

# New Features Enterprise Connect™ Data Access 15.0 for Microsoft Windows, Linux, and UNIX

Document ID: DC20173-01-1500-03

Last revised: August 2008

This document alphabetically lists and describes new features available for the 15.0 release of Enterprise Connect™ Data Access (ECDA), including the most current release of features in ESD #2, as noted.

<b>Topic</b>	<b>Page</b>
Create Server wizard	2
ECDA Option for Oracle failover in an Oracle RAC environment	3
Configuring failover in the Oracle RAC environment	4
New failover configuration parameters for ECDA Option for Oracle	5
Special considerations for usage	6
ECDA support for back-end drivers	7
Enhanced security with SySAM 2.0	7
Enhanced text update	8
Extended encryption login	8
ODBC Driver Manager architecture	9
Partial text updates for Microsoft SQL Server	10
Revised directory structure	10
Unsigned integer and bigint datatypes, date datatype, and time datatype	11
Unicode datatype support	12

## Create Server wizard

ECDA 15.0 now provides a GUI wizard that performs certain setup and configuration tasks, depending on your platform.

For Oracle

ECDA Option for Oracle does not use ODBC drivers. However, it does require a unique set of configuration settings and a pointer to a file that will establish a connection to the target Oracle database. The Create Server wizard can easily create a DirectConnect™ server by asking for the required configuration properties and for a pointer to the appropriate file. Then, it applies the information to create and start a DirectConnect server.

---

**Note** In ECDA Option for Oracle, you can connect to the target database without further configuration, as long as the settings are correct. However, if necessary, you can still refine or correct these settings by using DirectConnect Manager or by modifying them with a text editor.

---

For Linux and UNIX

As of version 15.0, ECDA allows you to use multiple versions of an ODBC driver manager for Linux and UNIX platforms. ECDA ships with a default driver manager called unixODBC Driver Manager, but because you are no longer tied to a single type of driver manager, you can obtain a driver manager supplied by another vendor. However, to work properly with another vendor's driver manager, the ECDA DirectConnect server needs an initial configuration to be able to link to the vendor driver manager you want. (If you provide none, it defaults to the unixODBC Driver Manager.)

To help you implement a vendor's driver manager, ECDA 15.0 provides a Create Server wizard. Although the wizard does not complete the setup, it does simplify the process of getting the DirectConnect server configured to run with the driver manager that you specify. Then, when the DirectConnect server is running, you can use DirectConnect Manager to complete the configuration and further refine the settings of the services. Continuing with DirectConnect Manager, you can test the connections to the target database to validate those settings.

For Windows

For Microsoft Windows products, ECDA will only use the driver manager that the operating system provides. The Create Server wizard will allow you to create a new server but it will not give you the ability to set up an alternative driver manager. (In this case, using the AddServer batch file may be a faster and easier approach.)

Connectivity limitations

For Linux, UNIX, and Windows platforms, the Create Server wizard does not allow you to:

- Create DSNs
- Modify the *odbc.ini* file for a specific ODBC driver
- Enable the connection to a target database

Refer to the vendor driver manager for utilities you can use to complete these tasks.

## ECDA Option for Oracle failover in an Oracle RAC environment

---

**Note** To use the RAC failover feature, you must have a minimum 10g version of Oracle.

---

Oracle provides technology called Real Application Clusters (RAC), which allows an Oracle database to run in a clustered environment; in other words, the database can be serviced by several Oracle instances spread across several host machines. In this clustered environment, a “virtual instance” will direct and redirect instance connections from one instance to another when failover occurs. If an instance fails, the client is failed over to a secondary instance in the RAC environment. The Sybase® Enterprise Connect Data Access Option for Oracle uses the Oracle feature called Transparent Application Failover (TAF) to provide this failover function.

When a heterogeneous environment includes an Oracle RAC environment, ECDA Option for Oracle support of TAF helps prevent an outage due to an Oracle instance failure, and as a result, reduces downtime in operations.

In support of Oracle RAC, ECDA Option for Oracle manages the failover from one instance to another as follows:

- It connects to an Oracle database instance participating in an Oracle RAC environment.
- When failure occurs, it sends the ECDA Option for Oracle client connection to alternate Oracle instances that are defined in the ECDA Option for Oracle *insnames.ora* file.

- It reports any error messages resulting from a failover to the client application.

---

**Note** If all backup instances are unavailable, an ORA-03113 error is sent back to the client application.

---

## Configuring failover in the Oracle RAC environment

To provide the failover capabilities in an Oracle RAC environment, you must configure the Oracle connection string in the ECDA Option for Oracle *tnsnames.ora* file to use Oracle TAF.

To enable failover, you enter multiple virtual listener addresses in the *tnsnames.ora* file for the RAC database configuration. The first address is used for the initial connection. As shown in the following code segment, when a failure occurs, the connections between the ECDA Option for Oracle and the Oracle database will try to reconnect to the subsequent virtual listener address entries (shown in bold typeface) until a connection is successful.

```
RACDB=
  (DESCRIPTION=
    (ADDRESS = (PROTOCOL = TCP) (HOST = host1) (PORT = 1521))
    (ADDRESS = (PROTOCOL = TCP) (HOST = host2) (PORT = 1521))
    (ADDRESS = (PROTOCOL = TCP) (HOST = host3) (PORT = 1521))
    (ADDRESS = (PROTOCOL = TCP) (HOST = host4) (PORT = 1521))
  )
```

In addition, you must add the `FAILOVER_MODE` section to the Oracle connection string. It has two configuration parameters that define the failover characteristics for a connection:

- `TYPE`, which has two options:
  - `SELECT` keeps track of any select statements issued in the current transaction and replays them after the failover connection is reestablished. (Sybase recommends this option.)
  - `SESSION` only establishes a new connection to the backup instance, and all work is lost.
- `METHOD`, which also has two options:
  - `BASIC` establishes a new connection to the backup instance only when the primary connection fails. (Sybase recommends this option.)

- PRECONNECT establishes a connection to the backup instance when the primary connection is made, reducing failover time. However, for every connection, two connections are created, so more resources are used.

This code segment shows the two recommended `FAILOVER_MODE` options in bold typeface:

```
(CONNECT_DATA =
  (SERVER = DEDICATED)
  (SERVICE_NAME = racdb.sybase.com)
  (FAILOVER_MODE =
    (TYPE = SELECT)
    (METHOD = BASIC)
  )
)
```

This code segment shows the example Oracle connection string combined in the `tnsnames.ora` file with all of the configuration parameters (in bold typeface) set to enable failover:

```
RACDB=
  (DESCRIPTION=
    (ADDRESS = (PROTOCOL = TCP) (HOST = host1) (PORT = 1521))
    (ADDRESS = (PROTOCOL = TCP) (HOST = host2) (PORT = 1521))
    (ADDRESS = (PROTOCOL = TCP) (HOST = host3) (PORT = 1521))
    (ADDRESS = (PROTOCOL = TCP) (HOST = host4) (PORT = 1521))
    (CONNECT_DATA =
      (SERVER = DEDICATED)
      (SERVICE_NAME = racdb.sybase.com)
      (FAILOVER_MODE =
        (TYPE = SELECT)
        (METHOD = BASIC)
      )
    )
  )
)
```

## New failover configuration parameters for ECDA Option for Oracle

The `FAILOVER_MODE` section has two parameters that ECDA Option for Oracle does **not** support in its `tnsnames.ora` file:

- DELAY
- RETRIES

However, two new configuration file parameters have been added to replace them:

- `fo_delay`, which defines the number of seconds to wait between failover retry attempts (default is 0).
- `fo_retries`, which defines the number of times the failover retry will be attempted (default is 0).

### Special considerations for usage

For users of the ECDA Option for Oracle, failover of connections in an Oracle RAC feature may require some specific actions and responses to failover error messages that are returned. The following subsections describe several conditions that may occur.

#### ALTER SESSION commands

Client applications to the ECDA Option for Oracle that have been failed over to a backup instance must replay all `ALTER SESSION` statements that have been issued during the session to the ECDA Option for Oracle. This is required because, when failover occurred, the connection will have been set to its default state.

#### Data Manipulation Language (DML) transactions

Oracle TAF does not automatically fail over DML transaction statements, such as `INSERT`, `UPDATE`, or `DELETE`. Because the ECDA Option for Oracle is basically a passthrough gateway, it does not keep track of any transaction boundaries; therefore, it has no knowledge of all of the statements that make up the transaction. As a result, when a failover has occurred, the next statement in the transaction will cause an error, causing the entire transaction to fail.

In response, the ECDA Option for Oracle client will need either to issue a `ROLLBACK` or to disconnect and reconnect (perform an implicit `ROLLBACK`) before the transaction can be reapplied.

## SELECT statements

If select statements are being processed at the time that failover occurs:

- If TYPE=SELECT in the *tnsnames.ora* file, failover will be successful.
- If TYPE=SESSION, failover does not occur and all work is lost.

## ECDA support for back-end drivers

Because Sybase no longer provides ODBC drivers for back-end access to the target database, you need to obtain your own driver from your database vendor. Sybase supports and certifies with these back-end drivers:

- IBM DB2 CLI ODBC Driver
- DataDirect Driver for DB2 and Microsoft SQL Server
- Microsoft Driver for SQL Server (on Microsoft Windows), which is downloadable and free

For more information, see the Enterprise Connect Data Access *Installation Guide* for your platform.

## Enhanced security with SySAM 2.0

ECDA 15.0 now provides Sybase Software Asset Manager (SySAM) 2.0. This upgrade is supported for both ECDA options (ODBC and Oracle) and performs the following security-related tasks:

- Verifies that a valid license exists for the software being used. If a license is not valid, the service does not start and the DirectConnect server logs the license error message.
- Allows Sybase customers with support contracts to access every Emergency Bug Fix (EBF) published before the end of their contract date.

---

**Note** If you download ECDA 15.0 software without a license, SySAM 2.0 allows a 30-day grace period for you to obtain a valid license through the Sybase Product Download Center on the Web.

---

For more information about SySAM 2.0, see the Enterprise Connect Data Access *Installation Guide* for your platform.

## Enhanced text update

**(ESD #2)** The Enhanced Text Update feature that has been implemented allows Replication Server® to include a WHERE clause with a WRITETEXT BULK statement for Microsoft SQL Server. This improves the performance of text, image, and ntext updates.

## Extended encryption login

**(ESD #2)** The OCS API provides the ability to support the public key encrypted password handshake between the Open Client™ and the Open Server™, making this functionality available for implementation in Enterprise Connect Data Access products.

Following is the process that occurs when any Open Client-based application (such as isql) issues a public key encrypted login request to an Open Server application (such as ECDA and the DirectConnect server):

- 1 The server sends a public key to the client, and waits for the client to send back the encrypted password.
- 2 When the DirectConnect server receives the encrypted password, it decrypts the password to plain text.
- 3 The DirectConnect server sends the plain text password to a remote server or database for authentication.

To request an extended encrypted login, use the isql -X command. In the following example, “testsrv” is a DirectConnect service name, “tester” is a user name, “sybase” is a password, and -X is the option for the encrypted login request:

```
isql -Stestsrv -Utester -Psybase -X
```



# ODBC Driver Manager architecture

---

**Note** This feature applies to UNIX and Linux platforms only.

---

ECDA uses ODBC drivers to access multiple database targets, including Microsoft SQL Server and DB2 UDB. An ODBC Driver Manager loads ODBC database drivers on behalf of the ECDA DirectConnect server.

With ECDA 12.6.1, Sybase linked with and shipped the DataDirect ODBC Driver Manager on Linux and UNIX platforms.

Now, with ECDA 15.0, Sybase replaces the DataDirect ODBC Driver Manager with the open source unixODBC Driver Manager. Sybase does continue to support the Data Direct Driver suite. If you want to use the DataDirect ODBC Driver suite, you must use the DataDirect Driver Manager. Using the new Create Server wizard, you can select and use a DataDirect Driver Manager library when you create a new server.

As a result of this more open architecture, Sybase will be able to formally certify other ODBC drivers as required. In the meantime, you may want to test and integrate with other ODBC drivers for additional database access to meet your business needs.

## Configuration property

A new service library configuration property called `ODBCDriverManager` specifies the load library for the desired ODBC Driver Manager. This configuration property can contain the library's full path name as well as library version extensions.

## Environment variables

ECDA 15.0 uses these library path environment variables to locate the libraries that the runtime loader uses to satisfy runtime dependencies:

- *Linux*: `LD_LIBRARY_PATH`
- *AIX*: `LIBPATH`
- *Solaris*: `LD_LIBRARY_PATH`
- *HP/UX*: `SHLIB_PATH`
- *HP/Itanium*: `LD_LIBRARY_PATH`

## ODBC Driver Manager environment variables

To find and load drivers, the driver manager uses configuration files, which contain specific information about target databases and driver location.

The environment variable identifier for the unixODBC driver manager is `ODBCSYSINI`, which points to the directory that contains these files:

- *odbc.ini* contains the Data Source Name (DSN) information, such as connection information, driver identifiers, and other driver-specific attributes.
- *odbcinst.ini* contains driver-specific information only.

For more information, see the Enterprise Connect Data Access *Installation Guide* for UNIX.

## Partial text updates for Microsoft SQL Server

**(ESD #2)** ECDA Option for ODBC now supports the Microsoft SQL Server feature that executes partial text updates to text, image, and ntext data. To replicate partial text updates, Open Client, Open Server, and Replication Server have added the partial text update feature. Now, the DirectConnect server supports the partial text update commands sent by Replication Server.

## Revised directory structure

ECDA 15.0 has been updated to use the Open Client and Open Server 15.0 libraries. This update allows ECDA products to synchronize with other Sybase 15.0 products.

In addition, ECDA 15.0 now has a directory structure that isolates it from other libraries, so that when the DirectConnect server is running, it does not conflict with other Sybase product libraries.

For detailed information about the directory structure, see the Enterprise Connect Data Access *Installation Guide* for your platform.

## Unsigned *integer* and *bigint* datatypes, date datatype, and time datatype

(ESD #2) Sybase Open Client and Open Server 15.0 and the later releases of Sybase Open Client and Open Server 12.5.1 support several new datatypes: CS\_BIGINT, CS\_UBIGINT (unsigned bigint), CS\_USMALLINT (unsigned smallint), CS\_UINT (unsigned integer), CS\_DATE, and CS\_TIME. The ECDA Option for Oracle, the ECDA option for DB2 UDB, and the ECDA option for Microsoft SQL Server now support these datatypes between the client and target databases.

The ECDA options for Microsoft SQL Server and DB2 UDB support these *express transfer* options between:

- **BigInt and Numeric** – allows transfer of data between the datatype columns by configuring the property `BigIntResults` to a value `Legacy`.
- **BigInt and BigInt** – allows transfer of data between the datatype columns. The transfer does not require any changes to be done to the database configuration property sets.
- **Numeric and Numeric** – allows transfer of data between the datatype columns.
- **Time and Time** – allows transfer of data from DB2 UDB TIME (source) to ASE TIME (target) columns when the precision value of the target column is greater than the precision of the source column. If the precision value of the source column is greater than that of the target column, express transfer fails. Transfer from ASE TIME column to DB2 UDB TIME column is not supported by express transfer. Transfer bulk supports this transfer. Since DB2 UDB Time column does not support milliseconds, transfer bulk truncates the milliseconds during the conversion from ASE to DB2 UDB time value.
- **Time and Char** – allows transfer from Time (source) to Char (target) columns only when target column has either equal or greater precision than the source column. Target column with smaller precision generates an error. For express transfer between Microsoft SQL Server Char column to DB2 UDB Time column requires IBM AS400 ODBC driver support. The Data Direct drivers generate an invalid data descriptor error. Alternatively, you can use transfer bulk for transferring data.
- **Unsigned (BigInt, Smallint, and Int) and Numeric** – allows transfer of data between the datatype columns.

Express transfer does not support data transfer between an Unsigned (BigInt, Smallint, and Int) datatype column and a Char datatype column.

The express transfer option, which initiates a direct transfer of data between two databases from the client application, has these exceptions:

- ASE DATE and TIME columns to DB2 UDB TIMESTAMP column are not supported.
- ASE TIME column to DB2 UDB TIME column is not supported.
- DB2 UDB TIME column does not accept values in fractional seconds. Therefore, all character values must be in the HH.MM.SS format.
- DB2 UDB TIME column to ASE TIME column; the value 24.00.00 is not supported.

However, you can use the *bulk copy transfer* option that supports these exceptions.

## Unicode datatype support

**(ESD #2)** The ECDA Option for ODBC and the ECDA Option for Oracle now supports several new unicode datatypes for Replication Server, CIS/ASE, and Open Client applications. These unicode datatypes include:

- Unichar - Unicode fixed length character
- Univarchar - Unicode variable length character
- Unitext - Unicode large character object

In the ECDA Option for ODBC all the transfer modes are fully supported for these datatypes.