name	Default Value	Required
ClientCert	None	Yes, if two-way SSL verification is enabled.

Description

The full path to the PEM file containing the client's SSL certificate.

Note: This setting is applicable only when two-way SSL is enabled.

Client Private Key File

Key Name	Default Value	Required
ClientPrivateKey	None	Yes, if two-way SSL verification is enabled.

Description

The full path to the PEM file containing the client's SSL private key.

If the private key file is protected with a password, then provide the password using the driver configuration option "Client Private Key Password" on page 59.

Note: This setting is applicable only when two-way SSL is enabled.

Client Private Key Password

Key Name	Default Value	Required
ClientPrivateKeyPassword	None	Yes, if two-way SSL verification is enabled and the client's private key file is protected with a password.

Description

The password of the private key file that is specified in the Client Private Key File field (the ClientPrivateKey key).

Convert Key Name to Lower Case

Key Name	Default Value	Required
LCaseSspKeyName	Selected (1)	No

Description

When this option is enabled (1), the driver converts server-side property key names to all lower case characters.

When this option is disabled (0), the driver does not modify the server-side property key names.

Database

Key Name	Default Value	Required
Schema	default	No

Description

The name of the database schema to use when a schema is not explicitly specified in a query. You can still issue queries on other schemas by explicitly specifying the schema in the query.

Note: To inspect your databases and determine the appropriate schema to use, type the **show databases** command at the Spark command prompt.

Decimal Column Scale

Key Name	Default Value	Required
DecimalColumnScale	10	No

Description

The maximum number of digits to the right of the decimal point for numeric data types.

Default String Column Length

Key Name	Default Value	Required
DefaultStringColumnLength	255	No

Description

The maximum number of characters that can be contained in STRING columns.

By default, the columns metadata for Spark does not specify a maximum length for STRING columns.

Delegation UID

Key Name	Default Value	Required
DelegationUID	None	No

Description

Use this option to delegate all operations against Spark to a user that is different than the authenticated user for the connection.

Note: This option is applicable only when connecting to a Shark Server 2 and Spark Thrift Server instance that supports this feature.

Driver Config Take Precedence

Key Name	Default Value	Required
DriverConfigTakePrecedence	Clear (0)	No

Description

When this option is enabled (1), driver-wide configurations take precedence over connection and DSN settings.

When this option is disabled (0), connection and DSN settings take precedence instead.

Enable SSL

Key Name	Default Value	Required
	Clear (0)	No

Description

When this option is enabled (1), the client verifies the Spark server using SSL.

When this option is disabled (0), SSL is disabled.

Note:

This option is applicable only when connecting to a Spark server that supports SSL.

Fast SQLPrepare

Key Name	Default Value	Required
FastSQLPrepare	Clear (0)	No

Description

When this option is enabled (1), the driver defers query execution to SQLExecute.

When this option is disabled (0), the driver does not defer query execution to SQLExecute.

When using Native Query mode, the driver will execute the HiveQL query to retrieve the result set metadata for SQLPrepare. As a result, SQLPrepare might be slow. If the result set metadata is not required after calling SQLPrepare, then enable Fast SQLPrepare.

Get Tables With Query

Key Name	Default Value	Required
GetTablesWithQuery	Clear (0)	No

Description

When this option is enabled (1), the driver uses the SHOW TABLES query to retrieve the names of the tables in a database.

When this option is disabled (0), the driver uses the GetTables Thrift API call to retrieve the names of the tables in a database.

Note: This option is applicable only when connecting to a Shark Server 2 or Spark Thrift Server instance.

Host

Key Name	Default Value	Required
HOST	None	Yes

Description

The IP address or host name of the Spark server.

Host FQDN

Key Name	Default Value	Required
KrbHostFQDN	None	Yes, if the authentication mechanism is Kerberos .

Description

The fully qualified domain name of the Shark Server 2 or Spark Thrift Server host.

You can set the value of Host FQDN to **_HOST** to use the Spark server host name as the fully qualified domain name for Kerberos authentication.

HTTP Path

Key Name	Default Value	Required
HTTPath	<p>/sparkhive2 if using Windows Azure HDInsight Service (6)</p> <p>/ if using non-Windows Azure HDInsight Service with Thrift Transport set to HTTP(2)</p>	No

Description

The partial URL corresponding to the Spark server.

Mechanism

Key Name	Default Value	Required
AuthMech	<p>No Authentication (0) if you are connecting to Spark Server 1.</p> <p>User Name (2) if you are connecting to Spark Server 2.</p>	No

Description

The authentication mechanism to use.

Select one of the following settings, or set the key to the corresponding number:

- No Authentication (0)
- Kerberos (1)
- User Name (2)
- User Name and Password (3)
- Windows Azure HDInsight Emulator (5)
- Windows Azure HDInsight Service (6)

Password

Key Name	Default Value	Required
PWD	None	Yes, if the authentication mechanism is User Name and Password, Windows Azure HDInsight Emulator , or Windows Azure HDInsight Service .

Description

The password corresponding to the user name that you provided in the User Name field (the UID key).

Port

Key Name	Default Value	Required
PORT	<ul style="list-style-type: none"> • non-HDInsight clusters: 10000 • Windows Azure HDInsight Emulator: 10001 • Windows Azure HDInsight Service: 443 	Yes

Description

The number of the TCP port that the Spark server uses to listen for client connections.

Realm

Key Name	Default Value	Required
KrbRealm	Depends on your Kerberos configuration.	No

Description

The realm of the Shark Server 2 or Spark Thrift Server host.

If your Kerberos configuration already defines the realm of the Shark Server 2 or Spark Thrift Server host as the default realm, then you do not need to configure this option.

Rows Fetched Per Block

Key Name	Default Value	Required
RowsFetchedPerBlock	10000	No

Description

The maximum number of rows that a query returns at a time.

Any positive 32-bit integer is a valid value, but testing has shown that performance gains are marginal beyond the default value of 10000 rows.

Save Password (Encrypted)

Key Name	Default Value	Required
N/A	Clear (0)	No

Description

This option is available only in the Windows driver. It appears in the Hortonworks Spark ODBC Driver DSN Setup dialog box and the SSL Options dialog box.

When this option is enabled (1), the specified password is saved in the registry.

When this option is disabled (0), the specified password is not saved in the registry.

Important: The password is obscured (not saved in plain text). However, it is still possible for the encrypted password to be copied and used.

Service Name

Key Name	Default Value	Required
		Yes, if the authentication mechanism is Kerberos .

Description

The Kerberos service principal name of the Spark server.

Show System Table

Key Name	Default Value	Required
ShowSystemTable	Clear (0)	No

Description

When this option is enabled (1), the driver returns the spark_system table for catalog function calls such as `SQLTables` and `SQLColumns`

When this option is disabled (0), the driver does not return the spark_system table for catalog function calls.

Socket Timeout

Key Name	Default Value	Required
SocketTimeout	30	No

Description

The number of seconds that an operation can remain idle before it is closed.

Note: This option is applicable only when asynchronous query execution is being used against Shark Server 2 or Spark Thrift Server instances.

Spark Server Type

Key Name	Default Value	Required
SparkServerType	Spark Thrift Server (3)	No

Description

Select **Shark Server** or set the key to 1 if you are connecting to a Shark Server instance.

Select **Shark Server 2** or set the key to 2 if you are connecting to a Shark Server 2 instance.

Select **Spark Thrift Server** or set the key to 3 if you are connecting to a Spark Thrift Server instance.

Thrift Transport

Key Name	Default Value	Required
ThriftTransport	Binary (0) if you are connecting to Spark Server 1. SASL (1) if you are connecting to Spark Server 2.	No

Description

The transport protocol to use in the Thrift layer.

Select one of the following settings, or set the key to the corresponding number:

- Binary (0)
- SASL (1)
- HTTP (2)

Trusted Certificates

Key Name	Default Value	Required
TrustedCerts	The cacerts.pem file in the lib folder or subfolder within the driver's installation directory. The exact file path varies depending on the version of the driver that is installed. For example, the path for the Windows driver is different from the path for the Mac OS X driver.	No

Description

The location of the PEM file containing trusted CA certificates for authenticating the Spark server when using SSL.

If this option is not set, then the driver will default to using the trusted CA certificates PEM file installed by the driver.

Note: This setting is applicable only when SSL is enabled.

Two Way SSL

Key Name	Default Value	Required
TwoWaySSL	Clear (0)	No

Description

When this option is enabled (1), the client and the Spark server verify each other using SSL. See also the driver configuration options "Client Certificate File" on page 58, "Client Private Key File" on page 58, and "Client Private Key Password" on page 59.

When this option is disabled (0), then the server does not verify the client. Depending on whether one-way SSL is enabled, the client may verify the server. For more information, see "Enable SSL" on page 61.

Note: This option is applicable only when connecting to a Spark server that supports SSL. You must enable SSL before Two Way SSL can be configured. For more information, see "Enable SSL" on page 61.

Unicode SQL Character Types

Key Name	Default Value	Required
UseUnicodeSqlCharacterTypes	Clear (0)	No

Description

When this option is enabled (1), the driver returns SQL_WVARCHAR for STRING and VARCHAR columns, and returns SQL_WCHAR for CHAR columns.

When this option is disabled (0), the driver returns SQL_VARCHAR for STRING and VARCHAR columns, and returns SQL_CHAR for CHAR columns.

Use Async Exec

Key Name	Default Value	Required
EnableAsyncExec	Clear (0)	No

Description

When this option is enabled (1), the driver uses an asynchronous version of the API call against Spark for executing a query.

When this option is disabled (0), the driver executes queries synchronously.

Use Native Query

Key Name	Default Value	Required
UseNativeQuery	Clear (0)	No

Description

When this option is enabled (1), the driver does not transform the queries emitted by an application, and executes HiveQL queries directly.

When this option is disabled (0), the driver transforms the queries emitted by an application and converts them into an equivalent form in HiveQL.

Note: If the application is Spark-aware and already emits HiveQL, then enable this option to avoid the extra overhead of query transformation.

Use Only SSPI Plugin

Key Name	Default Value	Required
UseOnlySSPI	Clear (0)	No

Description

When this option is enabled (1), the driver handles Kerberos authentication by using the SSPI plugin instead of MIT Kerberos by default.

When this option is disabled (0), the driver uses MIT Kerberos to handle Kerberos authentication, and only uses the SSPI plugin if the gssapi library is not available.

Important: This option is available only in the Windows driver.

User Name

Key Name	Default Value	Required
UID	For User Name authentication only, the default value is anonymous	No, if the authentication mechanism is User Name . Yes, if the authentication mechanism is User Name and Password, Windows Azure HDInsight Emulator, or Windows Azure HDInsight Service .

Description

The user name that you use to access Shark Server 2 or Spark Thrift Server.

Configuration Options Having Only Key Names

The following configuration options do not appear in the Windows user interface for the Hortonworks ODBC Driver with SQL Connector for Apache Spark and are only accessible when using a connection string or configuring a connection from a Linux/Mac OS X computer:

- "ADUserNameCase" on page 70
- "Driver" on page 71
- "ForceSynchronousExec" on page 71
- "http.header." on page 72
- "SSP_" on page 72

ADUserNameCase

Default Value	Required
Unchanged	No

Description

Use this option to control whether the driver changes the user name part of an AD Kerberos UPN to all upper case or all lower case. The following values are possible:

- **Upper:** Change the user name to all upper case.
- **Lower:** Change the user name to all lower case.
- **Unchanged:** Do not modify the user name.

Note: This option is applicable only when using Active Directory Kerberos from a Windows client machine to authenticate.

Driver

Default Value	Required
The default value varies depending on the version of the driver that is installed. For example, the value for the Windows driver is different from the value of the Mac OS X driver.	Yes

Description

The name of the installed driver (Hortonworks Spark ODBC Driver) or the absolute path of the Hortonworks ODBC Driver with SQL Connector for Apache Spark shared object file.

ForceSynchronousExec

Default Value	Required
0	No

Description

When this option is enabled (1), the driver is forced to execute queries synchronously when connected to an HDInsight cluster.

When this option is disabled (0), the driver is able to execute queries asynchronously when connected to an HDInsight cluster.

Note: This option is applicable only to HDInsight clusters.

http.header.

Default Value	Required
None	No

Description

Set a custom HTTP header by using the following syntax, where *HeaderKey* is the name of the header to set and *HeaderValue* is the value to assign to the header:

```
http.header.HeaderKey=HeaderValue
```

For example:

```
http.header.AUTHENTICATED_USER=john
```

After the driver applies the header, the `http.header.` prefix is removed from the DSN entry, leaving an entry of *HeaderKey=HeaderValue*

The example above would create the following custom HTTP header:

```
AUTHENTICATED_USER: john
```

Note:

The `http.header.` prefix is case-sensitive. This option is applicable only when you are using HTTP as the Thrift transport protocol. For more information, see "Thrift Transport" on page 67.

SSP_

Default Value	Required
None	No

Description

Set a server-side property by using the following syntax, where *SSPKey* is the name of the server-side property to set and *SSPValue* is the value to assign to the server-side property:

```
SSP_SSPKey=SSPValue
```

For example:

```
SSP_mapred.queue.names=myQueue
```

After the driver applies the server-side property, the `SSP_` prefix is removed from the DSN entry, leaving an entry of *SSPKey=SSPValue*

Note: The `SSP_` prefix must be upper case.



Architecting the Future of Big Data

Contact Us

If you have difficulty using the Hortonworks ODBC Driver with SQL Connector for Apache Spark, please contact our support staff. We welcome your questions, comments, and feature requests.

Please have a detailed summary of the client and server environment (OS version, patch level, Hadoop distribution version, Spark version, configuration, etc.) ready, before you call or write us. Supplying this information accelerates support.

By telephone:

USA: (855) 8-HORTON

International: (408) 916-4121

On the Internet:

Visit us at www.hortonworks.com

Third Party Trademarks and Licenses

Third Party Trademarks

Kerberos is a trademark of the Massachusetts Institute of Technology (MIT).

Linux is the registered trademark of Linus Torvalds in Canada, United States and/or other countries.

Microsoft, MSDN, Windows, Windows Azure, Windows Server, Windows Vista, and the Windows start button are trademarks or registered trademarks of Microsoft Corporation or its subsidiaries in Canada, United States and/or other countries.

Red Hat, Red Hat Enterprise Linux, and CentOS are trademarks or registered trademarks of Red Hat, Inc. or its subsidiaries in Canada, United States and/or other countries.

Apache Spark, Apache, and Spark are trademarks or registered trademarks of The Apache Software Foundation or its subsidiaries in Canada, United States and/or other countries.

All other trademarks are trademarks of their respective owners.

Third Party Licenses

Boost Software License - Version 1.0 - August 17th, 2003

Permission is hereby granted, free of charge, to any person or organization obtaining a copy of the software and accompanying documentation covered by this license (the "Software") to use, reproduce, display, distribute, execute, and transmit the Software, and to prepare derivative works of the Software, and to permit third-parties to whom the Software is furnished to do so, all subject to the following:

The copyright notices in the Software and this entire statement, including the above license grant, this restriction and the following disclaimer, must be included in all copies of the Software, in whole or in part, and all derivative works of the Software, unless such copies or derivative works are solely in the form of machine-executable object code generated by a source language processor.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE AND NON-INFRINGEMENT. IN NO EVENT SHALL THE COPYRIGHT HOLDERS OR ANYONE DISTRIBUTING THE SOFTWARE BE LIABLE FOR ANY DAMAGES OR OTHER LIABILITY, WHETHER IN CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

cURL

COPYRIGHT AND PERMISSION NOTICE

Copyright (c) 1996 - 2015, Daniel Stenberg, daniel@haxx.se.

All rights reserved.

Permission to use, copy, modify, and distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Except as contained in this notice, the name of a copyright holder shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization of the copyright holder.

Cyrus SASL

Copyright (c) 1994-2012 Carnegie Mellon University. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The name "Carnegie Mellon University" must not be used to endorse or promote products derived from this software without prior written permission. For permission or any other legal details, please contact

Office of Technology Transfer
Carnegie Mellon University
5000 Forbes Avenue
Pittsburgh, PA 15213-3890
(412) 268-4387, fax: (412) 268-7395
tech-transfer@andrew.cmu.edu

4. Redistributions of any form whatsoever must retain the following acknowledgment:

"This product includes software developed by Computing Services at Carnegie Mellon University (<http://www.cmu.edu/computing/>)."

CARNEGIE MELLON UNIVERSITY DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS, IN NO EVENT SHALL CARNEGIE MELLON UNIVERSITY BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

dtoa License

The author of this software is David M. Gay.

Copyright (c) 1991, 2000, 2001 by Lucent Technologies.

Permission to use, copy, modify, and distribute this software for any purpose without fee is hereby granted, provided that this entire notice is included in all copies of any software which is or includes a copy or modification of this software and in all copies of the supporting documentation for such software.

THIS SOFTWARE IS BEING PROVIDED "AS IS", WITHOUT ANY EXPRESS OR IMPLIED WARRANTY. IN PARTICULAR, NEITHER THE AUTHOR NOR LUCENT MAKES ANY REPRESENTATION OR WARRANTY OF ANY KIND CONCERNING THE MERCHANTABILITY OF THIS SOFTWARE OR ITS FITNESS FOR ANY PARTICULAR PURPOSE.

Expat

Copyright (c) 1998, 1999, 2000 Thai Open Source Software Center Ltd

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR

IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

ICU License - ICU 1.8.1 and later

COPYRIGHT AND PERMISSION NOTICE

Copyright (c) 1995-2014 International Business Machines Corporation and others

All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, provided that the above copyright notice(s) and this permission notice appear in all copies of the Software and that both the above copyright notice(s) and this permission notice appear in supporting documentation.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR HOLDERS INCLUDED IN THIS NOTICE BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

Except as contained in this notice, the name of a copyright holder shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization of the copyright holder.

All trademarks and registered trademarks mentioned herein are the property of their respective owners.

OpenSSL

Copyright (c) 1998-2011 The OpenSSL Project. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. All advertising materials mentioning features or use of this software must display the following acknowledgment:

"This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit. (<http://www.openssl.org/>)"

4. The names "OpenSSL Toolkit" and "OpenSSL Project" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact openssl-core@openssl.org.
5. Products derived from this software may not be called "OpenSSL" nor may "OpenSSL" appear in their names without prior written permission of the OpenSSL Project.
6. Redistributions of any form whatsoever must retain the following acknowledgment:

"This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit (<http://www.openssl.org/>)"

THIS SOFTWARE IS PROVIDED BY THE OpenSSL PROJECT "AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE OpenSSL PROJECT OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This product includes cryptographic software written by Eric Young(eay@cryptsoft.com).
This product includes software written by Tim Hudson (tjh@cryptsoft.com).

Original SSLeay License

Copyright (C) 1995-1998 Eric Young (eay@cryptsoft.com)

All rights reserved.

This package is an SSL implementation written by Eric Young (eay@cryptsoft.com). The implementation was written so as to conform with Netscapes SSL.

This library is free for commercial and non-commercial use as long as the following conditions are adhered to. The following conditions apply to all code found in this distribution, be it the RC4, RSA, lhash, DES, etc., code; not just the SSL code. The SSL documentation included with this distribution is covered by the same copyright terms except that the holder is Tim Hudson (tjh@cryptsoft.com).

Copyright remains Eric Young's, and as such any Copyright notices in the code are not to be removed. If this package is used in a product, Eric Young should be given attribution as the author of the parts of the library used. This can be in the form of a textual message at program startup or in documentation (online or textual) provided with the package.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. All advertising materials mentioning features or use of this software must display the following acknowledgement:

"This product includes cryptographic software written by Eric Young (eay@cryptsoft.com)"

The word 'cryptographic' can be left out if the routines from the library being used are not cryptographic related :-).

4. If you include any Windows specific code (or a derivative thereof) from the apps directory (application code) you must include an acknowledgement:

"This product includes software written by Tim Hudson (tjh@cryptsoft.com)"

THIS SOFTWARE IS PROVIDED BY ERIC YOUNG ``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The licence and distribution terms for any publically available version or derivative of this code cannot be changed. i.e. this code cannot simply be copied and put under another distribution licence [including the GNU Public Licence.]

rapidjson

Tencent is pleased to support the open source community by making RapidJSON available.

Copyright (C) 2015 THL A29 Limited, a Tencent company, and Milo Yip. All rights reserved.

If you have downloaded a copy of the RapidJSON binary from Tencent, please note that the RapidJSON binary is licensed under the MIT License.

If you have downloaded a copy of the RapidJSON source code from Tencent, please note that RapidJSON source code is licensed under the MIT License, except for the third-party components listed below which are subject to different license terms. Your integration of RapidJSON into your own projects may require compliance with the MIT License, as well as the other licenses applicable to the third-party components included within RapidJSON.

A copy of the MIT License is included in this file.

Other dependencies and licenses:

Open Source Software Licensed Under the BSD License:

The msinttypes r29
Copyright (c) 2006-2013 Alexander Chemeris
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- Neither the name of copyright holder nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS AND CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Open Source Software Licensed Under the JSON License:

json.org

Copyright (c) 2002 JSON.org
All Rights Reserved.

JSON_checker
Copyright (c) 2002 JSON.org
All Rights Reserved.

Terms of the JSON License:

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

The Software shall be used for Good, not Evil.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Terms of the MIT License:

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM,

DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Stringencoders License

Copyright 2005, 2006, 2007

Nick Galbreath -- nickg [at] modp [dot] com

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

Neither the name of the modp.com nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This is the standard "new" BSD license:

<http://www.opensource.org/licenses/bsd-license.php>

zlib License

Copyright (C) 1995-2013 Jean-loup Gailly and Mark Adler

This software is provided 'as-is', without any express or implied warranty. In no event will the authors be held liable for any damages arising from the use of this software.

Permission is granted to anyone to use this software for any purpose, including commercial applications, and to alter it and redistribute it freely, subject to the following restrictions:

1. The origin of this software must not be misrepresented; you must not claim that you wrote the original software. If you use this software in a product, an acknowledgment in the product documentation would be appreciated but is not required.
2. Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software.
3. This notice may not be removed or altered from any source distribution.

Jean-loup Gailly

Mark Adler

jloup@gzip.org

madler@alumni.caltech.edu

Apache License, Version 2.0

The following notice is included in compliance with the Apache License, Version 2.0 and is applicable to all software licensed under the Apache License, Version 2.0.

Apache License

Version 2.0, January 2004

<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work

constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
 - (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. **Disclaimer of Warranty.** Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. **Limitation of Liability.** In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. **Accepting Warranty or Additional Liability.** While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.



See the License for the specific language governing permissions and limitations under the License.

This product includes software that is licensed under the Apache License, Version 2.0 (listed below):

Apache Hive

Copyright © 2008-2015 The Apache Software Foundation

Apache Spark

Copyright © 2014 The Apache Software Foundation

Apache Thrift

Copyright © 2006-2010 The Apache Software Foundation

Apache ZooKeeper

Copyright © 2010 The Apache Software Foundation

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.