Quick Start: DataDirect Connect® Series *for* ODBC Drivers on UNIX and Linux

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](#)*. If you use a Motif GUI on Linux, you can use the UNIX ODBC Administrator to configure drivers. Refer to *Quick Start: DataDirect Connect Series for ODBC Drivers on Linux using a Motif GUI.* 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 32-bit DB2 driver, you would enter:

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

where *nn* represents the driver level number and *zz* represents the extension.

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, test loading tool returns an error message explaining why.

Connecting to a Database

The following procedure is applicable to all UNIX/Linux systems.

Defining a Data Source in the odbc.ini

The default `odbc.ini` file installed in the installation directory is a template in which you create data source definitions. You enter your site-specific database connection information using a text editor. Each data source definition must include the keyword `Driver=`, which is the full path to the driver.

The following examples show the minimum connection string options that must be set to complete a test connection, where xx represents iv for 32-bit or dd for 64-bit drivers, nn represents the driver level number, and zz represents the extension. The values for the options are samples and are not necessarily the ones you would use.

Click the following links for specific information about the odbc.ini data source entry of each driver:

Driver for Apache Hive	MySQL Wire Protocol	SQL Server Wire Protocol
DB2 Wire Protocol	Oracle	Sybase IQ Wire Protocol
Greenplum Wire Protocol	Oracle Wire Protocol	Sybase Wire Protocol
Impala Wire Protocol	PostgreSQL Wire Protocol	Driver for the Teradata Database
Informix	Progress OpenEdge Wire Protocol	
Informix Wire Protocol	Salesforce	

Driver for Apache Hive™

To configure a connection, you might enter:

```
[Apache Hive Wire Protocol]
Driver=ODBCHOME/lib/xxhive.zz
Database=default
HostName=HiveServer
PortNumber=10000
LogonID=yourid
```

`HostName`: Type either the name or the IP address of the server to which you want to connect.

`PortNumber`: Type the port number of the server listener. The default port number for the Apache Hive server is 10000. Because of reported concurrency issues, you might want to use a different port number.

`Database`: Type the name of the Apache Hive database to which you want to connect by default. The database must exist, or the connection attempt will fail.

`LogonID`: Type the default user ID that is used to connect to your 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.

To configure a connection to DB2 for Linux/UNIX/Windows, you might enter:

```
[DB2 Wire Protocol]
Driver=ODBCHOME/lib/xxdb2nn.zz
IpAddress=123.456.78.90
TcpPort=50000
Database=SAMPLE
```

To configure a connection to DB2 for i or z/OS, you might enter:

```
[DB2 Wire Protocol]
Driver=ODBCHOME/lib/xxdb2nn.zz
IpAddress=123.456.78.90
TcpPort=446
Location=V5R2L0C
Collection=userid
```

Connection Options

`IpAddress`: 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.

`TcpPort`: 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.

`Database`: This option is valid and required only if you are connecting to a DB2 database on Linux/UNIX/Windows. Enter the name of the database to which you want to connect.

`Location`: This option is valid and required only if you are connecting to a DB2 database on DB2 for i or z/OS. Enter 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.

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

Greenplum Wire Protocol

To configure a connection, you might enter:

```
[Greenplum Wire Protocol]
Driver=ODBCHOME/lib/xxgplmnn.zz
Database=Gplumdb1
HostName=GreenplumServer
PortNumber=5432
```

Connection Options

`Database`: The name of the database to which you want to connect by default.

`Hostname`: Either the name or the IP address of the server to which you want to connect.

`PortNumber`: The port number of the server listener. The default is `5432`.

Impala^{TM} Wire Protocol

To configure a connection, you might enter:

```
[Impala Wire Protocol]
Driver=ODBCHOME/lib/xximpala.zz
Database=Impala1
HostName=ImpalaServer
PortNumber=21050
```

Connection Options

`Database`: The name of the database to which you want to connect by default.

`Hostname`: Either the name or the IP address of the server to which you want to connect.

`PortNumber`: The port number of the server listener. The default is `21050`.

Informix

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

To configure a connection, you might enter:

```
[Informix]
Driver=ODBCHOME/lib/ivinfxx.nn
Database=Informix3
HostName=InformixHost
ServerName=InformixServer
Service=online
```

Connection Options

`Database`: The name of the database to which you want to connect by default.

`HostMame`: The name of the machine on which the Informix server resides.

`ServerName`: The name of the Informix server as it appears in the sqlhosts file.

`Service`: The name of the service as it appears on the host machine.

Informix Wire Protocol

To configure a connection, you might enter:

```
[Informix Wire Protocol]
Driver=ODBCHOME/lib/xxifclnn.zz
Database=Informix3
HostName=InformixHost
LogonID=JohnD
PortNumber=1500
ServerName=InformixServer
```

Connection Options

`Database`: The name of the database to which you want to connect by default.

`Hostname`: Either the name or the IP address of the server to which you want to connect.

`LogonID`: Your user name as specified on the Informix server.

`PortNumber`: The port number of the server listener.

`ServerNumber`: The name of the Informix server as it appears in the sqlhosts file.

MySQL Wire Protocol

To configure a connection, you might enter:

```
[MySQL Wire Protocol]
Driver=ODBCHOME/lib/xxmysqlnn.zz
Database=MySQL3
HostName=MySQLHost
PortNumber=3306
```

Connection Options

`Database`: The name of the database to which you want to connect by default.

`Hostname`: Either the name or the IP address of the server to which you want to connect.

`PortNumber`: The port number of the server listener. The default is `3306`.

Oracle

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

To configure a connection, you might enter:

```
[Oracle]
Driver=ODBCHOME/lib/ ivor8xx.nn
ServerName=OracleServer
```

Connection Options

`ServerName`: 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

To configure a connection, you might enter:

```
[Oracle Wire Protocol]
Driver=ODBCHOME/lib/xxorann.zz
EditionName=oracle 1
HostName=199.226.224.34
PortNumber=1521
ServiceName=TEST
```

Connection Options

`Hostname`: Either the name or the IP address of the server to which you want to connect.

`PortNumber`: The port number of your Oracle listener. Check with your database administrator for the number.

`ServiceName`: 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.

`SID`: The Oracle System Identifier that refers to the instance of Oracle running on the server. The default is ORCL.

NOTE: `SID` and `ServiceName` are mutually exclusive. Only one or the other can be specified in the data source; otherwise, an error is generated.

`EditionName`: Oracle 11*g* R2 and higher only. The name of the Oracle edition the driver uses 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

To configure a connection, you might enter:

```
[PostgreSQL Wire Protocol]
Driver=ODBCHOME/lib/xxpsqlnn.zz
Database=Pgredb1
HostName=PostgreSQLServer
PortNumber=5432
```

Connection Options

`Database`: The name of the database to which you want to connect by default.

`Hostname`: Either the name or the IP address of the server to which you want to connect.

`PortNumber`: The port number of the server listener. The default is `5432`.

Progress OpenEdge® Wire Protocol

To configure a connection, you might enter:

```
[Progress OpenEdge Wire Protocol]
Driver=ODBCHOME/lib/xxoenn.zz
Database=odb1
HostName=OpenEdgeServer
PortNumber=5432
```

Connection Options

`Database`: The name of the database to which you want to connect by default.

`Hostname`: Either the name or the IP address of the server to which you want to connect.

`PortNumber`: The port number of the server listener.

Salesforce

The driver requires a Java Virtual Machine (JVM): Java SE 7 or higher. Before you configure a data source for the Salesforce driver, you must set the library path environment variable for your operating system to the directory containing your JVM's `libjvm.so` [`sl`|`a`] file, and that directory's parent directory. The library path environment variable is:

- `LD_LIBRARY_PATH` on Linux, Oracle Solaris, and HP-UX Itanium

- `SHLIB_PATH` on HP-UX PA-RISC

- `LIBPATH` on AIX

To configure a connection, you might enter:

```
[Salesforce]
Driver=ODBCHOME/lib/ivsfrc27.so
HostName=test.salesforce.com
UserName=JohnDoe
Password=secret
```

Connection Options

`Host Name`: The default Salesforce instance is login.salesforce.com. If you are logging into a different Salesforce instance, type the root of the Salesforce URL.

`User Name`: Type your logon ID for Salesforce.

`Password`: Type your case-sensitive password for the Salesforce instance.

If your Salesforce instance requires a security token, you can append it to the password, for example, secretXaBARTsLZReM4Px47qPLOS, where secret is the password and the remainder of the value is the security token. Both the password and security token are case-sensitive.

`Fetch Size`: Smaller fetch sizes can improve the initial response time of the query. Larger fetch sizes improve overall fetch times at the cost of additional memory. Smaller fetch sizes can improve the initial response time of the query. Larger fetch sizes improve overall fetch times at the cost of additional memory.

`WS fetch size`: If set to `0`, the driver attempts to fetch up to a maximum of 2000 rows. This value typically provides the maximum throughput. Setting the value lower than 2000 can reduce the response time for returning the initial data.

SQL Server Wire Protocol

To configure a connection, you might enter:

```
[SQLServer1 Wire Protocol]
Driver=ODBCHOME/lib/xxsqlsnn.zz
HostName=123.123.0.12
Database=master
PortNumber=1433
```

Connection Options

`HostName`: 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.

`Database`: The name of the database to which you want to connect by default.

`PortNumber`: The port number of the server listener. The default is `1433`.

Sybase IQ Wire Protocol

To configure a connection, you might enter:

```
[Sybase IQ Wire Protocol]
Driver=ODBCHOME/lib/xxsyiqnn.zz
Database=master
NetworkAddress=123.226.224.34,2638
```

Connection Options

`Database`: 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.

`NetworkAddress`: 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 `123.226.224.34, 2638`.

If your network supports named servers, you can specify an address as: *server_name, port_number*. For example, you can enter `SyIQserver, 2638`.

User Name: The default user ID that is used to connect to your database. Your ODBC application may override this value.

Sybase Wire Protocol

To configure a connection, you might enter:

```
[Sybase Wire Protocol]
Driver=ODBCHOME/lib/xxasenn.zz
Database=master
NetworkAddress=123.226.224.34,5000
```

Connection Options

`Database`: 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.

`NetworkAddress`: 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 `123.226.224.34, 5000`.

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

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.

To configure a connection, you might enter:

```
[Teradata]
Driver=ODBCHOME/lib/xxterann.zz
DBCName=123.123.12.12
SecurityMechanism=TD2
SecurityParameter=5678
UserID=John
```

Connection Options

`DBCName`: 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.

`SecurityMechanism`: Enter TD2.

`SecurityParameter`: 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`: The default UserID for the Teradata database.

Testing the Connection

The DataDirect Connect Series *for* ODBC installation includes a program named example that can be used to connect to a data source and execute SQL. The application is located in the `ODBCHOME/example` directory. To run the program, type `example` and follow the prompts to enter your data source name, user name, and password. If successful, a `SQL>` prompt appears and you can type in SQL Statements such as

```
SELECT * FROM accounts
```

If example is unable to connect, the appropriate error message appears.

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 *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 `ODBCHOME` is your DataDirect Connect *for* ODBC installation directory:

```
ODBCHOME/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 Driver for Apache Hive, Greenplum Wire Protocol, Informix, Impala Wire Protocol, PostgreSQL Wire Protocol, Progress OpenEdge Wire Protocol, and the Driver for the Teradata database are not included in the Performance Wizard. To tune these drivers for performance, set the following options:

The Driver for Apache Hive

If you know the typical fetch size for your application	Controlled by `ArraySize=`The number of rows your application is configured to retrieve. Smaller array sizes can improve the initial response time of the query. Larger fetch sizes improve overall fetch times at the cost of additional memory.

Greenplum Wire Protocol Driver

If your application does not use threads	`ApplicationUsingThreads=0`
If you want to use DataDirect connection pooling	`ConnectionPooling=1`
If your application is configured for high availability that replays queries after a failure	Controlled by `FailoverMode`=Although high availability that replays queries after a failure provides increased levels of protection, it can adversely affect performance because of increased overhead.

Impala Wire Protocol Driver

If you know the typical fetch size for your application	Controlled by `ArraySize`=The number of rows your application is configured to retrieve. Smaller array sizes can improve the initial response time of the query. Larger fetch sizes improve overall fetch times at the cost of additional memory.
If your application retrieves images, pictures, long text, or long binary data larger than 1 MB	Controlled by `Default Buffer Size for Long/LOB Columns=` To improve performance, a buffer size can be set to accommodate the maximum size of the data. The buffer size should only be large enough to accommodate the maximum amount of data retrieved; otherwise, performance will be reduced.
If your application needs to access a database objects owned only by the current user	`UseCurrentSchema=1`

Informix Driver (Client)

If your application does not use threads	`ApplicationUsingThreads=0`
If your application does not issue SQLCancel	`CancelDetectInterval=0`

PostgreSQL Wire Protocol Driver

If your application does not use threads	`ApplicationUsingThreads=0`
If you want to use DataDirect connection pooling	`ConnectionPooling=1`
If your application is configured for high availability that replays queries after a failure	Controlled by `FailoverMode`=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 `EncryptionMethod`=Data encryption may adversely affect performance because of the additional overhead (mainly CPU usage) required to encrypt and decrypt data.

Progress OpenEdge Wire Protocol Driver

If your application is configured for high availability that replays queries after a failure	Controlled by `FailoverMode`=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 `EncryptionMethod`=Data encryption may adversely affect performance because of the additional overhead (mainly CPU usage) required to encrypt and decrypt data.

Salesforce Driver

If your application does not use threads	`ApplicationUsingThreads=0`
If you want to use DataDirect connection pooling	`ConnectionPooling=1`
If you know the typical fetch size for your application	Set `FetchSize` to suit your environment. Smaller fetch sizes can improve the initial response time of the query. Larger fetch sizes improve overall fetch times at the cost of additional memory.
If you know how many rows of data the driver attempts to fetch for each ODBC call for a Web service	Set `WSFetchSize=0` to fetch a maximum of 2000 rows for optimum throughput. Setting the value lower than 2000 can reduce the response time for returning the initial data.