

Configuring ODBC on Linux

ODBC driver managers use configuration files to define and configure ODBC data sources and drivers. To configure an ODBC connection for Linux, complete the following steps:

- [Step 1: Set Environment Variables](#)
- [Step 2: Define the ODBC Data Sources in `odbc.ini`](#)
- [Step 3: \(Optional\) Define the ODBC Driver in `.odbcinst.ini`](#)
- [Step 4: Configure the Drill ODBC Driver](#)

Sample Configuration Files

Before you connect to Drill through an ODBC client tool on Linux, copy the following configuration files in `/opt/mapr/drill/Setup` to your home directory unless the files already exist in your home directory:

- `mapr.drillodbc.ini`
- `odbc.ini`
- `odbcinst.ini`

In your home directory, rename the files as hidden files. Use `sudo` if necessary:

- `.mapr.drillodbc.ini`
 - `.odbc.ini`
 - `.odbcinst.ini`
-

Step 1: Set Environment Variables

1. Set the `ODBCINI` environment variable to point to the `.odbc.ini` in your home directory.

Example: `export ODBCINI=~/.odbc.ini`

2. Set the `MAPRDRILLINI` environment variable to point to `.mapr.drillodbc.ini` in your home directory.

Example: `export MAPRDRILLINI=~/.mapr.drillodbc.ini`

3. Set the `LD_LIBRARY_PATH` environment variable to point to your ODBC driver manager libraries.

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

You can have both 32- and 64-bit versions of the driver installed at the same time on the same computer.

Important

Do not include the paths to both 32- and 64-bit shared libraries in `'LD_LIBRARY_PATH'` at the same time.

Only include the path to the shared libraries corresponding to the driver matching the bitness of the client application you use to access Drill.

Step 2: Define the ODBC Data Sources in `.odbc.ini`

Define the ODBC data sources in the `~/.odbc.ini` configuration file for your environment. To use Drill in embedded mode, set the following properties:

```
ConnectionType=Direct
HOST=localhost
PORT=31010
ZKQuorum=
ZKClusterID=
```

To use Drill in distributed mode, set the following properties. (These properties are described in detail in the [Direct and ZooKeeper Quorum Properties](#) section.)

```
ConnectionType=ZooKeeper
HOST=
PORT=
```

```
ZKQuorum=<host name>:<port>,<host name>:<port> . . . <host name>:<port>
ZKClusterID=<cluster name in `drill-override.conf`>
```

The following Linux sample shows a possible configuration for using Drill in distributed mode.

```
[ODBC]
Trace=no

[ODBC Data Sources]
MapR Drill 32-bit=MapR Drill ODBC Driver 32-bit
MapR Drill 64-bit=MapR Drill ODBC Driver 64-bit

[MapR Drill 32-bit]
# This key is not necessary and only describes the data source.

# Description=MapR Drill ODBC Driver (32-bit) DSN
Description=MapR Drill ODBC Driver (32-bit) DSN

# Driver: The location where the ODBC driver is installed to.
Driver=/opt/mapr/drill/lib/32/libdrillodbc_sb32.so

# The DriverUnicodeEncoding setting is only used for SimbaDM
# When set to 1, SimbaDM runs in UTF-16 mode.
# When set to 2, SimbaDM runs in UTF-8 mode.
# DriverUnicodeEncoding=2

# Values for ConnectionType, AdvancedProperties, Catalog, Schema should be set here.
# If ConnectionType is Direct, include Host and Port. If ConnectionType is
ZooKeeper, include ZKQuorum and ZKClusterID
# They can also be specified on the connection string.
# AuthenticationType:No authentication;Plain;Kerberos;
ConnectionType=Direct
HOST=[HOST]
PORT=[PORT]
ZKQuorum=[Zookeeper Quorum]
ZKClusterID=[Cluster ID]
AuthenticationType=No Authentication
UID=[USERNAME]
PWD=[PASSWORD]
DelegationUID=
KrbServiceHost=mapr
KrbServiceName=
AdvancedProperties=CastAnyToVarchar=true;HandshakeTimeout=5;QueryTimeout=180;TimestampTZDisplayTimezone=

Catalog=DRILL
Schema=

[MapR Drill 64-bit]
# This key is not necessary and only describes the data source.
Description=MapR Drill ODBC Driver (64-bit) DSN

# Driver: The location where the ODBC driver is installed to.
Driver=/opt/mapr/drill/lib/64/libdrillodbc_sb64.so

# The DriverUnicodeEncoding setting is only used for SimbaDM
# When set to 1, SimbaDM runs in UTF-16 mode.
# When set to 2, SimbaDM runs in UTF-8 mode.
# DriverUnicodeEncoding=2

# Values for ConnectionType, AdvancedProperties, Catalog, Schema should be set here.
# If ConnectionType is Direct, include Host and Port. If ConnectionType is
ZooKeeper, include ZKQuorum and ZKClusterID
# They can also be specified on the connection string.
# AuthenticationType:No authentication;Plain;Kerberos;
ConnectionType=Direct
HOST=[HOST]
PORT=[PORT]
```

```

ZKQuorum=[Zookeeper Quorum]
ZKClusterID=[Cluster ID]
AuthenticationType=No Authentication
UID=[USERNAME]
PWD=[PASSWORD]
DelegationUID=
KrbServiceHost=mapr
KrbServiceName=
AdvancedProperties=CastAnyToVarchar=true;HandshakeTimeout=5;QueryTimeout=180;TimestampTZDisplayTimezone=
Catalog=DRILL
Schema=

```

Authentication Properties

If the Drillbit requires authentication, uncomment the `AuthenticationType`, add an `AuthenticationType`, and configure properties. If the Drillbit does not require authentication (or to configure no password protection), you can use the `No Authentication` option. You do not need to configure additional settings.

- **Kerberos**
 - See the [MIT Kerberos](#) documentation for installing and configuring a Kerberos environment, which is beyond the scope of the information provided here.
 - To specify the Kerberos mechanism:
 - Set the `AuthenticationType` to `Kerberos`.
 - Set the `KrbServiceHost` property to the FQDN of the Drill server host.
 - Set the `KrbServiceName` property to the Kerberos service principal name of the Drill server.
- **Plain (or Basic Authentication)**
 - Configure the `UID` to an appropriate name for accessing the Drill server.
 - Set the `PWD` property to the password corresponding to the `UID`.

Direct and ZooKeeper Quorum Properties

To use Drill in distributed mode, set `ConnectionType` to `Zookeeper`, get the `ZKQuorum` and `ZKClusterID` values from the `drill-override.conf` file, and define the `ZKQuorum` and `ZKClusterID` properties. The `drill-override.conf` is in the `/drill/drill-<version>/conf` directory. Format `ZKQuorum` as a comma separated list of ZooKeeper nodes in the following format:

```
<host name/ip address> : <port number>, <host name/ip address> : <port number>, . . .
```

For example:

- `ZKQuorum=centos23:5181,centos28:5181,centos29:5181`
- `ZKClusterID=docs41cluster-drillbits`

To use Drill in embedded mode, do not define the `ZKQuorum` and `ZKClusterID` properties. Start Drill using the `drill-localhost` command, set `ConnectionType` to `Direct`, and define `HOST` and `PORT` properties. For example:

- `HOST=<IP address of drillbit>:5181`
- `PORT=31010`

[Driver Configuration Options](#) describes configuration options available for controlling the behavior of DSNs using the Drill ODBC Driver.

Step 3: (Optional) Define the ODBC Driver in .odbcinst.ini

The `.odbcinst.ini` is an optional configuration file that defines the ODBC Drivers. This configuration file is optional because you can specify drivers directly in the `.odbc.ini` configuration file. The `.odbcinst.ini` file contains the following sample configurations.

Example

```

[ODBC Drivers]
MapR Drill ODBC Driver 32-bit=Installed
MapR Drill ODBC Driver 64-bit=Installed

[MapR Drill ODBC Driver 32-bit]
Description=MapR Drill ODBC Driver (32-bit)
Driver=/opt/mapr/lib/32/libdrillodbc_sb32.so

```

```
[MapR Drill ODBC Driver 64-bit]
Description=MapR Drill ODBC Driver (64-bit)
Driver=/opt/mapr/lib/64/libdrillodbc_sb64.so
```

Step 4: Configure the Drill ODBC Driver

Configure the Drill ODBC Driver for your environment by modifying the `.mapr.drillodbc.ini` configuration file. This configures the driver to work with your ODBC driver manager. The following sample shows a possible configuration, which you can use as is if you installed the default iODBC driver manager.

Example

```
. . .
[Driver]
DisableAsync=0
ErrorMessagesPath=/opt/mapr/drill/ErrorMessage
LogLevel=0
LogPath=[LogPath]
SwapFilePath=/tmp

## - Note that the path to your ODBC Driver Manager must be specified in
LD_LIBRARY_PATH.

. . .
```

Configuring `.mapr.drillodbc.ini`

To configure the Drill ODBC Driver in the `.mapr.drillodbc.ini` configuration file, complete the following steps:

1. Open the `.mapr.drillodbc.ini` configuration file in a text editor.
2. Edit the `DisableAsync` setting if you want to enable a synchronous ODBC connection for performance reasons. Change the default 0 to 1 to disable the asynchronous and enable the synchronous connection. A change in state occurs during driver initialization and is propagated to all driver DSNs.

Note: As of version 1.3.8 of the driver, the `DriverManagerEncoding` setting is automatically detected and set if necessary. The value depends on the driver manager used; it's typically UTF-16 or UTF-32. iODBC uses UTF-32 and unixODBC uses UTF-16.
3. Save the `.mapr.drillodbc.ini` configuration file.

Next Step

[Testing the ODBC Connection](#)