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

This document alphabetically lists and describes new features available in Sybase® Enterprise Connect™ Data Access (ECDA) version 15.0, including ESD #1, ESD #2, and ESD #3.

Create Server wizard

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

For Oracle

The 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 establishes a connection to the target Oracle database. You can create a DirectConnect™ server by providing the required configuration properties, and a pointer to the appropriate file to the Create Server wizard. The wizard applies the information to create and start a DirectConnect server. As long as the settings are correct, you can connect to the target database without further configuration. However, if necessary, you can still refine or correct these settings by either 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 you can obtain a driver manager supplied by another vendor. However, to work properly with a driver manager from another vendor, the ECDA DirectConnect server needs an initial configuration. If you do not provide an initial configuration, the DirectConnect server defaults to the unixODBC Driver Manager.

To help you implement a particular vendor's driver manager, ECDA 15.0 provides a Create Server wizard that simplifies configuring the DirectConnect server to run with the driver manager 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, and test the connections to the target database to validate those settings.

For Windows

For Microsoft Windows products, ECDA uses only the driver manager that the operating system provides. You can use the Create Server wizard to create a new server but you cannot set up an alternative driver manager. Using the AddServer batch file may be a faster and easier approach.

Connectivity limitations

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.

Datatype support for Unicode

(ESD #2) The ECDA Option for ODBC and the ECDA Option for Oracle support several new Unicode datatypes for Replication Server®, Adaptive Server® Enterprise/Component Integration Services (CIS), 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 transfer modes are fully supported for these datatypes.

Datatype support for date, time, and unsigned *integer, bigint, and smallint*

(ESD #2) Sybase Open Client and Open Server™ 15.0 and the later versions 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

- CS_TIME

The ECDA Option for Oracle and the ECDA Option for ODBC also support these datatypes between the client and target databases.

The ECDA Option for ODBC supports 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 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 Adaptive Server 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, but bulk copy transfer supports this transfer. Since DB2 UDB Time column does not support milliseconds, bulk copy transfer 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. Data Direct drivers generate an invalid data descriptor error. Alternatively, you can use bulk copy transfer 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, which supports these exceptions.

ECDA Option for Oracle failover in an Oracle RAC environment

To use the RAC failover feature, you must be using Oracle 10g or later.

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" directs and redirects instance connections from one instance to another when failover occurs. If an instance fails, the client fails over to a secondary instance in the RAC environment. ECDA 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, thus reducing operational downtime.

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 that is 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 *tnsnames.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 returned to the client application.

Configuring failover in the Oracle RAC environment

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

To enable failover, 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 try to reconnect to the subsequent virtual listener address entries 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. This section 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 that you use this option.

    - SESSION establishes a new connection only 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 that you use 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:

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

This code segment shows an example of the Oracle connection string combining the earlier code segments in the *tnsnames.ora* file with all of the configuration parameters 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)
   )
  )
 )
```

Failover configuration parameters for ECDA Option for Oracle

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

- DELAY
- RETRIES

However, these configuration file parameters replace them:

- fo_delay – defines the number of seconds to wait between failover retry attempts. Default is 0.
- fo_retries – defines the number of times the failover retry is attempted. Default is 0.

Special considerations for usage

For ECDA Option for Oracle users, 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 occurs, the connection is 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, ECDA Option for Oracle keeps no record of the statements that make up a transaction. As a result, when a failover occurs, the next statement in the transaction causes an error, which causes the entire transaction to fail.

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

SELECT statements

If select statements are processed when a failover occurs:

- If TYPE=SELECT in the *tnsnames.ora* file, failover is 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 must obtain a driver from your database vendor. Sybase supports and certifies 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

ECDA 15.0 uses Sybase Software Asset Manager (SySAM) 2.0, which is supported for both ECDA options—ODBC and Oracle—and performs these security-related tasks:

- Verifies that a valid license exists for the software being used. If a license is invalid, 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 during which you can obtain a valid license from the Sybase Product Download Center Web site at https://sybase.subscribenet.com.

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

Enhanced text update

(ESD #2) Enhanced text update 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 supports the public-key encrypted password handshake between Open Client and Open Server, making this functionality available for implementation in ECDA products.

This 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 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 this example, "testsrv" is the DirectConnect service name, "tester" is the user name, "sybase" is the password, and -X is the option for the encrypted login request:

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

Improved performance in ECDA Option for Oracle

(ESD #3) In the ECDA Option for Oracle, set the rep_sparse_parse configuration parameter to 1 to allow SQL statements in Oracle syntax to pass from Replication Server clients directly to Oracle without being parsed by ECDA Option for Oracle. This improves performance since most of the SQL statements going to Oracle from Replication Server use Oracle syntax.

The default value for this parameter is 0, which means the ECDA Option for Oracle parses all SQL statements from Replication Server to Oracle.

This parameter is service-specific and dynamic and it applies only to Replication Server clients.

See the *Enterprise Connect Data Access Option for Oracle 15.0 for Microsoft Windows, Linux, and UNIX Server Administration and Users Guide* to set configuration parameters.

ODBC driver manager architecture

This feature applies only to UNIX and Linux platforms.

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

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

As of ECDA 15.0, Sybase has replaced the DataDirect ODBC Driver Manager with the open source unixODBC Driver Manager. To use the DataDirect ODBC Driver suite, you must use the DataDirect Driver Manager. Use the Create Server wizard to 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 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:

Platform	Library path environment variable
HP Itanium	LD_LIBRARY_PATH
HP-UX	SHLIB_PATH
IBM AIX	LIBPATH
Linux	LD_LIBRARY_PATH
Sun Solaris	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 Linux and UNIX*.

Partial text updates for Microsoft SQL Server

(ESD #2) ECDA Option for ODBC supports the Microsoft SQL Server feature that executes partial text updates to text, image, and ntext data. Open Client, Open Server, and Replication Server support the partial text update feature. Additionally, the DirectConnect server supports 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.