Interaction Server Deployment Guide

Installing ODBC on Linux

12/19/2025

Contents

Installing ODBC on Linux

These directions apply primarily to Linux. Directions on deploying ODBC drivers for Windows are available on this page in the Interaction Server Deployment Guide.

Install unixODBC

Download

For compatibility purposes, the easiest and fastest way to do this is to download a prebuilt RPM or package that is compatible with your current Linux version.

> ### Important
> Microsoft ODBC driver 11 for SQL Server (Linux) is compatible with version 2.3.0 of unixODBC Driver Manager only. Genesys recommends using the latest version of ODBC Driver. For more information, see Installing the Driver Manager.

You can also

- Download, build and install unixODBC from the source provided at http://www.unixodbc.org/. Follow the instructions on the page.
- Build your own rpm. Information is at https://fedoraproject.org/wiki/How_to_create_an_RPM_package

> ### Important
> Genesys has no responsibility or license for unixODBC, which is a third-party package. Any and all of these unixODBC installation options are merely recommendations, and it is your responsibility to the choose one that fits your purpose best. There are dependencies for each of these options. Also, environments with alternative third-party drivers, database accelerators and ODBC Managers are not supported on compatibility issues.

Check

Once you have downloaded and installed unixODBC, check the **odbcinst.ini** (under **/etc** in the installation directory) and **odbc.ini** (in **$HOME/.odbc.ini**) files. This guide will refer to the unixODBC installation directory as the environment variable $UNIXODBC.

Install ODBC drivers

Oracle

Install the ODBC Driver

1. Download the latest version of the ODBC driver from http://www.oracle.com/technetwork/topics/ linuxx86-64soft-092277.html. You will need these two packages:

   - Instant Client Package - Basic: All files required to run OCI, OCCI, and JDBC-OCI applications

   - Instant Client Package - ODBC: Additional libraries for enabling ODBC applications

2. Unzip both packages to the same directory, which this guide will refer to as environment variable $DRIVER.

3. From the $DRIVER directory, run the following:

   ```
   ./odbc_update_ini.sh $UNIXODBC  $DRIVER [<Driver_Name>] [<DSN>]
   ```

   For example,

   ```
   ./odbc_update_ini.sh $UNIXODBC $DRIVER oracleodbc-12.1 ora
   ```

4. Set the three environment variables ORACLE_HOME, LD_LIBRARY_PATH, and TNS_ADMIN.

   - TNS_ADMIN is the directory containing the **TNSNAMES.ora** file, which describes Oracle connections. If such a file does not exist, you must create it.

   - ORACLE_HOME is the directory where the **bin** and **lib** directories are located for the Oracle client.

   - LD_LIBRARY_PATH is the load library path.

   For example,

   ```
   export ORACLE_HOME=$DRIVER
    export LD_LIBRARY_PATH= $UNIXODBC/lib/:$DRIVER/
    export TNS_ADMIN=/etc/oracle
   ```

Sample

This is a sample of **odbc.ini** with a configured DSN:

```
[ora]
Application Attributes = T
Attributes = W
BatchAutocommitMode = IfAllSuccessful
BindAsFLOAT = F
CloseCursor = F
DisableDPM = F
DisableMTS = T
Driver = OracleODBC-12.1
DSN = ora
EXECSchemaOpt =
EXECSyntax = T
Failover = T
FailoverDelay = 10
FailoverRetryCount = 10
FetchBufferSize = 64000
```

```
ForceWCHAR = F
Lobs = F
Longs = T
MaxLargeData = 0
MetadataIdDefault = F
QueryTimeout = T
ResultSets = T
ServerName = GENESYS_INX
SQLGetData extensions = F
Translation DLL =
Translation Option = 0
DisableRULEHint = T
UserID = SYSTEM
Password = system
StatementCache=F
CacheBufferSize=20
UseOCIDescribeAny=F
MaxTokenSize=8192
```

This is a sample of **odbcinst.ini:**

```
[OracleODBC-12.1]
Description=Oracle ODBC driver for Oracle 12g
Driver=/usr/lib/oracle/12.1/client64/lib/libsqora.so.12.1
Setup=
FileUsage=
CPTimeout=
CPReuse=
Driver Logging=7
```

In this example, the DSN name is ora. ServerName is the corresponding service name in **TNSNAMES.ora:**

```
GENESYS_INX =
  (DESCRIPTION =
    (ADDRESS = (PROTOCOL = TCP)(HOST = fakehost)(PORT = 1521))
    (CONNECT_DATA =
      (SERVER = DEDICATED)
      (SERVICE_NAME = genesys_inx)
    )
  )
```

MS SQL

> **Important**
>
> MS SQL officially supports only version 2.3.0 of unixODBC

1. Run

   ```
   export LD_LIBRARY_PATH= $UNIXODBC/lib
   ```

2. Follow the installation instructions given at http://www.microsoft.com/en-us/download/details.aspx?id=36437

If you want to use the MSSQL Driver with unixODBC version 2.3.1 or greater, you must perform the installation with the flag `-force`, and afterwards you must do the following:

1. Locate the file **libodbcinst.so.2** that was installed by unixODBC.

2. Make a symbolic link to it:

```
sudo ln -s libodbcinst.so.2 usr/lib64/libodbcinst.so.1
```

Odbcinst.ini should now contain the following:

```
[ODBC Driver 11 for SQL Server]
Description=Microsoft ODBC Driver 11 for SQL Server
Driver=/opt/microsoft/msodbcsql/lib64/libmsodbcsql-11.0.so.2270.0
Threading=1
UsageCount=1
```

3. Append a DSN entry name to **odbc.ini,** for example:

```
[sqltest]
Driver = ODBC Driver 11 for SQL Server
DSN = sqltest
Trace = No
ServerName = fakehost
UserID = genesys
Password = genesys
Database = genesys_inx
```

PostgreSQL

Information about PostgreSQL is provided in the Framework Database Connectivity Reference Guide and at http://www.postgresql.org/.

Test the Connection

To test the connection, run the command

```
isql -v [DSN name]
```

If the proper credentials are not configured in the **odbc.ini** file, you may have to add them as parameters of this command; for example,

```
isql -v [DSN name] [username] [password]
```

Related information can be found on the pages linked to this page in the Integrated Capture Points documentation.