

# Quick Start: DataDirect Connect® Series for ODBC Drivers on Linux Platforms with a Motif GUI

The Progress® DataDirect Connect® Series for ODBC includes the following products:

- DataDirect Connect for ODBC and DataDirect Connect XE for ODBC
- DataDirect Connect64 for ODBC and DataDirect Connect64 XE for ODBC

The following basic information enables you to connect with and test your driver immediately after installation. For installation instructions, see the [DataDirect Connect Series for ODBC Installation Guide](#). This Quick Start covers the following topics:

[Environment Setup](#)  
[Test Loading the Driver](#)  
[Connecting to a Database](#)  
[Testing the Connection](#)  
[Tuning the Drivers for Optimal Performance](#)

ODBCHOME in the following sections refers to your installation directory path determined at installation.

## Environment Setup

- 1 Check your permissions:** Log in as a user with full r/w/x permissions recursively on the entire DataDirect Connect Series for ODBC installation directory.
- 2 Determine which shell you are running:** From the login shell, execute the `echo $SHELL` command.
- 3 Run the DataDirect setup script to set variables:** Two scripts, `odbc.csh` and `odbc.sh`, are installed in the installation directory. For Korn, Bourne, and equivalent shells, execute `odbc.sh`. For a C shell, execute `odbc.csh`. After running the setup script, execute the `env` command to verify that the `ODBCHOME/lib` directory has been added to your shared library path.
- 4 Set ODBCINI variable:** DataDirect Connect Series for ODBC products install a default `odbc.ini` file, where your data sources reside, in the installation directory. You must set the `ODBCINI` environment variable to point to the path of the `odbc.ini` file. For example:

```
$ ODBCINI= /ODBCHOME/odbc.ini; export ODBCINI
```

## Test Loading the Driver

The `ivtestlib` (32-bit drivers) and `ddtestlib` (64-bit drivers) test loading tools are provided to verify that the driver can be loaded into memory; they are located in the `ODBCHOME/bin` directory. For example, to load the DB2 32-bit driver, you would enter:

```
$ ivtestlib ODBCHOME/lib/ivdb2nn.so
```

where `nn` represents the driver level number.

If the load is successful, the test loading tool returns a success message along with the version string of the driver. If the driver cannot be loaded, the test loading tool returns an error message explaining why.

# Connecting to a Database

The following procedure is applicable only to Linux systems using a Motif 2.0.3 GUI. The procedure describes using the DataDirect ODBC Data Source Administrator for Linux (the Linux ODBC Administrator).

- 1 To start the Linux ODBC Administrator, change to the ODBDHOME/tools directory. At a command prompt, enter:

```
./odbcadmin.
```

- 2 On the User DSN tab, click **Add** to display a list of installed drivers. Select the appropriate driver from the list and click **Finish** to display the driver Setup dialog box.
- 3 On the General tab, provide the required information; then, click **Apply**. Every General tab has the following options:
  - **Data Source Name:** Type a string that identifies this data source configuration in the system information, such as "Accounting."
  - **Description:** Type an optional long description of a data source name, such as "My Accounting Database."

Click the following links for specific information about the General tab of each driver:

[DB2 Wire Protocol](#)  
[Greenplum Wire Protocol](#)  
[Informix Wire Protocol](#)  
[MySQL Wire Protocol](#)

[Oracle](#)  
[Oracle Wire Protocol](#)  
[PostgreSQL Wire Protocol](#)  
[SQL Server Wire Protocol](#)

[Sybase Wire Protocol](#)  
[Driver for the Teradata Database](#)

## DB2 Wire Protocol

**IMPORTANT:** You must have the appropriate privileges for the driver to create and bind packages with your user ID. These privileges are BINDADD for binding packages, CREATEIN on the collection specified by the Package Collection option, and GRANT EXECUTE on the PUBLIC group for executing the packages. These are typically the permissions of a Database Administrator (DBA). If you do not have these privileges, someone that has a user ID with DBA privileges needs to create packages by connecting with the driver.

When connecting for the first time, the driver determines whether bind packages exist on the server. If packages do not exist, the driver creates them automatically using driver default values.

- **IP Address:** Type the IP address of the machine where the catalog tables are stored. Specify the address using the machine's numeric address or specify its host name. If you enter a host name, the driver must find this name (with the correct address assignment) in the HOSTS file on the workstation or in a DNS server. The default is localhost.
- **Tcp Port:** Type the port number that is assigned to the DB2 DRDA listener process on the server host machine. Specify either this port's numeric address or its service name. If you specify a service name, the driver must find this name (with the correct port assignment) in the SERVICES file on the workstation. The default is 50000.

On DB2 for i only, execute `NETSTAT` from a DB2 for i command line to determine the correct port number. Select option 3 to display a list of active ports on the DB2 for i machine. Find the entry for DRDA and press F-14 to toggle and display the port number. If DRDA is not currently listening, the DB2 for i command, `CHGDDMTCPA AUTOSTART(*YES) PWDRQD(*YES)` starts the listener and ensures that it is active at IPL.

- **Location Name:** This field is valid and required only if you are connecting to a DB2 database on DB2 for I or z/OS. Type the DB2 location name. Use the name defined during the local DB2 installation.

On z/OS only, your system administrator can determine the name of your DB2 location using the `DISPLAY DDF` command.

On DB2 for i only, your system administrator can determine the name of your DB2 location using the `WRKRDBDIRE` command. The name of the database that is listed as \*LOCAL is the value you should use.

NOTE: This field is disabled if the Database Name field is populated.

- **Collection:** This field is valid only if you are connecting to a DB2 database on DB2 for i z/OS. By default, the user ID is used for the value of Collection. The user ID should always be used on z/OS.

NOTE: This field is disabled if the Database Name field is populated.

- **Database Name:** This field is valid and required only if you are connecting to a DB2 database on Linux/UNIX/Windows. Type the name of the database to which you want to connect.

NOTE: This field is disabled if the Location Name or Collection fields are populated.

## Greenplum Wire Protocol

- **Host Name:** Type either the name or the IP address of the server to which you want to connect.
- **Port Number:** Type the port number of the server listener. The default is 5432.
- **Database Name:** Type the name of the database to which you want to connect by default.

## Informix Wire Protocol

- **Host Name:** Type either the name or the IP address of the server to which you want to connect.
- **Port Number:** Type the port number of the server listener.
- **Server Name:** Type the name of the Informix server as it appears in the sqlhosts file.
- **Database Name:** Type the name of the database to which you want to connect by default.
- **User Name:** Type your user name as specified on the Informix server.

## MySQL Wire Protocol

- **Host Name:** Type either the name or the IP address of the server to which you want to connect.
- **Port Number:** Type the port number of the server listener. The default is 3306.
- **Database Name:** Type the name of the database to which you want to connect by default.

## Oracle

IMPORTANT: You must have all components of your database client software installed and connecting properly; otherwise, the driver will not operate correctly.

Following is a list of connection options on the General Tab:

**Server Name:** Type the client connection string designating the server and database to be accessed. The information required varies depending on the client driver you are using.

## Oracle Wire Protocol

- **Host:** Type either the name or the IP address of the server to which you want to connect.

- **Port Number:** Type the port number of your Oracle listener. Check with your database administrator for the correct number.
- **SID:** Type the Oracle System Identifier that refers to the instance of Oracle running on the server. The default is `ORCL`.

This option and the Service Name option are mutually exclusive. If the Service Name option is specified, do not specify this option.

- **Service Name:** Type the Oracle service name that specifies the database used for the connection. The service name is a string that is the global database name—a name that is comprised of the database name and domain name, for example: `sales.us.acme.com`.

This option and the SID option are mutually exclusive. If the SID option is specified, do not specify this option.

- **Edition Name:** Oracle 11g R2 and higher only. Type the name of the Oracle edition that the driver is to use when establishing a connection. Oracle 11g R2 and higher allows your database administrator to create multiple editions of schema objects so that your application can still use those objects while the database is being upgraded. This option is only valid for Oracle 11g R2 and higher databases and tells the driver which edition of the schema objects to use.

## PostgreSQL Wire Protocol

- **Host Name:** Type either the name or the IP address of the server to which you want to connect.
- **Port Number:** Type the port number of the server listener. The default is `5432`.
- **Database Name:** Type the name of the database to which you want to connect by default.

## SQL Server Wire Protocol

- **Host Name:** Type either the name or the IP address of the server to which you want to connect.

If your network supports named servers, you can specify an address as: `server_name`. For example, you can enter `SSserver`.

You can also specify a named instance of Microsoft SQL Server. Specify this address as: `server_name\instance_name`. If only a server name is specified with no instance name, the driver uses the default named instance on the server.

- **Port Number:** Type the port number of the server listener. The default is `1433`.
- **Database Name:** Type the name of the database to which you want to connect by default.

## Sybase Wire Protocol

- **Network Address:** Type the IP address of the server to which you want to connect. Specify this address as: `IP_address, port_number`. For example, you can enter `199.226.224.34, 5000`.

If your network supports named servers, you can specify an address as: `server_name, port_number`. For example, you can enter `SybSserver, 5000`.

- **Database Name:** Type the name of the database to which you want to connect by default. If you do not specify a value, the default is the database defined by the system administrator for each user.

## Driver for the Teradata Database

**IMPORTANT:** You must have all components of your database client software installed and connecting properly; otherwise, the driver will not operate correctly.

Following is a list of items on the General Tab:

- **DBCName** or **Alias:** Type the IP address or the alias name of the Teradata Server. Using an IP address reduces the time it takes to connect, but if that address is not available at connection time, the connection fails and the driver does not attempt to fail over to another address.

Using an alias name increases the time it takes to connect because the driver must search a local hosts file to resolve the name to the IP address information, but it allows the driver to try and connect to alternate IP addresses if the first address fails. If you use an alias name, you must have or create a local hosts file that contains the alias names. The alias name cannot be more than eight characters long.

- **DBCName List:** Type the IP addresses or the alias names that are to appear in the drop-down list of the logon dialog box. Separate the names with commas. The same restrictions apply as described for the DBCName or Alias option.
- **Integrated Security:** Select this check box to enable the user to connect to the database through Single Sign On (SSO) using one of the authentication mechanisms that support SSO. When this check box is not selected (the default), UserID is required.
- **Security Mechanism:** Select TD2 from the drop-down list to specify the authentication mechanism used for connections to the data source.

Valid values are:

**Default**—uses TD2.

**KRB5**— uses Kerberos as the authentication mechanism on Windows clients working with Windows servers if the server is V2R6.0.

**KRB5C**— uses Kerberos Compatibility as the authentication mechanism on Windows clients working with Windows servers if the server is pre-V2R6.0.

**LDAP**—uses LDAP as the authentication mechanism.

**NTLM**— uses NTLM as the authentication mechanism on Windows clients working with Windows servers if the server is V2R6.0.

**NTLMC**— uses NTLM Compatibility as the authentication mechanism on Windows clients working with Windows servers if the server is pre-V2R6.0.

**TD1**—uses Teradata 1 as the authentication mechanism.

**TD2** (default)—uses Teradata 2 as the authentication mechanism.

- **Security Parameter:** Type a string of characters that is to be regarded as a parameter to the authentication mechanism. The string is ignored by the ODBC driver and is passed on to the TeraSSO function that is called to set the authentication mechanism. The characters [ ] { } ( ) , ; ? \* = ! @ must be enclosed in curly braces.
- **UserID:** Type the default UserID for the Teradata database.

## Testing the Connection

- 1 At the bottom of the Driver Setup dialog box, click **Test Connect** to attempt to connect to the data source using the connection properties that you specified. A logon dialog box appears. Note that the information you enter in the logon dialog box during a test connect is not saved. Click **OK**.

If the driver can connect, it releases the connection and displays a `connection established` message. If the driver cannot connect because of an improper environment or incorrect connection value, it displays an appropriate error message. Click **OK**.

- 2 Click **OK** or **Cancel** at the bottom of the Driver Setup dialog box. If you click **OK**, the values you have specified become the defaults when you connect to the data source.

## Tuning the Drivers for Optimal Performance

The drivers have connection options that directly affect performance. To tune the driver for optimal performance, run the DataDirect Connect Series *for* ODBC Performance Wizard, which is installed along with the product.

The Wizard leads you step-by-step through a series of questions about your application. Based on your answers, the Wizard provides the optimal settings for performance-related connection attributes.

The Wizard runs as an applet within a browser window. To start the Wizard that was installed with the product, launch the following file from your browser, where *installation\_directory* is your DataDirect Connect Series *for* ODBC installation directory:

```
installation_directory/wizards/index.html
```

NOTE: Security features set in your browser can prevent the Wizard from launching. Check with your system administrator before disabling any security features.

### Drivers Not Included in the Wizard

The Greenplum Wire Protocol, PostgreSQL Wire Protocol, and Driver for Teradata drivers are not included in the Performance Wizard. To tune these drivers for performance, set the following options:

#### Greenplum Wire Protocol Driver

If your application does not use threads	Disable <b>Application Using Threads</b> on the Advanced tab.
If you want to use DataDirect connection pooling	Enable <b>Connection Pooling</b> on the Pooling tab.
If your application is configured for high availability that replays queries after a failure	Controlled by <b>Failover Mode</b> on the Failover Tab. Although high availability that replays queries after a failure provides increased levels of protection, it can adversely affect performance because of increased overhead.

## PostgreSQL Wire Protocol Driver

If your application does not use threads	Disable <b>Application Using Threads</b> on the Advanced tab.
If you want to use DataDirect connection pooling	Enable <b>Connection Pooling</b> on the Pooling tab.
If your application is configured for high availability that replays queries after a failure	Controlled by <b>Failover Mode</b> on the Failover Tab. Although high availability that replays queries after a failure provides increased levels of protection, it can adversely affect performance because of increased overhead.
If your application requires encryption of data	Controlled by <b>Encryption Method</b> on the Security Tab. Data encryption may adversely affect performance because of the additional overhead (mainly CPU usage) required to encrypt and decrypt data.

© 2016. Progress Software Corporation. All rights reserved.

7/16, 7.1.6