



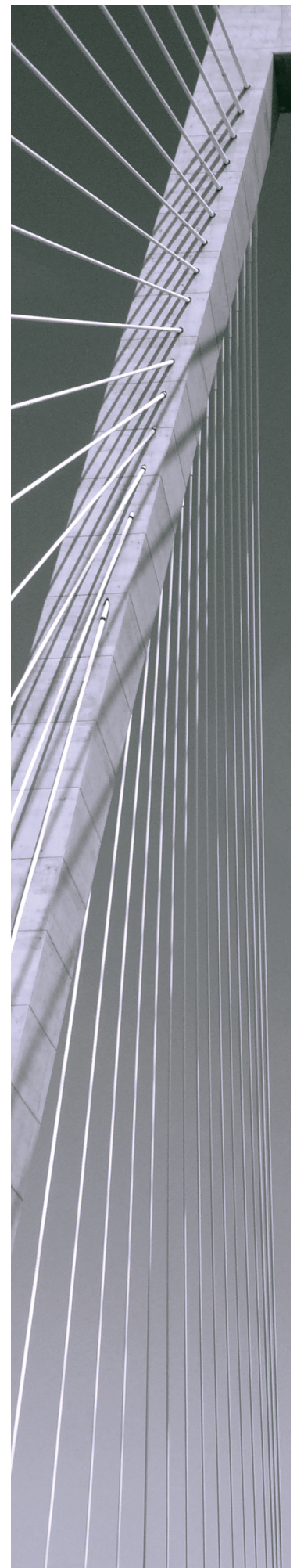
Simba MongoDB ODBC Driver with SQL Connector

Installation and Configuration Guide

Simba Technologies Inc.

Version 2.3.0

November 30, 2017



Copyright © 2017 Simba Technologies Inc. All Rights Reserved.

Information in this document is subject to change without notice. Companies, names and data used in examples herein are fictitious unless otherwise noted. No part of this publication, or the software it describes, may be reproduced, transmitted, transcribed, stored in a retrieval system, decompiled, disassembled, reverse-engineered, or translated into any language in any form by any means for any purpose without the express written permission of Simba Technologies Inc.

Trademarks

Simba, the Simba logo, SimbaEngine, and Simba Technologies are registered trademarks of Simba Technologies Inc. in Canada, United States and/or other countries. All other trademarks and/or servicemarks are the property of their respective owners.

Contact Us

Simba Technologies Inc.
938 West 8th Avenue
Vancouver, BC Canada
V5Z 1E5

Tel: +1 (604) 633-0008

Fax: +1 (604) 633-0004

www.simba.com

About This Guide

Purpose

The *Simba MongoDB ODBC Driver with SQL Connector Installation and Configuration Guide* explains how to install and configure the Simba MongoDB ODBC Driver with SQL Connector. The guide also provides details related to features of the driver.

Audience

The guide is intended for end users of the Simba MongoDB ODBC Driver, as well as administrators and developers integrating the driver.

Knowledge Prerequisites

To use the Simba MongoDB ODBC Driver, the following knowledge is helpful:

- Familiarity with the platform on which you are using the Simba MongoDB ODBC Driver
- Ability to use the data source to which the Simba MongoDB ODBC Driver is connecting
- An understanding of the role of ODBC technologies and driver managers in connecting to a data source
- Experience creating and configuring ODBC connections
- Exposure to SQL

Document Conventions

Italics are used when referring to book and document titles.

Bold is used in procedures for graphical user interface elements that a user clicks and text that a user types.

Monospace font indicates commands, source code, or contents of text files.

Note:

A text box with a pencil icon indicates a short note appended to a paragraph.

! Important:

A text box with an exclamation mark indicates an important comment related to the preceding paragraph.

Table of Contents

About the Simba MongoDB ODBC Driver	8
Windows Driver	9
Windows System Requirements	9
Installing the Driver on Windows	9
Creating a Data Source Name on Windows	10
Configuring Authentication on Windows	12
Configuring Advanced Options on Windows	14
Configuring Write-Back Options on Windows	16
Configuring SSL Verification on Windows	17
Exporting a Data Source Name on Windows	19
Importing a Data Source Name on Windows	19
Configuring Logging Options on Windows	19
Verifying the Driver Version Number on Windows	21
macOS Driver	23
macOS System Requirements	23
Installing the Driver on macOS	23
Verifying the Driver Version Number on macOS	24
Linux Driver	25
Linux System Requirements	25
Installing the Driver Using the RPM File	26
Installing the Driver Using the Tarball Package	27
Verifying the Driver Version Number on Linux	28
AIX Driver	29
AIX System Requirements	29
Installing the Driver on AIX	29
Configuring the ODBC Driver Manager on Non-Windows Machines	31
Specifying ODBC Driver Managers on Non-Windows Machines	31
Specifying the Locations of the Driver Configuration Files	32
Configuring ODBC Connections on a Non-Windows Machine	34
Creating a Data Source Name on a Non-Windows Machine	34
Configuring a DSN-less Connection on a Non-Windows Machine	37
Configuring Authentication on a Non-Windows Machine	40
Configuring SSL Verification on a Non-Windows Machine	41

Configuring Logging Options on a Non-Windows Machine	42
Testing the Connection on a Non-Windows Machine	44
Using a Connection String	46
DSN Connection String Example	46
DSN-less Connection String Examples	46
Features	51
Catalog Support	51
Double-Buffering	51
SQL Connector	52
Data Types	52
Schema Definitions	56
Virtual Tables	58
Write-back	61
Security and Authentication	63
Driver Configuration Options	64
Allow Self-Signed Certificates	65
Authentication Source	65
Batch Size	66
Binary Column Size	66
Certificate Authority File	66
Certificate Authority Directory	67
Certificate Revocation List File	67
Connect to Replica Set	67
Database	68
Documents to Fetch Per Block	68
Documents to Sample	68
Enable Double-Buffering	69
Enable Json Read/Write Mode	69
Enable Mixed Type Filter	70
Enable Passdown	71
Enable SSL	71
Encrypt Password	71
Expose Binary as SQL_LONGVARBINARY	72
Expose Strings as SQL_WVARCHAR	72
JSON Column Size	73
Journaled Writes	73

Installation and Configuration Guide

Local File	74
Log Level	74
Log Path	75
Max File Size	75
Max Number Files	76
Mechanism (Authentication)	76
Mechanism (Metadata)	77
No Cursor Timeout	77
Password	78
PEM Key File	78
PEM Key Password	79
Port	79
Read Preference	79
Replica Set Name	80
Sampling Method	80
Secondary Servers	81
Server	81
Service Name	81
Socket Timeout	82
Step Size	82
String Column Size	82
Timeout	83
UseLogPrefix	83
Username	84
Write Concern	85
Third-Party Trademarks	86
Third-Party Licenses	87

About the Simba MongoDB ODBC Driver

The Simba MongoDB ODBC Driver enables Business Intelligence (BI), analytics, and reporting on data that is stored in MongoDB databases. The driver complies with the ODBC 3.80 data standard and adds important functionality such as Unicode, as well as 32- and 64-bit support for high-performance computing environments on all platforms.

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 *Data Access Standards* on the Simba Technologies

website: <https://www.simba.com/resources/data-access-standards-glossary>. For complete information about the ODBC specification, see the *ODBC API Reference* from the Microsoft documentation: <https://docs.microsoft.com/en-us/sql/odbc/reference/syntax/odbc-api-reference>.

The Simba MongoDB ODBC Driver is available for Microsoft® Windows®, Linux, AIX, and macOS platforms.

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

 **Note:**

For information about how to use the driver in various BI tools, see the *Simba ODBC Drivers Quick Start Guide for Windows*: http://cdn.simba.com/docs/ODBC_QuickstartGuide/content/quick_start/intro.htm.

Windows Driver

Windows System Requirements

The Simba MongoDB ODBC Driver supports MongoDB versions 3.0, 3.2, and 3.4.

Install the driver on client machines where the application is installed. Each machine that you install the driver on must meet the following minimum system requirements:

- One of the following operating systems:
 - Windows 7, 8.1, or 10
 - Windows Server 2008 or later
- 600 MB of available disk space
- The machine does not have version 1.8.4 or earlier of the Simba MongoDB ODBC Driver installed.
- Visual C++ Redistributable for Visual Studio 2013 installed (with the same bitness as the driver that you are installing).
You can download the installation packages at <https://www.microsoft.com/en-ca/download/details.aspx?id=40784>.

To install the driver, you must have Administrator privileges on the machine.

Installing the Driver on Windows

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 that matches the bitness of the client application:

- `SimbaMongoDBODBC32.msi` for 32-bit applications
- `SimbaMongoDBODBC64.msi` for 64-bit applications

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

To install the Simba MongoDB ODBC Driver on Windows:

1. Depending on the bitness of your client application, double-click to run **SimbaMongoDBODBC32.msi** or **SimbaMongoDBODBC64.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**.
7. If you received a license file through email, then copy the license file into the `\lib` subfolder of the installation folder you selected above. You must have Administrator privileges when changing the contents of this folder.

Creating a Data Source Name on Windows

Typically, after installing the Simba MongoDB ODBC Driver, you need to create a Data Source Name (DSN).

Alternatively, for information about DSN-less connections, see [Using a Connection String](#) on page 46.

To create a Data Source Name on Windows:

1. Open the ODBC Administrator:
 - If you are using Windows 7 or earlier, click **Start**  > **All Programs** > **Simba MongoDB ODBC Driver 2.3** > **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.

 **Note:**

Make sure to select the ODBC Data Source Administrator that has the same bitness as the client application that you are using to connect to MongoDB.

2. In the ODBC Data Source Administrator, click the **Drivers** tab, and then scroll down as needed to confirm that the Simba MongoDB 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.

 **Note:**

It is recommended that you create a System DSN instead of a User DSN. Some applications load the data using a different user account, and might not be able to detect User DSNs that are created under another user account.

4. Click **Add**.
5. In the Create New Data Source dialog box, select **Simba MongoDB ODBC Driver** and then click **Finish**. The Simba MongoDB ODBC Driver DSN Setup dialog box opens.
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 **Server** field, type the name or IP address of the host where your MongoDB instance is running.
9. In the **Port** field, type the number of the TCP port that the server uses to listen for client connections.

 **Note:**

The default port used by MongoDB is 27017.

10. In the **Database** field, type the name of the database that you want to access.
11. If you are connecting to a replica set in your MongoDB implementation, select the **Connect to Replica Set** check box and then do the following:
 - a. In the **Replica Set Name** field, type the name of the replica set (this is a required field).
 - b. In the **Secondary Servers** field, type a comma-separated list of the servers in the replica set. You can indicate the TCP port that each server is using to listen for client connections by appending a colon (:) and the port number to the server name or IP address.
 - c. In the **Read Preference** drop-down list, select the appropriate option to specify how the driver routes read operations to the members of a replica set.
12. If the database that you are connecting to requires authentication, then use the options in the Authentication area to configure authentication as needed. For more information, see [Configuring Authentication on Windows](#) on page 12.
13. To configure advanced driver options including write-back options, click **Advanced Options**. For more information, see [Configuring Advanced Options on Windows](#) on page 14.
14. To configure client-server verification over SSL, click **SSL Options**. For more information, see [Configuring SSL Verification on Windows](#) on page 17.
15. To launch the Schema Editor application and create or customize the schema definition that the driver uses when connecting to the database, click **Schema Editor**.

 **Note:**

For information about how to use the Schema Editor, see the *Schema Editor User Guide* located in the installation directory of the driver.

- On Windows 7 or earlier, the guide is available from the **Simba MongoDB ODBC Driver** program group in the Start menu.
- On Windows 8 or later, you can search for the guide on the Start screen.

16. To configure logging behavior for the driver, click **Logging Options**. For more information, see [Configuring Logging Options on Windows](#) on page 19.
17. 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 Simba MongoDB ODBC Driver DSN Setup dialog box are correct. Contact your MongoDB server administrator as needed.

18. To save your settings and close the Simba MongoDB ODBC Driver DSN Setup dialog box, click **OK**.
19. To close the ODBC Data Source Administrator, click **OK**.

Configuring Authentication on Windows

Some MongoDB databases require authentication. You can configure the Simba MongoDB ODBC Driver to provide your credentials and authenticate the connection to the database using one of the following methods:

- [Using SCRAM-SHA-1](#) on page 12
- [Using Kerberos](#) on page 13
- [Using LDAP](#) on page 13

Using SCRAM-SHA-1

You can configure the driver to use the SCRAM-SHA-1 protocol to authenticate the connection. SCRAM-SHA-1 is the default authentication protocol used by MongoDB.

To configure SCRAM-SHA-1 authentication on Windows:

1. To access authentication options, open the ODBC Data Source Administrator where you created the DSN, select the DSN, and then click **Configure**.
2. In the **Mechanism** drop-down list, select **MongoDB User Name and Password**.

3. To use a database other than the admin database to check your credentials, type the name of the database in the **Authentication Source** field.
4. In the **Username** field, type an appropriate user name for accessing the MongoDB database.
5. In the **Password** field, type the password corresponding to the user name you typed above.
6. Encrypt your credentials by selecting one of the following:
 - If the credentials are used only by the current Windows user, select **Current User Only**.
 - Or, if the credentials are used by all users on the current Windows machine, select **All Users Of This Machine**.
7. To save your settings and close the dialog box, click **OK**.

Using Kerberos

You can configure the driver to use the Kerberos protocol to authenticate the connection.

Kerberos must be installed and configured before you can use this authentication mechanism. For information about how to install and configure Kerberos, see the MIT Kerberos Documentation: <http://web.mit.edu/kerberos/krb5-latest/doc/>.

To configure Kerberos authentication on Windows:

1. To access authentication options, open the ODBC Data Source Administrator where you created the DSN, select the DSN, and then click **Configure**.
2. In the **Mechanism** drop-down list, select **Kerberos**.
3. In the **Service Name** field, type the service name of the MongoDB server.
4. To save your settings and close the dialog box, click **OK**.

Using LDAP

You can configure the driver to use the LDAP protocol to authenticate the connection.

To configure LDAP authentication on Windows:

1. To access authentication options, open the ODBC Data Source Administrator where you created the DSN, select the DSN, and then click **Configure**.
2. In the **Mechanism** drop-down list, select **LDAP**.
3. In the **Username** field, type an appropriate user name for accessing the MongoDB database.
4. In the **Password** field, type the password corresponding to the user name you typed above.

5. Encrypt your credentials by selecting one of the following:
 - If the credentials are used only by the current Windows user, select **Current User Only**.
 - Or, if the credentials are used by all users on the current Windows machine, select **All Users Of This Machine**.
6. To save your settings and close the dialog box, click **OK**.

Configuring Advanced Options on Windows

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

To configure advanced options on Windows:

1. To access advanced options, open the ODBC Data Source Administrator where you created the DSN, then select the DSN, then click **Configure**, and then click **Advanced Options**.
2. To retrieve data using double-buffering instead of single-buffering, select the **Enable Double-Buffering** check box. You can configure the buffer size using the **Documents to fetch per block** field.
3. In the **Documents to fetch per block** field, type the maximum number of documents that a query returns at a time. This setting also determines the buffer size used when double-buffering is enabled.
4. To return MongoDB String data as SQL_WVARCHAR instead of SQL_VARCHAR, select the **Expose Strings as SQL_WVARCHAR** check box.
5. In the **String Column Size** field, type the maximum data length for String columns.
6. To return MongoDB Binary data as SQL_LONGVARBINARY instead of SQL_VARBINARY, select the **Expose Binary as SQL_LONGVARBINARY** check box.
7. In the **Binary Column Size** field, type the maximum data length for Binary columns.
8. To configure the driver to create a column for retrieving or storing documents as JSON-formatted strings:
 - a. Select the **Enable JSON Read/Write Mode** check box.
 - b. In the **JSON Column Size** field, type the default column length for the JSON fields.
9. To configure the driver to optimize joins between virtual tables and pass filtering and aggregation optimizations to the MongoDB database for handling, select the **Enable Passdown** check box.
10. To allow data values to pass filtering operations even if the values are stored as a different data type than the specified filter term, select the **Enable Mixed Type Filter** check box.

 **Note:**

Mixed type filtering requires the driver to scan the entire MongoDB collection. You can disable this feature to increase the driver's performance, but note that doing so also alters the results of your queries.

11. Use the options in the Metadata area to specify the schema definition to use when connecting to the database:
 - To configure the driver to use a schema definition stored in a JSON file, select **Local File** from the **Mechanism** drop-down list, and then click **Browse** and select the JSON file that you want to use.
 - To configure the driver to use a schema definition stored in the MongoDB database that you are connecting to, in the **Mechanism** drop-down list, select **Database**.

 **Note:**

For information about how to create a schema definition using the Schema Editor application, see the *Schema Editor User Guide* located in the installation directory of the driver.

- On Windows 7 or earlier, the guide is available from the **Simba MongoDB ODBC Driver** program group in the Start menu.
 - On Windows 8 or later, you can search for the guide on the Start screen.
12. Use the options in the Sampling area to configure how the driver samples data to generate temporary schema definitions:
 - a. In the **Sampling Method** list, select the sampling method to use. Select **Forward** to sample data sequentially starting from the first record in the database, or select **Backward** to sample sequentially from the last record, or select **Random** to sample a randomly selected set of records.
 - b. In the **Documents to sample** field, type the maximum number of documents that the driver can sample to generate the schema definition. To sample every document in the database, type **0**.

! Important:

- Make sure to configure the driver to sample all the necessary data. Documents that are not sampled do not get included in the schema definition, and consequently do not become available in ODBC applications.
- Typically, sampling a large number of documents results in a schema definition that is more accurate and better able to represent all the data in the database. However, the sampling process may take longer than expected when many documents are sampled, especially if the database contains complex, nested data structures.
- The sampling options set here are for fallback sampling by the driver only. The sampling behavior of the Schema Editor application is set separately.


13. In the **Step Size** field, type the interval at which the driver samples a record when scanning through the database. For example, if you set this option to **2**, then the driver samples every second record in the database starting from the first record.
14. To configure write-back behavior in the driver, click **Writeback Options**. For more information, see [Configuring Write-Back Options on Windows](#) on page 16.
15. To save your settings and close the Advanced Options dialog box, click **OK**.
16. To close the Simba MongoDB ODBC Driver DSN Setup dialog box, click **OK**.

Configuring Write-Back Options on Windows

You can configure write-back options to modify how the driver writes data to the MongoDB data store.

To configure write-back options on Windows:

1. To access write-back options, open the ODBC Data Source Administrator where you created the DSN, select the DSN, click **Configure**, click **Advanced Options**, and then click **Writeback Options**.
2. In the **Batch Size** field, type the maximum number of documents that the driver can handle at one time during a write operation.
3. Use the options in the Write Concern Settings area to configure how the driver reports the success of a write operation:
 - a. In the **Write Concern** field, type the total number of primary and secondary servers that must acknowledge a write operation in order for the driver to report a successful write operation.

 **Note:**

- If you use a value that is greater than 1, make sure to also specify an appropriate value in the **Timeout** field. Setting this option to a value greater than 1 without specifying a timeout interval may cause the driver to wait indefinitely for replica set members to come online.
- The process for acknowledging a write operation typically takes four times longer than an INSERT or UPDATE operation, but it is necessary if fault tolerance is important.

- b. In the **Timeout** field, type the maximum number of seconds that the driver waits for a secondary server to acknowledge a write operation before reporting that the operation has failed.
- c. To require the data to be committed to the journal before a write operation can be acknowledged, select the **Journalled Writes** check box.

 **Note:**

For detailed information about the write concern feature in MongoDB, see "Write Concern" in the *MongoDB Manual*:
<https://docs.mongodb.org/manual/core/write-concern/>.

4. To save your settings and close the Writeback Options dialog box, click **OK**.
5. Click **OK** to close the Advanced Options dialog box, and then click **OK** to close the Simba MongoDB ODBC Driver DSN Setup dialog box.

Configuring SSL Verification on Windows

If you are connecting to a MongoDB server that has Secure Sockets Layer (SSL) enabled, then you can configure the driver to connect to an SSL-enabled socket. When connecting to a server over SSL, the driver supports identity verification between the client and the server.

Configuring an SSL Connection without Identity Verification

You can configure a connection that uses SSL but does not verify the identity of the client or the server.

To configure an SSL connection without verification on Windows:

1. To access the 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**.
2. Select the **Enable SSL** check box.

3. Select the **Allow Self-Signed Certificates** check box.
4. To save your settings and close the dialog box, click **OK**.

Configuring One-way SSL Verification

You can configure one-way verification so that the client verifies the identity of the MongoDB server.

To configure one-way SSL verification on Windows:

1. To access the 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**.
2. Select the **Enable SSL** check box.
3. Choose one:
 - To verify the server using a certificate from a specific PEM file, in the **Certificate Authority File** field, specify the full path of the PEM file.
 - Or, to verify the server using certificates stored in multiple PEM files, in the **Certificate Authority Directory** field, specify the full path to the directory where the PEM files are located.
4. In the **Certificate Revocation List File** field, specify the full path of the PEM file containing the list of revoked certificates.
5. To save your settings and close the dialog box, click **OK**.

Configuring Two-way SSL Verification

You can configure two-way SSL verification so that the client and the MongoDB server verify each other.

To configure two-way SSL verification on Windows:

1. To access the 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**.
2. Select the **Enable SSL** check box.
3. In the **PEM Key File** field, specify the full path of the PEM file containing the certificate for verifying the client.
4. If the client certificate is protected with a password, type the password in the **PEM Key Password** field.
5. Choose one:
 - To verify the server using a certificate from a specific PEM file, in the **Certificate Authority File** field, specify the full path of the PEM file.

- Or, to verify the server using certificates stored in multiple PEM files, in the **Certificate Authority Directory** field, specify the full path to the directory where the PEM files are located.
6. In the **Certificate Revocation List File** field, specify the full path of the PEM file containing the list of revoked certificates.
 7. To save your settings and close the dialog box, click **OK**.

Exporting a Data Source Name on Windows

After you configure a DSN, you can export it to be used on other machines. When you export a DSN, all of its configuration settings are saved in a `.sdc` file. You can then distribute the `.sdc` file to other users so that they can import your DSN configuration and use it on their machines.

To export a Data Source Name on Windows:

1. Open the ODBC Data Source Administrator where you created the DSN, select the DSN, click **Configure**, and then click **Logging Options**.
2. Click **Export Configuration**, specify a name and location for the exported DSN, and then click **Save**.

Your DSN is saved as a `.sdc` file in the location that you specified.

Importing a Data Source Name on Windows

You can import a DSN configuration from a `.sdc` file and then use those settings to connect to your data source.

To import a Data Source Name on Windows:

1. Open the ODBC Data Source Administrator where you created the DSN, select the DSN, click **Configure**, and then click **Logging Options**.
2. Click **Import Configuration**, browse to select the `.sdc` file that you want to import the DSN configuration from, and then click **Open**.
3. Click **OK** to close the Logging Options dialog box.

The Simba MongoDB ODBC Driver DSN Setup dialog box loads the configuration settings from the selected `.sdc` file. You can now save this DSN and use it to connect to your data source.

Configuring Logging Options on Windows

To help troubleshoot issues, you can enable logging. In addition to functionality provided in the Simba MongoDB ODBC Driver, 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 settings for logging apply to every connection that uses the Simba MongoDB ODBC Driver, so make sure to disable the feature after you are done using it.

To enable driver logging on Windows:

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 logging level corresponding to the amount of information that you want to include in log files:

Logging Level	Description
OFF	Disables all logging.
FATAL	Logs severe error events that lead the driver to abort.
ERROR	Logs error events that might allow the driver to continue running.
WARNING	Logs events that might result in an error if action is not taken.
INFO	Logs general information that describes the progress of the driver.
DEBUG	Logs detailed information that is useful for debugging the driver.
TRACE	Logs all driver activity.

3. In the **Log Path** field, specify the full path to the folder where you want to save log files. You can type the path into the field, or click **Browse** and then browse to select the folder.
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 make sure that the new settings take effect.

The Simba MongoDB ODBC Driver produces a log file named `simbamongodboDBC_driver.log` at the location that you specify in the Log Path field.

If you enable the `UseLogPrefix` connection property, the driver prefixes the log file name with the user name associated with the connection and the process ID of the application through which the connection is made. For more information, see [UseLogPrefix](#) on page 83.


To disable driver logging on Windows:

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**.
4. Restart your ODBC application to make sure that the new settings take effect.


Verifying the Driver Version Number on Windows

If you need to verify the version of the Simba MongoDB ODBC Driver that is installed on your Windows machine, you can find the version number in the ODBC Data Source Administrator.

To verify the driver version number on Windows:

1. Open the ODBC Administrator:
 - If you are using Windows 7 or earlier, click **Start**  **> All Programs > Simba MongoDB ODBC Driver 2.3 > 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.

 **Note:**

Make sure to select the ODBC Data Source Administrator that has the same bitness as the client application that you are using to connect to MongoDB.

2. Click the **Drivers** tab and then find the Simba MongoDB ODBC Driver in the list of ODBC drivers that are installed on your system. The version number is displayed in the **Version** column.

macOS Driver

macOS System Requirements

The Simba MongoDB ODBC Driver supports MongoDB versions 3.0, 3.2, and 3.4.

Install the driver on client machines where the application is installed. Each machine that you install the driver on must meet the following minimum system requirements:

- macOS version 10.9, 10.10, or 10.11
- 500 MB of available disk space
- iODBC 3.52.7 or later
- Before you can use the Kerberos authentication mechanism, you must install Kerberos.

For information about how to install and configure Kerberos, see the MIT Kerberos Documentation: <http://web.mit.edu/kerberos/krb5-latest/doc/>.

Installing the Driver on macOS

The Simba MongoDB ODBC Driver is available for macOS as a .dmg file named `SimbaMongoDBODBC.dmg`. The driver supports both 32- and 64-bit client applications.

To install the Simba MongoDB ODBC Driver on macOS:

1. Double-click **SimbaMongoDBODBC.dmg** to mount the disk image.
2. Double-click **SimbaMongoDBODBC.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**.

 **Note:**

By default, the driver files are installed in the `/Library/simba/mongodb` directory.

6. To accept the installation location and begin the installation, click **Install**.
7. When the installation completes, click **Close**.

8. If you received a license file through email, then copy the license file into the `/lib` subfolder in the driver installation directory. You must have root privileges when changing the contents of this folder.

For example, if you installed the driver to the default location, you would copy the license file into the `/Library/simba/mongodb/lib` folder.

Next, configure the environment variables on your machine to make sure that the ODBC driver manager can work with the driver. For more information, see [Configuring the ODBC Driver Manager on Non-Windows Machines](#) on page 31.

Verifying the Driver Version Number on macOS

If you need to verify the version of the Simba MongoDB ODBC Driver that is installed on your macOS machine, you can query the version number through the Terminal.

To verify the driver version number on macOS:

- At the Terminal, run the following command:

```
pkgutil --info com.simba.mongodbodbc
```

The command returns information about the Simba MongoDB ODBC Driver that is installed on your machine, including the version number.

Linux Driver

The Linux driver is available as an RPM file and as a tarball package.

Linux System Requirements

The Simba MongoDB ODBC Driver supports MongoDB versions 3.0, 3.2, and 3.4.

Install the driver on client machines where the application is installed. 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) 6 or 7
 - CentOS 6 or 7
 - SUSE Linux Enterprise Server (SLES) 11 or 12
 - Debian 7 or 8
 - Ubuntu 14.04 or 16.04
- 800 MB of available disk space
- One of the following ODBC driver managers installed:
 - iODBC 3.52.7 or later
 - unixODBC 2.3.0 or later
- All of the following `libsasl` libraries installed:
 - `cyrus-sasl-2.1.22-7` or later
 - `cyrus-sasl-gssapi-2.1.22-7` or later
 - `cyrus-sasl-plain-2.1.22-7` or later

 **Note:**

If the package manager in your Linux distribution cannot resolve the dependencies automatically when installing the driver, then download and manually install the packages.

- Before you can use the Kerberos authentication mechanism, you must install Kerberos.
For information about how to install and configure Kerberos, see the MIT Kerberos Documentation: <http://web.mit.edu/kerberos/krb5-latest/doc/>.

To install the driver, you must have root access on the machine.

Installing the Driver Using the RPM File

On 64-bit editions of Linux, 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 to install and use the version of the driver that matches the bitness of the client application:

- `simbamongodb-[Version]-[Release].[LinuxDistro].i686.rpm` for the 32-bit driver
- `simbamongodb-[Version]-[Release].[LinuxDistro].x86_64.rpm` for the 64-bit driver

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

The placeholders in the file names are defined as follows:

- `[Version]` is the version number of the driver.
- `[Release]` is the release number for this version of the driver.
- `[LinuxDistro]` is one of the following values indicating which Linux distribution the installer is optimized for:

Value	Platform for Installation
e15	CentOS 5 or 6 RHEL 5 or 6
e17	CentOS 7 RHEL 7
suse11	SLES 11
suse12	SLES 12

Make sure to install the driver using the RPM that is optimized for your Linux distribution. Otherwise, you may encounter errors when using the driver.

To install the Simba MongoDB ODBC Driver using the RPM File:

1. Log in as the root user, and then navigate to the folder containing the RPM package for the driver.
2. Depending on the Linux distribution that you are using, run one of the following commands from the command line, where `[RPMFileName]` is the file name of the RPM package:


- If you are using Red Hat Enterprise Linux or CentOS, run the following command:

```
yum --nogpgcheck localinstall [RPMFileName]
```

- Or, if you are using SUSE Linux Enterprise Server, run the following command:

```
zypper install [RPMFileName]
```

The Simba MongoDB ODBC Driver files are installed in the `/opt/simba/mongodb` directory.

 **Note:**

If the package manager in your Linux distribution cannot resolve the `libsasl` dependencies automatically when installing the driver, then download and manually install the packages.

3. If you received a license file through email, then copy the license file into the `/opt/simba/mongodb/lib/32` or `/opt/simba/mongodb/lib/64` folder, depending on the version of the driver that you installed. You must have root privileges when changing the contents of this folder.

Next, configure the environment variables on your machine to make sure that the ODBC driver manager can work with the driver. For more information, see [Configuring the ODBC Driver Manager on Non-Windows Machines](#) on page 31.

Installing the Driver Using the Tarball Package

The Simba MongoDB ODBC Driver is available as a tarball package named `SimbaMongoDBODBC-[Version].[Release]-Linux.tar.gz`, where `[Version]` is the version number of the driver and `[Release]` is the release number for this version of the driver. The package contains both the 32-bit and 64-bit versions of the driver.

On 64-bit editions of Linux, 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 that matches the bitness of the client application. You can install both versions of the driver on the same machine.

To install the Simba MongoDB ODBC Driver using the tarball package:

1. Log in as the root user, and then navigate to the folder containing the tarball package.
2. Run the following command to extract the package and install the driver:

```
tar --directory=/opt -zxvf [TarballName]
```

Where *[TarballName]* is the name of the tarball package containing the driver.

The Simba MongoDB ODBC Driver files are installed in the `opt/simba/mongodb` directory.

3. If you received a license file through email, then copy the license file into the `opt/simba/mongodb/lib/32` or `opt/simba/mongodb/lib/64` folder, depending on the version of the driver that you installed. You must have root privileges when changing the contents of this folder.

Next, configure the environment variables on your machine to make sure that the ODBC driver manager can work with the driver. For more information, see [Configuring the ODBC Driver Manager on Non-Windows Machines](#) on page 31.

Verifying the Driver Version Number on Linux

If you need to verify the version of the Simba MongoDB ODBC Driver that is installed on your Linux machine, you can query the version number through the command-line interface if the driver was installed using an RPM file.

To verify the driver version number on Linux:

- Depending on your package manager, at the command prompt, run one of the following commands:
 - `yum list | grep SimbaMongoDBODBC`
 - `rpm -qa | grep SimbaMongoDBODBC`

The command returns information about the Simba MongoDB ODBC Driver that is installed on your machine, including the version number.

AIX Driver

AIX System Requirements

The Simba MongoDB ODBC Driver supports MongoDB versions 3.0, 3.2, and 3.4.

Install the driver on client machines where the application is installed. Each machine that you install the driver on must meet the following minimum system requirements:

- IBM AIX 7.1 (64-bit only)
- GCC 4.9 or later
- 150 MB of available disk space
- One of the following ODBC driver managers installed:
 - iODBC 3.52.7 or later
 - unixODBC 2.3.0 or later
- Before you can use the Kerberos authentication mechanism, you must install Kerberos.
For information about how to install and configure Kerberos, see the MIT Kerberos Documentation: <http://web.mit.edu/kerberos/krb5-latest/doc/>.

To install the driver, you must have root access on the machine.

Note:

The Schema Editor application is not supported on AIX, so you cannot create and save new schema definitions. When connecting to a database, the AIX driver must use a temporary schema definition that is generated during connection time or a persistent schema definition that was created on another platform.

Installing the Driver on AIX

The Simba MongoDB ODBC Driver is available for 64-bit editions of AIX as a tarball package named `SimbaMongoDBODBC-[Version].[Release]-AIX.tar.gz`, where `[Version]` is the version number of the driver, and `[Release]` is the release number for this version of the driver. The AIX driver does not support 32-bit client applications.

To install the Simba MongoDB ODBC Driver on AIX:

1. Log in as the root user, and then navigate to the folder containing the tarball package.

2. Run the following command to extract the package and install the driver:

```
tar --directory=/opt -zxvf [TarballName]
```

Where *[TarballName]* is the name of the tarball package containing the driver.

The Simba MongoDB ODBC Driver files are installed in the `opt/simba/mongodb` directory.

3. If you received a license file through email, then copy the license file into the `opt/simba/mongodb/lib/64` folder. You must have root privileges when changing the contents of this folder.

Next, configure the environment variables on your machine to make sure that the ODBC driver manager can work with the driver. For more information, see [Configuring the ODBC Driver Manager on Non-Windows Machines](#) on page 31.

Configuring the ODBC Driver Manager on Non-Windows Machines

To make sure that the ODBC driver manager on your machine is configured to work with the Simba MongoDB ODBC Driver, do the following:

- Set the library path environment variable to make sure that your machine uses the correct ODBC driver manager. For more information, see [Specifying ODBC Driver Managers on Non-Windows Machines](#) on page 31.
- If the driver configuration files are not stored in the default locations expected by the ODBC driver manager, then set environment variables to make sure that the driver manager locates and uses those files. For more information, see [Specifying the Locations of the Driver Configuration Files](#) on page 32.

After configuring the ODBC driver manager, you can configure a connection and access your data store through the driver. For more information, see [Configuring ODBC Connections on a Non-Windows Machine](#) on page 34.

Specifying ODBC Driver Managers on Non-Windows Machines

You need to make sure that your machine uses the correct ODBC driver manager to load the driver. To do this, set the library path environment variable.

macOS

If you are using a macOS machine, then set the `DYLD_LIBRARY_PATH` environment variable to include the paths to the ODBC driver manager libraries. For example, if the libraries are installed in `/usr/local/lib`, then run the following command to set `DYLD_LIBRARY_PATH` for the current user session:

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

For information about setting an environment variable permanently, refer to the macOS shell documentation.

Linux

If you are using a Linux machine, then set the `LD_LIBRARY_PATH` environment variable to include the paths to the ODBC driver manager libraries. For example, if the libraries are installed in `/usr/local/lib`, then run the following command to set `LD_LIBRARY_PATH` for the current user session:

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

For information about setting an environment variable permanently, refer to the Linux shell documentation.

AIX

If you are using an AIX machine, then set the LIBPATH environment variable to include the paths to the ODBC driver manager libraries. For example, if the libraries are installed in `/usr/local/lib`, then run the following command to set LIBPATH for the current user session:

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

For information about setting an environment variable permanently, refer to the AIX shell documentation.

Specifying the Locations of the Driver Configuration Files

By default, ODBC driver managers are configured to use hidden versions of the `odbc.ini` and `odbcinst.ini` configuration files (named `.odbc.ini` and `.odbcinst.ini`) located in the home directory, as well as the `simba.mongodbodbc.ini` file in the `lib` subfolder of the driver installation directory. If you store these configuration files elsewhere, then you must set the environment variables described below so that the driver manager can locate the files.

If you are using iODBC, do the following:

- Set ODBCINI to the full path and file name of the `odbc.ini` file.
- Set ODBCINSTINI to the full path and file name of the `odbcinst.ini` file.
- Set SIMBA_MONGODB_ODBC_INI to the full path and file name of the `simba.mongodbodbc.ini` file.

If you are using unixODBC, do the following:

- Set ODBCINI to the full path and file name of the `odbc.ini` file.
- Set ODBCYSINI to the full path of the directory that contains the `odbcinst.ini` file.
- Set SIMBA_MONGODB_ODBC_INI to the full path and file name of the `simba.mongodbodbc.ini` file.

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

For iODBC:

```
export ODBCINI=/usr/local/odbc/odbc.ini
export ODBCINSTINI=/usr/local/odbc/odbcinst.ini
export SIMBA_MONGODB_ODBC_INI=/etc/simba.mongodbodbc.ini
```

For unixODBC:

```
export ODBCINI=/usr/local/odbc/odbc.ini
export ODBCYSINI=/usr/local/odbc
export SIMBA_MONGODB_ODBC_INI=/etc/simba.mongodbodbc.ini
```

To locate the `simba.mongodbodbc.ini` file, the driver uses the following search order:

1. If the `SIMBA_MONGODB_ODBC_INI` environment variable is defined, then the driver searches for the file specified by the environment variable.
2. The driver searches the directory that contains the driver library files for a file named `simba.mongodbodbc.ini`.
3. The driver searches the current working directory of the application for a file named `simba.mongodbodbc.ini`.
4. The driver searches the home directory for a hidden file named `.simba.mongodbodbc.ini` (prefixed with a period).
5. The driver searches the `/etc` directory for a file named `simba.mongodbodbc.ini`.

Configuring ODBC Connections on a Non-Windows Machine

The following sections describe how to configure ODBC connections when using the Simba MongoDB ODBC Driver on non-Windows platforms:

- [Creating a Data Source Name on a Non-Windows Machine](#) on page 34
- [Configuring a DSN-less Connection on a Non-Windows Machine](#) on page 37
- [Configuring Authentication on a Non-Windows Machine](#) on page 40
- [Configuring SSL Verification on a Non-Windows Machine](#) on page 41
- [Configuring Logging Options on a Non-Windows Machine](#) on page 42
- [Testing the Connection on a Non-Windows Machine](#) on page 44

Creating a Data Source Name on a Non-Windows Machine

When connecting to your data store using a DSN, you only need to configure the `odbc.ini` file. Set the properties in the `odbc.ini` file to create a DSN that specifies the connection information for your data store. For information about configuring a DSN-less connection instead, see .

If your machine is already configured to use an existing `odbc.ini` file, then update that file by adding the settings described below. Otherwise, copy the `odbc.ini` file from the `Setup` subfolder in the driver installation directory to the home directory, and then update the file as described below.

To create a Data Source Name on a non-Windows machine:

1. In a text editor, open the `odbc.ini` configuration file.

 **Note:**

If you are using a hidden copy of the `odbc.ini` file, you can remove the period (.) from the start of the file name to make the file visible while you are editing it.

2. In the `[ODBC Data Sources]` section, add a new entry by typing a name for the DSN, an equal sign (=), and then the name of the driver.

For example, on a macOS machine:

```
[ODBC Data Sources]
```

```
Sample DSN=Simba MongoDB ODBC Driver
```

As another example, for a 64-bit driver on a Linux or AIX machine:

```
[ODBC Data Sources]
Sample DSN=Simba MongoDB ODBC Driver 64-bit
```

3. Create a section that has the same name as your DSN, and then specify configuration options as key-value pairs in the section:
 - a. Set the `Driver` property to the full path of the driver library file that matches the bitness of the application.

For example, on a macOS machine:

```
Driver=/Library/simba/mongodb/lib/libmongodbc_
sbu.dylib
```

As another example, for a 64-bit driver on a Linux or AIX machine:

```
Driver=/opt/simba/mongodb/lib/64/libmongodbc_
sb64.so
```

- b. Set the `Server` property to the IP address or host name of the server, and then set the `Port` property to the number of the TCP port that the server uses to listen for client connections.

For example:

```
Server=192.168.222.160
Port=27017
```

- c. Set the `Database` property to the name of the database that you want to access.

For example:

```
Database=TestData
```

- d. If authentication is required to access the server, then specify the authentication mechanism and your credentials. For more information, see [Configuring Authentication on a Non-Windows Machine](#) on page 40.
 - e. If you want to connect to the server through SSL, then enable SSL and specify the certificate information. For more information, see [Configuring SSL Verification on a Non-Windows Machine](#) on page 41.
 - f. If you are using a specific schema definition for this connection, do one of the following:

- To use a schema definition that is stored in the database, set the `LoadMetadataTable` property to 1.
- Or, to use a schema definition that is stored in a JSON file, set the `LoadMetadataTable` property to 0 and the `LocalMetadataFile` property to the full path and name of the JSON file.

For example, if your schema definition is stored in a JSON file:


```
LoadMetadataTable=0
LocalMetadataFile=/localhome/simba/schemas/mongodb_
schema.json
```

 **Note:**

- If you connect to the data store without specifying a schema definition, and the data store does not already contain a valid schema definition, then the driver automatically generates a temporary schema definition in order to support the connection.
- For information about how to create a schema definition using the Schema Editor application, see the *Schema Editor User Guide* located in the installation directory of the driver. Note that the Schema Editor is not supported on AIX.

- g. Optionally, set additional key-value pairs as needed to specify other optional connection settings. For detailed information about all the configuration options supported by the Simba MongoDB ODBC Driver, see [Driver Configuration Options](#) on page 64.

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

 **Note:**

If you are storing this file in its default location in the home directory, then prefix the file name with a period (.) so that the file becomes hidden. If you are storing this file in another location, then save it as a non-hidden file (without the prefix), and make sure that the `ODBCINI` environment variable specifies the location. For more information, see [Specifying the Locations of the Driver Configuration Files](#) on page 32.

For example, the following is an `odbc.ini` configuration file for macOS containing a DSN that connects to MongoDB without authentication and without a specific schema definition:

```
[ODBC Data Sources]
Sample DSN=Simba MongoDB ODBC Driver
[Sample DSN]
```

```
Driver=/Library/simba/mongodb/lib/libmongodbc_sbu.dylib
Server=192.168.222.160
Port=27017
Database=TestData
```

As another example, the following is an `odbc.ini` configuration file for a 64-bit driver on a Linux or AIX machine, containing a DSN that connects to MongoDB without authentication and without a specific schema definition:

```
[ODBC Data Sources]
Sample DSN=Simba MongoDB ODBC Driver 64-bit
[Sample DSN]
Driver=/opt/simba/mongodb/lib/64/libmongodbc_sb64.so
Server=192.168.222.160
Port=27017
Database=TestData
```

You can now use the DSN in an application to connect to the data store.

Configuring a DSN-less Connection on a Non-Windows Machine

To connect to your data store through a DSN-less connection, you need to define the driver in the `odbcinst.ini` file and then provide a DSN-less connection string in your application.

If your machine is already configured to use an existing `odbcinst.ini` file, then update that file by adding the settings described below. Otherwise, copy the `odbcinst.ini` file from the `Setup` subfolder in the driver installation directory to the home directory, and then update the file as described below.

To define a driver on a non-Windows machine:

1. In a text editor, open the `odbcinst.ini` configuration file.

 **Note:**

If you are using a hidden copy of the `odbcinst.ini` file, you can remove the period (.) from the start of the file name to make the file visible while you are editing it.

2. In the `[ODBC Drivers]` section, add a new entry by typing a name for the driver, an equal sign (=), and then `Installed`.

For example:

```
[ODBC Drivers]
Simba MongoDB ODBC Driver=Installed
```

3. Create a section that has the same name as the driver (as specified in the previous step), and then specify the following configuration options as key-value pairs in the section:

- a. Set the `Driver` property to the full path of the driver library file that matches the bitness of the application.

For example, on a macOS machine:

```
Driver=/Library/simba/mongodb/lib/libmongodbc_
sbu.dylib
```

As another example, for a 64-bit driver on a Linux or AIX machine:

```
Driver=/opt/simba/mongodb/lib/64/libmongodbc_
sb64.so
```

- b. Optionally, set the `Description` property to a description of the driver.

For example:

```
Description=Simba MongoDB ODBC Driver
```

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

 **Note:**

If you are storing this file in its default location in the home directory, then prefix the file name with a period (.) so that the file becomes hidden. If you are storing this file in another location, then save it as a non-hidden file (without the prefix), and make sure that the `ODBCINSTINI` or `ODBCSYSINI` environment variable specifies the location. For more information, see [Specifying the Locations of the Driver Configuration Files](#) on page 32.

For example, the following is an `odbcinst.ini` configuration file for macOS:

```
[ODBC Drivers]
Simba MongoDB ODBC Driver=Installed
[Simba MongoDB ODBC Driver]
Description=Simba MongoDB ODBC Driver
Driver=/Library/simba/mongodb/lib/libmongodbc_
sbu.dylib
```

As another example, the following is an `odbcinst.ini` configuration file for both the 64-bit driver on Linux or AIX:

```
[ODBC Drivers]
Simba MongoDB ODBC Driver 64-bit=Installed
[Simba MongoDB ODBC Driver 64-bit]
Description=Simba MongoDB ODBC Driver (64-bit)
Driver=/opt/simba/mongodb/lib/64/libmongodbc_sb64.so
```

You can now connect to your data store by providing your application with a connection string where the `Driver` property is set to the driver name specified in the `odbcinst.ini` file, and all the other necessary connection properties are also set. For more information, see "DSN-less Connection String Examples" in [Using a Connection String](#) on page 46.

For instructions about configuring specific connection features, see the following:

- [Configuring Authentication on a Non-Windows Machine](#) on page 40
- [Configuring SSL Verification on a Non-Windows Machine](#) on page 41

For detailed information about all the connection properties that the driver supports, see [Driver Configuration Options](#) on page 64.

 **Note:**

- If you connect to the data store without specifying a schema definition, and the data store does not already contain a valid schema definition, then the driver automatically generates a temporary schema definition in order to support the connection. For information about specifying a schema definition, see [Mechanism \(Metadata\)](#) on page 77 and [Local File](#) on page 74.
- For information about how to create a schema definition using the Schema Editor application, see the *Schema Editor User Guide* located in the installation directory of the driver. Note that the Schema Editor is not supported on AIX.

Configuring Authentication on a Non-Windows Machine

Some MongoDB databases require authentication. You can configure the Simba MongoDB ODBC Driver to authenticate your connection to provide your credentials and authenticate the connection to the database using one of the following methods:

- [Using SCRAM-SHA-1](#) on page 40
- [Using Kerberos](#) on page 40
- [Using LDAP](#) on page 41

You can set the connection properties described below in a connection string or in a DSN (in the `odbc.ini` file). Settings in the connection string take precedence over settings in the DSN.

Using SCRAM-SHA-1

You can configure the driver to use the SCRAM-SHA-1 protocol to authenticate the connection. SCRAM-SHA-1 is the default authentication protocol used by MongoDB.

To configure SCRAM-SHA-1 authentication on a non-Windows machine:

1. Set the `AuthMechanism` property to `SCRAM-SHA-1`.
2. To use a database other than the admin database to check your credentials, set the `AuthSource` property to the name of the database.
3. Set the `UID` property to an appropriate user name for accessing the MongoDB database.
4. Set the `PWD` property to the password corresponding to the user name you typed above.

Using Kerberos

You can configure the driver to use the Kerberos protocol to authenticate the connection.

Kerberos must be installed and configured before you can use this authentication mechanism. For information about how to install and configure Kerberos, see the MIT Kerberos Documentation: <http://web.mit.edu/kerberos/krb5-latest/doc/>.

To configure Kerberos authentication on a non-Windows machine:

1. Set the `AuthMechanism` property to `GSSAPI`.
2. Set the `gssapiServiceName` property to the service name of the MongoDB server.

Using LDAP

You can configure the driver to use the LDAP protocol to authenticate the connection.

To configure LDAP authentication on a non-Windows machine:

1. Set the `AuthMechanism` property to `PLAIN`.
2. Set the `UID` property to an appropriate user name for accessing the MongoDB database.
3. Set the `PWD` property to the password corresponding to the user name you typed above.

Configuring SSL Verification on a Non-Windows Machine

If you are connecting to a MongoDB server that has Secure Sockets Layer (SSL) enabled, then you can configure the driver to connect to an SSL-enabled socket. When connecting to a server over SSL, the driver supports identity verification between the client and the server.

You can set the connection properties described below in a connection string or in a DSN (in the `odbc.ini` file). Settings in the connection string take precedence over settings in the DSN.

Configuring an SSL Connection without Identity Verification

You can configure a connection that uses SSL but does not verify the identity of the client or the server.

To configure an SSL connection without verification on a non-Windows machine:

1. Set the `SSL` property to `1`.
2. Set the `sslAllowInvalidCertificates` property to `1`.

Configuring One-way SSL Verification

You can configure one-way verification so that the client verifies the identity of the MongoDB server.

To configure one-way SSL verification on a non-Windows machine:

1. Set the `SSL` property to `1`.
2. Choose one:

- To verify the server using a certificate from a specific `.pem` file, set the `sslCAFile` property to the full path of the PEM file.
 - Or, to verify the server using certificates stored in multiple `.pem` files, set the `sslCADir` property to the full path of the directory where the PEM files are located.
3. Set the `sslCRLFile` to the full path of the `.pem` file containing the list of revoked certificates.

Configuring Two-way SSL Verification

You can configure two-way SSL verification so that the client and the MongoDB server verify each other.

To configure two-way SSL verification on a non-Windows machine:

1. Set the `SSL` property to `1`.
2. Set the `sslPEMKeyFile` property to the full path of the `.pem` file containing the certificate for verifying the client.
3. If the client certificate is protected with a password, set the `sslPEMKeyPwd` property to the password.
4. Choose one:
 - To verify the server using a certificate from a specific `.pem` file, set the `sslCAFile` property to the full path of the PEM file.
 - Or, to verify the server using certificates stored in multiple `.pem` files, set the `sslCADir` property to the full path of the directory where the `.pem` files are located.
5. Set the `sslCRLFile` to the full path of the `.pem` file containing the list of revoked certificates.

Configuring Logging Options on a Non-Windows Machine

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.

Logging is configured through driver-wide settings in the `simba.mongodbodbc.ini` file, which apply to all connections that use the driver.

To enable logging on a non-Windows machine:

1. Open the `simba.mongodboDBC.ini` configuration file in a text editor.
2. To specify the level of information to include in log files, set the `LogLevel` property to one of the following numbers:


LogLevel Value	Description
0	Disables all logging.
1	Logs severe error events that lead the driver to abort.
2	Logs error events that might allow the driver to continue running.
3	Logs events that might result in an error if action is not taken.
4	Logs general information that describes the progress of the driver.
5	Logs detailed information that is useful for debugging the driver.
6	Logs all driver activity.

3. Set the `LogPath` key to the full path to the folder where you want to save log files.
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 `simba.mongodboDBC.ini` configuration file.
7. Restart your ODBC application to make sure that the new settings take effect.

The Simba MongoDB ODBC Driver produces a log file named `simbamongodboDBC_driver.log` at the location you specify using the `LogPath` key.

If you set the `UseLogPrefix` property to 1, then each file name is prefixed with `[UserName]_[ProcessID]_`, where `[UserName]` is the user name associated with the connection and `[ProcessID]` is the process ID of the application through which the connection is made. For more information, see [UseLogPrefix](#) on page 83.

To disable logging on a non-Windows machine:

1. Open the `simba.mongodboDBC.ini` configuration file in a text editor.
2. Set the `LogLevel` key to 0.
3. Save the `simba.mongodboDBC.ini` configuration file.
4. Restart your ODBC application to make sure that the new settings take effect.

Testing the Connection on a Non-Windows Machine

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 make sure 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 the connection string for connecting to your data store, and then press ENTER. For more information, see [Using a Connection String](#) on page 46.

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

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 make sure 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 `isql` or `iusql` by using the corresponding 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.

Using a Connection String

For some applications, you might 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 sections 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 might 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 64.

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.

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.

! Important:

When you connect to the data store using a DSN-less connection string, the driver does not encrypt your credentials.

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

- *[MongoDBDatabase]* is the database that you want to access.
- *[PortNumber]* is the number of the TCP port that the MongoDB server uses to listen for client connections.

- *[ServerInfo]* is the IP address or host name of the MongoDB server to which you are connecting.
- *[ServiceName]* is the Kerberos service principal name of the MongoDB server.
- *[SetName]* is the name of the replica set that you want to access.
- *[YourPassword]* is the password corresponding to your user name.
- *[YourUserName]* is the user name that you use to access the MongoDB server.

Connecting to a Standard MongoDB Server Without Authentication

The following is the format of a DSN-less connection string for a standard connection to a MongoDB server that does not require authentication:

```
Driver=Simba MongoDB ODBC Driver;Server=[ServerInfo];  
Port=[PortNumber];Database=[MongoDBDatabase];
```

For example:

```
Driver=Simba MongoDB ODBC Driver;Server=192.168.222.160;  
Port=27017;Database=TestData;
```

Connecting to a Standard MongoDB Server that Requires SCRAM-SHA-1 Authentication

The following is the format of a DSN-less connection string for a standard connection to a MongoDB server that requires SCRAM-SHA-1 authentication:

```
Driver=Simba MongoDB ODBC Driver;Server=[ServerInfo];  
Port=[PortNumber];Database=[MongoDBDatabase];  
authMechanism=SCRAM-SHA-1;UID=[YourUserName];  
PWD=[YourPassword];
```

For example:

```
Driver=Simba MongoDB ODBC Driver;Server=192.168.222.160;  
Port=27017;Database=TestData;authMechanism=SCRAM-SHA-1;  
UID=simba;PWD=simba;
```

Connecting to a Standard MongoDB Server that Requires Kerberos Authentication

The following is the format of a DSN-less connection string for a standard connection to a MongoDB server that requires Kerberos authentication:

```
Driver=Simba MongoDB ODBC Driver;Server=[ServerInfo];  
Port=[PortNumber];Database=[MongoDBDatabase];  
authMechanism=GSSAPI;gssapiServiceName=[ServiceName];
```

For example:

```
Driver=Simba MongoDB ODBC Driver;Server=192.168.222.160;  
Port=27017;Database=TestData;authMechanism=GSSAPI;  
gssapiServiceName=mongodb;
```

Connecting to a Standard MongoDB Server that Requires LDAP Authentication

The following is the format of a DSN-less connection string for a standard connection to a MongoDB server that requires LDAP authentication:

```
Driver=Simba MongoDB ODBC Driver;Server=[ServerInfo];  
Port=[PortNumber];Database=[MongoDBDatabase];  
authMechanism=PLAIN;UID=[YourUserName];PWD=[YourPassword];
```

For example:

```
Driver=Simba MongoDB ODBC Driver;Server=192.168.222.160;  
Port=27017;Database=TestData;authMechanism=PLAIN;  
UID=simba;PWD=simba;
```

Connecting to a Replica Set Without Authentication

The following is the format of a DSN-less connection string for connecting to a replica set that does not require authentication:

```
Driver=Simba MongoDB ODBC Driver;Server=[ServerInfo];  
Port=[PortNumber];Database=[MongoDBDatabase];  
EnableReplicaSet=1;SecondaryServers=[Server1]:[PortNumber1],  
[Server2]:[PortNumber2];ReplicaSet=[SetName];
```

For example:

```
Driver=Simba MongoDB ODBC Driver;Server=192.168.222.160;  
Port=27017;Database=TestData;EnableReplicaSet=1;  
SecondaryServers=192.168.222.165:27017,  
192.168.222.231:27017;ReplicaSet=TestSet;
```


Connecting to a Replica Set that Requires MongoDB User Name and Password Authentication

The following is the format of a DSN-less connection string for connecting to a replica set that requires MongoDB User Name and Password authentication:

```
Driver=Simba MongoDB ODBC Driver;Server=[ServerInfo];
Port=[PortNumber];Database=[MongoDBDatabase];
UID=[YourUserName];PWD=[YourPassword];EnableReplicaSet=1;
SecondaryServers=[Server1]:[PortNumber1], [Server2]:
[PortNumber2];ReplicaSet=[SetName];
```

For example:

```
Driver=Simba MongoDB ODBC Driver;Server=192.168.222.160;
Port=27017;Database=TestData;UID=simba;PWD=simba;
EnableReplicaSet=1;SecondaryServers=192.168.222.165:27017,
192.168.222.231:27017;ReplicaSet=TestSet;
```

Connecting to a Replica Set that Requires Kerberos Authentication

The following is the format of a DSN-less connection string for connecting to a replica set that requires Kerberos authentication:

```
Driver=Simba MongoDB ODBC Driver;Server=[ServerInfo];
Port=[PortNumber];Database=[MongoDBDatabase];
authMechanism=GSSAPI;gssapiServiceName=[ServiceName];
EnableReplicaSet=1;SecondaryServers=[Server1]:[PortNumber1],
[Server2]:[PortNumber2];ReplicaSet=[SetName];
```

For example:

```
Driver=Simba MongoDB ODBC Driver;Server=192.168.222.160;
Port=27017;Database=TestData;authMechanism=GSSAPI;
gssapiServiceName=ServiceName;EnableReplicaSet=1;
SecondaryServers=192.168.222.165:27017,
192.168.222.231:27017;ReplicaSet=TestSet;
```

Connecting to a Replica Set that Requires LDAP Authentication

The following is the format of a DSN-less connection string for connecting to a replica set that requires LDAP authentication:

```
Driver=Simba MongoDB ODBC Driver;Server=[ServerInfo];
Port=[PortNumber];Database=[MongoDBDatabase];
```

```
authMechanism=PLAIN;UID=[YourUserName];PWD=[YourPassword];  
EnableReplicaSet=1;SecondaryServers=[Server1]:[PortNumber1],  
[Server2]:[PortNumber2];ReplicaSet=[SetName];
```

For example:

```
Driver=Simba MongoDB ODBC Driver;Server=192.168.222.160;  
Port=27017;Database=TestData;authMechanism=PLAIN;  
UID=simba;PWD=simba;EnableReplicaSet=1;  
SecondaryServers=192.168.222.165:27017,  
192.168.222.231:27017;ReplicaSet=TestSet;
```

Features

For more information on the features of the Simba MongoDB ODBC Driver, see the following:

- [Catalog Support](#) on page 51
- [Double-Buffering](#) on page 51
- [SQL Connector](#) on page 52
- [Data Types](#) on page 52
- [Schema Definitions](#) on page 56
- [Virtual Tables](#) on page 58
- [Write-back](#) on page 61
- [Security and Authentication](#) on page 63

Catalog Support

The Simba MongoDB ODBC Driver supports catalogs by using the name of the MongoDB database as the catalog. Doing so allows the driver to work easily with various ODBC applications.

The driver supports queries against MongoDB databases outside of the one that you are connected to, as well as joins between tables that belong to different databases. To work with data from other databases, in your SQL statements, specify the database (or catalog) that each table belongs to using the syntax *DatabaseName.TableName*.

Double-Buffering

The Simba MongoDB ODBC Driver is capable of using double-buffering to improve driver performance during SELECT operations.

The impact of double-buffering depends on how the transfer speed of your network compares to the data processing speed of the driver. If the transfer speed is significantly higher, then enabling double-buffering allows the driver to make full use of the network's capabilities. Conversely, if the transfer speed is considerably lower, the additional processes involved in double-buffering might cause a decrease in performance.

To make optimal use of double-buffering, you need to set an appropriate buffer size. A buffer size that is too small might decrease performance, while a buffer size that is too large might diminish the performance improvements from double-buffering. If the

transfer speed of the network is slow, the additional time spend processing a large buffer size might cause a decrease in performance.

For information about configuring double-buffering, see [Enable Double-Buffering](#) on page 69.

SQL Connector

The SQL Connector feature of the driver allows applications to execute standard SQL queries against MongoDB. It converts SQL-92 queries to MongoDB API calls and processes the results.

Data Types

The Simba MongoDB ODBC Driver supports many MongoDB data types, and converts data between MongoDB and SQL data types as needed. The mapping between each data type is determined by the schema definition, which you can create by using the Schema Editor application that is installed with the driver.



Note:

The Schema Editor is not supported on AIX. On all other platforms, information about how to use the Schema Editor can be found in the *Schema Editor User Guide* located in the installation directory of the driver.

- On Windows 7 or earlier, the guide is available from the **Simba MongoDB ODBC Driver** program group in the Start menu.
- On Windows 8 or later, you can search for the guide on the Start screen.

The following table lists the supported data type mappings.

To support complex data types such as objects and arrays, the driver renormalizes the data into virtual tables. For more information, see [Virtual Tables](#) on page 58.

MongoDB Type	SQL Type
Binary	SQL_BIGINT SQL_BIT
Boolean	SQL_DOUBLE SQL_INTEGER SQL_VARCHAR

MongoDB Type	SQL Type
Date	SQL_TIMESTAMP
DBPointer	SQL_VARCHAR
Decimal	SQL_DECIMAL
JavaScript	SQL_VARCHAR
JavaScript (with scope)	SQL_VARCHAR <div style="border: 1px solid black; padding: 5px; margin-top: 10px;"> <p>! Important:</p> <p>MongoDB cannot represent JavaScript (with scope) data as a string. The driver converts the data to type SQL_VARCHAR, but returns the following string as the value: <code>Unsupported JavaScript with Scope.</code></p> </div>
MaxKey	SQL_VARCHAR
MinKey	SQL_VARCHAR
NumberDouble	SQL_BIGINT
NumberInt	SQL_DOUBLE
	SQL_INTEGER
NumberLong	SQL_VARCHAR
ObjectID	SQL_VARCHAR
Regular Expression	SQL_VARCHAR
String	SQL_VARCHAR
Symbol	SQL_VARCHAR
Timestamp	SQL_VARCHAR
Undefined	SQL_VARCHAR

MongoDB Type	SQL Type
UUID	SQL_BINARY
Array	N/A The data is renormalized into a virtual table.
Object	N/A The data is renormalized into a virtual table.

Different rows in the sampled data might have the same field assigned to different data types. The driver resolves this mixed data typing by specifying a single type for the field in the schema definition. The specified type is the first data type from this list that appears in the sampled data.

- Array**

Fields that have Object as one of the data types are also returned as Array data.

- Binary**

- String**

Fields that have any of the following types as one of the data types, but not Array or Binary, are also returned as String data:

- Date
- Timestamp
- DBPointer
- JavaScript
- JavaScript (with scope)
- Symbol
- Regular Expression
- MaxKey
- MinKey
- Undefined
- OID

- NumberDouble**

- NumberLong**

- NumberInt**

For example, the driver treats the following field named `f` as MongoDB data of type Array:

```
{f : {g1 : 1}}
{f : [1, 2, 3]}
```

Data types that do not have a direct mapping from MongoDB to ODBC are represented as type VARCHAR in ODBC. The detected MongoDB type is used during INSERT and UPDATE operations.

Embedded Documents

The driver renormalizes embedded documents into columns. For example, consider the following JSON document:

```
{"contact": {"address": {"street": "1-123 Broadway", "city": "Vancouver"}}, "phone": "+12345678"}
```

When generating the schema definition, the driver identifies the following columns in the document, all of type String:

contact_address_street	contact_address_city	phone
1-123 Broadway	Vancouver	+12345678

The driver is able to work with these columns as if they were standard table columns.

Arrays

The ODBC interface does not natively support collection-based data types, so the Simba MongoDB ODBC Driver implements two options for accessing and interacting with collection-based data. Depending on preference, arrays in MongoDB can be renormalized into virtual tables or columns.

By default, arrays are renormalized into virtual tables. To view array elements as columns instead, use the **Move to Parent** option in the Schema Editor. You can create the column in a parent virtual table or in the base table.

Note:

The Schema Editor is not supported on AIX. On all other platforms, information about how to use the Schema Editor can be found in the *Schema Editor User Guide* located in the installation directory of the driver.

- On Windows 7 or earlier, the guide is available from the **SimbaMongoDB ODBC Driver** program group in the Start menu.
- On Windows 8 or later, you can search for the guide on the Start screen.

Arrays as Virtual Tables

The driver can renormalize MongoDB arrays into virtual tables. For more information, see [Virtual Tables](#) on page 58.

Arrays as Columns

The driver can also renormalize MongoDB arrays into columns. Consider the following JSON document:

```
{"values": ["hello", 1, {"v1": {"v2": "this is an embedded document"}}]}
```

The driver can represent the array elements using the following columns, where values_1 is of type Double and the other two are of type String:

values_0	values_1	values_2_v1_v2
Hello	1.0	this is an embedded document

The driver works with these columns as if they were standard table columns.

The column names include the index of the array element that the column represents, starting with an index of 0. In other words, the first element of the array uses a suffix of _0, the second element uses _1, and so on.

Schema Definitions

To enable consistent support for your MongoDB data, you must configure the driver to use a schema definition from a JSON file or the database. You can use the Schema Editor application to create a schema definition and then save it in a JSON file or the database.

Note:

The Schema Editor is not supported on AIX. On all other platforms, information about how to use the Schema Editor can be found in the *Schema Editor User Guide* located in the installation directory of the driver.

- On Windows 7 or earlier, the guide is available from the **SimbaMongoDB ODBC Driver** program group in the Start menu.
- On Windows 8 or later, you can search for the guide on the Start screen.

When the driver connects to a database without a specified schema definition, it automatically generates a temporary schema definition using the settings defined in

the Sampling area of the Advanced Options dialog box or the `SamplingStrategy`, `SamplingLimit`, and `SamplingStepSize` properties. However, temporary schema definitions do not persist after the connection is closed, and the driver may generate different schema definitions during subsequent connections to the same database due to variances in the data that gets sampled.

! Important:

When creating a schema definition or connecting to the database without specifying one, make sure to configure the driver to sample all the necessary data. Documents that are not sampled do not get included in the schema definition, and consequently do not become available in ODBC applications. If the schema definition was created through the Schema Editor, you can use the Schema Editor to sample additional documents and add them to the schema definition. Note that the Schema Editor is not supported on AIX.

Mapping MongoDB Data

MongoDB is able to store data that follows different rules of data typing and structure compared to traditional relational data, and the tables may contain complex data such as nested arrays or arrays of differently-typed elements. Because traditional ODBC toolsets may not support these data structures, the Simba MongoDB ODBC Driver generates a schema definition that maps the MongoDB data to an ODBC-compatible format.

The Simba MongoDB ODBC Driver does the following when generating a schema definition:

1. Samples the data in the database in order to detect its structure and determine the data mappings that best support the data.
2. Assigns a MongoDB data type to each column.
3. Maps each MongoDB data type to the SQL data type that is best able to represent the greatest number of values.
4. Renormalizes single-level objects into columns.
5. For each array and nested object in the database, the driver generates a virtual table to expand the data, and saves these virtual tables as part of the schema definition. For more information about virtual tables, see [Virtual Tables](#) on page 58.

During this sampling process, the driver defines data types for each column, but does not change the data types of the individual cells in the database. As a result, columns may contain mixed data types. During read operations, values are converted to match the SQL data type of the column so that the driver can work with all the data in the column consistently.

Virtual Tables

One advantage of the MongoDB design is the ability to store data that is denormalized into a fewer number of tables. However, the ODBC interface does not natively support accessing denormalized data types such as arrays and objects. By expanding the data contained within arrays and objects into virtual tables, the Simba MongoDB ODBC Driver allows you to directly interact with the data but leave the storage of the data in its denormalized form in MongoDB.

If any columns are mapped to a denormalized data type during the sampling process, then the driver creates the following tables and saves them as part of the schema definition:

- A base table, which contains only the normal data from the real table.
- A virtual table for each column of denormalized data, expanding the nested data.

Virtual tables refer to the data in the real table, enabling the driver to access the denormalized data. By querying the virtual tables, you can access the contents of arrays and objects via ODBC. When you write or modify data in a virtual table, the data in the real table in the MongoDB database is updated.

The base table and virtual tables appear as additional tables in the list of tables that exist in the database, and are named using the following conventions:

- The base table uses the same name as the real table that it represents.
- In the ODBC layer, the name of each virtual table is formed using the name of the real table that contains the array or object, an underscore (`_`), and the name of the array or object.
- In the MongoDB layer, the name of each virtual table is formed using the name of the collection that the data comes from, a period (`.`), and then the name of the array or object followed by a set of closed square brackets (`[]`) for each hierarchy level in which the array or object is nested.
- If a virtual table or column has the same name as an actual table or column in the database, then the driver appends `_1` to the virtual table or column name as a suffix. The number increments as necessary until the name is unique.

For example, consider the example MongoDB table named `CustomerTable` shown below.

_id	Customer Name	Invoices	Service Level	Contacts	Ratings
1111	ABC	[[{"invoice_id": "123", "item": "toaster", "price": "456", "discount": "0.2"}, {"invoice_id": "124", "item": "oven", "price": "1235", "discount": "0.2"}]]	Silver	[[{"type": "primary", "name": "John Johnson"}, {"type": "invoicing", "name": "Jill Jilliamson"}]]	[5,6]
2222	XYZ	[[{"invoice_id": "135", "item": "fridge", "price": "12543", "discount": "0.0"}]]	Gold	[[{"type": "primary", "name": "Jane Doe"}]]	[1,2]

CustomerTable has two columns that have an array of Objects in each cell, Invoices and Contacts, and one column that has an array of Scalar types, Ratings. Multiple virtual tables would be generated for this single source table. The first table is the base table, which is shown below.

_id	Customer Name	Service Level
1111	ABC	Silver
2222	XYZ	Gold

The base table contains all of the data of the original table, but the data from the arrays has been omitted and will be expanded in the virtual tables.

The following three tables show the virtual tables that represent the original arrays in the example.

_id	Invoices_dim1_idx	invoice_id	item	price	discount
1111	0	123	toaster	456	0.2
1111	1	124	oven	1235	0.2
2222	0	135	fridge	12543	0.0

_id	Contacts_dim1_idx	type	name
1111	0	primary	John Johnson
1111	1	invoicing	Jill Jillichamson
2222	0	primary	Jane Doe

_id	Ratings_dim1_idx	Ratings_value
1111	0	5
1111	1	6
2222	0	1
2222	1	2

Each of these tables contain the following:

- A reference back to the original primary key column corresponding to the row of the original array (via the `_id` column)
- An indication of the position of the data within the original array (using the `Invoices_dim1_idx`, `Contacts_dim1_idx`, and `Ratings_dim1_idx` columns)
- The expanded data for each element within the array:
 - **Invoices:** `invoice_id`, `item`, `price` and `discount`
 - **Contacts:** `type` and `name`
 - **Ratings:** `Ratings_value`

You can select, insert, and update data in the base tables and virtual tables as if they were standard relational tables, and the driver will handle the storage details within MongoDB.

For example, to append 7 to the Ratings array in the CustomerTable where `_id = 1111`, execute the following statement:

```
INSERT INTO CustomerTable_Ratings (_id, Ratings_value)
VALUES (1111, 7)
```

Some operations may be processed differently or not supported for certain types of data. For more information, see [Write-back](#) on page 61.

Write-back

The Simba MongoDB ODBC Driver supports Data Manipulation Language (DML) statements such as INSERT, UPDATE, and DELETE. The driver does not support SQL subqueries or ODBC transactions.

! Important:

Writing data to the MongoDB database may change the typing of the data. If the existing data is of a type that is different from the column data type specified in the schema definition, then the existing data will be replaced by new data that is of the column data type.

When the driver samples the data and generates the schema definition, the driver defines a SQL data type and a MongoDB data type for each column in the base tables and virtual tables, but does not change the data types of the individual values in the database. As a result, columns may contain mixed data types. The driver supports DML statements on mixed data types. When you execute a write operation, the driver will attempt to complete the operation using the column data type specified in the schema definition.

The data provided in DML statements must match the column data types. For example, a String value cannot be inserted in a column that is defined as an Integer column in the schema definition.

The driver handles each DML statement as described below.

! Important:

The CREATE, ALTER, and DROP statements are not supported for tables.

INSERT

Each row in MongoDB needs to have a unique ID represented by the `_id` column. If not provided during insertion, MongoDB auto-generates a unique ID for each row. The `_id` field is exposed as a valid column in the driver and can be auto-generated when issuing INSERT statements through the driver. For example, consider the following table.

<code>_id</code>	<code>sample_column</code>
"517024D6CC79814E3FEBD352"	1
"5170ED77E49CC93A918DE316"	2

To insert a document with an auto-generated value for `_id` (data type: `jstOID`), issue the following command:

```
INSERT INTO sample_table_1(sample_column) VALUES(3)
```

The following table shows the table after the insertion.

<code>_id</code>	<code>sample_column</code>
"517024D6CC79814E3FEBD352"	1
"5170ED77E49CC93A918DE316"	2
"51710FFCE49CC93A918DE322"	3

The value for the `_id` column can also be inserted using the `INSERT` statements, as in the following examples:

```
INSERT INTO sample_table_2(_id, sample_column) VALUES(1,1)
```

```
INSERT INTO sample_table_2 VALUES(1,1)
```

UPDATE

When updating rows, special care needs to be taken to avoid duplicate values for the `_id` column. As mentioned before, `_id` needs to be unique across all rows. When an `UPDATE` statement tries to set a value for the `_id` column and matches multiple rows, only one of the rows is updated with the new values, and the driver return an error for the remaining rows. `UPDATE` is not executed atomically.

DELETE

The `DELETE` statement is not supported for virtual tables.


The driver considers a table valid as long as the table contains some data. If a table is completely empty, then the driver is not able to access the table. Consider the following example:

```
DELETE FROM sample_table_3
```

The command removes all data from `sample_table_3`. Therefore, `sample_table_3` is invalid. Any users attempting to access the table receive an error.

Security and Authentication

To protect data from unauthorized access, some MongoDB data stores require connections to be authenticated with user credentials or the SSL protocol. The Simba MongoDB ODBC Driver provides full support for these authentication protocols.

 **Note:**

In this documentation, "SSL" refers to both TLS (Transport Layer Security) and SSL (Secure Sockets Layer). The driver supports up to TLS 1.2. The SSL version used for the connection is the highest version that is supported by both the driver and the server.

The driver provides a mechanism that enables you to authenticate your connection using the SCRAM-SHA-1 protocol (which MongoDB uses by default), the Kerberos protocol, or the LDAP protocol. For detailed configuration instructions, see [Configuring Authentication on Windows](#) on page 12 or [Configuring Authentication on a Non-Windows Machine](#) on page 40.

Additionally, the driver supports the following types of SSL connections:

- No identity verification
- One-way authentication
- Two-way authentication

It is recommended that you enable SSL whenever you connect to a server that is configured to support it. SSL encryption protects data and credentials when they are transferred over the network, and provides stronger security than authentication alone. For detailed configuration instructions, see .

Driver Configuration Options

Driver Configuration Options lists the configuration options available in the Simba MongoDB ODBC Driver alphabetically by field or button label.

When creating or configuring a connection from a Windows machine, the fields and buttons are available in the following dialog boxes:

- Simba MongoDB ODBC Driver DSN Setup
- Advanced Options
- SSL Options
- Writeback Options

When using a connection string or configuring a connection from a non-Windows machine, use the key names provided.

The following configuration options are available:

- [Allow Self-Signed Certificates](#) on page 65
- [Authentication Source](#) on page 65
- [Batch Size](#) on page 66
- [Binary Column Size](#) on page 66
- [Certificate Authority File](#) on page 66
- [Certificate Authority Directory](#) on page 67
- [Certificate Revocation List File](#) on page 67
- [Connect to Replica Set](#) on page 67
- [Database](#) on page 68
- [Documents to Fetch Per Block](#) on page 68
- [Documents to Sample](#) on page 68
- [Enable Double-Buffering](#) on page 69
- [Enable Json Read/Write Mode](#) on page 69
- [Enable Mixed Type Filter](#) on page 69
- [Max File Size](#) on page 75
- [Max Number Files](#) on page 76
- [Mechanism \(Authentication\)](#) on page 76
- [Mechanism \(Metadata\)](#) on page 77
- [No Cursor Timeout](#) on page 77
- [Password](#) on page 78
- [PEM Key File](#) on page 78
- [PEM Key Password](#) on page 79
- [Port](#) on page 79
- [Read Preference](#) on page 79
- [Replica Set Name](#) on page 80
- [Sampling Method](#) on page 80
- [Secondary Servers](#) on page 81
- [Server](#) on page 81
- [Service Name](#) on page 81
- [Socket Timeout](#) on page 82
- [Step Size](#) on page 82
- [String Column Size](#) on page 82
- [Timeout](#) on page 83

- 70
- [Enable Passdown](#) on page 71
- [Enable SSL](#) on page 71
- [Encrypt Password](#) on page 71
- [Expose Binary as SQL_LONGVARBINARY](#) on page 72
- [JSON Column Size](#) on page 73
- [Local File](#) on page 74
- [Log Level](#) on page 74
- [Log Path](#) on page 75
- [Username](#) on page 84
- [Write Concern](#) on page 85

Allow Self-Signed Certificates

Key Name	Default Value	Required
sslAllowInvalidCertificates	Clear (0)	No

Description

This option specifies whether the driver allows self-signed SSL certificates from the server.

- Enabled (1): The driver authenticates the MongoDB server even if the server is using a self-signed certificate.
- Disabled (0): The driver does not allow self-signed certificates from the server.

 **Note:**

This setting is only applicable when SSL is enabled.

Authentication Source

Key Name	Default Value	Required
AuthSource	admin	No

Description

The name of the MongoDB database that you want to use to check your credentials for authentication.

Batch Size

Key Name	Default Value	Required
DmlBatchSize	500	No

Description

The maximum number of documents that the driver can handle at one time during a write operation.

Binary Column Size

Key Name	Default Value	Required
DefaultBinaryColumnLength	32767	No

Description

The maximum data length for Binary columns.

Note:

The maximum value that you can set for this option is 2147483647.

Certificate Authority File

Key Name	Default Value	Required
sslCAFile	None	No

Description

The full path of the .pem file that you use to verify the server.

Note:

To configure the driver to access multiple `.pem` files for SSL verification, use the Certificate Authority Directory option or the `sslCADir` key instead.

Certificate Authority Directory

Key Name	Default Value	Required
<code>sslCADir</code>	None	No

Description

The full path of the directory containing the `.pem` files that you use to verify the server. This setting enables the driver to access multiple `.pem` files for SSL verification.

Note:

To specify a single `.pem` file, use the Certificate Authority File option or the `sslCAFile` key instead.

Certificate Revocation List File

Key Name	Default Value	Required
<code>sslCRLFile</code>	None	No

Description

The full path of the `.pem` file containing the list of revoked certificates.

Connect to Replica Set

Key Name	Default Value	Required
<code>EnableReplicaSet</code>	Clear (0)	No

Description

This option specifies whether the driver can access replica sets in your MongoDB implementation.

- Enabled (1): The driver can access replica sets in your MongoDB implementation.
- Disabled (0): The driver cannot access replica sets.

Database

Key Name	Default Value	Required
Database	test	Yes

Description

The name of the MongoDB database that you want to access.

Documents to Fetch Per Block

Key Name	Default Value	Required
BatchSize	4096	No

Description

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

This setting also determines the buffer size used when double-buffering is enabled.

Documents to Sample

Key Name	Default Value	Required
SamplingLimit	100	No

Description

The maximum number of records that the driver can sample to generate a temporary schema definition. When this option is set to 0, the driver samples every document in the database.

! Important:

- Make sure to configure the driver to sample all the necessary data. Documents that are not sampled do not get included in the schema definition, and consequently do not become available in ODBC applications.
- Typically, sampling a large number of documents results in a schema definition that is more accurate and better able to represent all the data in the database. However, the sampling process may take longer than expected when many documents are sampled, especially if the database contains complex, nested data structures.

Enable Double-Buffering

Key Name	Default Value	Required
EnableDoubleBuffer	Selected (1)	No

Description

This option specifies whether the driver retrieves the data using double-buffering. For more information about double-buffering, see [Double-Buffering](#) on page 51.

- Enabled (1): The driver retrieves the data using double-buffering.
- Disabled (0): The driver retrieves the data using single-buffering.

Enable Json Read/Write Mode

Key Name	Default Value	Required
UseJsonColumn	0	No

Description

This option specifies whether the driver reports a special column named DocumentAsJson that retrieves or stores whole documents as Json-formatted strings.

- Enabled (1): The driver reports the Json column.
- Disabled (0): The driver does not report the Json column.

 **Note:**

Only enable this option as needed. The process of converting documents into Json strings decreases the performance of the driver.

See also the driver configuration option [JSON Column Size](#) on page 73.

Enable Mixed Type Filter

Key Name	Default Value	Required
<code>EnableMixedTypeFilter</code>	Selected (1)	No

Description

This option specifies whether the driver uses mixed type filtering, which allows a value to pass the filter even if it is not the same data type as the specified filter term.

- Enabled (1): When filtering data, the driver retrieves all values that match the specified filter term, even if the values are stored as different data types than the filter term. For example, if the data store contains "1000" as a NumberInt value and also as a String value, when you execute a query that filters for "1000" as a String value, the driver returns both the NumberInt and String values.
- Disabled (0): When filtering data, the driver only retrieves values that match the data type of the specified filter term. For example, if the data store contains "1000" as a NumberInt value and also as a String value, when you execute a query that filters for "1000" as a String value, the driver only returns the String value.

 **Note:**

Mixed type filtering requires the driver to scan the entire MongoDB collection. You can disable this feature to increase the driver's performance, but note that doing so also alters the results of your queries.

Enable Passthrough

Key Name	Default Value	Required
EnablePassthroughOptimization	Selected (1)	No

Description

This option specifies whether the driver optimizes joins between virtual tables, and passes filtering and aggregation optimizations to the MongoDB database for handling.

- Enabled (1): The driver optimizes joins between virtual tables, and passes filtering and aggregation optimizations to the MongoDB database for handling.
- Disabled (0): The driver does not optimize joins, and does not pass filtering and aggregation optimizations to the MongoDB database.

Enable SSL

Key Name	Default Value	Required
SSL	Clear(0)	No

Description

This option specifies whether the driver uses SSL to connect to the server.

- Enabled (1): The driver uses SSL to connect to the server.
- Disabled (0): The driver does not use SSL to connect to the server.

Encrypt Password

Key Name	Default Value	Required
N/A	All Users Of This Machine	No

Description

This option specifies how the driver encrypts the credentials that are saved in the DSN:

- **Current User Only:** The credentials are encrypted, and can only be used by the current Windows user.
- **All Users Of This Machine:** The credentials are encrypted, but can be used by any user on the current Windows machine.

! Important:

This option is available only when you configure a DSN using the Simba MongoDB ODBC Driver DSN Setup dialog box in the Windows driver. When you connect to the data store using a connection string, the driver does not encrypt your credentials.

Expose Binary as SQL_LONGVARBINARY

Key Name	Default Value	Required
UseSqlLongVarbinary	1	No

Description

This option specifies whether the driver returns Binary columns as data of type SQL_LONGVARBINARY or SQL_VARBINARY.

- Enabled (1): The driver returns Binary columns as SQL_LONGVARBINARY data.
- Disabled (0): The driver returns Binary columns as SQL_VARBINARY data.

Expose Strings as SQL_WVARCHAR

Key Name	Default Value	Required
UseSqlWvarchar	Selected (1)	No

Description

This option specifies how the String data type is mapped to SQL.

- Enabled (1): The MongoDB String data type is mapped to SQL_WVARCHAR.
- Disabled (0): The MongoDB String data type is mapped to SQL_VARCHAR.

JSON Column Size

Key Name	Default Value	Required
JsonColumnSize	1023	No

Description

The default column length for JSON fields.

See also the driver configuration option [Enable Json Read/Write Mode](#) on page 69.

Journalled Writes

Key Name	Default Value	Required
WriteConcernJournalled	Clear(0)	No

Description

This option specifies whether the driver requires that the data from a write operation to be committed to the journal before the write operation can be acknowledged.

- Enabled (1): The driver requires that the data from a write operation to be committed to the journal before the write operation can be acknowledged.
- Disabled (0): The driver does not require that the data from a write operation to be committed to the journal before the write operation can be acknowledged.

Note:

This option is only applicable when the Write Concern option or the `WriteConcern` key is enabled.

Local File

Key Name	Default Value	Required
LocalMetadataFile	None	Yes, if the metadata mechanism is set to Local File or LoadMetadataTable=0

Description

The full path of a local JSON file containing the schema definition that you want the driver to use when connecting to MongoDB.

Log Level

Key Name	Default Value	Required
LogLevel	OFF (0)	No

Description

Use this property to enable or disable logging in the driver and to specify the amount of detail included in log files.

! Important:

- Only enable logging long enough to capture an issue. Logging decreases performance and can consume a large quantity of disk space.
- The settings for logging apply to every connection that uses the Simba MongoDB ODBC Driver, so make sure to disable the feature after you are done using it.
- This option is not supported in connection strings. To configure logging for the Windows driver, you must use the Logging Options dialog box. To configure logging for a non-Windows driver, you must use the `simba.mongodbodbc.ini` file.

Set the property to one of the following values:

- OFF (0): Disable all logging.
- FATAL (1): Logs severe error events that lead the driver to abort.

- ERROR (2): Logs error events that might allow the driver to continue running.
- WARNING (3): Logs events that might result in an error if action is not taken.
- INFO (4): Logs general information that describes the progress of the driver.
- DEBUG (5): Logs detailed information that is useful for debugging the driver.
- TRACE (6): Logs all driver activity.

When logging is enabled, the driver produces a log file named `mongodbodbc_driver.log` in the location specified in the Log Path (`LogPath`) property.

Log Path

Key Name	Default Value	Required
<code>LogPath</code>	None	Yes, if logging is enabled.

Description

The full path to the folder where the driver saves log files when logging is enabled.

! Important:

This option is not supported in connection strings. To configure logging for the Windows driver, you must use the Logging Options dialog box. To configure logging for a non-Windows driver, you must use the `simba.mongodbodbc.ini` file.

Max File Size

Key Name	Default Value	Required
<code>LogFileSize</code>	20	No

Description

The maximum size of each log file in megabytes (MB). After the maximum file size is reached, the driver creates a new file and continues logging.

! Important:

This option is not supported in connection strings. To configure logging for the Windows driver, you must use the Logging Options dialog box. To configure logging for a non-Windows driver, you must use the `simba.mongodbc.ini` file.

Max Number Files

Key Name	Default Value	Required
LogFileCount	50	No

Description

The maximum number of log files to keep. After the maximum number of log files is reached, each time an additional file is created, the driver deletes the oldest log file.

! Important:

This option is not supported in connection strings. To configure logging for the Windows driver, you must use the Logging Options dialog box. To configure logging for a non-Windows driver, you must use the `simba.mongodbc.ini` file.

Mechanism (Authentication)

Key Name	Default Value	Required
AuthMechanism	No Authentication (None)	No

Description

! Important:

This section describes the Mechanism option from the Simba MongoDB ODBC Driver DSN Setup dialog box. For information about the Mechanism option from the Advanced Options dialog box, see [Mechanism \(Metadata\)](#) on page 77.

The authentication mechanism to use, selected from the following:

- No Authentication (`None`): The driver does not authenticate the connection.
- MongoDB User Name and Password (`SCRAM-SHA-1`): The driver authenticates the connection using the SCRAM-SHA-1 protocol, which is the default authentication protocol used by MongoDB.
- Kerberos (`GSSAPI`): The driver authenticates the connection using the Kerberos protocol.
- LDAP (`PLAIN`): The driver authenticates the connection using the LDAP protocol.

Mechanism (Metadata)

Key Name	Default Value	Required
<code>LoadMetadataTable</code>	Database (1)	No

Description

! Important:

This section describes the Mechanism option from the Advanced Options dialog box. For information about the Mechanism option from the Simba MongoDB ODBC Driver DSN Setup dialog box, see [Mechanism \(Authentication\)](#) on page 76.

This option specifies where the driver looks for the schema definition.

- Database (1): The driver loads the schema definition from the MongoDB database.
- Local File (0): The driver loads the schema definition from the JSON file specified in the Local File field or the `LocalMetadataFile` key.

No Cursor Timeout

Key Name	Default Value	Required
<code>NoCursorTimeout</code>	False	No

Description

This option specifies whether the driver allows active cursors on the data source server to expire.

- `False`: The data source server will time out idle cursors after the threshold inactivity period set on the server.
- `True`: The driver prevents the data source server from timing out idle cursors.

! Important:

When set to `True`, there is a risk that if the driver should quit or lose the connection to the server unexpectedly, the cursor will remain open on the server indefinitely. You can adjust the threshold for idle cursor timeouts on the MongoDB server, see <https://docs.mongodb.com/v3.0/reference/parameters/> for details.

Password

Key Name	Default Value	Required
PWD	None	Yes, if the authentication mechanism is MongoDB User Name and Password (SCRAM-SHA-1) or LDAP (PLAIN).

Description

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

PEM Key File

Key Name	Default Value	Required
<code>sslPEMKeyFile</code>	None	No

Description

The full path of the `.pem` file containing the certificate for verifying the client.

PEM Key Password

Key Name	Default Value	Required
sslPEMKeyPwd	None	No

Description

The password of the client certificate file that is specified in the PEM Key File field (`sslPEMKeyFile`).

Port

Key Name	Default Value	Required
Port	27017	Yes

Description

The TCP port that the MongoDB server uses to listen for client connections.

Read Preference

Key Name	Default Value	Required
ReadPreference	Primary (<code>primary</code>)	No

Description

Specify how the driver routes read operations to the members of a replica set. The following values are possible:

- Primary (`primary`)
- Primary preferred (`primaryPreferred`)
- Secondary (`secondary`)
- Secondary preferred (`secondaryPreferred`)
- Nearest (`nearest`)

Note:

- The values are case-sensitive.
- This option is only available when the Connect to Replica Set option or the `EnableReplicaSet` key is enabled.

For more information, see "Replication" in the *MongoDB Manual*:
<http://docs.mongodb.org/manual/replication/>.

Replica Set Name

Key Name	Default Value	Required
ReplicaSet	None	Yes, if connecting to a replica set.

Description

The name of the replica set for the driver to access.

Note:

This option is only available when the Connect to Replica Set option or the `EnableReplicaSet` key is enabled.

Sampling Method

Key Name	Default Value	Required
SamplingStrategy	Forward	No

Description

This options specifies how the driver samples data when generating a temporary schema definition.

- `Forward`: The driver samples data starting from the first record in the database, then samples the next record, and so on.
- `Backwards`: The driver samples data starting from the last record in the database, then samples the preceding record, and so on.

- **Random:** The driver selects sample records from the data source at random until the `SamplingLimit` is reached.

**Note:**

The random sampling strategy is only supported by MongoDB Server 3.2 or higher.

Secondary Servers

Key Name	Default Value	Required
<code>SecondaryServers</code>	None	No

Description

A comma-separated list of the servers that you need to use when connecting to a replica set. You can indicate the TCP port that the server is using to listen for client connections by appending a colon (:) and the port number to the server name or IP address.

Server

Key Name	Default Value	Required
<code>Server</code>	None	Yes

Description

The host name or IP address of the MongoDB server.

Service Name

Key Name	Default Value	Required
<code>gssapiServiceName</code>	<code>mongodb</code>	Yes, if the authentication mechanism is Kerberos.

Description

The Kerberos service principal name of the MongoDB server.

Socket Timeout

Key Name	Default Value	Required
SocketTimeoutMS	-1	No

Description

The amount of time, in milliseconds, before an attempt to send or receive on a socket times out. The default -1 setting turns the option off, so that timeout does not occur.

Step Size

Key Name	Default Value	Required
SamplingStepSize	1	No

Description

The interval at which the driver samples records when scanning through the database to generate a temporary schema definition. For example, if you set this option to 2, then the driver samples every second record in the database.

If the Sampling Method is set to Random, this setting is ignored.

String Column Size

Key Name	Default Value	Required
DefaultStringColumnLength	255	No

Description

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

Note:

The maximum value that you can set for this option is 2147483647.

Timeout

Key Name	Default Value	Required
WriteConcernTimeout	0	No

Description

The maximum number of seconds that the driver waits for a secondary server to acknowledge a write operation before reporting that the operation has failed.

When this option is set to 0, the driver does not time out. Instead, the driver waits until all secondary servers acknowledge the write operation, and then reports that the operation has succeeded.

Note:

This option is only applicable when the Write Concern option or the `WriteConcern` key is enabled.

! Important:

If you set the Write Concern option or key to a value that is greater than 1, make sure that this option is not set to 0. This may cause the driver to wait indefinitely for replica set members to come online.

UseLogPrefix

Key Name	Default Value	Required
UseLogPrefix	0	No

Description

This option specifies whether the driver includes a prefix in the names of log files so that the files can be distinguished by user and application.

! Important:

To configure this option for the Windows driver, you create a value for it in one of the following registry keys:

- For a 32-bit driver installed on a 64-bit machine: `HKEY_LOCAL_MACHINE\SOFTWAREWow6432Node\Simba\Simba MongoDB ODBC Driver\Driver`
- Otherwise: `HKEY_LOCAL_MACHINE\SOFTWARE\Simba\Simba MongoDB ODBC Driver\Driver`

Use `UseLogPrefix` as the value name, and either `0` or `1` as the value data.

To configure this option for a non-Windows driver, you must use the `simba.mongodbodbc.ini` file.

Set the property to one of the following values:

- `1`: The driver prefixes log file names with the user name and process ID associated with the connection that is being logged.

For example, if you are connecting as a user named "jdoe" and using the driver in an application with process ID 7836, the generated log file would be named `jdoe_7836_.`

- `0`: The driver does not include the prefix in log file names.

Username

Key Name	Default Value	Required
UID	None	Yes, if the authentication mechanism is MongoDB User Name and Password (SCRAM-SHA-1) or LDAP (PLAIN).

Description

The user name that you use to access the MongoDB instance.

Write Concern

Key Name	Default Value	Required
WriteConcern	1	No

Description

The total number of primary and secondary servers that must acknowledge a write operation in order for the driver to report a successful write operation.

When this option is set to 0, the driver does not require write operations to be acknowledged.

! Important:

- If you set this option to a value that is greater than 1, make sure to also specify an appropriate value for the Timeout setting or the `WriteConcernTimeout` key. Setting this option to a value greater than 1 without specifying a timeout interval may cause the driver to wait indefinitely for replica set members to come online.
- The process for acknowledging a write operation typically takes four times longer than an INSERT or UPDATE operation, but it is necessary if fault tolerance is important.

Third-Party Trademarks

Debian is a trademark or registered trademark of Software in the Public Interest, Inc. or its subsidiaries in Canada, United States and/or other countries.

IBM and AIX are trademarks or registered trademarks of IBM Corporation or its subsidiaries in Canada, United States, and/or other countries.

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.

Mac, macOS, Mac OS, and OS X are trademarks or registered trademarks of Apple, Inc. or its subsidiaries in Canada, United States and/or other countries.

Microsoft, MSDN, Windows, 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.

SUSE is a trademark or registered trademark of SUSE LLC or its subsidiaries in Canada, United States and/or other countries.

Ubuntu is a trademark or registered trademark of Canonical Ltd. or its subsidiaries in Canada, United States and/or other countries.

MongoDB and Mongo are trademarks or registered trademarks of MongoDB, Inc. or its subsidiaries in Canada, the United States and/or other countries.

All other trademarks are trademarks of their respective owners.

Third-Party Licenses

The licenses for the third-party libraries that are included in this product are listed below.

Cyrus SASL License

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 License

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 NOINFRINGEMENT. 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 License

Copyright (c) 1998-2016 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 License

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>

Oracle Binary Code License Agreement for the Java SE Platform Products and JavaFX

ORACLE AMERICA, INC. ("ORACLE"), FOR AND ON BEHALF OF ITSELF AND ITS SUBSIDIARIES AND AFFILIATES UNDER COMMON CONTROL, IS WILLING TO LICENSE THE SOFTWARE TO YOU ONLY UPON THE CONDITION THAT YOU ACCEPT ALL OF THE TERMS CONTAINED IN THIS BINARY CODE LICENSE AGREEMENT AND SUPPLEMENTAL LICENSE TERMS (COLLECTIVELY "AGREEMENT"). PLEASE READ THE AGREEMENT CAREFULLY. BY SELECTING THE "ACCEPT LICENSE AGREEMENT" (OR THE EQUIVALENT) BUTTON AND/OR BY USING THE SOFTWARE YOU ACKNOWLEDGE THAT YOU HAVE READ THE TERMS AND AGREE TO THEM. IF YOU ARE AGREEING TO THESE TERMS ON BEHALF OF A COMPANY OR OTHER LEGAL ENTITY, YOU REPRESENT THAT YOU HAVE THE LEGAL AUTHORITY TO BIND THE LEGAL ENTITY TO THESE TERMS. IF YOU DO NOT HAVE SUCH AUTHORITY, OR IF YOU DO NOT WISH TO BE BOUND BY THE TERMS, THEN SELECT THE "DECLINE LICENSE AGREEMENT" (OR THE EQUIVALENT) BUTTON AND YOU MUST NOT USE THE SOFTWARE ON THIS SITE OR ANY OTHER MEDIA ON WHICH THE SOFTWARE IS CONTAINED.

1. DEFINITIONS. "Software" means the software identified above in binary form that you selected for download, install or use (in the version You selected for download, install or use) from Oracle or its authorized licensees, any other machine readable materials (including, but not limited to, libraries, source files, header files, and data files), any updates or error corrections provided by Oracle, and any user manuals, programming guides and other documentation provided to you by Oracle under this Agreement. "General Purpose Desktop Computers and Servers" means computers, including desktop and laptop computers, or servers, used for general computing functions under end user control (such as but not specifically limited to email, general purpose Internet browsing, and office suite productivity tools). The use of Software in systems and solutions that provide dedicated functionality (other than as mentioned above) or designed for use in embedded or function-specific software applications, for example but not limited to: Software embedded in or bundled with industrial control systems, wireless mobile telephones, wireless handheld devices, kiosks, TV/STB, Blu-ray Disc devices, telematics and network control switching equipment, printers and storage management systems, and other related systems are excluded from this definition and not licensed under this Agreement. "Programs" means (a) Java technology applets and applications intended to run on the Java Platform, Standard Edition platform on Java-enabled General Purpose Desktop Computers and Servers; and (b) JavaFX technology applications intended to run on the JavaFX Runtime on JavaFX-enabled General Purpose Desktop Computers and Servers. "Commercial Features" means those features identified in Table 1-1 (Commercial Features In Java SE Product Editions) of the Java SE documentation accessible at

<http://www.oracle.com/technetwork/java/javase/documentation/index.html>.
“README File” means the README file for the Software accessible at
<http://www.oracle.com/technetwork/java/javase/documentation/index.html>.

2. LICENSE TO USE. Subject to the terms and conditions of this Agreement including, but not limited to, the Java Technology Restrictions of the Supplemental License Terms, Oracle grants you a non-exclusive, non-transferable, limited license without license fees to reproduce and use internally the Software complete and unmodified for the sole purpose of running Programs. THE LICENSE SET FORTH IN THIS SECTION 2 DOES NOT EXTEND TO THE COMMERCIAL FEATURES. YOUR RIGHTS AND OBLIGATIONS RELATED TO THE COMMERCIAL FEATURES ARE AS SET FORTH IN THE SUPPLEMENTAL TERMS ALONG WITH ADDITIONAL LICENSES FOR DEVELOPERS AND PUBLISHERS.
3. RESTRICTIONS. Software is copyrighted. Title to Software and all associated intellectual property rights is retained by Oracle and/or its licensors. Unless enforcement is prohibited by applicable law, you may not modify, decompile, or reverse engineer Software. You acknowledge that the Software is developed for general use in a variety of information management applications; it is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use the Software in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure its safe use. Oracle disclaims any express or implied warranty of fitness for such uses. No right, title or interest in or to any trademark, service mark, logo or trade name of Oracle or its licensors is granted under this Agreement. Additional restrictions for developers and/or publishers licenses are set forth in the Supplemental License Terms.
4. DISCLAIMER OF WARRANTY. THE SOFTWARE IS PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND. ORACLE FURTHER DISCLAIMS ALL WARRANTIES, EXPRESS AND IMPLIED, INCLUDING WITHOUT LIMITATION, ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NONINFRINGEMENT.
5. LIMITATION OF LIABILITY. IN NO EVENT SHALL ORACLE BE LIABLE FOR ANY INDIRECT, INCIDENTAL, SPECIAL, PUNITIVE OR CONSEQUENTIAL DAMAGES, OR DAMAGES FOR LOSS OF PROFITS, REVENUE, DATA OR DATA USE, INCURRED BY YOU OR ANY THIRD PARTY, WHETHER IN AN ACTION IN CONTRACT OR TORT, EVEN IF ORACLE HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. ORACLE'S ENTIRE LIABILITY FOR DAMAGES HEREUNDER SHALL IN NO EVENT EXCEED ONE THOUSAND DOLLARS (U.S. \$1,000).
6. TERMINATION. This Agreement is effective until terminated. You may terminate this Agreement at any time by destroying all copies of Software. This Agreement will terminate immediately without notice from Oracle if you fail to comply with any provision of this Agreement. Either party may terminate this Agreement

immediately should any Software become, or in either party's opinion be likely to become, the subject of a claim of infringement of any intellectual property right. Upon termination, you must destroy all copies of Software.

7. EXPORT REGULATIONS. You agree that U.S. export control laws and other applicable export and import laws govern your use of the Software, including technical data; additional information can be found on Oracle's Global Trade Compliance web site (<http://www.oracle.com/us/products/export>). You agree that neither the Software nor any direct product thereof will be exported, directly, or indirectly, in violation of these laws, or will be used for any purpose prohibited by these laws including, without limitation, nuclear, chemical, or biological weapons proliferation.
8. TRADEMARKS AND LOGOS. You acknowledge and agree as between you and Oracle that Oracle owns the ORACLE and JAVA trademarks and all ORACLE- and JAVA-related trademarks, service marks, logos and other brand designations ("Oracle Marks"), and you agree to comply with the Third Party Usage Guidelines for Oracle Trademarks currently located at <http://www.oracle.com/us/legal/third-party-trademarks/index.html>. Any use you make of the Oracle Marks inures to Oracle's benefit.
9. U.S. GOVERNMENT LICENSE RIGHTS. If Software is being acquired by or on behalf of the U.S. Government or by a U.S. Government prime contractor or subcontractor (at any tier), then the Government's rights in Software and accompanying documentation shall be only those set forth in this Agreement.
10. GOVERNING LAW. This agreement is governed by the substantive and procedural laws of California. You and Oracle agree to submit to the exclusive jurisdiction of, and venue in, the courts of San Francisco, or Santa Clara counties in California in any dispute arising out of or relating to this agreement.
11. SEVERABILITY. If any provision of this Agreement is held to be unenforceable, this Agreement will remain in effect with the provision omitted, unless omission would frustrate the intent of the parties, in which case this Agreement will immediately terminate.
12. INTEGRATION. This Agreement is the entire agreement between you and Oracle relating to its subject matter. It supersedes all prior or contemporaneous oral or written communications, proposals, representations and warranties and prevails over any conflicting or additional terms of any quote, order, acknowledgment, or other communication between the parties relating to its subject matter during the term of this Agreement. No modification of this Agreement will be binding, unless in writing and signed by an authorized representative of each party.

SUPPLEMENTAL LICENSE TERMS

These Supplemental License Terms add to or modify the terms of the Binary Code License Agreement. Capitalized terms not defined in these Supplemental Terms shall have the same meanings ascribed to them in the Binary Code License Agreement.

These Supplemental Terms shall supersede any inconsistent or conflicting terms in the Binary Code License Agreement, or in any license contained within the Software.

- A. **COMMERCIAL FEATURES.** You may not use the Commercial Features for running Programs, Java applets or applications in your internal business operations or for any commercial or production purpose, or for any purpose other than as set forth in Sections B, C, D and E of these Supplemental Terms. If You want to use the Commercial Features for any purpose other than as permitted in this Agreement, You must obtain a separate license from Oracle.
- B. **SOFTWARE INTERNAL USE FOR DEVELOPMENT LICENSE GRANT.** Subject to the terms and conditions of this Agreement and restrictions and exceptions set forth in the README File incorporated herein by reference, including, but not limited to the Java Technology Restrictions of these Supplemental Terms, Oracle grants you a non-exclusive, non-transferable, limited license without fees to reproduce internally and use internally the Software complete and unmodified for the purpose of designing, developing, and testing your Programs.
- C. **LICENSE TO DISTRIBUTE SOFTWARE.** Subject to the terms and conditions of this Agreement and restrictions and exceptions set forth in the README File, including, but not limited to the Java Technology Restrictions and Limitations on Redistribution of these Supplemental Terms, Oracle grants you a non-exclusive, non-transferable, limited license without fees to reproduce and distribute the Software, provided that (i) you distribute the Software complete and unmodified and only bundled as part of, and for the sole purpose of running, your Programs, (ii) the Programs add significant and primary functionality to the Software, (iii) you do not distribute additional software intended to replace any component(s) of the Software, (iv) you do not remove or alter any proprietary legends or notices contained in the Software, (v) you only distribute the Software subject to a license agreement that: (a) is a complete, unmodified reproduction of this Agreement; or (b) protects Oracle's interests consistent with the terms contained in this Agreement and that includes the notice set forth in Section H, and (vi) you agree to defend and indemnify Oracle and its licensors from and against any damages, costs, liabilities, settlement amounts and/or expenses (including attorneys' fees) incurred in connection with any claim, lawsuit or action by any third party that arises or results from the use or distribution of any and all Programs and/or Software. The license set forth in this Section C does not extend to the Software identified in Section G.
- D. **LICENSE TO DISTRIBUTE REDISTRIBUTABLES.** Subject to the terms and conditions of this Agreement and restrictions and exceptions set forth in the README File, including but not limited to the Java Technology Restrictions and Limitations on Redistribution of these Supplemental Terms, Oracle grants you a non-exclusive, non-transferable, limited license without fees to reproduce and distribute those files specifically identified as redistributable in the README File ("Redistributables") provided that: (i) you distribute the Redistributables complete and unmodified, and only bundled as part of Programs, (ii) the Programs add

significant and primary functionality to the Redistributables, (iii) you do not distribute additional software intended to supersede any component(s) of the Redistributables (unless otherwise specified in the applicable README File), (iv) you do not remove or alter any proprietary legends or notices contained in or on the Redistributables, (v) you only distribute the Redistributables pursuant to a license agreement that: (a) is a complete, unmodified reproduction of this Agreement; or (b) protects Oracle's interests consistent with the terms contained in the Agreement and includes the notice set forth in Section H, (vi) you agree to defend and indemnify Oracle and its licensors from and against any damages, costs, liabilities, settlement amounts and/or expenses (including attorneys' fees) incurred in connection with any claim, lawsuit or action by any third party that arises or results from the use or distribution of any and all Programs and/or Software. The license set forth in this Section D does not extend to the Software identified in Section G.

- E. DISTRIBUTION BY PUBLISHERS. This section pertains to your distribution of the Java™ SE Development Kit Software (“JDK”) with your printed book or magazine (as those terms are commonly used in the industry) relating to Java technology (“Publication”). Subject to and conditioned upon your compliance with the restrictions and obligations contained in the Agreement, Oracle hereby grants to you a non-exclusive, nontransferable limited right to reproduce complete and unmodified copies of the JDK on electronic media (the “Media”) for the sole purpose of inclusion and distribution with your Publication(s), subject to the following terms: (i) You may not distribute the JDK on a stand-alone basis; it must be distributed with your Publication(s); (ii) You are responsible for downloading the JDK from the applicable Oracle web site; (iii) You must refer to the JDK as Java™ SE Development Kit; (iv) The JDK must be reproduced in its entirety and without any modification whatsoever (including with respect to all proprietary notices) and distributed with your Publication subject to a license agreement that is a complete, unmodified reproduction of this Agreement; (v) The Media label shall include the following information: “Copyright [YEAR], Oracle America, Inc. All rights reserved. Use is subject to license terms. ORACLE and JAVA trademarks and all ORACLE- and JAVA-related trademarks, service marks, logos and other brand designations are trademarks or registered trademarks of Oracle in the U.S. and other countries.” [YEAR] is the year of Oracle's release of the Software; the year information can typically be found in the Software's “About” box or screen. This information must be placed on the Media label in such a manner as to only apply to the JDK; (vi) You must clearly identify the JDK as Oracle's product on the Media holder or Media label, and you may not state or imply that Oracle is responsible for any third-party software contained on the Media; (vii) You may not include any third party software on the Media which is intended to be a replacement or substitute for the JDK; (viii) You agree to defend and indemnify Oracle and its licensors from and against any damages, costs, liabilities, settlement amounts and/or expenses (including attorneys' fees) incurred in connection with any claim, lawsuit or action by any third party that arises or results from the use or distribution of the JDK and/or the

Publication; ; and (ix) You shall provide Oracle with a written notice for each Publication; such notice shall include the following information: (1) title of Publication, (2) author(s), (3) date of Publication, and (4) ISBN or ISSN numbers. Such notice shall be sent to Oracle America, Inc., 500 Oracle Parkway, Redwood Shores, California 94065 U.S.A , Attention: General Counsel.

- F. JAVA TECHNOLOGY RESTRICTIONS. You may not create, modify, or change the behavior of, or authorize your licensees to create, modify, or change the behavior of, classes, interfaces, or subpackages that are in any way identified as "java", "javax", "sun", "oracle" or similar convention as specified by Oracle in any naming convention designation.
- G. LIMITATIONS ON REDISTRIBUTION. You may not redistribute or otherwise transfer patches, bug fixes or updates made available by Oracle through Oracle Premier Support, including those made available under Oracle's Java SE Support program.
- H. COMMERCIAL FEATURES NOTICE. For purpose of complying with Supplemental Term Section C.(v)(b) and D.(v)(b), your license agreement shall include the following notice, where the notice is displayed in a manner that anyone using the Software will see the notice:

Use of the Commercial Features for any commercial or production purpose requires a separate license from Oracle. "Commercial Features" means those features identified Table 1-1 (Commercial Features In Java SE Product Editions) of the Java SE documentation accessible at <http://www.oracle.com/technetwork/java/javase/documentation/index.html>

- I. SOURCE CODE. Software may contain source code that, unless expressly licensed for other purposes, is provided solely for reference purposes pursuant to the terms of this Agreement. Source code may not be redistributed unless expressly provided for in this Agreement.
- J. THIRD PARTY CODE. Additional copyright notices and license terms applicable to portions of the Software are set forth in the THIRDPARTYLICENSEREADME file accessible at <http://www.oracle.com/technetwork/java/javase/documentation/index.html>. In addition to any terms and conditions of any third party opensource/freeware license identified in the THIRDPARTYLICENSEREADME file, the disclaimer of warranty and limitation of liability provisions in paragraphs 4 and 5 of the Binary Code License Agreement shall apply to all Software in this distribution.
- K. TERMINATION FOR INFRINGEMENT. Either party may terminate this Agreement immediately should any Software become, or in either party's opinion be likely to become, the subject of a claim of infringement of any intellectual property right.
- L. INSTALLATION AND AUTO-UPDATE. The Software's installation and auto-update processes transmit a limited amount of data to Oracle (or its service provider) about those specific processes to help Oracle understand and optimize

them. Oracle does not associate the data with personally identifiable information. You can find more information about the data Oracle collects as a result of your Software download at <http://www.oracle.com/technetwork/java/javase/documentation/index.html>.

For inquiries please contact: Oracle America, Inc., 500 Oracle Parkway,
Redwood Shores, California 94065, USA.

Last updated 02 April 2013

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 Log4j

Copyright © 1999-2014 The Apache Software Foundation

This product includes software developed at The Apache Software Foundation (<http://www.apache.org/>).

ResolverUtil.java

Copyright 2005-2006 Tim Fennell

Dumbster SMTP test server

Copyright 2004 Jason Paul Kitchen

TypeUtil.java

Copyright 2002-2012 Ramnivas Laddad, Juergen Hoeller, Chris Beams

libbson

Copyright © 2013-2015 MongoDB, Inc.

MongoDB C Driver

Copyright © 2013-2015 MongoDB, Inc.

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.