Infor LN ODBC Driver Administration Guide

Publication information

Infor LN Release: 10.5

Publication Date: September 5, 2016

Document code: esinstlndbcag

Contents

About this guide

This document describes the administration of the Infor LN ODBC Driver for these connectors:

- Database Connection Driver for ODBC
- Database Connection Driver for JDBC

Note that Infor LN ODBC Driver supports earlier Infor product versions, such as Infor Baan IVc and Infor Baan 5.0. That is why the term Infor ERP is used in this document.

Intended audience

This document is intended for Infor ERP Enterprise database administrators and system administrators.

Related documents

You can find the documents in the product documentation section of the Infor Xtreme Support portal, as described in "Contacting Infor".

- *Infor Enterprise Server Installation Guide*

Contacting Infor

If you have questions about Infor products, go to the Infor Xtreme Support portal.

If we update this document after the product release, we will post the new version on this website. We recommend that you check this website periodically for updated documentation.

If you have comments about Infor documentation, contact documentation@infor.com.

Introduction

1

Infor LN ODBC Driver is a new driver that replaces the Infor Integration's ODBC/JDBC driver.

Infor Integration was built with Java 1.1 tools and Visual Studio 2000. It was a 32 bit application. Infor Integration was installed as a client application together with DBConnection (Infor ES Database Connector), which is a small part of the portingset.

Most of the Infor Integration functionality became obsolete, except the functionality for the ODBC/JDBC-drivers.

All irrelevant functionality is removed from the Infor Integration software to create a new driver. The Infor LN ODBC Driver supports:

- 32-bit
- 64-bit
- Unicode

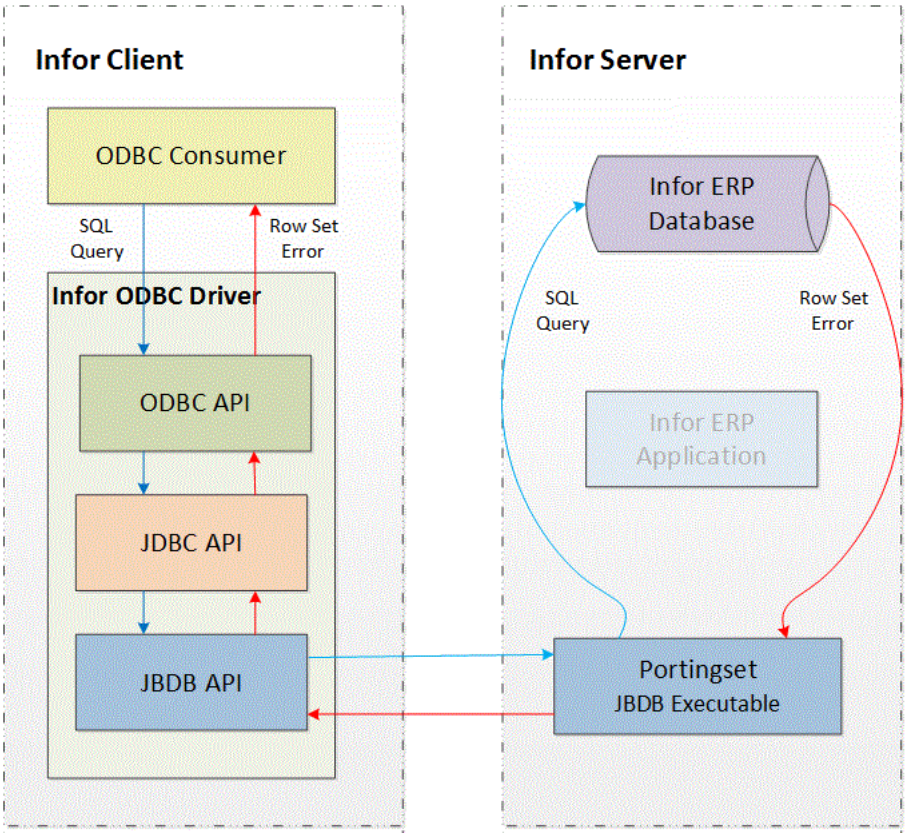
Driver for ODBC

Consider the Infor LN ODBC Driver as the ODBC driver for Infor servers. This ODBC driver exposes databases on Infor servers as ODBC data sources to Infor clients on Windows platforms. The LNDBC can be used on Windows platforms and is restricted to read-only access of a database on an Infor ERP server.

The connection between the client and the server is established through LN database connection driver software. On the client side, the LNDBC forms the interface between the ODBC API and JBDB API that communicate directly with JBDB of the portingset on the ERP server.

On the server side, Infor ES Porting Set establishes the connection between JBDB and the Infor ERP database. JBDB is the database driver that converts the Database Model messages into SQL queries on the Infor ERP database. Vice versa, JBDB converts the response of the database to Database Model messages. Note that the JBDB on the ERP Server determines the syntax of the SQL commands the ODBC consumer can use on the client.

This diagram illustrates the architecture of an integration in which an ODBC client queries an Infor ERP Server, using the LN ODBC Driver.



An ODBC-compliant client application (an ODBC consumer) uses the Connector for ODBC to call an Infor Server in this way:

- Connect to a data source
- Submit a SQL query for execution (call ODBC API)
- Retrieve results (a record set or an error)
- Disconnect data source

Scenario for ODBC integrations

The distribution scenarios can be identified as:

- Infor LN ODBC Driver and Infor DB Driver on the same system.

  This is the most common scenario, only the portingset transport between Infor DB Driver and Database Driver is used.

- All components on the same system

  This scenario can only be used if the ODBC client application runs on the same system as the Enterprise Server application

Driver for JDBC

Besides ODBC also JDBC is supported. This is the Java variant of ODBC and can be used on Windows and UNIX systems.

On how to use JDBC, see "JDBC" on page 23.

HTML help for Infor LN ODBC Driver

The installer places the `lnodbc.chm` file in the destination folder that contains the HTML help for Infor LN ODBC Driver.

MBCS vs Unicode

The ODBC driver for Infor Integration did not support Unicode. The LN ODBC Driver does. You can use it with both MBCS clients and Unicode clients. If the database contains Unicode characters we recommend that you use a Unicode client.

Prerequisites

2

Install java 32-bit and java 64 bit version 1.8 before you install ODBC Driver.

Installing the Infor LN ODBC Driver

3

Use the Infor Enterprise Server Installer to install the 32-bit and 64-bit Infor LN ODBC Driver. One ODBC system DSN will be configured for both drivers with the same parameters.

See *Infor Enterprise Server Installation Guide*.

Installation process

If you plan to use a BaanLoginSSL connection create the keystore or truststore first. See "Creating Keystore (and Truststore) for BaanLogin SSL connection" on page 13.

To install the Infor LN ODBC driver:

1. Start the installer.
2. Select Infor LN ODBC Driver
3. Click **Next**.
4. Specify the connect information to the LN Enterprise Server host.
   See online help for the connection parameters.
   After the installation you can add more connections, see the online help.

Creating Keystore (and Truststore) for BaanLogin SSL connection

A keystore is required for a BaanLoginSSL connection.

To define the mutual SSL trust certificates for the BaanLogin SSL connection, you must create two keystores. For example:

- `lndbc.jks`
- `ln-server.p12`

To create a keystore for the lndbc server on the lndbc server:

1 Run this command:

```
keytool -v -genkeypair -keystore <lndbc.jks>
    -keypass changeit -storepass <changeit> -storetype JKS -alias
    <lndbc-server> -dname CN=<lndbc server.mydomain.com> -validity
7300
    -keyalg RSA -keysize 2048 -sigalg SHA256withRSA
```

Run the command in a folder with full access permissions. Change the names between **<>** to applicable names.

2 Retrieve the **<**`ln-server.p12`**>** file from the `<ln_server\bse\security>` folder. Place the file in the same folder as where `lndbc.jks` is created.

3 Run these commands:

```
keytool -v -exportcert -keystore <lndbc.jks>
    -storepass <changeit> -storetype JKS -alias <lndbc-server> -file

    <lndbc-cert.cer>
```

and

```
keytool -v -exportcert -keystore <ln-server.p12>
    -storepass <changeit> -storetype PKCS12 -alias <ln-server> -file

    <ln-server-cert.cer>
```

4 Import the certificate in 'other' keystore:

```
keytool -v -importcert -noprompt -keystore
    <lndbc.jks> -storepass <changeit> -keypass <changeit>
    -storetype JKS -alias <ln-server> -file
    <ln-server-cert.cer>
```

and

```
keytool -v -importcert -noprompt -keystore
    <ln-server.p12> -storepass <changeit> -keypass <changeit>
    -storetype PKCS12 -alias <lndbc-server> -file
    <lndbc-cert.cer>
```

5 Place the updated `ln-server.p12` file back in the folder:`ln_server\bse\security`

The self-signed certificates are generated with a validity of several years.

During installation or later in the odbc administrator you can set the keystore and truststore locations to the `lndbc.jks`.

Set their passphrases to *<changit>*

You can start the odbc administrator in one of these folders:

- `<WinDir>\system32\odbcad32.exe`
- `<WinDir>\syswow64\odbcad32.exe`

ODBC configuration

The ODBC configuration is done during installation. You can add more DSNs later.

To test the configuration

1. Start the ODBC Data Source Administrator.

   You can find the `odbcad32.exe` file for the 64 bit version in:

   `C:\windows\system`

   For the 32 bit version go to:

   `C:\windows\syswow64`
2. Select the **System DSN** tab.
3. Select the driver you want to test and click **Configure**.
4. Click **Test Connection** to validate the connection.

   A connection check is done with this statement:

   ```
   select * from ttaad100 where company_nr=0
   ```

   Then data for columns 1, 2 and 5 is retrieved: the name, description and package combination. Only three rows are shown.

Runtime

4

In your LNDBC destination you can find documentation about the use of the LN Database Connection Driver and run a program to test the installation.

Your LNDBC destination folder is for example:

`C:\Infor\ese\lndbc\lndbc`

Open a command prompt and run the `odbctest` (32/64)(A) file to test the installation.

You can use one of these executables:

- `odbctest64.exe`: Tests the odbc driver with a 64-bit unicode application.
- `Odbctest64a.exe`: Tests the odbc driver with a 64-bit MBCS application.
- `odbctest32.exe`: Tests the odbc driver with a 32-bit unicode application.
- `Odbctest32a.exe`: Tests the odbc driver with a 32-bit MBCS application.

Running one of these excutables from a command prompt without parameters will show you the usage.

Open the `lnodbc.chm` file for the documentation.

Logging

5

If you are not using a BaanLoginSSL connection and the windows variable `LNDBC_USERHOME` is not set, you can find log files in the user folder.

For example:

`C:\Users\<your name>\.lndbclog`

If you are using a BaanLoginSSL connection, you must set the windows variable `LNDBC_USERHOME`. Specify a folder with write permissions for all LNDBC users, for example:

`LNDBC_USERHOME=c:\lndbc`

The log files look similar to:

`C:\lndbc\.lndbclog\`*`jdoe_ODBC_Driver_1472573769455`*`.log`

For each user a file is created with the username, application name and a number that increases every time a log file for this user is created. The format is <username_appname_number>.log

Two log files are created if:

- An error occurs before connection and the log level is set to 0
- The log level in `lndbc-cfg.properties` is set to 1 or 2

The first log file contains logging before the user connects to the database. The second log file contains all logging after the user connects.

ODBC tracing

To activate tracing:

1 Start the ODBC Data Source Administrator.

For 32 bit drivers double-click `odbcad32.exe` in:

`C:\windows\syswow64\`

For 64 bit drivers double-click `odbcad32.exe` in:

`C:\windows\system32\`

2 Click the **System DSN** tab.
3 Select **Data Source Infor_LN_ODBC**.
4 Click the **Tracing** tab.
5 In the Custom Trace dll for 32 bit you must specify: `c:\windows\syswow64\odbctrac.dll`
Changes for 64 bit drivers are not required.
6 Click **Start Tracing Now**.
7 Run the application that connects to the database through ODBC and run a query.
8 Click **Stop Tracing**.
You will find the ODBC logging in the log file path as specified in the **Tracing** tab of the ODBC Data Source Administrator.

LNDBC Tracing

After running a query for the first time, the `lndbc-cfg.properties` file is created in either:

`C:\Users\<yourname>\AppData\Roaming\Infor\lndbc\lndbc-cfg.properties`

or:

`<LNDBC_USERHOME>\lndbc\lndbc-cfg.properties`

1 Open the `lndbc-cfg.properties` file.
2 Set Loglevel= 0, 1 or 2

```
lndbc-cfg.properties:
# Loglevel
# Loglevel=0         logs only errors
# Loglevel=1         logs also information
# Loglevel=2         logs all
# Perprocess
# Perprocess=true    each connection has its own logfile
# Perprocess=false   all connections log in the same file
Loglevel=2
Perprocess=true
```

Instead of setting the log level in `lndbccfg.properties`, you can add the LOG_LEVEL parameter to the connection string when connecting to LNDBC. For example: LOG_LEVEL=2. This overrules the log level as set in the `lndbccfg.properties` file.

Online help

6

The destination folder of LN ODBC Driver contains more extensive documentation about the use of LN ODBC Driver.

1 Navigate to, for example:

```
C:\Infor\ese\lndbc\lndbc
```

2 Open the file `lnodbc.chm`

JDBC

7

JDBC is the java variant of ODBC. A JDBC Data Source Administrator does not exist. A client java application must be written to call the jdbc functions.

An example URL for an On-premise system:

```
jdbc:infor://BSE_PROTOCOL=BaanLogin;BSE_PORT=7150;BSE_HOSTNAME=
<hostname>;BSE=<bse>;
BSE_OSUSER=<name>;BSE_OSPASSWORD=<password>
```

Or

```
jdbc:infor://BSE_PROTOCOL=BaanLoginSSL;BSE_PORT=7150;BSE_HOSTNAME=
<hostname>;BSE=<bse>;KEYSTORE=<key
store location + file
name>;KEYSTORE_PASSPHRASE=<password>;TRUSTSTORE=<trust
store location + file
name>;TRUST_STORE_PASSPHRASE=<password>;BSE_UPN=<UPN>>;MULTIPLE_DOMAINS=
1
```

Or

```
jdbc:infor://BSE_PROTOCOL=BaanLoginSSL;BSE_PORT=7150;BSE_HOSTNAME=
<hostname>;BSE=<bse>;KEYSTORE=<key
store location + file
name>;KEYSTORE_PASSPHRASE=<password>;TRUSTSTORE=<trust
store location + file
name>;TRUST_STORE_PASSPHRASE=<password>;BSE_SAM_USER=<SAM User>;
MULTIPLE_DOMAINS=0
```

An example URL for a Multitenant system:

```
jdbc:infor://BSE_PROTOCOL=BaanLoginSSL;BSE_PORT=7150;BSE_HOSTNAME=
<hostname>;BSE=<bse>;KEYSTORE=<key
store location + file
name>;KEYSTORE_PASSPHRASE=<password>;TRUSTSTORE=<trust
store location + file
```

```
name>;TRUST_STORE_PASSPHRASE=<password>BSE_IDENTITY2=<identity2>;BSE_
TENANT=<tenant>
```

A simplified example on how to execute a query.

To activate the JDBC driver, the JDBC client must call:

```
Class.forName("com.infor.lndbc.jdbc.JDBCDriver")
```

The JDBC client receives a Connection instance by calling:

```
Connection connection =
    DriverManager.getConnection(url)
```

The URL can look similar to this:

```
 jdbc:infor://BSE_HOSTNAME=<hostname>;BSE=<bse>;BSE_PROTOCOL=baanlogin;
BSE_PORT=7150;BSE_OSUSER=<name>;BSE_OSPASSWORD=<password>
```

A query can look similar to this:

```
String queryText = "select * from ttaad200 where
    company_nr=0"
```

To retrieve metadata from the connection:

```
DatabaseMetaData databaseMetaData =
connection.getMetaData();
boolean readOnly = databaseMetaData.isReadOnly();
System.out.println("databaseMetaData.isReadOnly()" +
readOnly);
```

To execute a query:

```
 Statement statement = connection.createStatement();
ResultSet resultSet = statement.executeQuery(queryText);
statement.close();
Object o = resultSet.getObject(1);
System.out.print(print(o.toString()));
```

See java.sql documentaton at: https://docs.oracle.com/javase/tutorial/jdbc.

Troubleshooting

8

During the installation you can run into issues. some of those issues and their solutions are explained here.

Updating java

The ODBC Driver uses java native interface. The `dll` for this can be found in:

`C:\Program Files\Java\jre<version>\bin\server\jvm.dll`

The windows PATH variable contains the location for this `jvm.dll`, for example:

`C:\Program Files\Java\jre1.8.0_73\bin\server`

If java is updated and the old version is removed, the new version number can be changed. This means that the `PATH` variable must be updated to represent the version.

For example for 64-bit ODBC drivers:

`C:\Program Files\Java\`*`jre1.8.0_74`*`\bin\server`

For example for 32-bit ODBC drivers:

`C:\Program Files (x86)\Java\`*`jre1.8.0_74`*`\bin\client`

When the driver cannot find the correct java version. A message is displayed:

```
SQLDriverConnect failed
IM003 (Specified driver could not be loaded
due to system error 126: The specified module could not be found.
(Infor LN ODBC Driver,
C:\Infor\ese\lndbc\lndbc\lnodbc.dll).) 160
```

See your windows documentation on how to update the `PATH` variable.

Installed files and settings

A

After installation the files and settings are installed for the ODBC Driver.

Infor LN ODBC driver

This table shows the components installed with the ODBC Driver:

Components	Description
`C:\Infor\ese\lndbc\lndbc odbctest64.exe`	File (Unicode enabled) to test the 64-bit ODBC connection. Run `odbctest64.exe` on a command prompt to see the options.
`C:\Infor\ese\lndbc\lndbc odbctest64A.exe`	File (No Unicode enabled) to test the 64-bit ODBC connection. Run `odbctest64A.exe` on a command prompt to see the options.
`C:\Infor\ese\lndbc\lndbc odbctest32.exe`	File (Unicode enabled) to test the 32-bit ODBC connection. Run `odbctest32.exe` on a command prompt to see the options
`C:\Infor\ese\lndbc\lndbc odbctest32A.exe`	File (Unicode enabled) to test the 32-bit ODBC connection. Run `odbctest32A.exe` on a command prompt to see the options
`C:\Infor\ese\lndbc\lndbc lndbcstd.dll C:\Infor\ese\lndbc\lndbc lndbc-std32.dll`	Dll to convert c++ function calls to java and vice versa for 64 bit and 32 bit respectively.
`C:\Infor\ese\lndbc\lndbc lnodbc.dll`	C++ driver for ODBC
`C:\Infor\ese\lndbc\lndbc lndbc.jar C:\Infor\ese\lndbc\lndbc lnodbc32.dll`	Java driver for JDBC/ODBC
`C:\Infor\ese\lndbc\lndbc lndbc3p.jar`	Third party components used by `lndbc.jar`

Components	Description
`C:\Infor\ese\lndbc\lndbc Jbdb2.jar`	Java component to connect with JBDB executable of the Infor ES Portingset.
`C:\Infor\ese\lndbc\lndbc lndbcinfo.bat`	Shows copyright and version information for the ODBC Driver (and JDBC driver).
`C:\Infor\ese\lndbc\lndbc lndbc-cfg.properties`	Contains configuration settings for logging.
`C:\Infor\ese\lndbc\lndbc lnodbc.chm`	Help file
`C:\Infor\ese\lndbc\lndbc lndbc-cfg.properties`	Contains configuration settings

Registry settings for Infor LN ODBC driver

These settings are specified:

- `HKEY_LOCAL_MACHINE\SOFTWARE\ODBC\ODBC.INI\Infor LN DBC`:
  - BSE <bse>
  - BSE_HOSTNAME <hostname>
  - BSE_OSPASSWORD <encrypted password>
  - BSE_OSUSER <user>
  - BSE_PORT 7150
  - BSE_PROPERTIES
  - BSE_PROTOCOL <protocol>
  - Driver C:\Infor\ese\lndbc\lndbc\lnodbc.dll
  - KEY_STORE <keystore>
  - KEYSTORE_PASSPHRASE <keystore passphrase>
  - TRUST_STORE <truststore>
  - TRUST_STORE_PASSPHRASE <trust store passphrase>
  - MULTIPLE_DOMAINS
- `HKEY_LOCAL_MACHINE\SOFTWARE\ODBC\ODBC.INI\ODBC Data Sources`:
  - Infor LN ODBC Infor LN ODBC Driver
- `HKEY_LOCAL_MACHINE\SOFTWARE\ODBC\ODBCINST.INI\Infor LN ODBC Driver`:
  - APILEvel:1
  - ConnectFunctions:YYY
  - Driver: C:\Infor\ese\lndbc\lndbc\lnodbc.dll
  - DriverODBCVer 03:00

- FileUsage : 0
- Setup: C:\Infor\ese\lndbc\lndbc\lnodbc.dll
- SQLLevel: 1
- UsageCount: 0x00000001 (1)

- `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\SessionManager\Environment`:
  - Path=%Path%; C:\Infor\ese\lndbc\lndbc;C:\Infor\ese\lndbc\lndbc; C:\Program Files\Java\jre1.8.0_77\bin\server;C:\Program Files (x86)\Java\jre1.8.0_77\bin\client
  - ClassPath= %ClassPath%; C:\Infor\ese\lndbc\lndbc; C:\Infor\ese\lndbc\lndbc\lndbc.jar; C:\Infor\ese\lndbc\lndbc\lndbc3p.jar

Glossary