Configuring Heterogeneous Replication server for MSSQL & Azure

EDCA setup, doc version 2.1

Setup and configuration guide to replicating from Sybase to SQL Server: Garrett Devine, Email: Garrett.devine@dbxperts.co.uk

Contents

Recommendations First

Sybase recommends that ECDA for ODBC, and the target database reside on the same machine.

System requirements

Repserver, 512MB RAM, 380MB disk
ECDA, 512MB RAM, 300MB disk

Configuring ECDA Option for ODBC – on Replicate

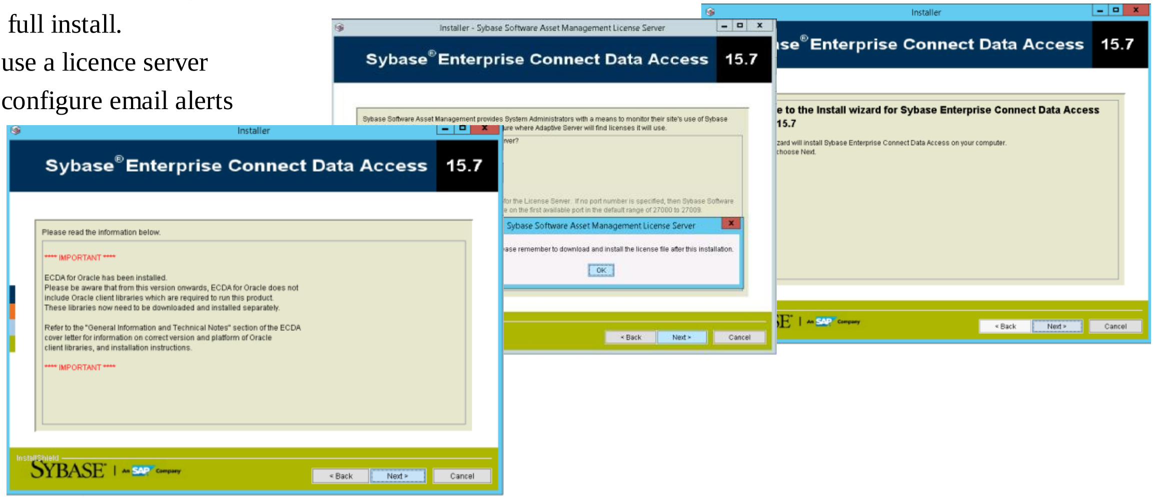
1. Open ODBC Driver 32-bit manager from %systemroot%\SysWOW64\odbcad32.exe (this is the 32-bit version). You must use the 32-bit version.
2. Add System ODBC DNS connection for SQL Server driver called ***ECDA_TRAIN1***
3. SQL Server name is VM_TRAIN\MSSQL_TRAIN1
4. Add SQL server login to SQL Server called ecda_user, with sysadmin role (for now)
5. Use "SQL Server authentication", userID=ecda_user, password=ecda_user. (sysadmin role, password does not expire)

*note: in some clients systems, we have found that we had to setup the ODBC connection using a Domain User account and use a User DNS.

**note: if setting up an ODBC connection to Azure, see the notes further down this document. You must specify the name of the DB you want to connect to.

Installing ECDA

1. Extract the EDCA package to a temp location on the SQL Server node and run setup
2. Choose full install.
3. Do not use a licence server
4. Do not configure email alerts



Installer
Sybase® Enterprise Connect Data Access 15.7
< Back
Next >
Cancel
Installer - Sybase Software Asset Management License Server
Sybase® Enterprise Connect Data Access 15.7
Installer
Sybase® Enterprise Connect Data Access 15.7
InstallShield
SYBASE | An SAP Company
OK

e to the Install wizard for Sybase Enterprise Connect Data Access 15.7

zard will install Sybase Enterprise Connect Data Access on your computer. choose Next.

Sybase Software Asset Management provides System Administrators with a means to monitor their site's use of Sybase ure where Adaptive Server will find licenses it will use.

ver?

for the License Server. If no port number is specified, then Sybase Software e on the first available port in the default range of 27000 to 27009.

Sybase Software Asset Management License Server

ase remember to download and install the license file after this installation.

Please read the information below.

**** IMPORTANT ****

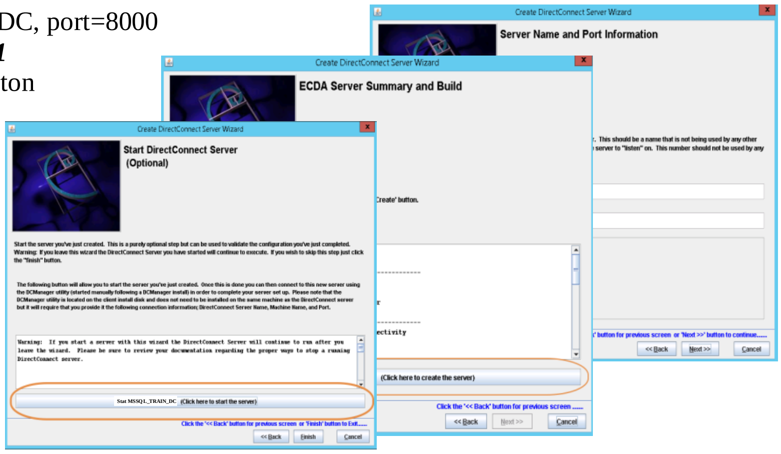
ECDA for Oracle has been installed.
Please be aware that from this version onwards, ECDA for Oracle does not include Oracle client libraries which are required to run this product.
These libraries now need to be downloaded and installed separately.

Refer to the "General Information and Technical Notes" section of the ECDA cover letter for information on correct version and platform of Oracle client libraries, and installation instructions.

**** IMPORTANT ****

Configuring and starting the DirectConnect server

1. In CMD window, execute "%SYBASE%\DC-15_0\DCWizard/DCWizard.bat"
2. Select the ECDA Option for ODBC
3. Set server name= MSSQL_TRAIN1_DC, port=8000
4. Enter service name = ***ECDA_TRAIN1***
5. Finally, select the „Create Server" button
6. Start the service



Create DirectConnect Server Wizard
Server Name and Port Information
<< Back
Next >>
Cancel
Create DirectConnect Server Wizard
ECDA Server Summary and Build
<< Back
Next >>
Cancel
Create DirectConnect Server Wizard
Start DirectConnect Server (Optional)
<< Back
Finish
Cancel

. This should be a name that is not being used by any other server to "listen" on. This number should not be used by any

' button for previous screen or 'Next >>' button to continue......

Create' button.

ectivity

(Click here to create the server)

Click the '<< Back' button for previous screen ......

Start the server you've just created. This is a purely optional step but can be used to validate the configuration you've just completed. Warning: If you leave this wizard the DirectConnect Server you have started will continue to execute. If you wish to skip this step just click the "finish" button.

The following button will allow you to start the server you've just created. Once this is done you can then connect to this new server using the DCManager utility (started manually following a DCManager install) in order to complete your server set up. Please note that the DCManager utility is located on the client install disk and does not need to be installed on the same machine as the DirectConnect server but it will require that you provide it the following connection information; DirectConnect Server Name, Machine Name, and Port.

```
Warning:  If you start a server with this wizard the DirectConnect Server will continue to run after you
leave the wizard.  Please be sure to review your documentation regarding the proper ways to stop a running
DirectConnect server.
```

Stat MSSQL_TRAIN_DC (Click here to start the server)

Click the '<< Back' button for previous screen or 'Finish' button to Exit......

7. Edit C:\sybase\DC-15_0\servers\MSSQL_TRAIN1_DC\cfg\dcany.cfg to contain the following:-

```
[Service Library]
{Client Interaction}

[ECDA_TRAIN1]
{ACS Required}
ConnectionSpec1=ECDA_TRAIN1
{Client Interaction}
EnableAtStartup=yes
TransactionMode=short
SendWarningMessages=yes
{Target Interaction}
Allocate=connect
SQLTransformation=passthrough
ReturnNativeError=yes
{Catalog Stored Procedures}
CSPColumnODBCVersion=3
```

Note1: the section name and ConnectionSpec1 settings are set to the name of our service. This name **must** match the name given to the ODBC connection.

Adding licence file

SAP provide a pdf of the licence file. Copy and paste the contents of the file into

C:\Sybase\SYSAM-2_0\licenses\SySAMLicenseServer.lic

Stop DC server (close MS-DOS box)

Start DC server by opening new MD-DOS box and running

```
C:\sybase\DC-15_0\bin>DCStart.bat -SMSSQL_TRAIN1_DC
```

Testing

1. Start DirectConnect service, `C:\sybase\DC-15_0\bin\DCStart.bat -SMSSQL_TRAIN1_DC`
2. Connect using "isql" from a MS-DOS session

```
C:\sybase\DC-15_0>DC_SYBASE.bat
C:\sybase\DC-15_0>isql -Usa -Pecda_user -SECDA_TRAIN1
```

Verify the connection to the replicate Microsoft SQL Server database by obtaining the DBMS name and version number:

```
select @@sqldbmsname
go

SQLDbmsName
---------------------
Microsoft SQL Server
```

Well done. You have configured the DirectConnect server!

Tip: If you see the error "The specified DSN contains an architecture mismatch between the Driver and Application", change to using the 32bit ODBC driver, by running %systemroot%\SysWOW64\odbcad32.exe

Get replicating !

Assuming your Sybase repserver is already installed.

We will add a database, and then define a simple replication definition (repdef), which we will subscribe to in MSSQL Server.

Creating the primary database

```
ASE
disk init name="pubs2_data01",physname="C:\sybase_15\pubs2_data01.dat",size="20M"
go
sp_diskdefault pubs2_data01, defaulton
go
sp_diskdefault master, defaultoff
go

cd $SYBASE/ASE-15_0/scripts ISQL -iinstpbs2

MSSQL
Create small database using wizard, called pubs2
```

Create a replication maintenance user in Microsoft SQL Server

1. In the Sybase serer (primary)

   ```
   sp_addlogin rep_maint, rep_maint_ps,pubs2
   go
   sp_role ‘grant’, replication_role,rep_maint
   go
   use
   pubs2
   go
   sp_addalias rep_maint,dbo
   go
   ```

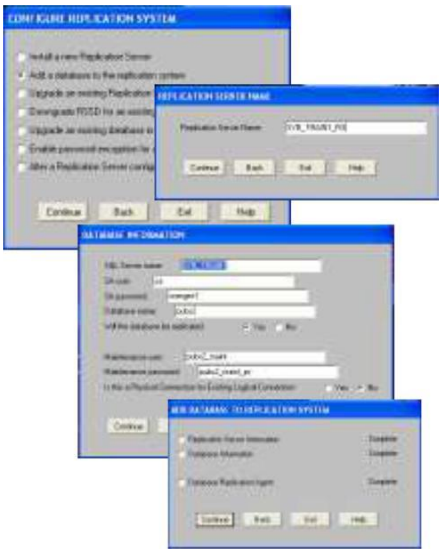
2. In MSSQL server (replicate)

   ```
   CREATE LOGIN rep_maint WITH PASSWORD='rep_maint_ps',
   DEFAULT_DATABASE=pubs2, CHECK_EXPIRATION=OFF,
   CHECK_POLICY=OFF
   go
   Use pubs2
   Go
   CREATE USER rep_maint FOR LOGIN rep_maint;
   EXEC sp_addrolemember 'db_owner', 'rep_maint';
   ```

Add databases to Replication System

Primary DB

Use *rs_init* to add primary database. This will configure the rep_agent and rs_* objects to the ASE database and add the connection to the repserver.


CONFIGURE REPLICATION SYSTEM
Install a new Replication Server
Add a database to the replication system
REPLICATION SERVER NAME
Replication Server Name
SYB_TRAIN1_RS
Continue
Back
Exit
Help
DATABASE INFORMATION
Will the database be replicated?
Yes
No
ADD DATABASE TO REPLICATION SYSTEM
Complete

1. Start *rs_init* and select to “Add a database” from the replication configuration dialogue
2. Give a server name of SYB_TRAIN1_RS
3. In the Database Information dialogue box fill in all the details and make sure you answer “yes” to “Will the database be replicated”. Accept the suggested user** name and password for the maintenance user.
4. Finally, once you are happy that all the information, select “Continue” from the “Add database to replication system” dialogue box.

   **Notes: you may need to drop the rep_maint alias, add it as a normal user using sp_adduser, run the above script and finally add the rep_maint user back in as an alias to dbo.

Replicate DB

Add entry to the sql.ini file in the Sybase Replication Server, so the repserver knows how to connect to the replicate database on the SQL Server box. Example below (change IP address).

```
[ECDA_TRAIN1]
master=TCP,10.1.5.255,8000
query=TCP,10.1.5.255,8000
```

With the DC still running, log into the RS server

```
create connection to ECDA_TRAIN1.pubs2
using profile rs_ase_to_msss;standard
set username rep_maint
set password "rep_maint_ps"
go
admin who
go
```

You should see similar entries to this:-

```
    28 DSI EXEC   Awaiting Command     105(1) ECDA_TRAIN1.pubs2
    27 DSI        Awaiting Message     105 ECDA_TRAIN1.pubs2
    26 SQM        Awaiting Message     105:0 ECDA_TRAIN1.pubs2
```

If your connections show as DOWN, see **trouble shooting** section at the end of this document.

Create the Replication Definition

```
In ASE,

use pubs2
go
-- get the repdef from the table
-- stored proc can be found here
http://www.sypron.nl/misctools.html
1>sp_gen_repdef @servername=SYB_TRAIN1, @tablename=publishers, @minimal=1
2>go

create replication definition publishers_repdef --add the '_repdef' bit
with primary at SYB_TRAIN1.pubs2
with all tables named 'publishers'
(
  pub_id char(4),
  pub_name varchar(40),
  city varchar(20),
  state char(2)
)
primary key (pub_id)
replicate minimal columns
```

Run the above output into the RS!

Top Tip: prevent issues with tables containing reserved words. You cannot use some words in SQL Server for column names, like 'percent' or 'status' without enclosing them in quotes. We need to make repserver aware by altering the connection out to the replicate DB.

```
alter connection to ECDA_TRAIN1.pubs2
set dsi_quoted_identifier to 'on'
go
```

Synchronise the table data between ASE & MSSQL

In ASE,

```
-- First, reverse engineer the table, using Ed Barlows excellent stored
proc, sp__revtable or use the Sybase ddlgen tool.
1> sp__revtable publishers
2> go
 -- Table_DDL
 ----------------------------------------------------------------------
 CREATE TABLE  publishers
 (
  pub_id     char(4)                      NOT NULL,
  pub_name   varchar(40)                  NULL,
  city       varchar(20)                  NULL,
  state      char(2)                      NULL
 )
 -- Index_DDL
 ----------------------------------------------------------------------
```

```
create unique clustered index pubind on dbo.publishers (pub_id)
```

Run the above output into MSSQL, pubs2 database, to create the empty table.

Use BCP to copy data:-

```
bcp pubs2..publishers out publishers.txt -SSYB_TRAIN1 -Usa -Pecda_user –c
```

```
msbcp pubs2..publishers in publishers.txt -Usa -Pecda_user -SVM-TRAIN\MSSQL_TRAIN1 –c
```

*note, I have changed the name of the Microsoft bcp.exe to msbcp.exe, to differentiate between the two versions of this utility. You only need to do this if both databases servers are on the same host (our test environment).

Create Subscription to our new repdef

In RS,

```
create subscription publishers_sub
for publishers_repdef
with replicate at ECDA_TRAIN1.pubs2
without materialization
```

```
check subscription publishers_sub
for publishers_repdef
with replicate at ECDA_TRAIN1.pubs2
```

Mark the table for replication

This step is vital, yet very simple. I have lost track of the number of times I have forgotten to do this!
In ASE,

```
sp_setreptable publishers, true
```

Insert sample row

```
insert into publishers values('9999', 'Fancypants publishing', 'London', 'NA')
```

Now in ASE & MSSQL compare the returned rows ‘select * from publishers’ in Sybase and SQL Server

If they are the same, congratulations you have installed replication between Sybase and MSSQL, using a simple repdef-subscription model. If the results are not the same, go back and check each step, one at a time.

Azure Database setup

This is very similar to a normal SQL Server. You follow the procedure above with some minor exceptions. When replicating to an Azure Database, the database does not sit on an Operating System that we can access, so we cannot install the EDCA on the same host that the database resides on. We choose to install the EDCA on the Sybase replication server but a VM within Azure would be an equally valid location.

Azure Firewall Access

First thing to do once you have created your DB in Azure is to check you can connect from your network to the Azure Database, using Management Studio. You need to open the Azure Firewall to allow access from your network. This is documented below.
https://docs.microsoft.com/en-us/azure/sql-database/sql-database-firewall-configure

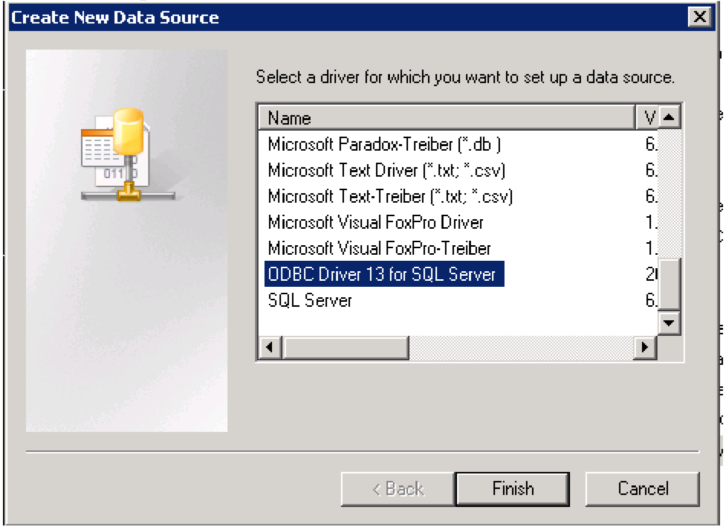
Download updated ODBC driver

You need to use the latest ODBC Driver to connect and use the Azure database. At the time of writing, this was ‘Microsoft® ODBC Driver 13 for SQL Server® - Windows + Linux‘ and can be downloaded from here, https://www.microsoft.com/en-us/download/details.aspx?id=50420&751be11f-ede8-5a0c-058c-2ee190a24fa6=True&e6b34bbe-475b-1abd-2c51-b5034bcdd6d2=True

If you do not use this newer driver, you will be prevented from using your named database and will be stuck using the default database (master).
Select the file English\X64\msodbcsql.msi, download it, install it. Simple.

Configuring ECDA Option for ODBC for Azure

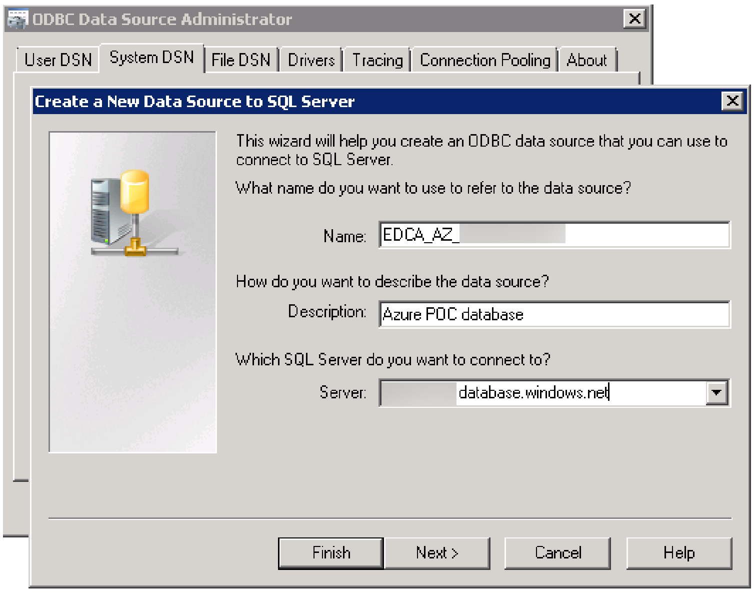
Select the updated ODBC Driver 13 for SQL Server and fill in the connection details.



Create New Data Source

Select a driver for which you want to set up a data source.

Name	V
Microsoft Paradox-Treiber (*.db)	6.
Microsoft Text Driver (*.txt; *.csv)	6.
Microsoft Text-Treiber (*.txt; *.csv)	6.
Microsoft Visual FoxPro Driver	1.
Microsoft Visual FoxPro-Treiber	1.
ODBC Driver 13 for SQL Server	2(
SQL Server	6.



ODBC Data Source Administrator

Create a New Data Source to SQL Server

This wizard will help you create an ODBC data source that you can use to connect to SQL Server.

What name do you want to use to refer to the data source?

Name: EDCA_AZ_[redacted]

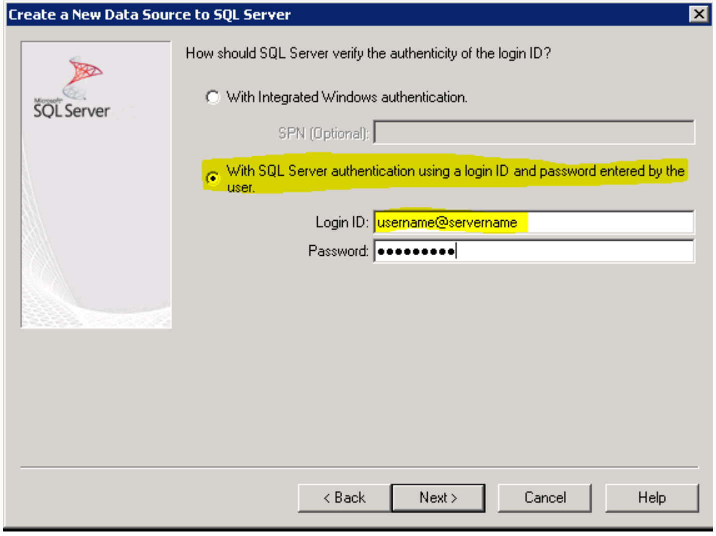
How do you want to describe the data source?

Description: Azure POC database

Which SQL Server do you want to connect to?

Server: [redacted]database.windows.net

The account you choose will also need to be in the master database. I usually setup an account with db_owner role in the target database and add as a standard user into the master database. The @servername, shown below, is optional.



How should SQL Server verify the authenticity of the login ID?

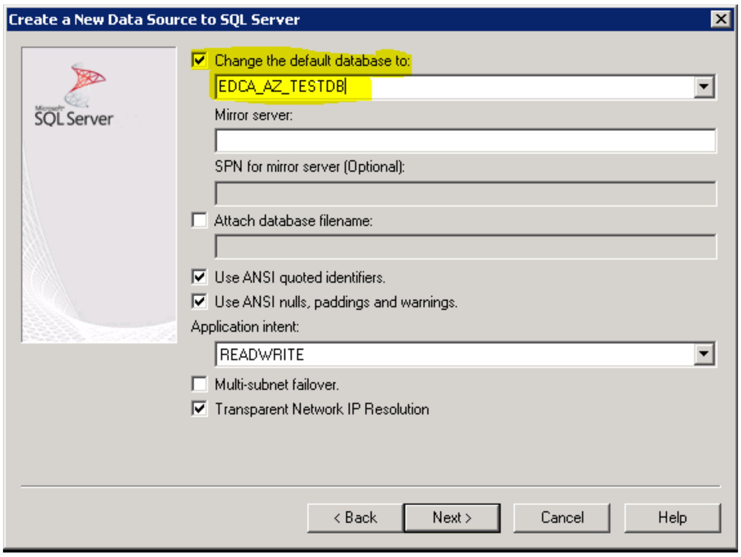
○ With Integrated Windows authentication.

SPN (Optional):

● With SQL Server authentication using a login ID and password entered by the user.

Login ID: username@servername

Password: ●●●●●●●●●



☑ Change the default database to:

EDCA_AZ_TESTDB

Mirror server:

SPN for mirror server (Optional):

□ Attach database filename:

☑ Use ANSI quoted identifiers.

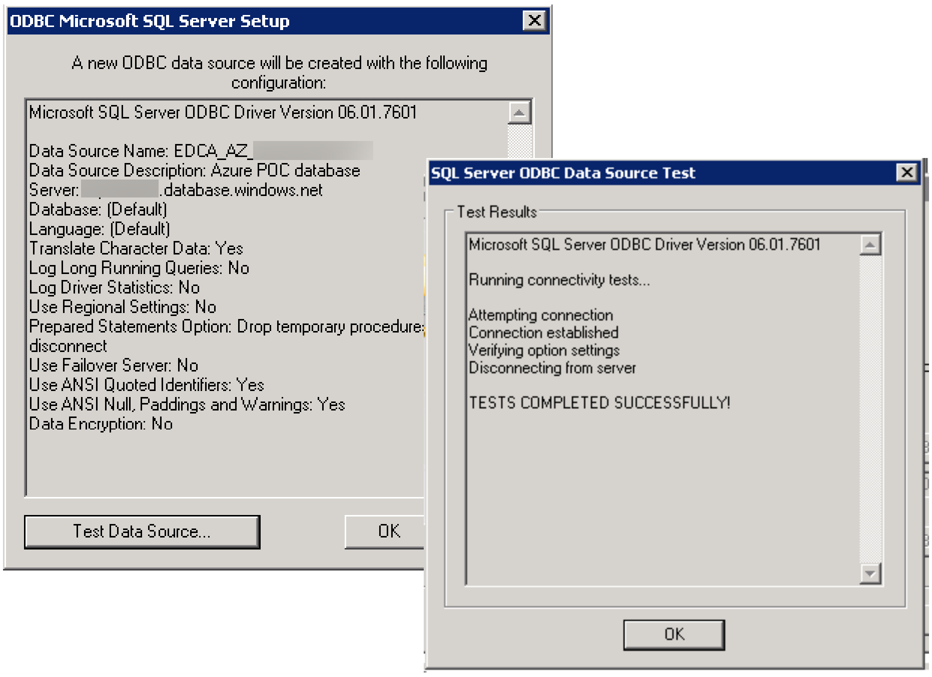
☑ Use ANSI nulls, paddings and warnings.

Application intent:

READWRITE

□ Multi-subnet failover.

☑ Transparent Network IP Resolution


ODBC Microsoft SQL Server Setup
Test Data Source...
OK
SQL Server ODBC Data Source Test
Test Results
OK

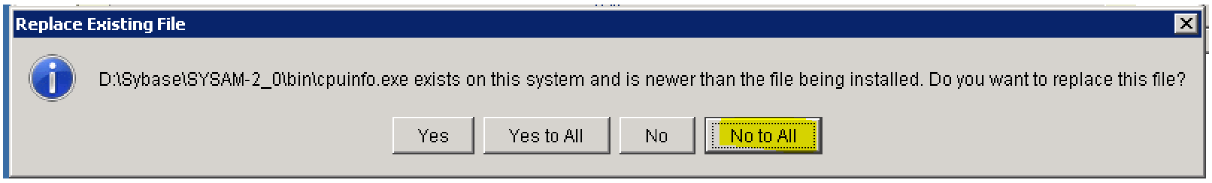
A new ODBC data source will be created with the following configuration:

```
Microsoft SQL Server ODBC Driver Version 06.01.7601

Data Source Name: EDCA_AZ_[redacted]
Data Source Description: Azure POC database
Server: [redacted].database.windows.net
Database: (Default)
Language: (Default)
Translate Character Data: Yes
Log Long Running Queries: No
Log Driver Statistics: No
Use Regional Settings: No
Prepared Statements Option: Drop temporary procedure
disconnect
Use Failover Server: No
Use ANSI Quoted Identifiers: Yes
Use ANSI Null, Paddings and Warnings: Yes
Data Encryption: No
```

```
Microsoft SQL Server ODBC Driver Version 06.01.7601

Running connectivity tests...

Attempting connection
Connection established
Verifying option settings
Disconnecting from server

TESTS COMPLETED SUCCESSFULLY!
```

Installing ECDA for Azure

1. Extract the EDCA package to a temp location on the SQL Server node and run setup
2. Choose full install.
3. If installing into the same directory as the SAP Replication server, choose to Not overwrite existing, newer files.



Replace Existing File

D:\Sybase\SYSAM-2_0\bin\cpuinfo.exe exists on this system and is newer than the file being installed. Do you want to replace this file?

Configuring and starting the DirectConnect server for Azure

1. Follow previous instructions for MSSQL
2. Set server name = UAT_DC_AZURE, port=8000
3. Name Service as EDCA_AZ_UATDB (matched ODBC name)
4. Create and start the server

Testing for Azure

1. Start DirectConnect service, `C:\sybase\DC-15_0\bin\DCStart.bat -S AZURE_TRAIN1_DC`
2. Connect using “isql‟ from a MS-DOS session

```
C:\sybase\DC-15_0>DC_SYBASE.bat
C:\sybase\DC-15_0>isql -Uusername -P<password> -S EDCA_AZ_TESTDB
```

   Verify the connection to the replicate Microsoft SQL Server database by obtaining the DBMS name and version number:

```
       select @@sqldbmsname
       go

       SQLDbmsName
       ---------------------
       Microsoft SQL Server
```

Well done. You have configured the DirectConnect server!

Replicating to Azure

When creating the connection to the Azure database, we have to use the ‘display_only’ method to print out the commands we need to run on the Azure database. See the Troubleshooting section below for more details. You must create the objects in the Azure database (rs_info, rs_update_lastcommit, etc) before you run the ‘create connection’ command.

```
create connection to EDCA_AZ_TESTDB.AZURE_DB
using profile rs_ase_to_msss;standard
set username username@servername
set password "<password>”
display_only
go
```

You may also need to change a the “grant all” statement to separate “grant SELECT, grant DELETE, etc” statements, as “grant all” is depreciated from some versions of SQL Server.

Testing replication to Azure

Create or use an existing table to test the replication follow the examples above to create a repdef, sub and mark the table for replication. Then test inserts, updates and deletes to the table and check the results in the Azure database.

Supplementary Note I - How to automatically start DirectConnect server at boot time?

Installing a DirectConnect server as a Windows service
DirectConnect no longer automatically creates the server as a Windows service. However, you can run a DirectConnect server as a Windows service. The following describes how to register, configure, start, stop, and remove DirectConnect as a Windows service. I will examine 2 methods of achiving this. **NSSM** and **ServiceWrapper**. A third method using srvany has been discounted, as it is more complex than our needs require.

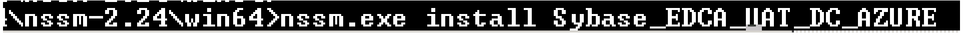
NSSM

First create a batch file that sets up the Sybase environment and then starts the Direct Connect server. I copied /Sybase/SYBASE.bat as a good starting place. Example file name is STARTDC_ECDA.bat

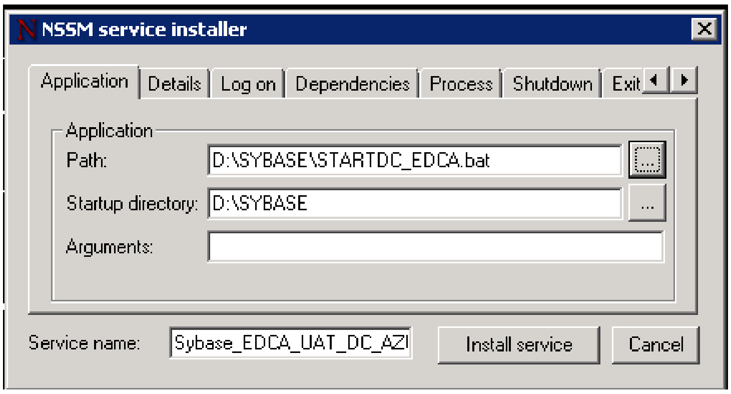
```
rem -- ECDA is needed for Sybase > SQL Server replication.
rem -- this scripts starts the Direct Connect instance NFUAT_DC_AZURE
rem #~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~##

REM
REM SAP Product Environment variables
REM
set SAP_JRE7_32=D:\SYBASE\Shared\SAPJRE-7_1_027_32BIT
set SAP_JRE7=D:\SYBASE\Shared\SAPJRE-7_1_027_64BIT
set SAP_JRE7_64=D:\SYBASE\Shared\SAPJRE-7_1_027_64BIT
set SYBASE_OCS=OCS-16_0
set INCLUDE=D:\SYBASE\OCS-16_0\include;%INCLUDE%
set LIB=D:\SYBASE\OCS-16_0\lib;%LIB%
REM
REM Replace dll, lib3p, and lib3p64 with devdll, devlib3p, and devlib3p64 when
debugging
REM
set PATH=D:\SYBASE\OCS-16_0\bin;D:\SYBASE\OCS-16_0\dll;D:\SYBASE\OCS-
16_0\lib3p64;D:\SYBASE\OCS-16_0\lib3p;%PATH%
set SYBASE=D:\SYBASE
set SYBASE_REP=REP-15_5
set PATH=D:\SYBASE\REP-15_5\bin;%PATH%
set PATH=D:\SYBASE\REP-15_5\connector\lib;%PATH%
set PATH=D:\SYBASE\REP-15_5\lib3p64\log4cxx\lib;%PATH%
set PATH=D:\SYBASE\REP-15_5\lib3p64\boost\lib;%PATH%
set PATH=D:\SYBASE\REP-15_5\lib3p64\cyrussasl\lib;%PATH%
set PATH=D:\SYBASE\REP-15_5\lib3p64\cyrussasl\lib\sasl2;%PATH%
set PATH=D:\SYBASE\REP-15_5\lib3p64\openssl\lib;%PATH%
set PATH=D:\SYBASE\REP-15_5\lib3p64\thrift\lib;%PATH%
set PATH=D:\SYBASE\REP-15_5\lib64;%PATH%


rem now start Direct Connect server for UAT_DC_AZURE
D:\sybase\DC-15_0\bin\DCStart.bat -SUAT_DC_AZURE
```

Download a Service Manager, like NSSM from http://nssm.cc/ and extract to your preferred location. I suggest saving to C:\Program Files\nssm-2.24

Open MS-DOS and to use the GUI version of the tool, run (change your service name to suit)



```
\nssm-2.24\win64>nssm.exe install Sybase_EDCA_UAT_DC_AZURE
```



NSSM service installer

Application
Path: D:\SYBASE\STARTDC_EDCA.bat
Startup directory: D:\SYBASE
Arguments:
Service name: Sybase_EDCA_UAT_DC_AZI

Select the batch file to start and the 'Install service'. Go to 'services' and start the new service.

ServiceWrapper

http://infocenter-archive.sybase.com/help/index.jsp?topic=/com.sybase.dc35394_0110/html/di_inst_win/di_inst_win41.htm

Example:

```
ServiceWrapper.exe --install SYBDC_MSSQL_TRAIN1_DC --username=<username> --password=<password>
C:\sybase\DC-15_0\bin\DCStart.bat -S MSSQL_TRAIN1_DC
```

Make sure account as ‘start as a service’ permission

Supplementary Note II – adding additional ‘Services’

You can have multiple database connections against a single Direct Connect server. These are called (rather confusingly, Services). The following is an example of how to add a new “Service” to an existing DC server.

1. Create new ODBC connection to target server (MSSQL) and specify the name of the target database.
2. Next, to help the DC know how to connect to the new service, add the following to the SQL.INI file. Note that the Port number is the same for the DC and all services that run under it

```
[EDCA_TRAIN2]
query=NLWNSCK, vm-train, 8000
win3_query=NLWNSCK, vm-train, 8000
```

3. Finally, configure the new “Service” by adding the following to the C:\Sybase\DC-15_0\servers\<server_name>\cfg\dcany.cfg file.

```
[ECDA_TRAIN2]
{ACS Required}
ConnectionSpec1=ECDA_TRAIN2
{Client Interaction}
EnableAtStartup=yes
TransactionMode=short
SendWarningMessages=yes
{Target Interaction}
Allocate=connect
SQLTransformation=passthrough
ReturnNativeError=yes
{Catalog Stored Procedures}
CSPColumnODBCVersion=3
```

4. Restart the DirectConnect server

Supplementary Note III – recommended config values from Sybase engineering

Below are the appropriate DC/ECDA config settings for a replication environment. Please compare with your current settings and update accordingly.

Allocate

Controls when an access service allocates conversations with the target database
system. Syntax Allocate=[connect | request]
Default connect

Values

connect specifies an access service to allocate the conversation when the client connects, and holds it open for the duration of the client connection.

request specifies an access service to allocate a new conversation each time the client application sends a request, and deallocates the conversation after each request.

Note: There is a large performance penalty when using the request setting.

TransactionMode

Specifies whether the access service or the client application manages commit and rollback statements.

Syntax TransactionMode=[short | long]

Default short
Values:-

long specifies the access service to give commitment control to the client application.
short specifies the access service to issue a commit or a rollback after each request.

Comment: The access service holds open the connection to the data source until the client application issues a commit or rollback, or until the ClientIdleTimeout value is exceeded.

SQLTransformation

Specifies the mode the access service uses for SQL transformation.
Syntax SQLTransformation=[passthrough | sybase | tsql0 | tsql1 | tsql2]

Default passthrough
Values
passthrough specifies an access service to send all SQL statements to the database system as received, without transformation. A client application uses passthrough mode to gain direct access to DBMS capabilities. *sybase* specifies an access service to perform SQL transformation of selected statements. It also allows the use of multi-part table names with the view command in SQL statements.

CSPColumnODBCVersion

Specifies ODBC version that catalog stored procedures results conform to.
This affects interoperability with ASE/CIS.

Syntax CSPColumnODBC Version = [2 | 3]

Default
3 Values
2 specifies ASE/CIS version 12.0.
3 specifies ASE/CIS version 12.5 and later.
This property affects interoperability with ASE/CIS.

ReturnNativeError

Allows a non-localized native error message and a native error severity to be returned to the client.

Syntax ReturnNativeError = [yes | no]

Default
no Values
yes specifies non-localized native error messages are returned to the client.
no specifies non-localized native error messages are not returned to the client.

SendWarningMessages

Specifies whether an access service returns warning messages to the client application.

Syntax SendWarningMessages=[no | yes]

Default
no Values
no specifies the access service not to return warning messages to the client application. *yes* specifies the access service to return warning messages to the client application.

Supplementary Note IV – Tracing

“TraceOpenServer =59” in server.cfg file & restart DC

Then see resulting files:
%SYBASE%\%SYBASE_ECON%\SERVERNAME\log\srv.log
%SYBASE%\%SYBASE_ECON%\ SERVERNAME\log\SERVERNAME.trc

Supplementry Note V – Sample dcany.cfg file

SAP support recently provided me with this sample dcany.cfg file that they use in-house

```
[Service Library]
{Logging}
LogSvcLibStatistics=0
{Client Interaction}
SvclibDescription=Access Service Library for ODBC.

[dcmssql_DEMO2]
{Logging}
LogConnectionStatistics=yes
{ACS Required}
ConnectionSpec1=dcmssql_DEMO2
{Client Interaction}
EnableAtStartup=yes
quoted_identifier =on
{Target Interaction}
SQLTransformation=sybase
{Tracing}
TraceTarget=yes
TraceEvents=yes
TraceInterface=yes
```

Troubleshooting

The online documentation for this option is not brilliant, so when issue occur, they can prove difficult to fix. SAP tech support are very helpful but you may want to try a few things first.

- Can you connect using ISQL? Use ISQL to test connections from the replication server host to the ECDA service. Make sure you have added in the ECDA details into the SQL.ini file local to the host you are testing from.

- Is the Direct Connect server running? You can start this as a batch file or see notes above to get it added as a service.

- Connections DOWN when running ‘admin who’ in repserver. If these are down when you run CREATE CONNECTION, then it could be an issue with the script that is silently run when you create a connection using a ‘profile’. There is a way you can step through each command the above ‘profile’ script executes. Re-run the CREATE CONNECTION command but add the ‘**display_only**’ option on the last line. It returns the list of commands it will execute at the replicate, including creating the rs_info and rs_lastcommit tables. You can take the SQL it spits out and run it manually from a ISQL session into the SQL Server database, then resume the connection.

  ```
  create connection to ECDA_TRAIN1.pubs2
  using profile rs_ase_to_msss;standard
  set username rep_maint
  set password "rep_maint_ps"
  display_only
  go
  ```

- This gives you all the commands you need:-

  ```
  -- I normally run these from the repserver using an ISQL session like the one below.
  >> isql -S ECDA_TRAIN1 -Urep_maint -Prep_maint_ps
  Copy and paste “your” output into the SQL Server session. Do this first and create the
  connection in the repserver once you have done this bit.

  drop table rs_info
  go
  create table rs_info (rskey varchar (20), rsval varchar (20))
  go
  insert into rs_info values ('charset_name', 'iso_1')
  insert into rs_info values ('sortorder_name', 'bin_iso_1')
  go
  drop table rs_lastcommit
  go
  create table rs_lastcommit (origin int, origin_qid binary(36), secondary_qid
  binary(36), origin_time datetime, dest_commit_time datetime)
  ```

```
go
create unique clustered index rs_lastcommit_idx on rs_lastcommit(origin)
go
drop procedure rs_update_lastcommit
go
create procedure rs_update_lastcommit @origin int,@origin_qid
binary(36),@secondary_qid binary(36),@origin_time datetime as update rs_lastcommit set
origin_qid = @origin_qid, secondary_qid = @secondary_qid,origin_time =
@origin_time,dest_commit_time = getdate() where origin = @origin if (@@rowcount = 0)
begin insert rs_lastcommit (origin, origin_qid, secondary_qid,origin_time,
dest_commit_time) values (@origin, @origin_qid, @secondary_qid, @origin_time,
getdate()) end
go
drop table rs_ticket_history
go
create table rs_ticket_history (cnt numeric(8,0) identity,h1 varchar(10),h2
varchar(10),h3 varchar(10),h4 varchar(50),pdb varchar(30),prs varchar(30),rrs
varchar(30),rdb varchar(30),pdb_t datetime, exec_t datetime,dist_t datetime,rsi_t
datetime,dsi_t datetime,rdb_t datetime default getdate(),exec_b numeric(22,0),rsi_b
numeric(22,0),dsi_tnx numeric(22,0),dsi_cmd numeric(22,0),ticket varchar(1024))
go
create unique index rs_ticket_idx on rs_ticket_history(cnt)
go
grant all on rs_ticket_history to public
commit
go
drop procedure rs_send_repserver_cmd
go
CREATE PROCEDURE rs_send_repserver_cmd @rs_api VARCHAR(8000) AS declare @cmd
VARCHAR(8000), @sql varchar(50) BEGIN if (patindex('rs_rcl', lower(@rs_api)) > 0)
begin print 'The Replication Server command should not contain the keyword ''rs_rcl'''
return(1) end select @cmd = 'rs_rcl ''' + replace(@rs_api, '''', '''''') + ''' rs_rcl'
if ('rs_rcl' != substring (@cmd, datalength(@cmd) - 5, 6)) begin print 'The
Replication Server command is too long.'  print 'Please split it into two or more
commands' return (1) end set @sql = 'rs_marker' exec @sql @cmd END
commit
go
drop table rs_threads
go
create table rs_threads(id int,seq int CONSTRAINT PK_rs_threads PRIMARY KEY
CLUSTERED(id ASC))
go
sp_indexoption 'rs_threads','disallowpagelocks',TRUE
go
grant select on rs_threads to public
commit
go


--now that all our database objects have been created in your replicate database, you
can now go ahead and create the connection using the commands that were outputted.
Note, this DOES not use the ‘profile’ clause.


create connection to ECDA_TRAIN1.pubs2
set error class to rs_msss_error_class
set function string class to rs_msss_function_class
set username to rep_maint
set password to rep_maint_ps
set batch to 'off'
set dsi_dataserver_make to ase
set dsi_connector_type to ctlib
set dsi_do_decompression to 'on'
go
```

- When setting up replication to Azure, make sure you disable **batch** and **dynamic SQL** for the EDCA connection.

```
alter connection to ECDA_TRAIN1.pubs2
Set dsi_bulk_copy to ‘off’
go
alter connection to ECDA_TRAIN1.pubs2
set dynamic_sql to ‘off’
go
```

Errors you will receive for these settings do not at first glance look anything to do with the issue.

```
Error you get for dsi_bluk_copy
Message: 195, State 0, Severity 15 -- '[[Message Iteration=1|Data Source Name=
ECDA_TRAIN1|SQLState=42000|Native Error=195|Message=[Microsoft][ODBC Driver 13 for SQL
Server][SQL Server]'char_convert' is not a recognized SET option.]'
Message: 207, State 0, Severity 19 -- '[[Message Iteration=1|Data Source Name=
ECDA_TRAIN1|SQLState=42S22|Native Error=207|Message=[Microsoft][ODBC Driver 13 for SQL
Server][SQL Server]Invalid column name 'sysstat2'.
```

Errors you get for dynamic_sql

```
Message: 30291, State 0, Severity 19 -- '[[Message Iteration=1|Data Source Name=
ECDA_TRAIN1|SQL Function=INSERT|SQLState=24000|Native Error=0|Message=[Microsoft][ODBC
Driver 13 for SQL Server]Invalid cursor state]'.
```