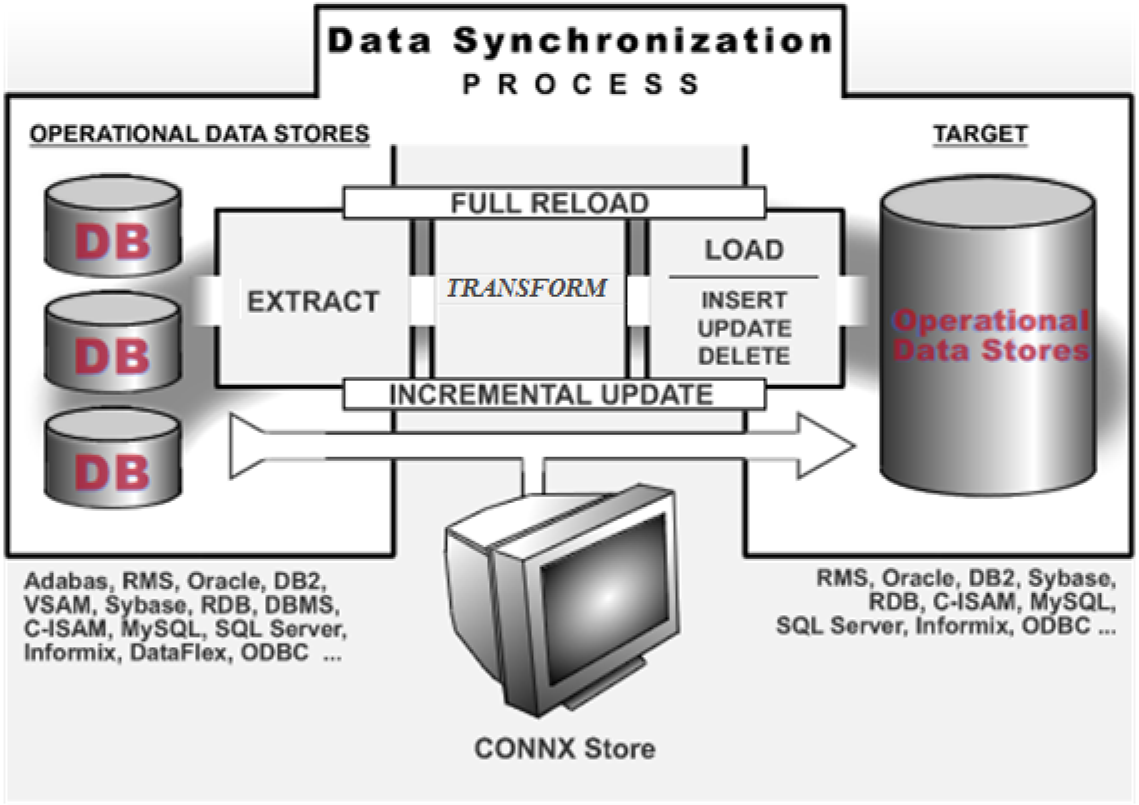
CONNX Data Synchronization 14.6

Table Of Contents

DataSync

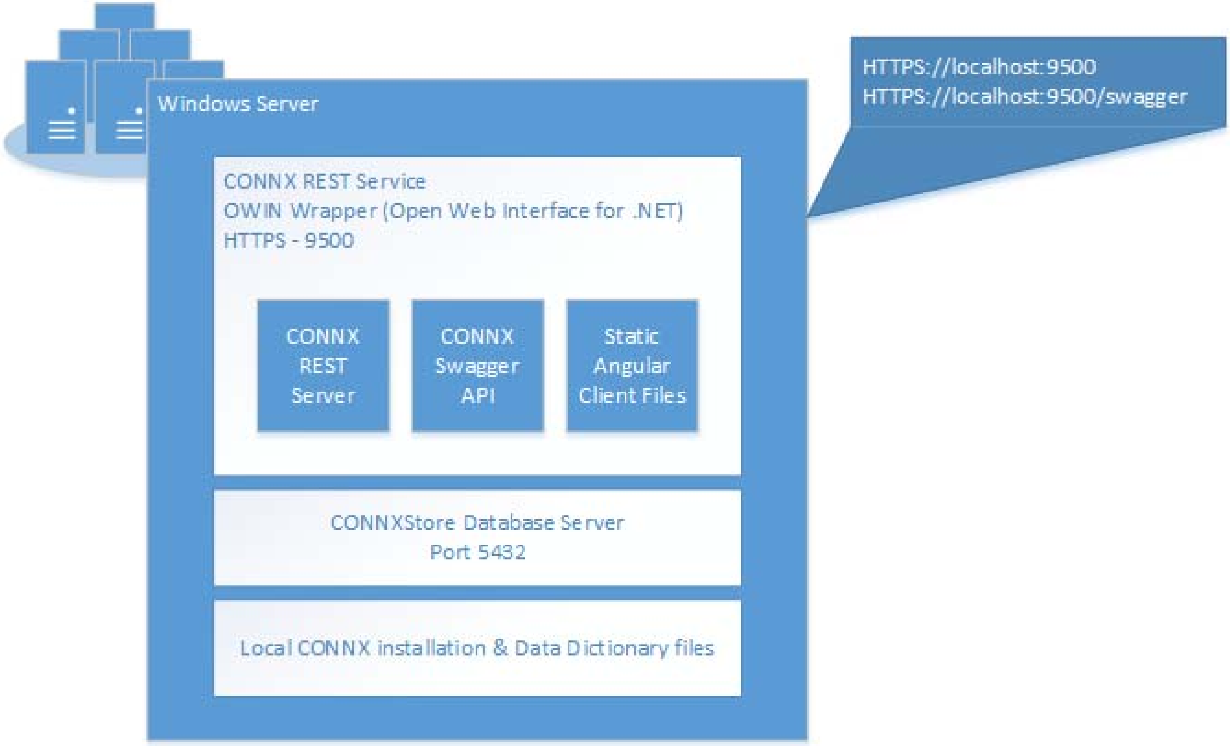
Introduction

The **CONNX Data Synchronization Tool** pumps data to a data mart or a data warehouse. With CONNX Data Synchronization, users can move a complete snapshot of a pre-defined list of tables, views or complex transformations into the Data Warehouse. The tool can also incrementally update the current snapshot of data through a simple-to-use scheduling interface. This accomplishes the goal of "near real-time" data synchronization.


Data Synchronization
PROCESS
OPERATIONAL DATA STORES
TARGET
DB
DB
DB
FULL RELOAD
EXTRACT
TRANSFORM
LOAD
INSERT
UPDATE
DELETE
INCREMENTAL UPDATE
Operational Data Stores
Adabas, RMS, Oracle, DB2, VSAM, Sybase, RDB, DBMS, C-ISAM, MySQL, SQL Server, Informix, DataFlex, ODBC ...
RMS, Oracle, DB2, Sybase, RDB, C-ISAM, MySQL, SQL Server, Informix, ODBC ...
CONNX Store

DataSync Server Architecture

The CONNX REST Service runs on a Windows Server and can be accessed via a web browser via HTTPS. An OWIN Web interface feeds an Angular client to the browser which then interacts with the CONNX server via REST API calls. This architecture not only supports the built in angular client supplied by DataSync, but also provides tremendous power to businesses who wish to build their own custom automated tasks via REST calls. Every REST API used by the Angular client is fully documented in swagger pages that can be accessed from the CONNX Server.


Windows Server
CONNX REST Service
OWIN Wrapper (Open Web Interface for .NET)
HTTPS - 9500
CONNX REST Server
CONNX Swagger API
Static Angular Client Files
CONNXStore Database Server
Port 5432
Local CONNX installation & Data Dictionary files
HTTPS://localhost:9500
HTTPS://localhost:9500/swagger

System Requirements

The system requirements for the CONNX Data Synchronization Tool are listed in the table below:

REST/Web Server

Required	**Recommended**
Windows 2012 Server or above / Windows 10	Windows 2012 Server or above / Windows 10
Server	
200 GB Free Space	200+ GB Free Space
1 GHz or higher CPU	4+ core - 2 GHz or higher CPU
2 GB RAM	8 GB RAM
standard disk drive controller	Raid controller or mirrored drive controller
100 MB/sec network card or WAN connection	1 GB network card or LAN connection

DataSync Client

The Chrome and Firefox browsers are supported running on a Windows or Linux platform.

Migrating from an Earlier Version of DataSync

Starting with version 14.5, CONNX DataSync is now a REST/Web server and can be accessed from any machine on the network via a browser interface. The stand alone version of DataSync has been deprecated but will still be shipped with CONNX as DataSync Classic.

Below are points to be considered when upgrading from previous versions of DataSync Classic:

DataSync uses a migration utility to help upgrade the CONNXSTORE database to include necessary changes to the database. A program called "DatapumpUpgrade" is run during the install to perform this task. In certain cases (when upgrading from CONNX 8.8, 8.9, and 9.0 SP1), the upgrade cannot work because the CDD must first be saved in the CONNX 11 (and later) format. The upgrade will succeed when the necessary actions listed in the table below are completed.

The following matrix displays actions that should be taken when upgrading to CONNX DataSync 14.5:

Migration from CONNX VERSION	ACTION
8.8, 8.9, and 9.0 SP1	Save the CDD in the current format by opening the CONNX Data Dictionary and re-saving it Re-create DataSync schedules (necessary because of third-party control changes)
9.0 SP2 to 10.5 build 8331	Re-create DataSync schedules (necessary because of third-party control changes)
11.0 to 14.0	DataSync now provides a REST interface which is accessed from a browser. The previous native Windows version of the GUI has been deprecated and is available on the Windows Start menu as DataSync Classic. When the DataSync REST Server first starts, it will use the CDD that was in use by the previous version of DataSync. Any new CDDs activated with the REST server will be located in the connx32\datasync\cdd folder.

Summary of Product Functions

DataSync provides several unique product functions.

- **Full reloads on demand**

  Purges all prior snapshot data in the data warehouse, replacing it in the data warehouse with a completely new snapshot.
- **Incremental updates on demand**

  Detects which records have been added, deleted, or changed and inserts, updates, or deletes the corresponding data in the data warehouse snapshot tables at a fraction of the usual time. The incremental update runs up to 10 times faster than the full update, a significant decrease in the time it takes to synchronize.
- **Scheduled incremental updates**

  Detects which records have been added, deleted, or changed and inserts, updates, or deletes the corresponding data in the data warehouse snapshot tables. Each file in the synchronization process may be scheduled either independently or as a group.
- **Scheduled full reloads**

  Purges all prior snapshot data in the data warehouse. A completely new snapshot of the data is placed in the data warehouse. Each file in the synchronization process may be scheduled either independently or as a group.
- **Datamart model schema creation**

  Automatically performs a complete analysis of all source database schemas and performs complete, applicable transformations for the target datamart schema.
- **Error detection and correction**

  Provides options for automatically detecting error conditions and resyncing until a successful sync has been performed.
- **Task triggering**

  Additional tasks can be triggered based on the status of a synchronization. A task can be as simple as sending an e-mail to notify that a sync was successful or as complex as calling an ETL (Extract, Transform, Load) tool to perform additional transformations against the synchronized data.
- **Transformations**

  Transforms can be created to change or manipulate data that is being synced from the source to the target. Transformations, such as column expressions, column mapping, aggregation, lookups, unions, etc. can be developed. This enables DataSync to have ETL (Extract, Transform, Load) functionality.

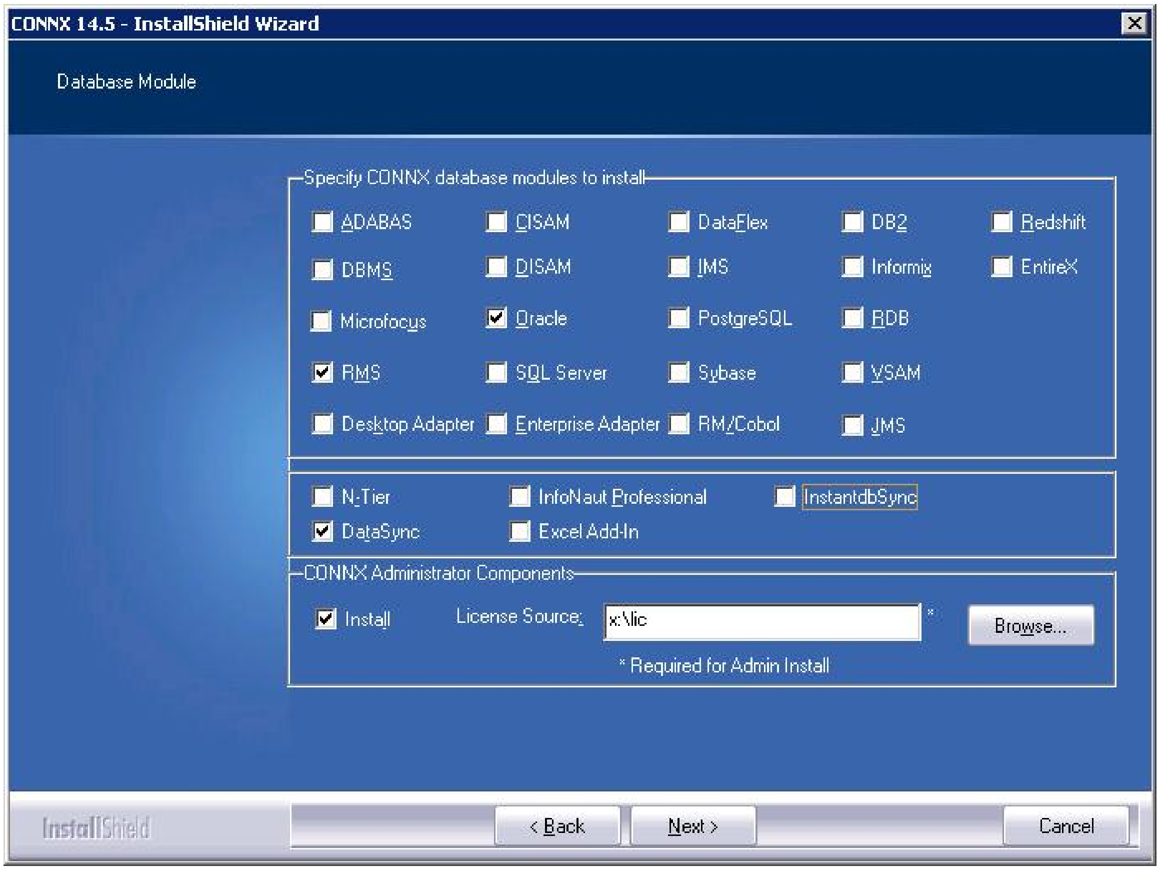
Installing DataSync

Prior to installing DataSync, please read Installing SSL Certificate for use with REST server. If you intend to use your own SSL certificate, it should be bound to port 9500 prior to running the DataSync installation procedure. If the DataSync installer does not find an SSL certificate bound to port 9500, it will create a self-signed certificate and bind it to port 9500 automatically.

DataSync is installed during a CONNX installation. When the Database Module screen is shown, DataSync should be checked. It is necessary to have the appropriate DataSync license. See the **CONNX Installation Guide** for more information.

1. Install CONNX. The **DataSync** module must be licensed and selected before continuing with installation. Click the **Next** button.

2. ***Important:*** *If you already have one or more CONNX licenses, you must select the check box for each database for which you are licensed. Failure to do so will cause the license for that database to uninstall.*



Specify CONNX database modules to install

☐ ADABAS	☐ CISAM	☐ DataFlex	☐ DB2	☐ Redshift
☐ DBMS	☐ DISAM	☐ IMS	☐ Informix	☐ EntireX
☐ Microfocus	☑ Oracle	☐ PostgreSQL	☐ RDB	
☑ RMS	☐ SQL Server	☐ Sybase	☐ VSAM	
☐ Desktop Adapter	☐ Enterprise Adapter	☐ RM/Cobol	☐ JMS	

☐ N-Tier	☐ InfoNaut Professional	☐ InstantdbSync
☑ DataSync	☐ Excel Add-In	

CONNX Administrator Components

☑ Install   License Source: x:\lic *

* Required for Admin Install

3. Once CONNX and DataSync are installed, you may need to **restart** your computer. Please do so if prompted.
4. Confirm that the **CONNXStore Database Server Service**, the **CONNXDataSync REST Service** and the **CONNX DataSync Schedule Service** are running by going to the **Start** menu. Point to **Settings,** and then click **Control Panel.** Click **Administrative Tools,** and then click **Services.** The **Status** column for both services should read **Started.** Close **Control Panel.**

Methods of Synchronization

DataSync offers four options to synchronize the source data:

Automatic	This is the default option. Except for the first synchronization, when a Full Reload is performed, subsequent synchronizations will attempt to use a CRC algorithm to detect and synchronize only changed, inserted, or deleted records. This can cause dramatic improvement in synchronization times.
Incremental Update	Performs an Incremental Update, if possible. If not, a Full Reload synchronization will be performed.
Full Reload	This will do a Full Reload synchronization from the source to the target.
Incremental Only	In some cases, an Automatic synchronization will revert to a Full Reload; for example, when the number of records changed have exceeded a threshold. When Incremental Only is selected, under NO circumstances will the data be dropped and reloaded to the target. This is advantageous when source data is operational 24x7.

Default Port numbers

The following is a list of 32 and 64 bit components and the default ports they listen on:

Component Name	Default Port
32 bit Enterprise Server Service (Windows)	6500
64 bit Enterprise Server Service (Windows)	6502
License Server (Windows, Unix)	7501
DataSync Rest Server	9500

If CONNX is installed in an environment where a firewall is present, these ports need to be opened.

Glossary of Icons

Below is a list of the icons used in the DataSync Administrator, their meanings and where they can be found.

- **Connection State**

  Indicates ***connected to a CDD*** or ***not connected to a CDD*** respectively.
  The *connection state button* appears in the upper right section of the main page. See Managing Connections for information about connecting to a CDD.

- **Help**

  Clicking this button opens a menu to the main help system.
  The *help button* appears in the upper right section of the main page.

- **Gear**

  Clicking this button opens a menu to Utilities and Settings.
  The *gear button* appears in the upper right section of the main page.

- **User**

  Clicking this button opens the user management menu. Only a user with a role of Administrator will have the User Management option. All other users will only have the option to Logout.
  The *user button* appears in the upper right section of the main page.

- **Search Filter**

  The search filter is a button represented by a funnel icon. When pressed, it will activate the ability to enter search criteria on a list to the purpose of filtering the results in the list.
  The *Search Filter* can be found on table lists found in the Overview, Tables, Transformations, Groups and Schedules tabs.

- **Status**

  The status icons indicate success or error after a sync task finishes.
  The *status icons* can be found on table lists found in the Overview, Tables, Transformations, Groups and Schedules tabs.

- **Edit**

  The edit button allows an item to be edited.
  The *edit button* can be found on the table lists in the Transformations, Groups and Schedules tabs

- **Delete**

  The delete button allows an item to be deleted.
  The *delete button* can be found on the table lists in the Transformations, Groups and Schedules tabs

- **Duplicate**

  The duplicate button allows transformations to be duplicated
  The *duplicate button* can be found on the Transformations tab

- **Import**

  The import button allows transformation definitions to be imported from an XML file.
  The *import button* can be found on the Transformations tab

- **Export**

  The export button allows transformation definitions to be exported to an XML file.
  The *export button* can be found on the Transformations tab

- **More Information Chevron**

  ⌄

  This chevron indicates that a table row can be expanded to show more information. There are times such as when an error message is present that all the information will not fit in a single row. Clicking on this chevron will display the extra information below the row.
  The *more information chevron* can be found on table lists found in the Overview, Tables, Transformations, Groups and Schedules tabs.

- **Less Information Chevron**

  ›

  This chevron indicates that a table row can be collapsed to show less information. Clicking this chevron will collapse any previously expanded rows of information.
  The *less information chevron* can be found on table lists found in the Overview, Tables, Transformations, Groups and Schedules tabs.

- **Up and Down arrows**

  ↑ ↓

  These arrows are used to move a selected row up or down in a list.
  The *up and down arrows* can be found on table lists in the Column Mapper

- **Right and Left arrows**

  → ←

  These arrows are used to move selected listbox items from one listbox to another.
  The *right and left arrows* can be found on the Group Properties dialog.

- **Download**

  ⤓

  The download button is used to download a CDD from the DataSync Server to the local browser. The *download button* can be found on the Select CDD screen in the Manage Connections stepper.

- **Cancel Task**

  ⊗

  Cancels the currently running synchronization task.
  The *Cancel Task* button can be found on the Current Syncs page

Installing an SSL Certificate for use with the REST Server

DataSync uses the HTTPS protocol to communicate between the REST server and the client running in a browser. The HTTPS protocol requires a valid SSL certificate for the REST server. During the installation of DataSync, a self-signed SSL certificate is installed on the server, and bound to the default port of 9500.

When using DataSync in production there are 3 likely scenarios about SSL certificate usage.

- Scenario 1 – Continued usage of the self-signed SSL certificate created by DataSync.
- Scenario 2 – Usage of SSL certificate created by an internal corporate Trusted Root Authority.
- Scenario 3 – Usage of SSL certificate created by an external Trusted Root Authority. (for example, IdenTrust, DigiCert, Sectigo, GoDaddy, etc – see the Wikipedia page on Certificate authority for more details)

Continued usage of the self-signed SSL certificates created by DataSync

The DataSync installation creates a self-signed certificate. Since the certificate is self-signed and the Trusted Root Certificate is not recognized by any other PC, accessing DataSync from other PCs in the environment requires additional steps on each of these PCs.

After installation, the self-signed certificate is installed and bound to the default port of 9500. If the installer detects that a certificate has already been bound to port 9500, the existing certificate is left alone and not replaced.

The self-signed certificate created by the installation can be found in your 64bit installation directory, under the openssl folder.
The name of the file is called **DataSyncWebAPI.crt**.

The following steps will enable another PC to access the DataSync Angular client.

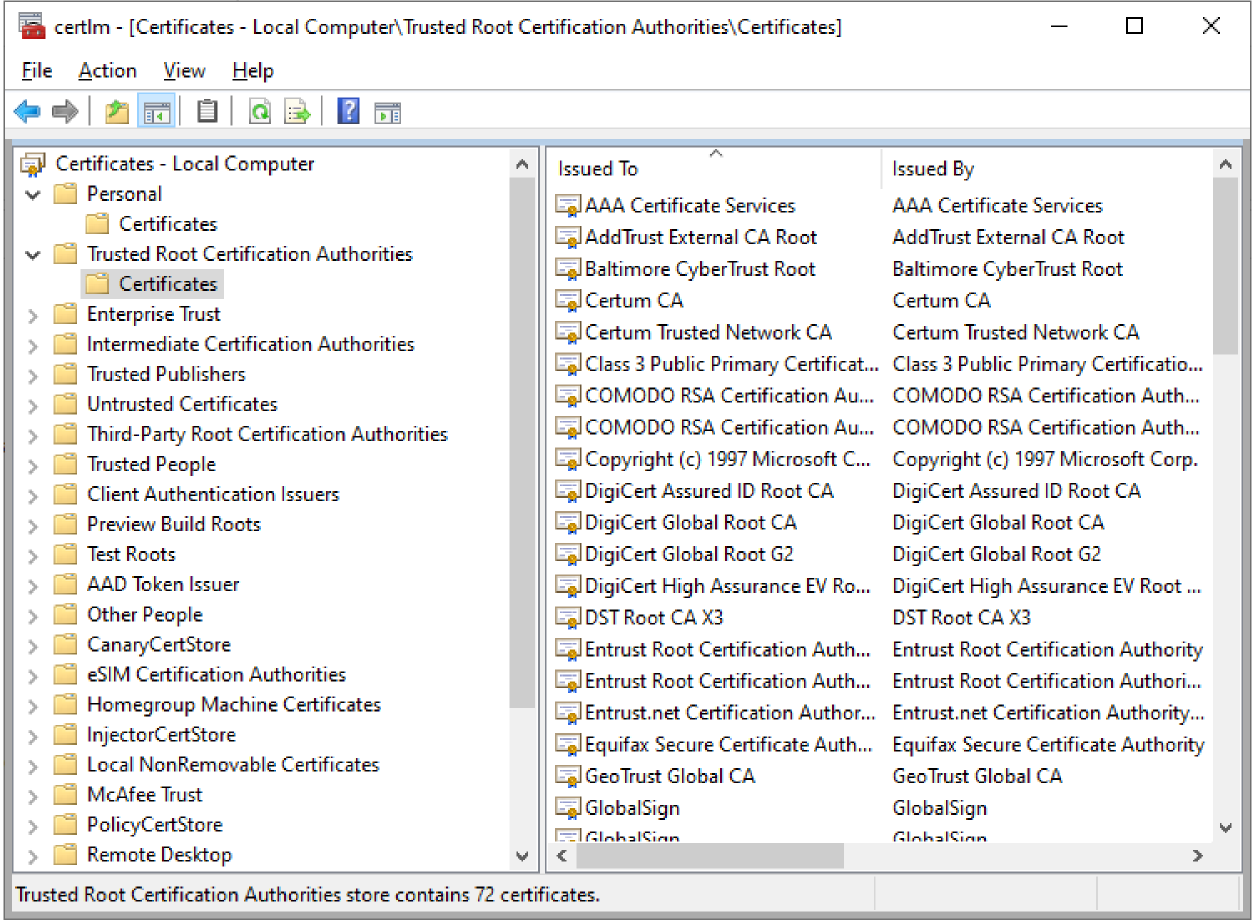
Step 1: Install the certificate under Trusted Root Certification Authorities

1.1. Copy the DataSyncWebAPI.crt from the DataSync server to the new PC.

1.2. Right-click ‘Start’ and select ‘Run’.

1.3. Type in ‘**certlm.msc**’ and right mouse click on the item and select “run as administrator”.

1.4. Expand ‘Trusted Root Certification Authorities’ in the left pane, right click on ‘Certificates’, select ‘All Tasks’ and then ‘Import’.


certlm - [Certificates - Local Computer\Trusted Root Certification Authorities\Certificates]
File
Action
View
Help

- Certificates - Local Computer
  - Personal
    - Certificates
  - Trusted Root Certification Authorities
    - Certificates
  - Enterprise Trust
  - Intermediate Certification Authorities
  - Trusted Publishers
  - Untrusted Certificates
  - Third-Party Root Certification Authorities
  - Trusted People
  - Client Authentication Issuers
  - Preview Build Roots
  - Test Roots
  - AAD Token Issuer
  - Other People
  - CanaryCertStore
  - eSIM Certification Authorities
  - Homegroup Machine Certificates
  - InjectorCertStore
  - Local NonRemovable Certificates
  - McAfee Trust
  - PolicyCertStore
  - Remote Desktop

Issued To	Issued By
AAA Certificate Services	AAA Certificate Services
AddTrust External CA Root	AddTrust External CA Root
Baltimore CyberTrust Root	Baltimore CyberTrust Root
Certum CA	Certum CA
Certum Trusted Network CA	Certum Trusted Network CA
Class 3 Public Primary Certificat...	Class 3 Public Primary Certificatio...
COMODO RSA Certification Au...	COMODO RSA Certification Auth...
COMODO RSA Certification Au...	COMODO RSA Certification Auth...
Copyright (c) 1997 Microsoft C...	Copyright (c) 1997 Microsoft Corp.
DigiCert Assured ID Root CA	DigiCert Assured ID Root CA
DigiCert Global Root CA	DigiCert Global Root CA
DigiCert Global Root G2	DigiCert Global Root G2
DigiCert High Assurance EV Ro...	DigiCert High Assurance EV Root ...
DST Root CA X3	DST Root CA X3
Entrust Root Certification Auth...	Entrust Root Certification Authority
Entrust Root Certification Auth...	Entrust Root Certification Authori...
Entrust.net Certification Author...	Entrust.net Certification Authority...
Equifax Secure Certificate Auth...	Equifax Secure Certificate Authority
GeoTrust Global CA	GeoTrust Global CA
GlobalSign	GlobalSign
GlobalSign	GlobalSign

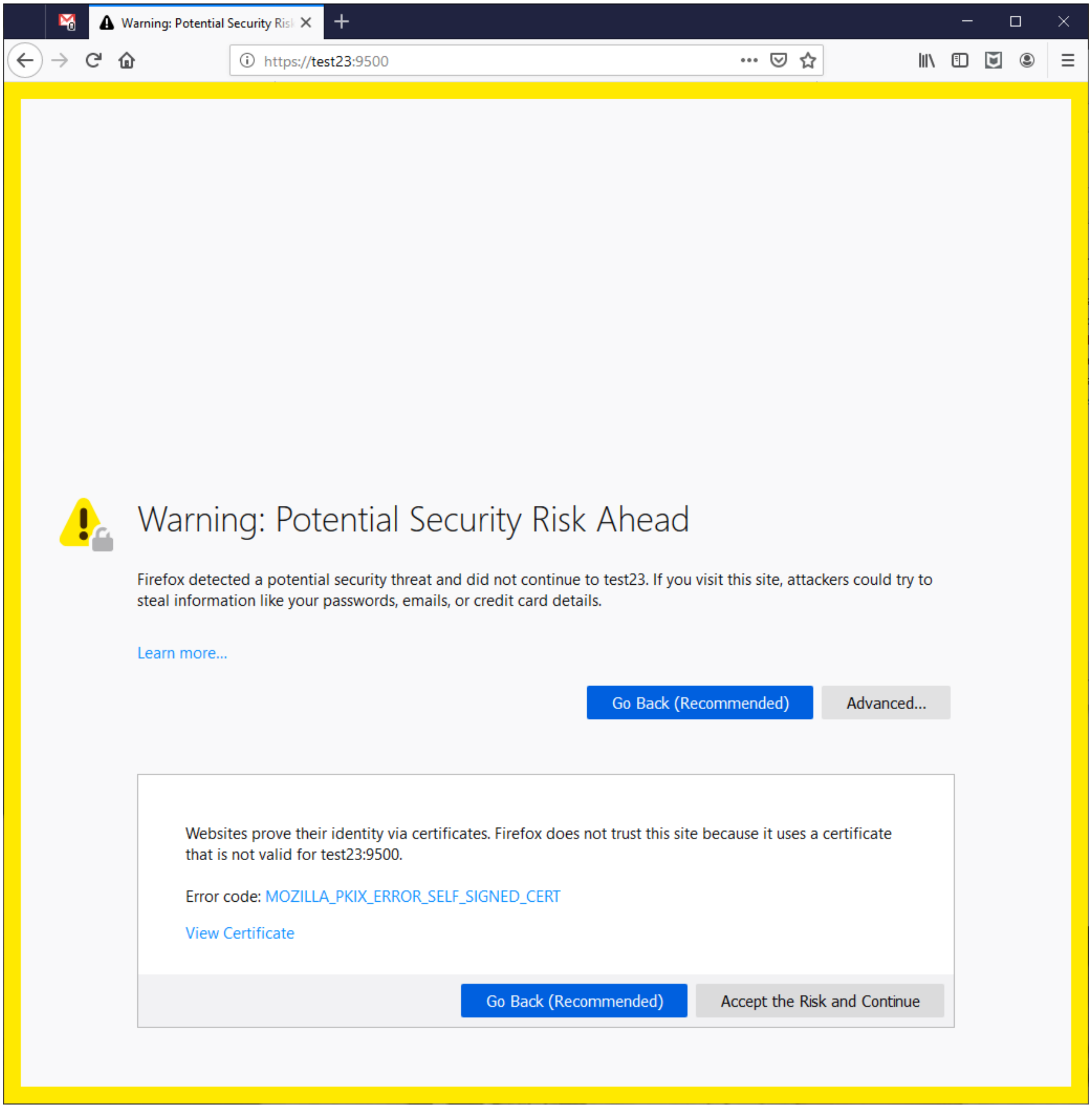
Trusted Root Certification Authorities store contains 72 certificates.

1.5. Follow the steps in the ‘Certificate Import Wizard’ to complete the certificate installation.

Step 2: Configure Firefox for use with Self-Signed Certificates

Since the trusted root certificate is self-signed, the Firefox browser will still issue a security warning. Please follow the instructions in the section Configuring Web Browsers for use with Self-Signed Certificates.

Depending on the Trusted Root Certificate, a freshly installed Firefox browser may display the following warning page when CONNX DataSync is accessed:



Warning: Potential Security Risk Ahead

Firefox detected a potential security threat and did not continue to test23. If you visit this site, attackers could try to steal information like your passwords, emails, or credit card details.

Websites prove their identity via certificates. Firefox does not trust this site because it uses a certificate that is not valid for test23:9500.

Error code: MOZILLA_PKIX_ERROR_SELF_SIGNED_CERT

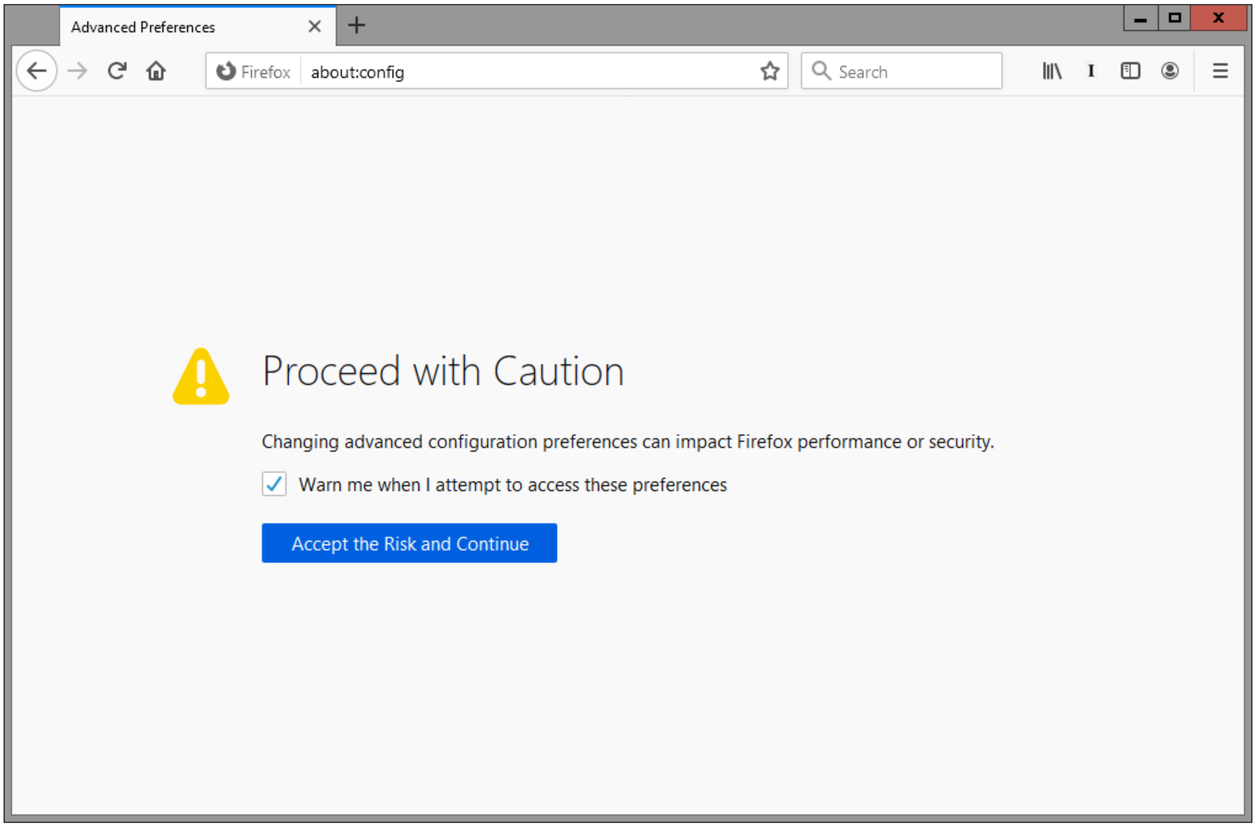
If this warning is displayed, you can then press the advanced button and accept the risk of going to the page and continue on to the DataSync site.

To permanently suppress this message and eliminate the need to add an exception for the certificate, Firefox must be configured to allow enterprise root certificates:

Note: This requires Firefox 68.8 or higher.

2.1. Open the Firefox browser.

2.2. In the address edit box type the following: about:config, this will take you to the following page:



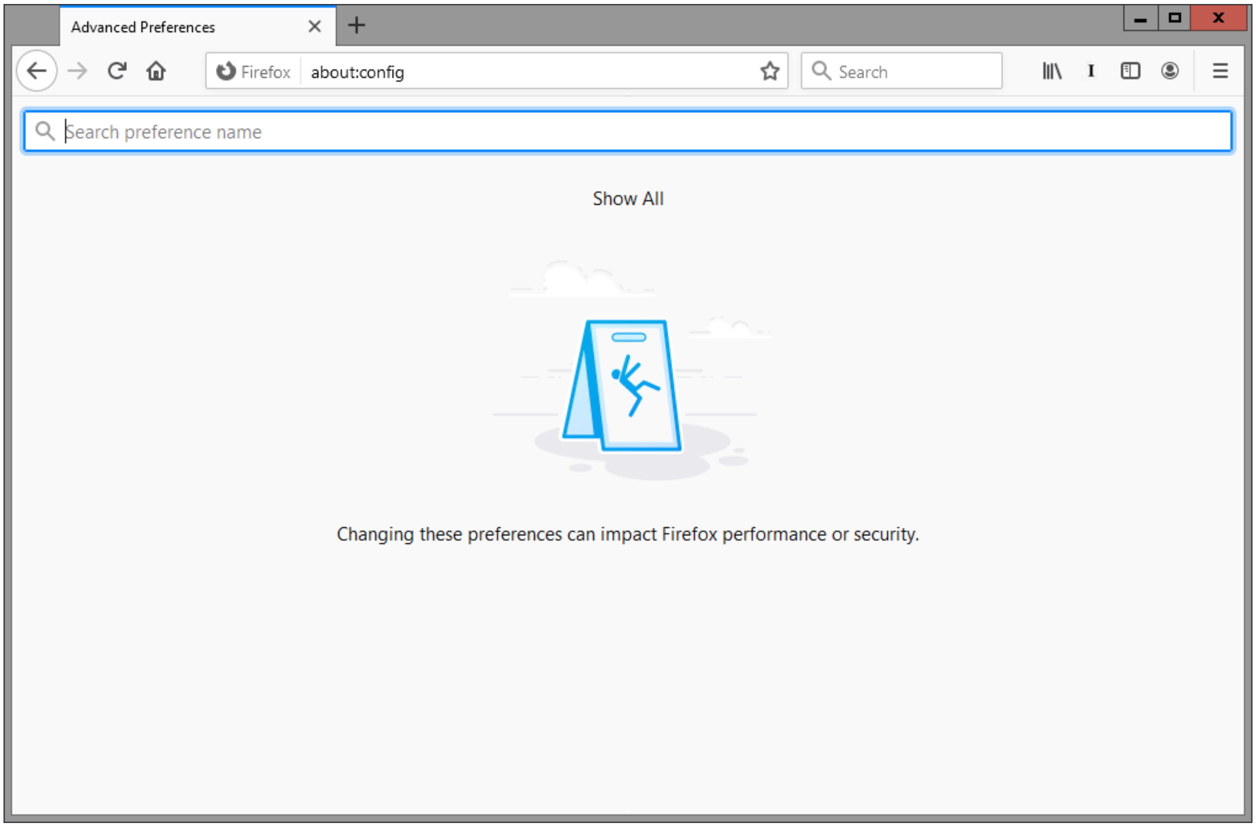
Proceed with Caution

Changing advanced configuration preferences can impact Firefox performance or security.

☑ Warn me when I attempt to access these preferences

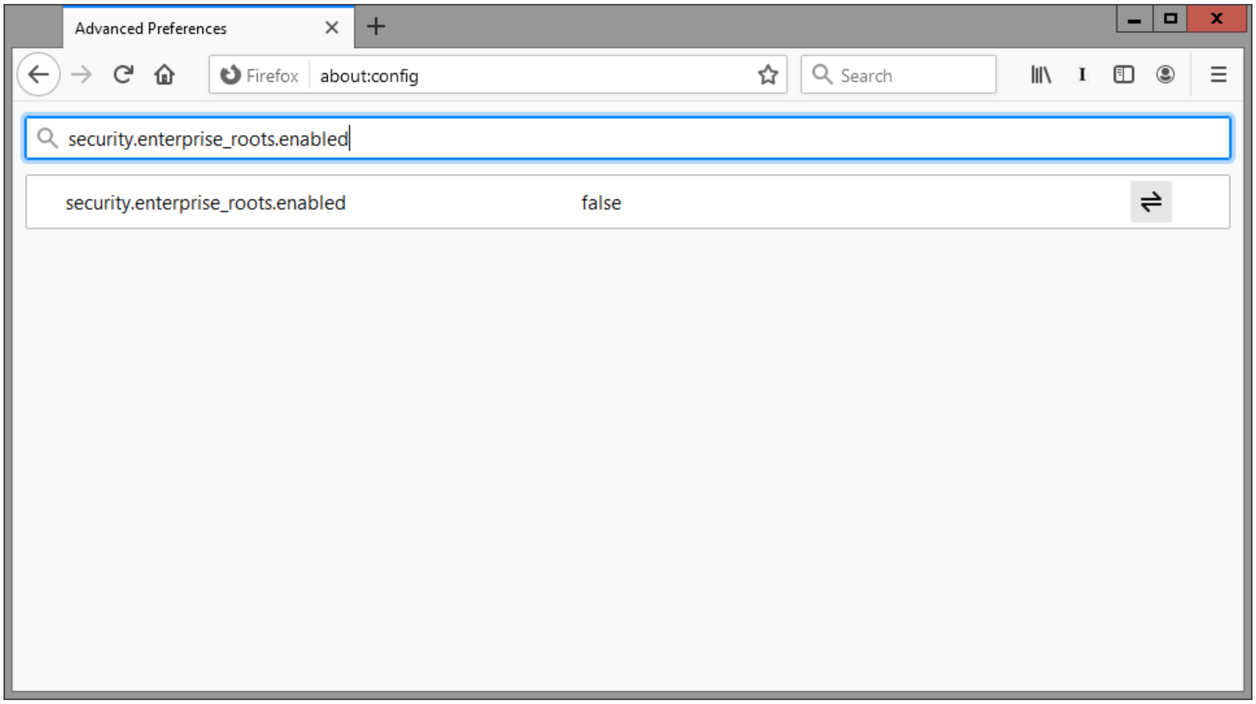
2.3. Accept the risk and continue.

2.4. This will take you to the following page:



Changing these preferences can impact Firefox performance or security.

2.5. In the search field type the following: security.enterprise_roots.enabled

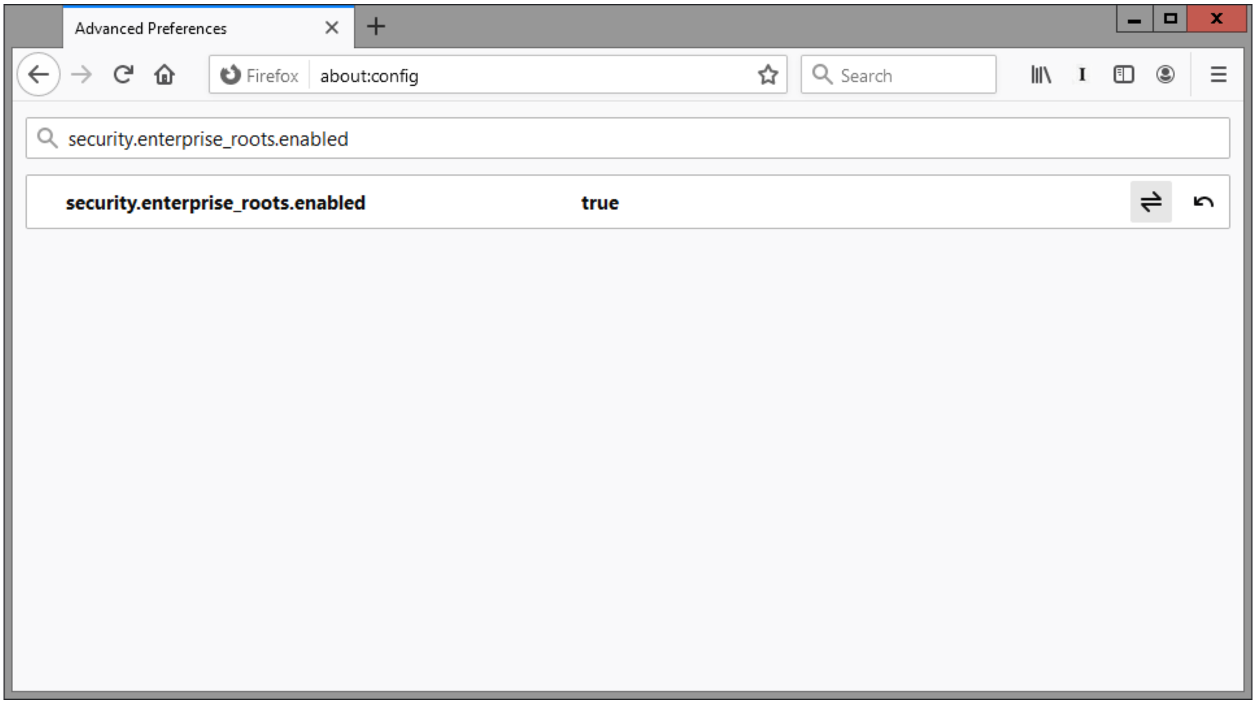


security.enterprise_roots.enabled

security.enterprise_roots.enabled	false

Your screen may be slightly different depending on your operating system. The default value for this setting is false.

2.6. Double-click on this entry to toggle the value from false to true. **Note:** Firefox may display a value of true for this setting even though it has never been changed. In this case, change the value to false and then back to true again.



security.enterprise_roots.enabled

security.enterprise_roots.enabled true

2.7. **Close and restart the browser** for the change to take place.

2.8. You should now be able to open DataSync without the security warning appearing.

Usage of SSL certificate created by an internal corporate Trusted Root Authority

When using SSL certificates created by an internal corporate Trusted Root Authority, the following prerequisite steps are required:

- The IT organization has already distributed and installed the internal Trusted Root Authority certificate to all PCs in the organization.
- The IT organization has already configured the supported web browsers to recognize the internal Trusted Root Authority certificate as valid.

Step 1: Obtain an SSL certificate for the DataSync server from the IT organization.

We recommend that the SSL certificate should have subject alternative names such as:

Localhost

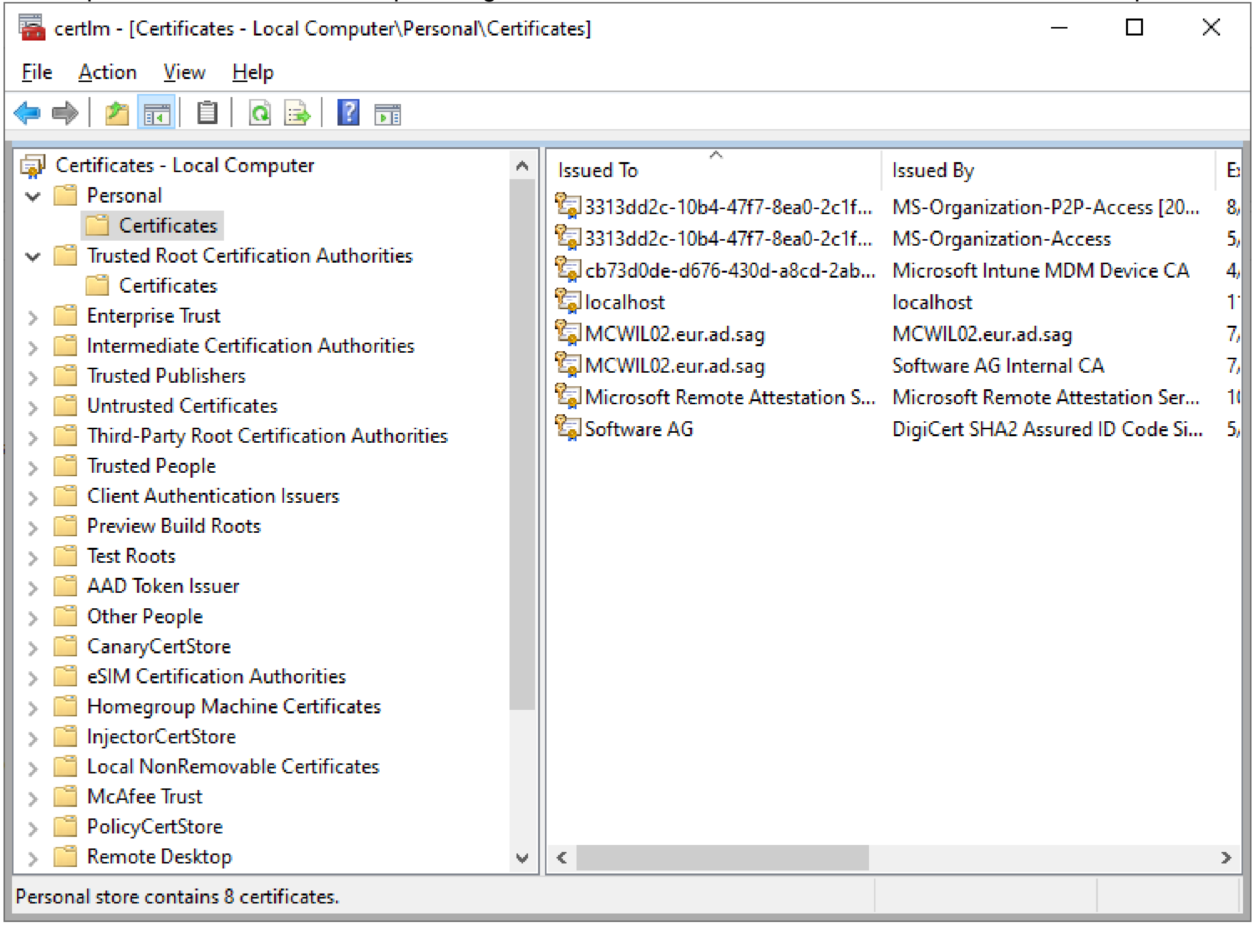
<servername>

<servername>.<fully qualified domain name>

Step 2: Install the SSL Certificate on the DataSync Server.

2.1. Type in '**certlm.msc**' and right mouse click on the item and select "run as administrator".

2.2. Expand ‘Personal’ in the left pane, right click on ‘Certificates’, select ‘All Tasks’ and then ‘Import’.


certlm - [Certificates - Local Computer\Personal\Certificates]
File
Action
View
Help

Certificates - Local Computer
- Personal
  - Certificates
- Trusted Root Certification Authorities
  - Certificates
- Enterprise Trust
- Intermediate Certification Authorities
- Trusted Publishers
- Untrusted Certificates
- Third-Party Root Certification Authorities
- Trusted People
- Client Authentication Issuers
- Preview Build Roots
- Test Roots
- AAD Token Issuer
- Other People
- CanaryCertStore
- eSIM Certification Authorities
- Homegroup Machine Certificates
- InjectorCertStore
- Local NonRemovable Certificates
- McAfee Trust
- PolicyCertStore
- Remote Desktop

Issued To	Issued By	E:
3313dd2c-10b4-47f7-8ea0-2c1f...	MS-Organization-P2P-Access [20...	8,
3313dd2c-10b4-47f7-8ea0-2c1f...	MS-Organization-Access	5,
cb73d0de-d676-430d-a8cd-2ab...	Microsoft Intune MDM Device CA	4,
localhost	localhost	1'
MCWIL02.eur.ad.sag	MCWIL02.eur.ad.sag	7,
MCWIL02.eur.ad.sag	Software AG Internal CA	7,
Microsoft Remote Attestation S...	Microsoft Remote Attestation Ser...	1(
Software AG	DigiCert SHA2 Assured ID Code Si...	5,

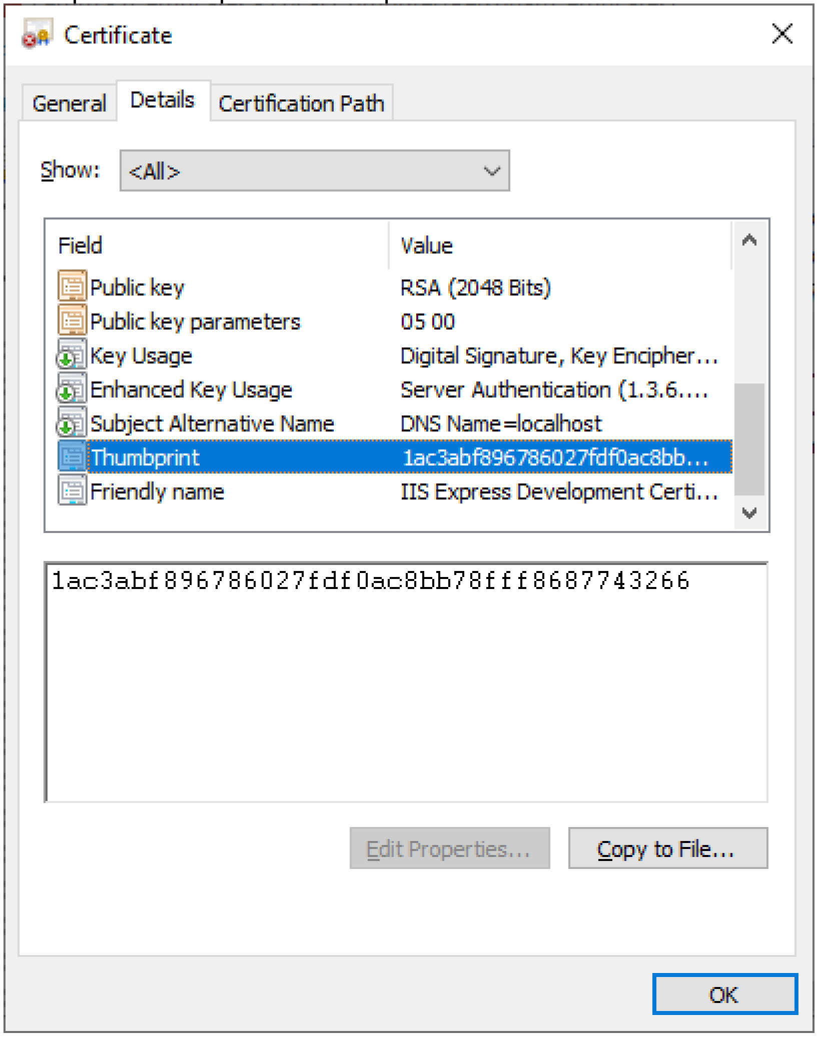
Personal store contains 8 certificates.

2.2. Follow the steps in the ‘Certificate Import Wizard’ to complete the certificate installation.

Step 3: Bind the personal certificate to port 9500

3.1. Double click on the installed personal certificate and select Details. Scroll to the bottom of the details list where you will see a field named Thumbprint. Select Thumbprint and you will see the thumbprint value expanded in the text window. Depending on the version of Windows, there may be spaces between each byte of the thumbprint. Copy and paste the thumbprint into a text editor. If there are spaces in the thumbprint, remove them in the editor.

Example of thumbprint without spaces:

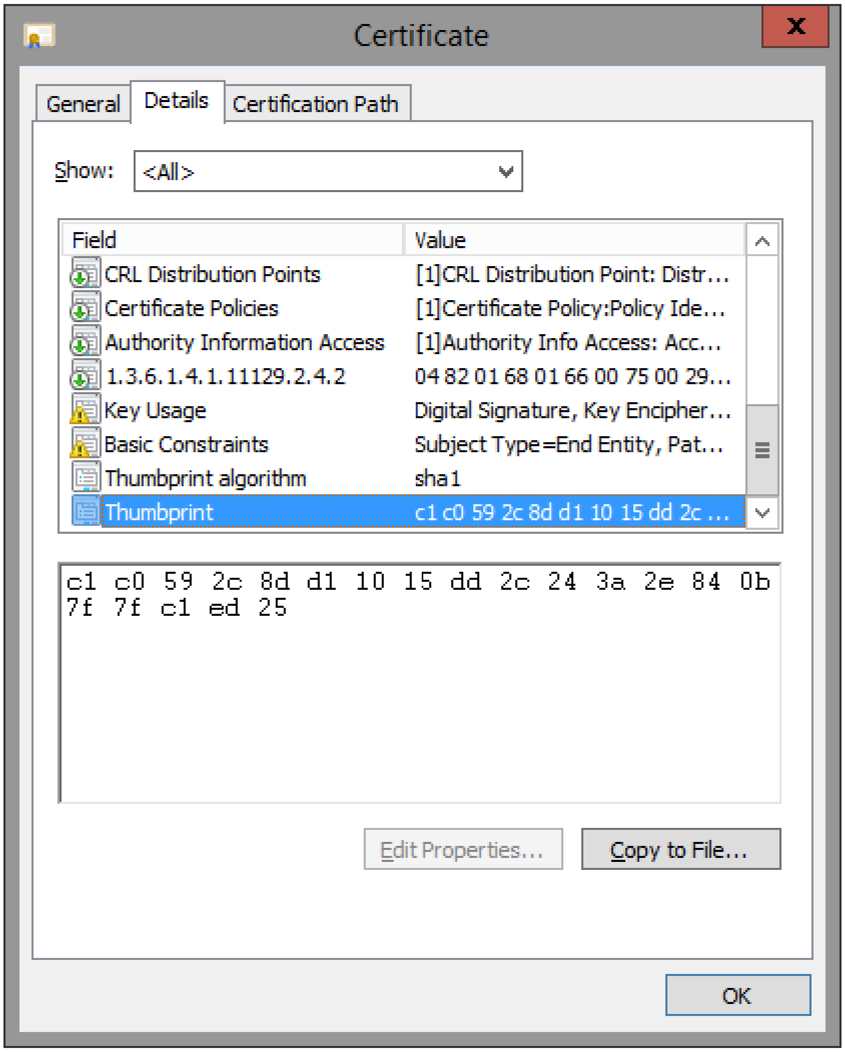


Show: <All>

Field	Value
Public key	RSA (2048 Bits)
Public key parameters	05 00
Key Usage	Digital Signature, Key Encipher...
Enhanced Key Usage	Server Authentication (1.3.6....
Subject Alternative Name	DNS Name=localhost
Thumbprint	1ac3abf896786027fdf0ac8bb...
Friendly name	IIS Express Development Certi...

```
1ac3abf896786027fdf0ac8bb78fff8687743266
```

Example of thumbprint with spaces:



Show: <All>

Field	Value
CRL Distribution Points	[1]CRL Distribution Point: Distr...
Certificate Policies	[1]Certificate Policy:Policy Ide...
Authority Information Access	[1]Authority Info Access: Acc...
1.3.6.1.4.1.11129.2.4.2	04 82 01 68 01 66 00 75 00 29...
Key Usage	Digital Signature, Key Encipher...
Basic Constraints	Subject Type=End Entity, Pat...
Thumbprint algorithm	sha1
Thumbprint	c1 c0 59 2c 8d d1 10 15 dd 2c ...

```
c1 c0 59 2c 8d d1 10 15 dd 2c 24 3a 2e 84 0b
7f 7f c1 ed 25
```

3.2. Prior to attempting to bind the certificate to port 9500, check to make sure there isn't a certificate already bound. Note that if the CONNX installation program was run prior to following these instructions, a self-signed certificate will have been created and automatically bound to port 9500. To check for previous bindings start a Windows DOS command prompt as Administrator and run the following command:

netsh http show sslcert ipport=0.0.0.0:9500

If this command shows a binding that has a Certificate Hash (thumbprint) other than the one you intend to use, the binding needs to be deleted. This will be the case if the CONNX installer already created and bound a self-signed certificate. To delete the binding, run the following command from the same Windows DOS command prompt:

netsh http delete sslcert ipport=0.0.0.0:9500

3.3. Start a Windows DOS command prompt as Administrator (or if step 3.2 was needed, use the command prompt already open) and run the following command to bind the certificate to the port 9500.

netsh http add sslcert ipport=0.0.0.0:9500 certhash=THUMBPRINT_FROM_PREVIOUS_STEP appid={87c9b46f-ae61-4a10-be41-52c7b89956fa}

Note: if you need to replace the certificate, you will first need to unbind this one. To do this, issue the netsh http delete command from step 3.2 above.

You should now be able to connect to the CONNX DataSync server with your browser without receiving any warnings. As stated earlier, if the Trusted Root Certificate is self-signed, the Firefox browser will still issue a warning.

Usage of SSL certificate created by an external Trusted Root Authority

Using SSL Certificates created by external Trusted Root Authorities (for example, IdenTrust, DigiCert, Sectigo, GoDaddy, etc), requires the minimal amount of steps as all web browsers are already configured to recognize these certificate authorities.

Step 1: Obtain/Purchase an SSL certificate for the DataSync server from an external Trusted Certificate Authority.

We recommend that the SSL certificate should have subject alternative names such as:

Localhost

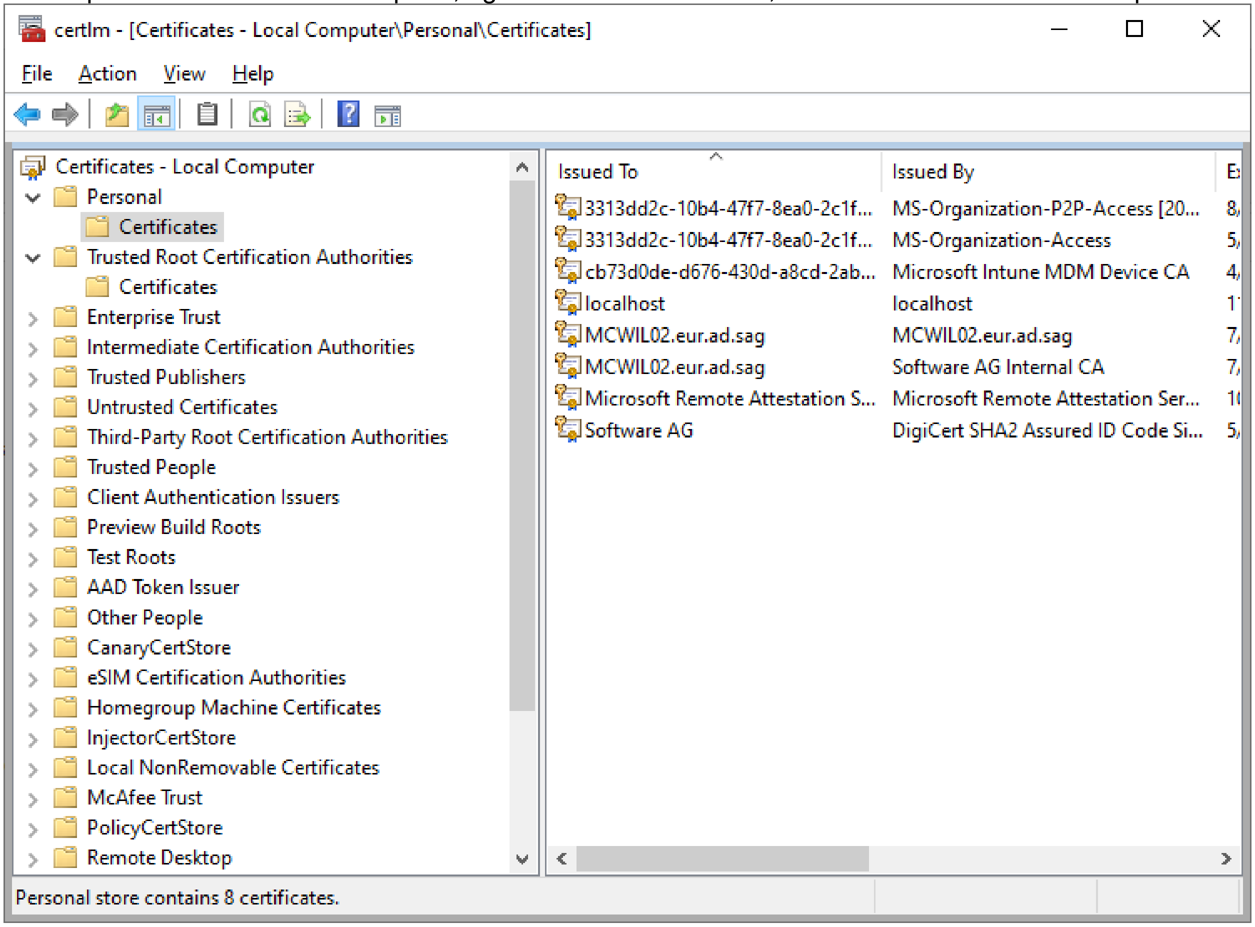
<servername>

<servername>.<fully qualified domain name>

Step 2: Install the SSL Certificate on the DataSync Server.

2.1. Type in '**certlm.msc**' and right mouse click on the item and select "run as administrator".

2.2. Expand 'Personal' in the left pane, right click on 'Certificates', select 'All Tasks' and then 'Import'.


certlm - [Certificates - Local Computer\Personal\Certificates]
File
Action
View
Help

Certificates - Local Computer
- Personal
  - Certificates
- Trusted Root Certification Authorities
  - Certificates
- Enterprise Trust
- Intermediate Certification Authorities
- Trusted Publishers
- Untrusted Certificates
- Third-Party Root Certification Authorities
- Trusted People
- Client Authentication Issuers
- Preview Build Roots
- Test Roots
- AAD Token Issuer
- Other People
- CanaryCertStore
- eSIM Certification Authorities
- Homegroup Machine Certificates
- InjectorCertStore
- Local NonRemovable Certificates
- McAfee Trust
- PolicyCertStore
- Remote Desktop

Issued To	Issued By	E
3313dd2c-10b4-47f7-8ea0-2c1f...	MS-Organization-P2P-Access [20...	8,
3313dd2c-10b4-47f7-8ea0-2c1f...	MS-Organization-Access	5,
cb73d0de-d676-430d-a8cd-2ab...	Microsoft Intune MDM Device CA	4,
localhost	localhost	1
MCWIL02.eur.ad.sag	MCWIL02.eur.ad.sag	7,
MCWIL02.eur.ad.sag	Software AG Internal CA	7,
Microsoft Remote Attestation S...	Microsoft Remote Attestation Ser...	1(
Software AG	DigiCert SHA2 Assured ID Code Si...	5,

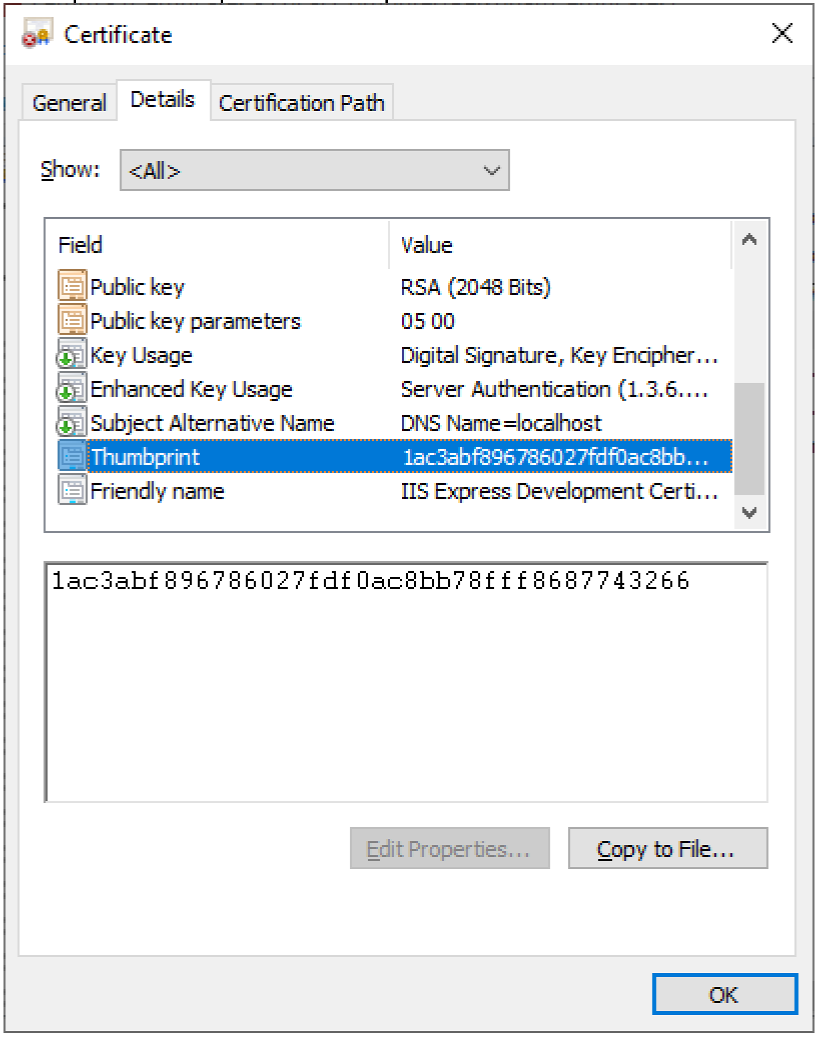
Personal store contains 8 certificates.

2.2. Follow the steps in the 'Certificate Import Wizard' to complete the certificate installation.

Step 3: Bind the personal certificate to port 9500

3.1. Double click on the installed personal certificate and select Details. Scroll to the bottom of the details list where you will see a field named Thumbprint. Select Thumbprint and you will see the thumbprint value expanded in the text window. Depending on the version of Windows, there may be spaces between each byte of the thumbprint. Copy and paste the thumbprint into a text editor. If there are spaces in the thumbprint, remove them in the editor.

Example of thumbprint without spaces:

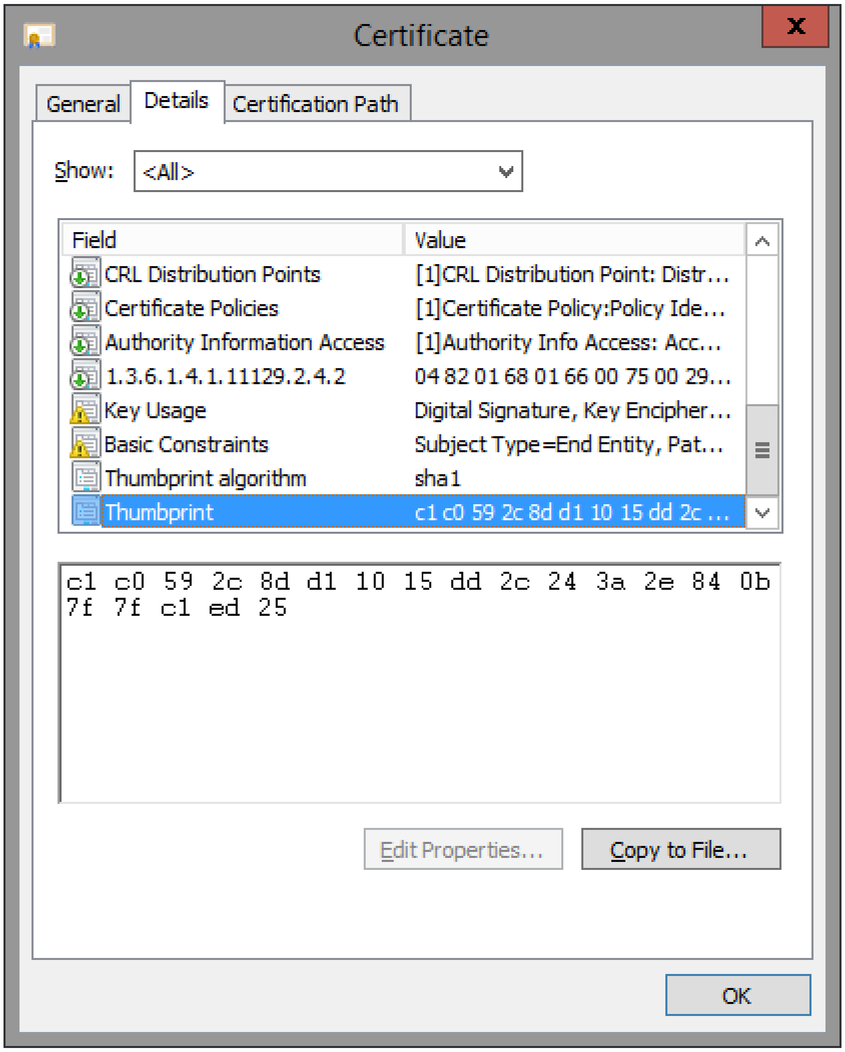


Show: <All>

Field	Value
Public key	RSA (2048 Bits)
Public key parameters	05 00
Key Usage	Digital Signature, Key Encipher...
Enhanced Key Usage	Server Authentication (1.3.6....
Subject Alternative Name	DNS Name=localhost
Thumbprint	1ac3abf896786027fdf0ac8bb...
Friendly name	IIS Express Development Certi...

```
1ac3abf896786027fdf0ac8bb78fff8687743266
```

Example of thumbprint with spaces:



Show: <All>

Field	Value
CRL Distribution Points	[1]CRL Distribution Point: Distr...
Certificate Policies	[1]Certificate Policy:Policy Ide...
Authority Information Access	[1]Authority Info Access: Acc...
1.3.6.1.4.1.11129.2.4.2	04 82 01 68 01 66 00 75 00 29...
Key Usage	Digital Signature, Key Encipher...
Basic Constraints	Subject Type=End Entity, Pat...
Thumbprint algorithm	sha1
Thumbprint	c1 c0 59 2c 8d d1 10 15 dd 2c ...

```
c1 c0 59 2c 8d d1 10 15 dd 2c 24 3a 2e 84 0b
7f 7f c1 ed 25
```

3.2. Prior to attempting to bind the certificate to port 9500, check to make sure there isn't a certificate already bound. Note that if the CONNX installation program was run prior to following these instructions, a self-signed certificate will have been created and automatically bound to port 9500. To check for previous bindings start a Windows DOS command prompt as Administrator and run the following command:

netsh http show sslcert ipport=0.0.0.0:9500

If this command shows a binding that has a Certificate Hash (thumbprint) other than the one you intend to use, the binding needs to be deleted. This will be the case if the CONNX installer already created and bound a self-signed certificate. To delete the binding, run the following command from the same Windows DOS command prompt:

netsh http delete sslcert ipport=0.0.0.0:9500

3.3. Start a Windows DOS command prompt as Administrator (or if step 3.2 was needed, use the command prompt already open) and run the following command to bind the certificate to the port 9500.

netsh http add sslcert ipport=0.0.0.0:9500 certhash=THUMBPRINT_FROM_PREVIOUS_STEP appid={87c9b46f-ae61-4a10-be41-52c7b89956fa}

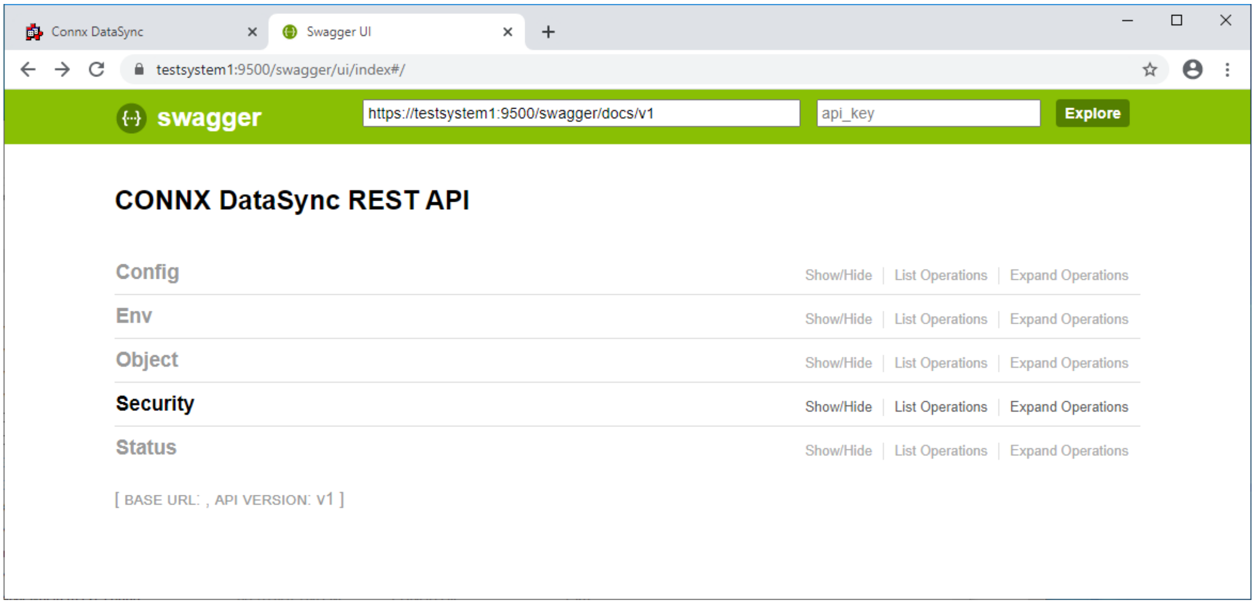
Note: if you need to replace the certificate, you will first need to unbind this one. To do this, issue the netsh http delete command from step 3.2 above.

You should now be able to connect to the CONNX DataSync server with your browser without receiving any warnings. As stated earlier, if the Trusted Root Certificate is self-signed, the Firefox browser will still issue a warning.

REST API Documentation

Accessing Swagger Page

The DataSync REST API is documented with Swagger. The swagger page can be found at **https://*<DataSync Server>*:*<Port>*/swagger** where *<DataSync Server>* is the name of the server where the DataSync REST Server is installed and *<Port>* is the port number it is listening on. Port 9500 is the default.



testsystem1:9500/swagger/ui/index#/

https://testsystem1:9500/swagger/docs/v1

api_key

CONNX DataSync REST API

Config

Env

Object

Security

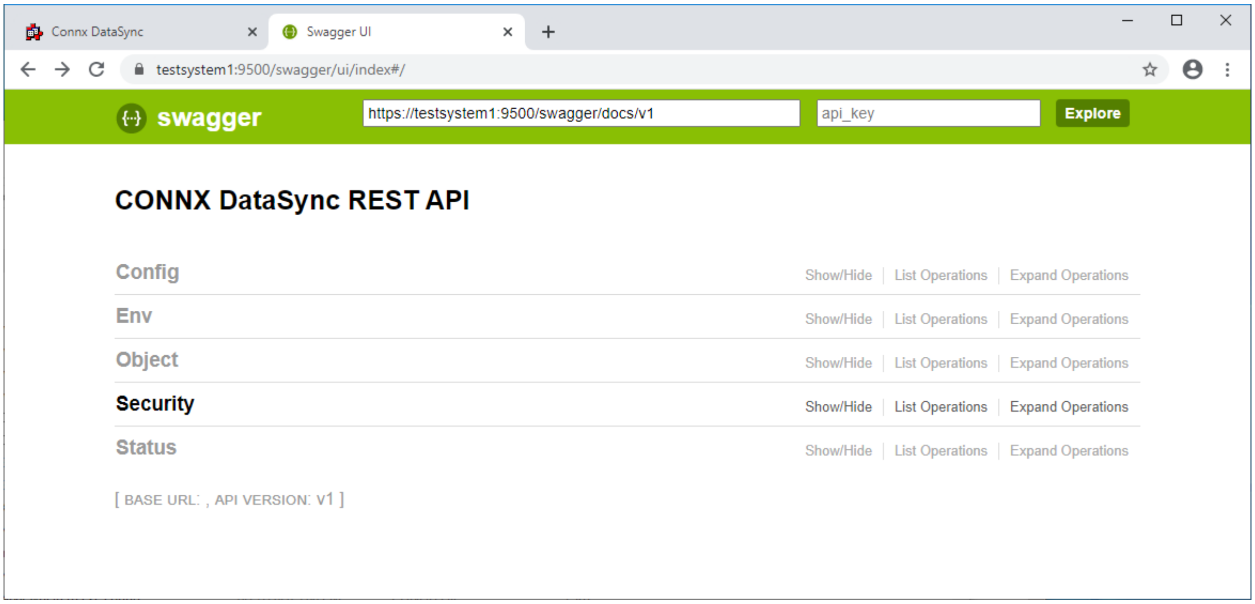
Status

[BASE URL: , API VERSION: v1]

Swagger supports a Try It Out feature that can be used to test the APIs. To use the Try It Out feature for any of the APIs, you first need to log in. See Using Swagger for an example of logging on to the DataSync Server with the Swagger Try It Out feature.

Using Swagger

Swagger supports a Try It Out feature that can be used to test the APIs. To use the Try It Out feature for any of the APIs, you first need to log in. Below is an example of how to log in using the Try It Out feature in swagger:



testsystem1:9500/swagger/ui/index#/

https://testsystem1:9500/swagger/docs/v1

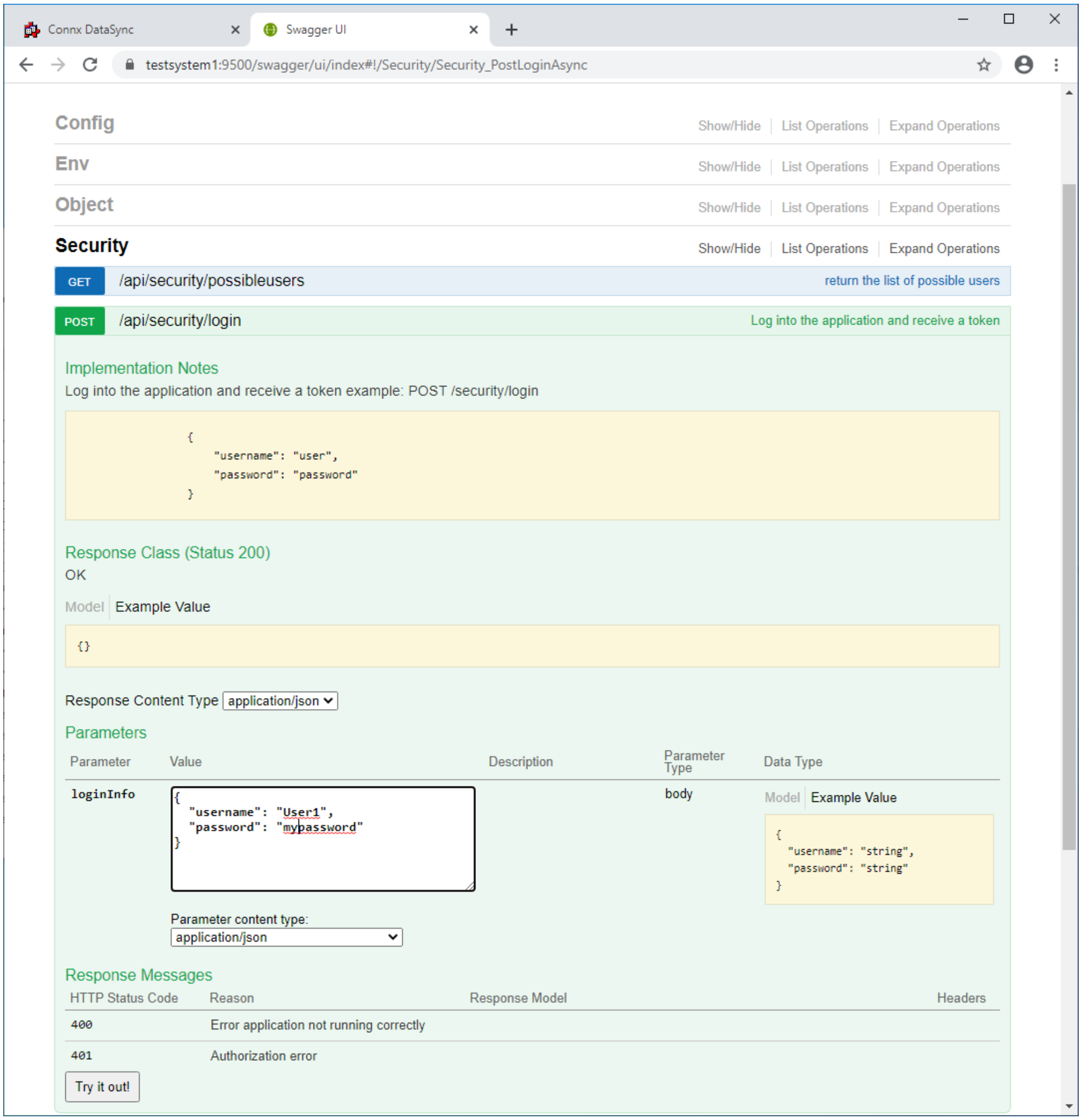
api_key

CONNX DataSync REST API

[BASE URL: , API VERSION: v1]

Connx DataSync
Swagger UI
swagger
Explore
Config
Show/Hide | List Operations | Expand Operations
Env
Show/Hide | List Operations | Expand Operations
Object
Show/Hide | List Operations | Expand Operations
Security
Show/Hide | List Operations | Expand Operations
Status
Show/Hide | List Operations | Expand Operations

1. Navigate to the swagger page and click Security to expand the list of APIs and select /api/security/login POST.


Connx DataSync
Swagger UI
testsystem1:9500/swagger/ui/index#!/Security/Security_PostLoginAsync

Config — Show/Hide | List Operations | Expand Operations

Env — Show/Hide | List Operations | Expand Operations

Object — Show/Hide | List Operations | Expand Operations

Security — Show/Hide | List Operations | Expand Operations

GET /api/security/possibleusers — return the list of possible users

POST /api/security/login — Log into the application and receive a token

Implementation Notes

Log into the application and receive a token example: POST /security/login

```
{
  "username": "user",
  "password": "password"
}
```

Response Class (Status 200)

OK

Model | Example Value

```
{}
```

Response Content Type: application/json

Parameters

Parameter	Value	Description	Parameter Type	Data Type
loginInfo	`{ "username": "User1", "password": "mypassword" }` Parameter content type: application/json		body	Model | Example Value `{ "username": "string", "password": "string" }`

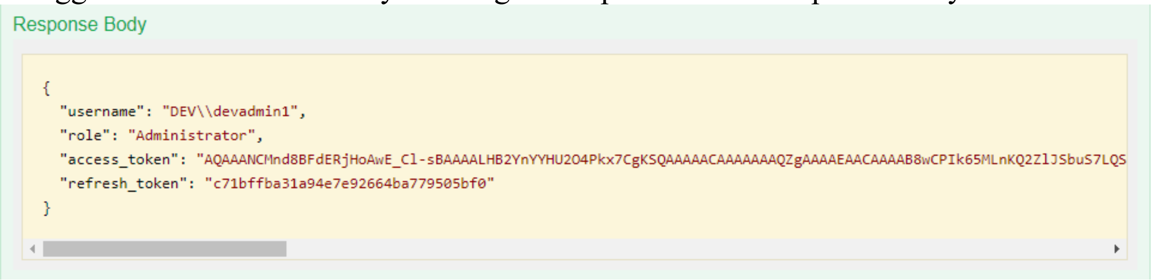
Response Messages

HTTP Status Code	Reason	Response Model	Headers
400	Error application not running correctly		
401	Authorization error		

Try it out!

2. In the Parameters section, click on the yellow Example Value box. This will populate the loginInfo with correctly formatted input parameters. For username, put in a valid Windows Active Directory user name. Note that swagger requires an escape for the backslash character so if you enter the user name in the form of domain\user, you will need to enter it as domain\\user when using swagger.
3. Click Try it Out!

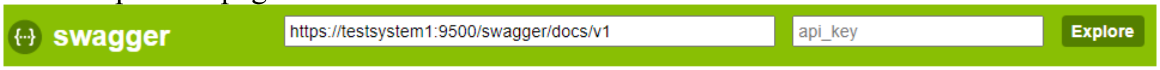
4. Swagger will do a POST and you will get a response in the Response Body



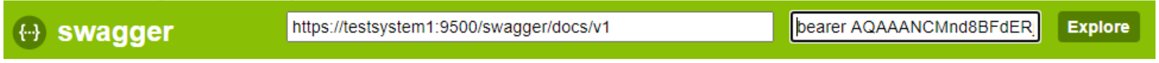
```
{
  "username": "DEV\\devadmin1",
  "role": "Administrator",
  "access_token": "AQAAANCMnd8BFdERjHoAwE_Cl-sBAAAALHB2YnYYHU2O4Pkx7CgKSQAAAAACAAAAAAAQZgAAAAEAACAAAAB8wCPIk65MLnKQ2Zl JSbuS7LQS
  "refresh_token": "c71bffba31a94e7e92664ba779505bf0"
}
```

Copy the access token into your copy buffer. You will need to select the entire string as it scrolls off the screen to the right. Do not copy the double quotes.

5. At the top of the page is a banner



6. In the text box that says api_key, enter the word bearer followed by the access_token from the response body



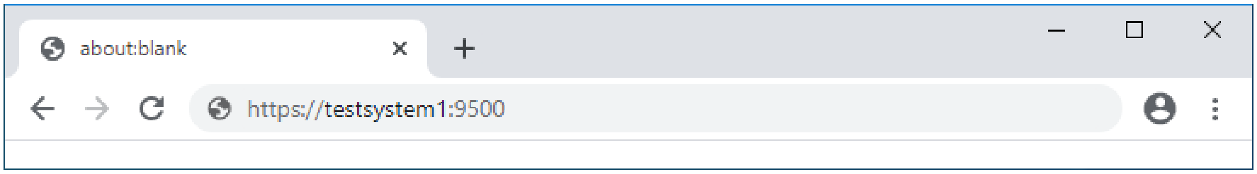
https://testsystem1:9500/swagger/docs/v1

bearer AQAAANCMnd8BFdER

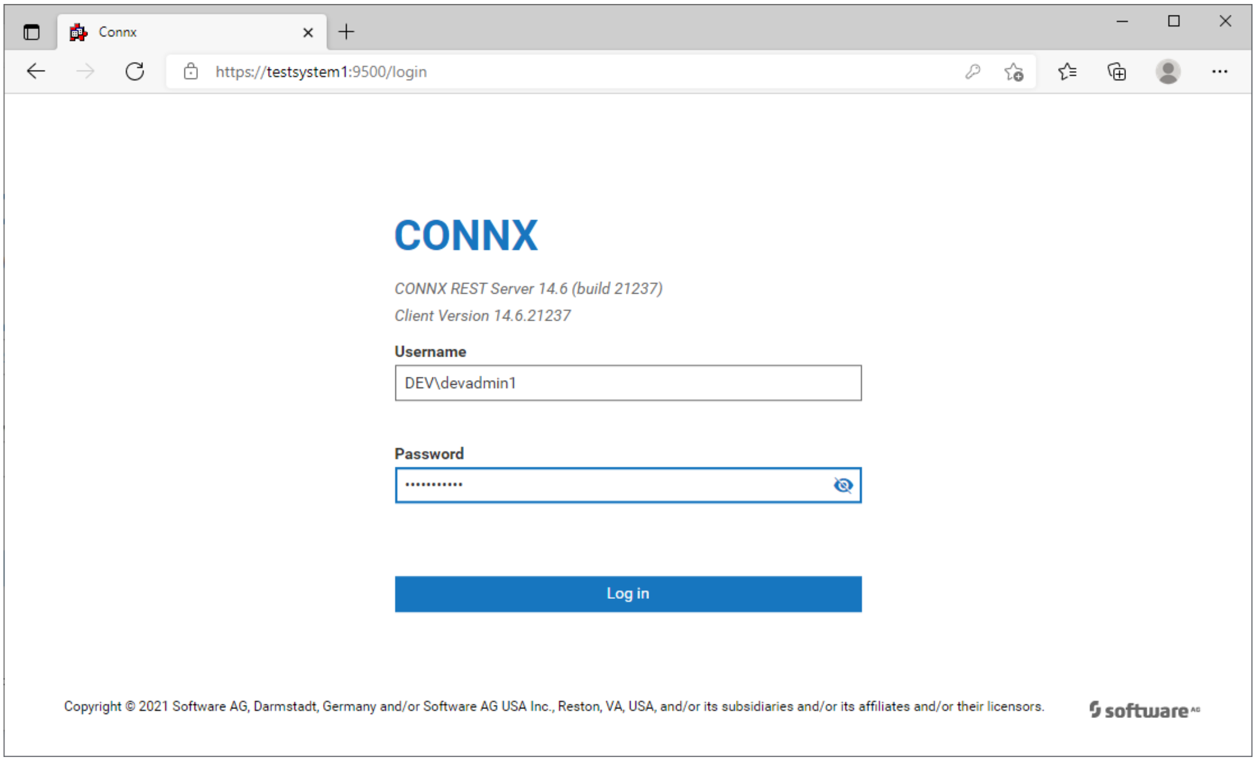
7. Press the enter key. This access token will be used on all subsequent calls to the REST server.

Logging in to the DataSync Server

To connect to the DataSync Server, open a browser session in either Chrome or Firefox. Enter the URL HTTPS://*<DataSync server name>*:*<port>* where *<DataSync server name>* is the name of the server where the DataSync REST/Web Server is installed and *<port>* is the port number it is listening on. (The default port is 9500) **Note:** For best results, we recommend the browser be run in its maximized setting.



Enter a valid Windows User ID and password. The User ID can specify a domain in the form *<DomainName>**<UserID>* If no domain name is specified, the domain the DataSync server is on will be used. In the below example, the domain is DEV and the User ID is devadmin1.



CONNX REST Server 14.6 (build 21237)

Client Version 14.6.21237

Username: DEV\devadmin1

Password: ••••••••••••

Copyright © 2021 Software AG, Darmstadt, Germany and/or Software AG USA Inc., Reston, VA, USA, and/or its subsidiaries and/or its affiliates and/or their licensors.

The first user to log on to a newly installed DataSync server will automatically become a member of the Administrators group. This user can then be used to manage additional users. Please see **Managing Users** for information on user management tasks.

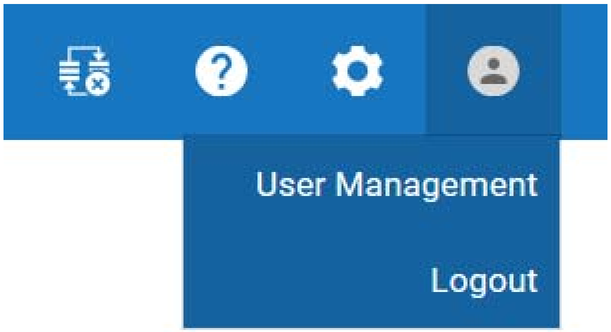
When the first user logs into DataSync, a message may appear in the upper right corner informing the user that there is no CDD connection. If DataSync is being upgraded from a previously installed version, the CDD that was in use with the previous version will automatically be connected and this message will not appear. Please see **Managing Connections** for information on uploading and activating CDDs for use with DataSync. The first user login will be automatically taken to the **Managing Users** page. Subsequent logins will go to the **Overview Summary Page**.

System Management

Managing Users

The User Management page can be accessed from the User Menu at the top of the screen. Some functions on this screen use icons. The description of these icons will have a hyperlink to the Glossary of Icons which contains a picture of the icon and an explanation of its purpose.

Note: A user role of Administrator is required to access the User Management functions. Users without Administrator privileges will not have User Management on their menu.

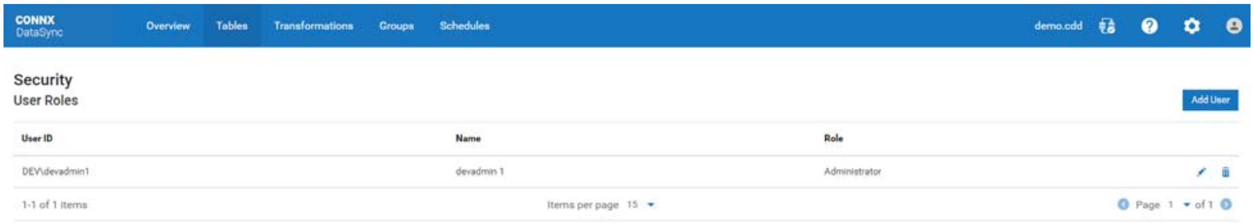


DataSync supports four User Roles. The roles are hierarchical meaning each higher role has all the permissions of the role below it plus additional permissions. The roles and their permissions are as follows:

Role Name	Role Activities
Administrator	1. All the activities of Operator 2. User Management
Designer	1. All the activities of Operator 2. Add/Update/Delete Transforms, Groups and Schedules 3. Change System Settings
Operator	1. All the activities of Monitor 2. Ability to run sync jobs
Monitor	1. View running sync jobs 2. View job history 3. View tables, transforms, groups and schedules

For a comprehensive list of each API and the required security to execute it, refer to the **REST API Documentation**

The first User ID used to log into DataSync is automatically registered as an Administrator. After this first log in, you will be taken directly to the User Management page within DataSync

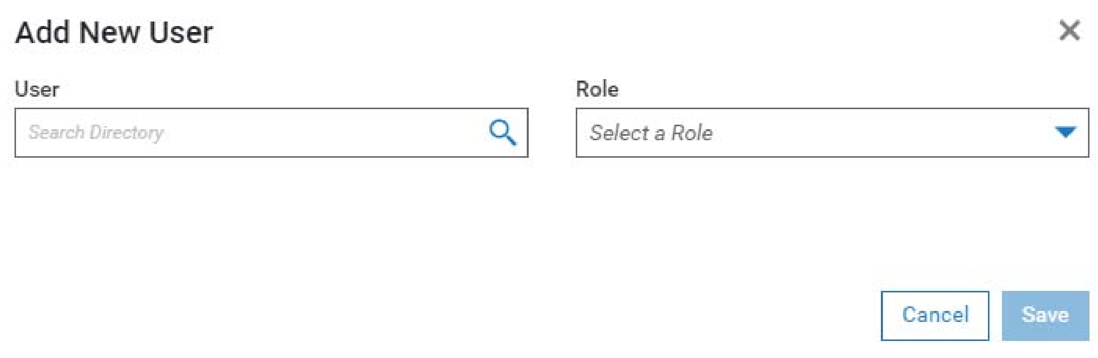


Security
User Roles

User ID	Name	Role
DEV\devadmin1	devadmin 1	Administrator

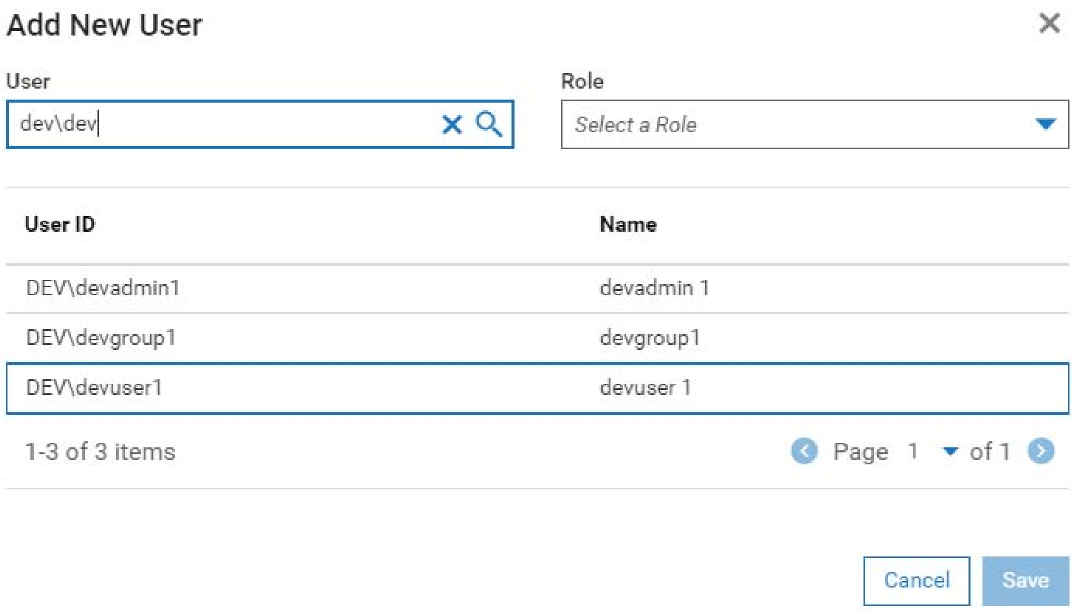
1-1 of 1 items   Items per page 15   Page 1 of 1

- **User Roles table**
  The User Roles table shows the list of users authorized to use DataSync and what their role is. Below is a description of each column in the table:
  - **User ID**
    The domain and user id as it appears in Windows Active Directory
  - **Name**
    This is the value in the Full Name Active Directory field
  - **Role**
    The Role of this user.
  - **Edit**
    Clicking the Edit button will take you to the Edit User dialog. This dialog will allow the user's role to be modified. There must always be at least one Administrator in the system. If an Administrator is the only Administrator in the system, attempting to change the Role to a non-Administrator role will result in an error.
  - **Delete**
    Clicking the Delete button will remove the user from the list of authorized DataSync users. There must always be at least one Administrator in the system. If an Administrator is the only Administrator in the system, attempting to delete the user will result in an error.
- **Add User**
  The Add User button will open Add New User dialog



Start typing the User ID or User Name in the **User** field and a search of the Active Directory will start. As more characters are entered, the list of possible users will narrow. When a domain forest is present and you wish to search for a user in a related domain other than the domain the

DataSync server is on, the domain must be specified. See the example below:



Add New User

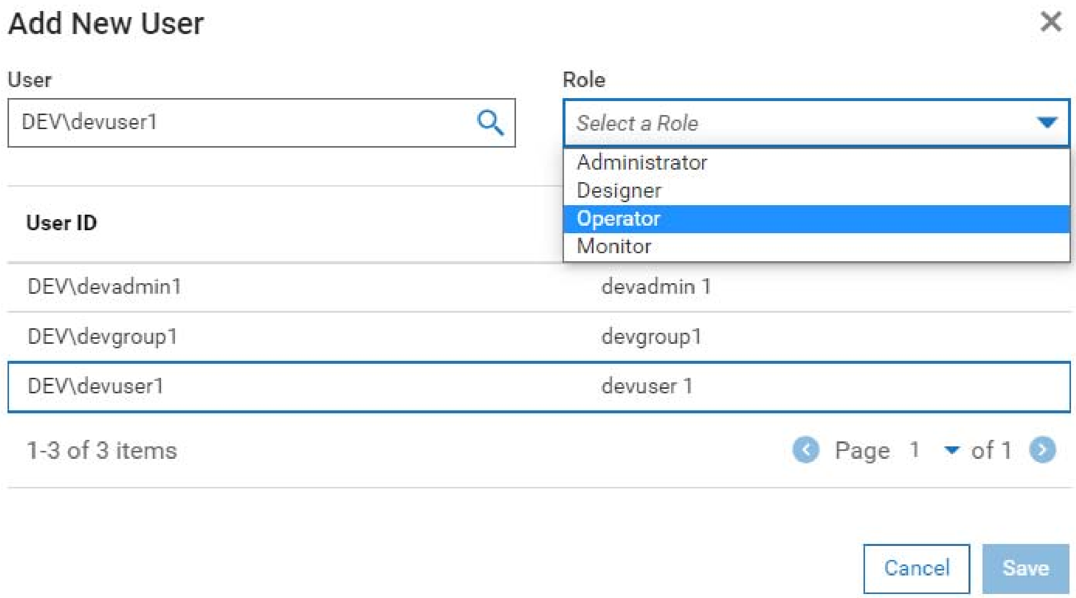
User
dev\dev

Role
Select a Role

User ID	Name
DEV\devadmin1	devadmin 1
DEV\devgroup1	devgroup1
DEV\devuser1	devuser 1

1-3 of 3 items   Page 1 of 1

In this example, the domain "dev" is a child of the domain the DataSync server is on. In this case, entering "dev\" causes DataSync to start the search in the "dev" domain. In this example, the search found three possible users. The user DEV\devuser1 will be used.



Add New User

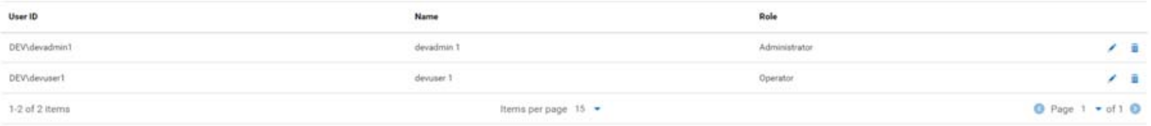
User
DEV\devuser1

Role
Select a Role

User ID	
DEV\devadmin1	devadmin 1
DEV\devgroup1	devgroup1
DEV\devuser1	devuser 1

1-3 of 3 items   Page 1 of 1

Select a role from the **Role** dropdown and click Save to return to the User Roles list



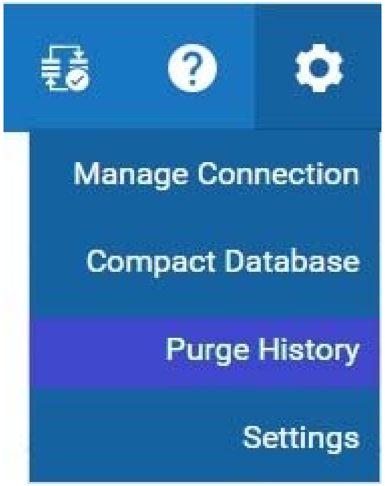
User ID	Name	Role
DEV\devadmin1	devadmin 1	Administrator
DEV\devuser1	devuser 1	Operator

1-2 of 2 items   Items per page 15   Page 1 of 1

Purge History feature

If the CONNX Data Synchronization tool runs for a lengthy period of time, historical information that accumulates may decrease its high performance levels. Use the following procedure to purge the historical information.

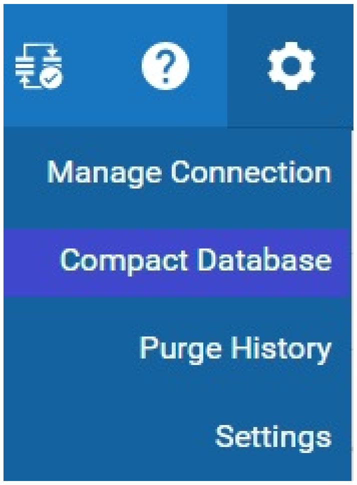
- On the **Gear** menu, select **Purge History.** The historical information is removed from the log.



Compact Database feature

To compress the CONNXStore database and restore lost performance levels, use the Compact Database feature.

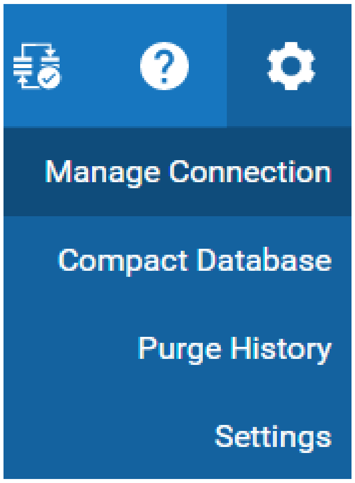
- On the **Gear** menu, select **Compact Database.** The contents of the CONNXStore database are compressed.



Managing Connections

Managing Connections

Connections are managed with the **Connect to a CDD** dialog. This dialog is access either by clicking the Gear icon and selecting Manage Connection or by clicking the Connection State Icon



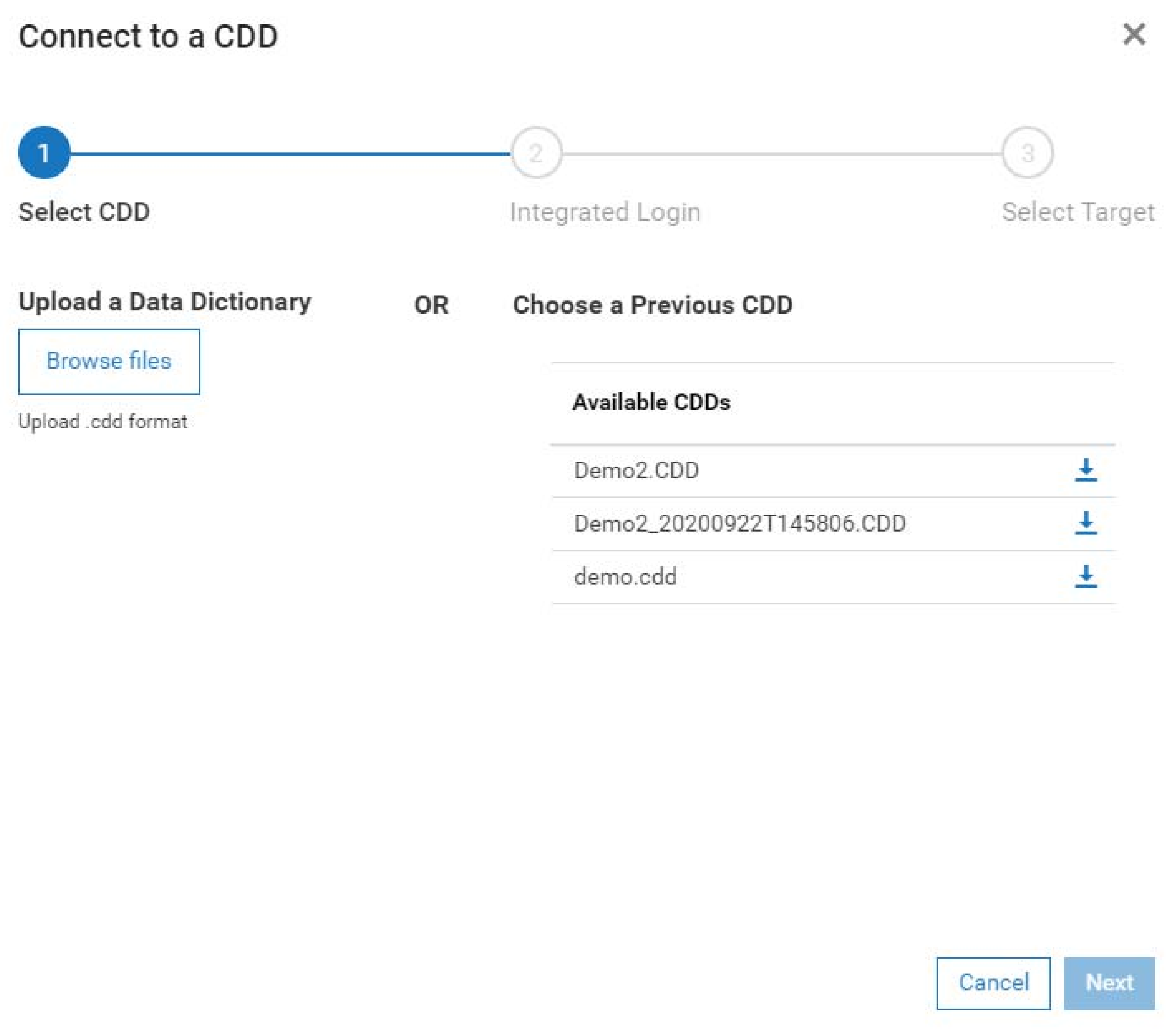
Before a CDD can be used by the DataSync REST Server, it must be uploaded to the server and then activated. CDDs are stored on the DataSync server in the connx32\datasync\cdd directory. To aid in the process of managing CDDs, a stepper dialog is used. See Step 1: Select CDD to begin the process.

Notes:

- When upgrading the DataSync Server from a version of DataSync prior to 14.5, the new DataSync server will use the active CDD from the old version (DataSync Classic). This CDD will remain in the location it was in with the old version.
- The use of both DataSync Server and DataSync Classic on the same server is not recommended. If they are used together, be aware that they share several internal components. Activating a new CDD in the DataSync Server will cause DataSync Classic to use that CDD as well. The same is true of changing the CDD in DataSync Classic. Doing this will cause the DataSync Server to use that CDD the next time it is started which could lead to confusion and possibly error situations.

Select CDD

The Select CDD step provides functionality to upload a new CDD to the DataSync Server, Download a CDD from the DataSync Server to the local machine where the client is running in a browser or chose a CDD that already exists on the DataSync Server for activation.



Connect to a CDD

Upload a Data Dictionary

Upload .cdd format

OR

Choose a Previous CDD

Available CDDs	
Demo2.CDD	⤓
Demo2_20200922T145806.CDD	⤓
demo.cdd	⤓

- **Upload a Data Dictionary**
  Before a CDD can be activated, it must be on the DataSync Server. A CDD can be created and edited on a local machine and then uploaded to the server when it is ready for use. To upload a CDD, click the Browse files button and then select the CDD from the File Open dialog. Once the CDD has been successfully uploaded, it will appear in the Available CDDs list.
- **Choose a Previous CDD**
  The Available CDDs list is a list of CDDs that reside in the connx32\datasync\cdd directory on DataSync server. Any time a CDD is uploaded, it will be added to this list.

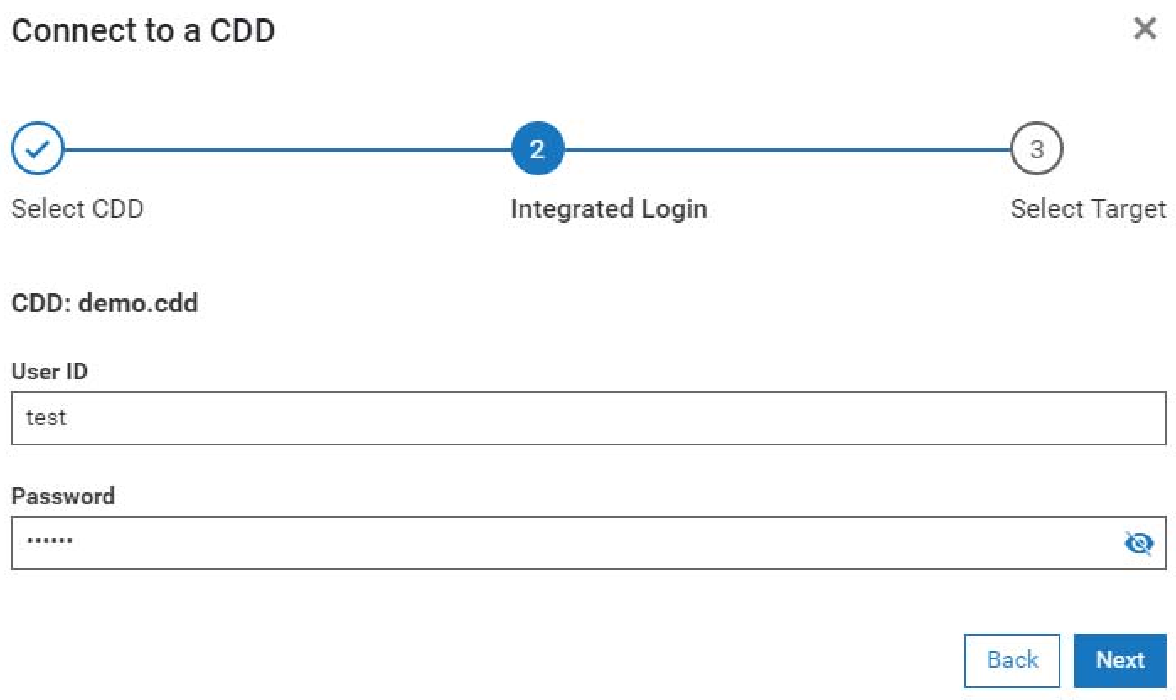
If a CDD is uploaded and there is a name conflict, the previous CDD of the same name will be renamed with and underscore followed by a timestamp. In the above example, the CDD Demo2.CDD was uploaded when there was already a CDD with that name. In this case, the original Demo2.CDD was renamed to Demo2_20200922T145806.CDD.

- **Download a CDD**
  When multiple users are editing CDDs, it is recommended that a CDD be downloaded from the server before making edits to it. This keeps the primary copy of the CDD on the server. If edits need to be made, the CDD should be downloaded, edited and then uploaded again. In this case, the previous version will be renamed with a timestamp. If you wish to revert to a previous version of the CDD, that CDD can be activated.

Once a CDD has been selected from the Available CDDs list, press Next to proceed to Step 2: Integrated Login

Integrated Login

The Integrated Login step collects the CONNX User ID and Password for the CDD. The DataSync Server will not be able to prompt for back end credentials in order to tie the CDD user and password to the back end user id and password so a CDD user will need to be created prior to activating the CDD. This can be accomplished by connecting to the CDD with InfoNaut and letting InfoNaut automatically set up the CDD user or an administrator can edit the CDD and add the desired users and associate the appropriate back end credentials with each user id. See the CONNX User Guide for more information on CDD Management.



Connect to a CDD

CDD: demo.cdd

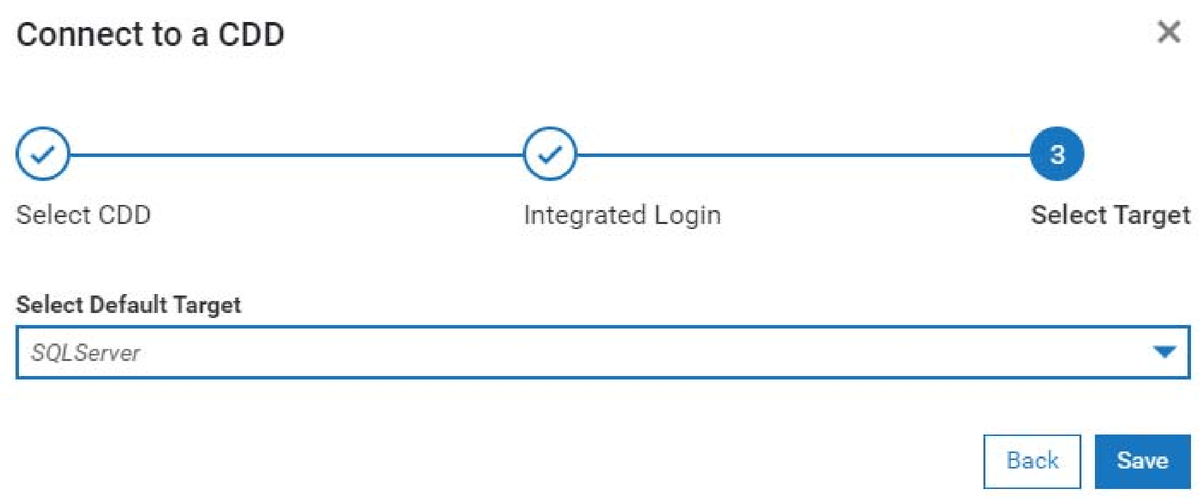
User ID: test

Password: ••••••

Once the User ID and Password have been entered, click Next to continue to Step 3: Select Target

Select Target

The final step is to select a default target database from the **Select Default Target** dropdown. The value can be changed at any time in the Sync Behavior section of the Setting Menu.



Connect to a CDD

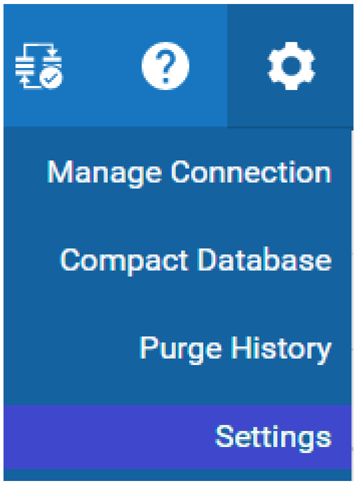
Select Default Target: SQLServer

After selecting a default target database, click **Save** and the CDD will become the active CDD.

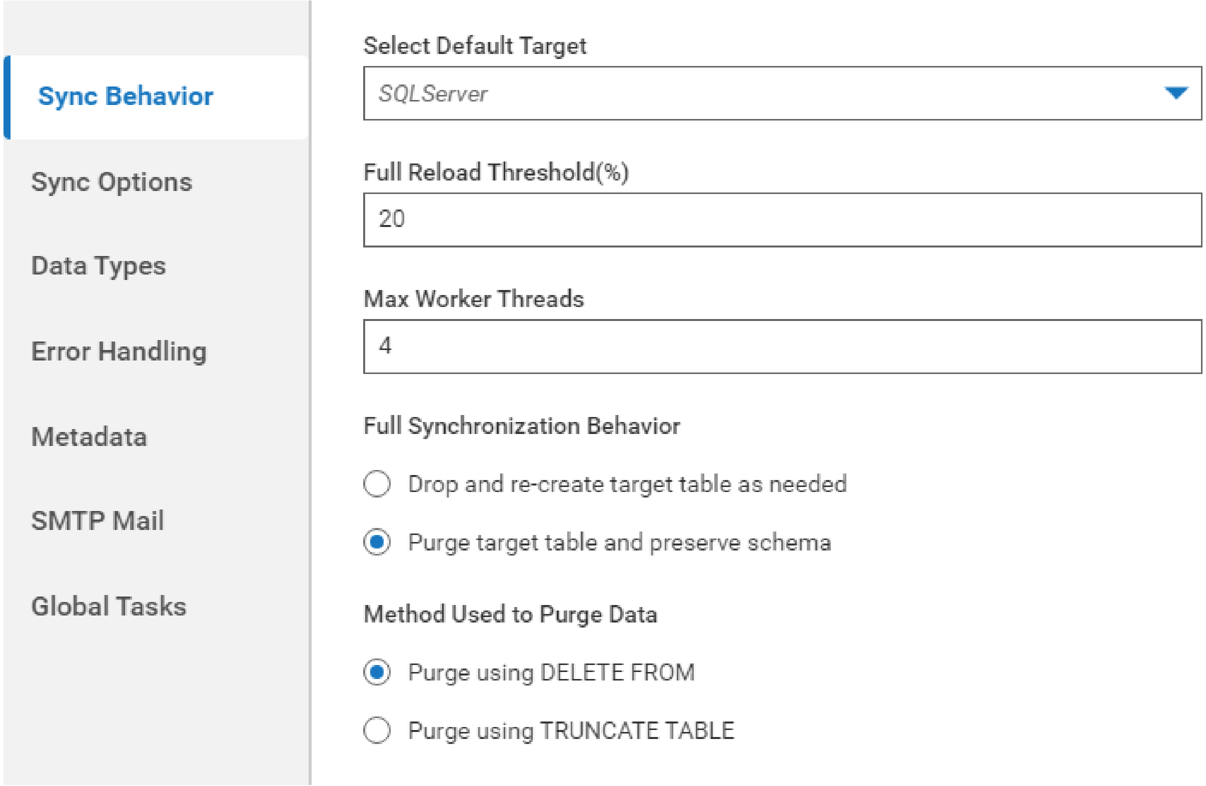
Settings Menu

Sync Behavior

DataSync settings are accessed from the Gear menu



The Sync Behavior page controls how DataSync behaves during the synchronization process



Select Default Target: SQLServer

Full Reload Threshold(%): 20

Max Worker Threads: 4

Full Synchronization Behavior

○ Drop and re-create target table as needed

● Purge target table and preserve schema

Method Used to Purge Data

● Purge using DELETE FROM

○ Purge using TRUNCATE TABLE

- **Select Default Target**
  This drop down box contains the list of all the databases defined in the CDD. The selected database will be used as the target for Table Syncs and the default target for Transformation Syncs. The target database for transformations can be overridden when defining transformations.

- **Full Reload Threshold(%)**
  The percentage amount that appears in Full Reload Threshold is the number that will determine if it is more efficient to perform a full reload or an incremental update based on the percentage of rows changed in the source table. This number should be very small, in the range of 3 to 20 percent.
- **Max Worker Threads**
  Enter a number in the Max Worker Threads text box. (Recommended = 4 threads per CPU on the Data Sync machine) The maximum number is the number of threads on the PC (and also the number of servers on the host database server) that can be used when processing the synchronization. For example, if there are 1000 tables of the same size, 1 out of 3 threads would process about 333 tables each. Increasing the number of threads uses more resources on the PC and on the host database server, but can also increase performance.

  Take note that increasing the thread count to a level that is too high can severely impact performance and possibly cause failures. It is strongly recommended that you do not exceed ten threads per CPU available on the Data Synchronization machine and also to consider the system's total memory, since each thread can allocate large blocks of memory when processing large tables.

  In order to test the effects of multiple threads, create a group and then synchronize the group.

  Tip: *A few tests with different threading values can determine a good level for Max Worker Threads for your system through examination of the group synchronization time.*
- **Full Synchronization Behavior**
  There are two methods to process a full synchronization. One is to drop and recreate the target table every time. The other is to delete the data in the target table. When deleting data, we can either truncate the data or delete the data. **Truncation** is typically much faster, but not all systems support truncate. If you select truncation, and the target does not support truncation, you will receive a warning and **Delete** will be used instead. An advantage to dropping and recreating the table is that changes in the source table will automatically be reflected in the target and so the system is self-healing. However, there may be customizations on the target server. In this case, we want to purge the data instead. **Truncation**, while faster, is not typically logged and therefore the user will have to redo a failed sync. **Deletion** works on every system, but is much slower than dropping and adding the schema or truncation.

  If the target database does not support truncation, a warning appears for every table selected for deletion even though the synchronization continues and succeeds.

  There are two options for **Full Synchronization Behavior:**

  1. **Drop and recreate target table as needed** is the default behavior.
  2. Select **Purge target table and preserve schema** if:

     You do not want to delete indices in the target table that are not in the source table. This could happen if the target table is used for specialized reporting
     or
     Your security policy does not allow you to drop tables

Method Used to Purge Data:

There are two options for **Purge target table and preserve schema**:

1. **Purge using DELETE FROM**
   Choose the Delete feature when all deletes are logged, and can be rolled back in case something goes wrong.

   If there is an error on a Truncate command, the database is left in the state of the error, as the command is permanent. If a Delete command has problems, the database rolls back to the previous state on transaction failure.
2. **Purge using TRUNCATE TABLE** (for target servers where truncate is available)
   The CONNX Data Synchronization tool may perform a Truncate more quickly than a Delete. Choose Truncate if:

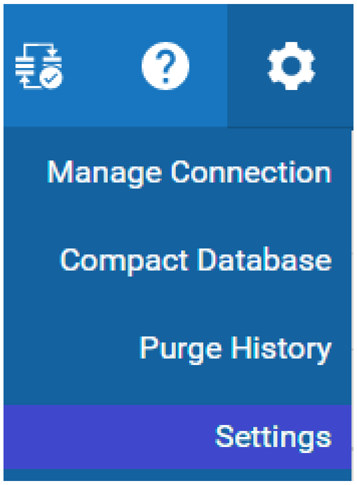
   You do not want to delete indexes in the target table that are not in the source table. This could happen if the target table is used for specialized reporting.
   or
   Your security policy does not allow you to drop tables.

   Truncate does not activate triggers, referential integrity constraints, or other database restrictions. Since database integrity is maintained on the source database for the synchronization, choose Truncate if performance is a concern.

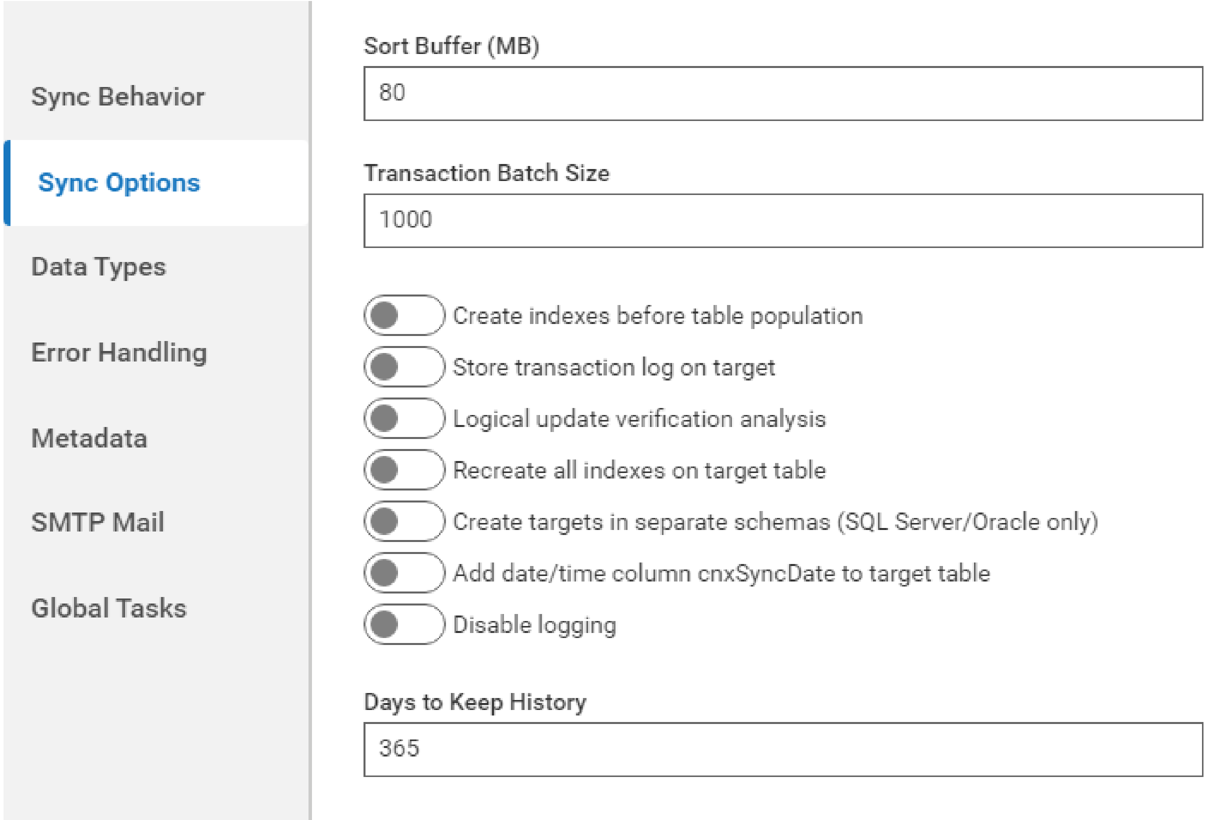
   If an unsupported database is chosen, the sync succeeds, but a warning message is issued.

Sync Options

DataSync settings are accessed from the Gear menu



The Sync Options page controls configuration settings for DataSync



Sort Buffer (MB)
80

Transaction Batch Size
1000

○ Create indexes before table population
○ Store transaction log on target
○ Logical update verification analysis
○ Recreate all indexes on target table
○ Create targets in separate schemas (SQL Server/Oracle only)
○ Add date/time column cnxSyncDate to target table
○ Disable logging

Days to Keep History
365

- **Sort Buffer**
  In the Sort buffer text box, enter the maximum amount of memory in megabytes used per thread to sort files. The number should reflect the amount of memory available on the machine used as

the sorting buffer and the number of threads used. The actual amount used may be much smaller if the tables being synchronized are small. The default sort buffer of 80 megabytes should be compatible with most systems that follow our recommended guidelines for minimal machine configuration.

- **Transaction Batch Size**
  Transaction Batch Size refers to how many records are inserted into the target database in a single transaction. Some database systems have problems if too many records are involved in a transaction, and DataSync offers the ability to set this field manually. 1000 is the default.
- **Create indexes before table population**
  To reorder the synchronization process so that indexes are created first, select Create indexes before table population. This can have either a negative or positive effect on performance and space requirements of a full synchronization, depending upon the database system. For instance, if the target is Oracle, it will be faster to create the indexes after the load of the data. Inserting rows with an existing index updates the indexes on every insert, whereas creation of an index at the end of the synchronization is somewhat more efficient.

  On the other hand, with RMS as the target system, if indexes are created as the synchronization completes, then two entire copies of the file will exist on the target system, as a sequential file is converted into an indexed file with a CONVERT/FDL command. The best choice strongly depends on the type of database system you are using and on the amount of resources available for performing the synchronizations.
- **Store transaction log on target**
  To create a transaction log which is stored in the target database, select the Store Transaction Log on Target option. If this option is not selected, the transaction log is not created. A transaction log is a table that contains the unique index information for each row that was inserted, updated, or deleted along with a flag that tells what sort of action was performed on each row. The transaction log table will be created on the target database and will have the name of the source table with the characters _ACD appended to the end. By using a transaction log, systems that need an audit process can tell exactly what was changed and what sort of operation was performed on the data. It should be noted, however, that the creation of a transaction log may slow synchronizations.
- **Logical update verification analysis**
  Logical Update Verification Analysis is used in conjunction with the Store Transaction Log on Target option.

  For each change verified, the target table transaction log will change the 'C' (for changed) in the ACD column of the transaction log to 'V' (for verified). If Logical Update Verification Analysis is selected, for each row that has had a change detected via CRC differences, a column by column search is performed to confirm a difference.

  Logical Update Verification Analysis should only be used in special cases, where a careful analysis has shown that more data is being processed than is necessary. Selecting this option makes the synchronizations run more slowly, since for every change detected, every single column in the record is compared from source to target - one a column at a time. If a post-process step that is very slow must be performed on the changes (for example some expensive ETL operations) then logical update verification analysis might be worthwhile. Some examples might include the following:
  - Multiple physical values with different internal representations but the same physical meaning, such as:

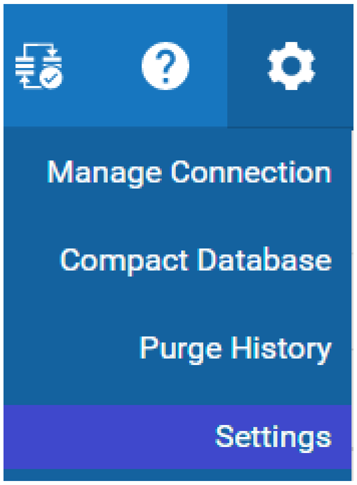
    Positive and negative zero on one's complement machines
    Positive signed and unsigned decimal numbers (e.g. signed + 1 and unsigned + 1 have the same meaning but different internal representation

  - Insignificant changes. If, for example, the very last bit of a floating point number changed in the mantissa, the actual meaning of what is stored may not really have changed. For instance, someone might have edited a field that contained the value "2.1" as a floating point number. Though no changes were made, the value was saved again into the database. It is possible that the number will differ slightly due to the nature of floating point. A comparison shows that both the source and target numbers are intended as 2.1, but a CRC comparison shows that some bit has changed and consider them as completely different.
  - Masked values. If, for example, a null-terminated string is stored in the source database, anything after the null string terminator is a physical difference that causes a different CRC, but the data stored may be the same. For instance, the string of bytes 'D', 'o', 'g', 0, 'f', 'o', 'o', 'd',0 will be interpreted simply as "Dog" since a null terminator (zero) follows the 'g'. The string of bytes 'D', 'o', 'g', 0, 'f', 'i', 'g', 'h', 't', 0 will also be interpreted simply as "Dog" since a null terminator (zero) follows the first 'g'. Anything beyond the null terminator has no meaning as far as the database is concerned. But CRC values will indicate that there has been a change.

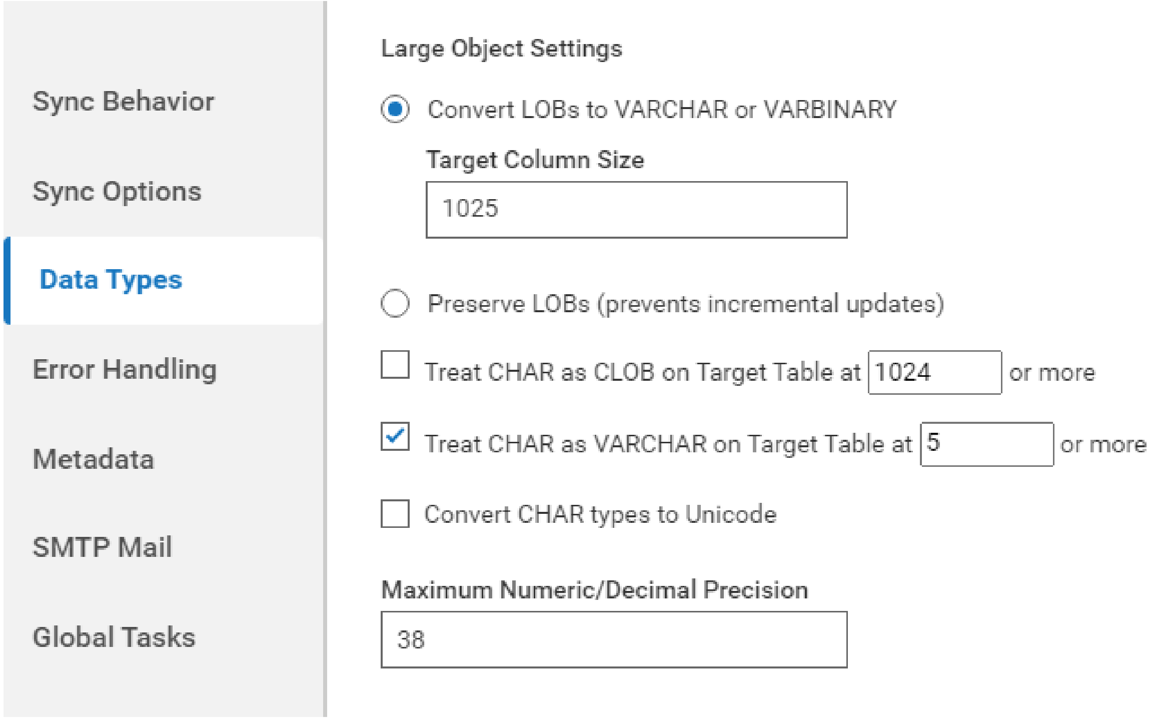
- **Recreate all indexes on target table**
  By default, only the unique index used to perform incremental updates is created on the target table after synchronization. Select Recreate all indexes on target table to recreate all indexes from the source table on the target table.
- **Create targets in separate schemas (SQL Server/Oracle only)**
  If you are using SQL Server or Oracle is the target database, select Create target tables in separate schemas to create target tables under different owners. DataSync creates a user for the purpose of creating tables with identical table names from different source catalogs. For example, if you want to synchronize both the test and production version of a table into a single SQL Server or Oracle database, CONNX fully qualifies the target name (product.table or test.table) so that there is no name collision.
- **Add date/time column cnxSyncDate to target table**
  This option will append a date and time column named cnxSyncDate to the end of each row in the synchronized target tables. The date and time of the last synchronization operation performed against the target row appears in the new column.
- **Disable logging**
  This option will prevent log information from being written to the datasync.log file. The datasync.log file is stored in the datasync folder.
- **Days to Keep History**
  Synchronization records that are older than the specified value will be purged from the database. This helps keeps the database size smaller. The default is 365 days.

Data Types

DataSync settings are accessed from the Gear menu



The Data Types page controls how DataSync interprets certain data types



Large Object Settings

● Convert LOBs to VARCHAR or VARBINARY

Target Column Size

1025

○ Preserve LOBs (prevents incremental updates)

□ Treat CHAR as CLOB on Target Table at 1024 or more

☑ Treat CHAR as VARCHAR on Target Table at 5 or more

□ Convert CHAR types to Unicode

Maximum Numeric/Decimal Precision

38

- **Convert LOBs to VARCHAR or VARBINARY**
  This setting tells DataSync to change LOBs to either VARCHAR or VARBINARY depending on the data. Enter the target column size (default is 1024). This is the default setting. When set, Preserve LOBs will not be set.
- **Preserve LOBs**
  This setting tells DataSync to send LOBs to the target without converting them to VARCHAR or

VARBINARY. When set, incremental updates will not be allowed and only full loads will be supported. When set, Convert LOBs to VARCHAR or VARBINARY will not be set.

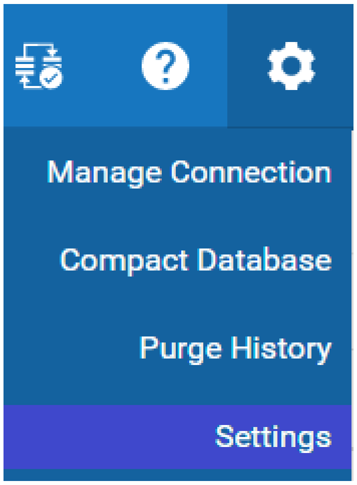
- **Treat CHAR as CLOB on Target Table**
  If your target data base does not support very large character fields,select Treat CHAR as CLOB on Target Table and enter the field size. The default value is 1024 but there is no minimum or maximum value.
- **Treat CHAR as VARCHAR on Target Table**
  If character fields are long and not fully populated, select Treat CHAR as VARCHAR on Target Table and enter the field size.

  Note: The field should contain at least 5 characters as there is minimal space savings for short CHAR fields.

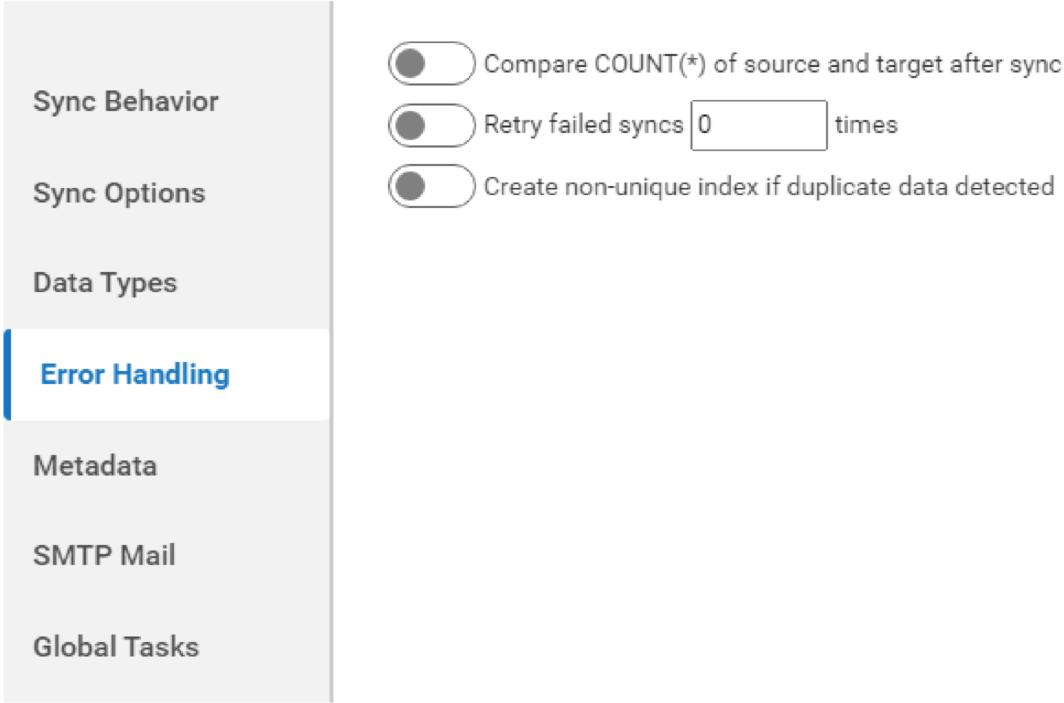
  Example: If the source table has a CHAR(100) field that averages 10 characters in length, it will be transformed into a VARCHAR(100) field on the target table after the synchronization. This could save an average of 90 characters per row.
- **Maximum Number/Decimal Precision**
  Enter the maximum numeric or decimal precision value in Maximum Numeric/Decimal Precision.

Error Handling

DataSync settings are accessed from the Gear menu



The Error Handling page controls how DataSync handles certain types of errors



○ Compare COUNT(*) of source and target after sync

○ Retry failed syncs 0 times

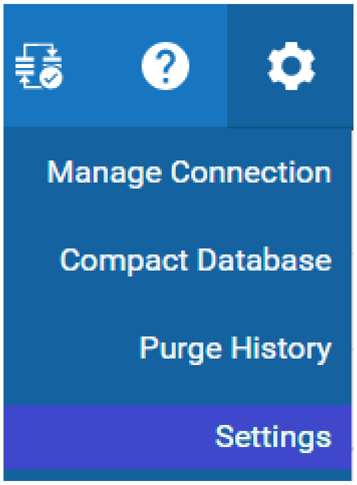
○ Create non-unique index if duplicate data detected

- **Compare COUNT(*) of source and target after sync**
  To compare source and target table counts, select Compare COUNT(*) of source and target after sync.
  Note: This option can only be selected if the sync is performed during a "batch window" and the source database remains static during that time.

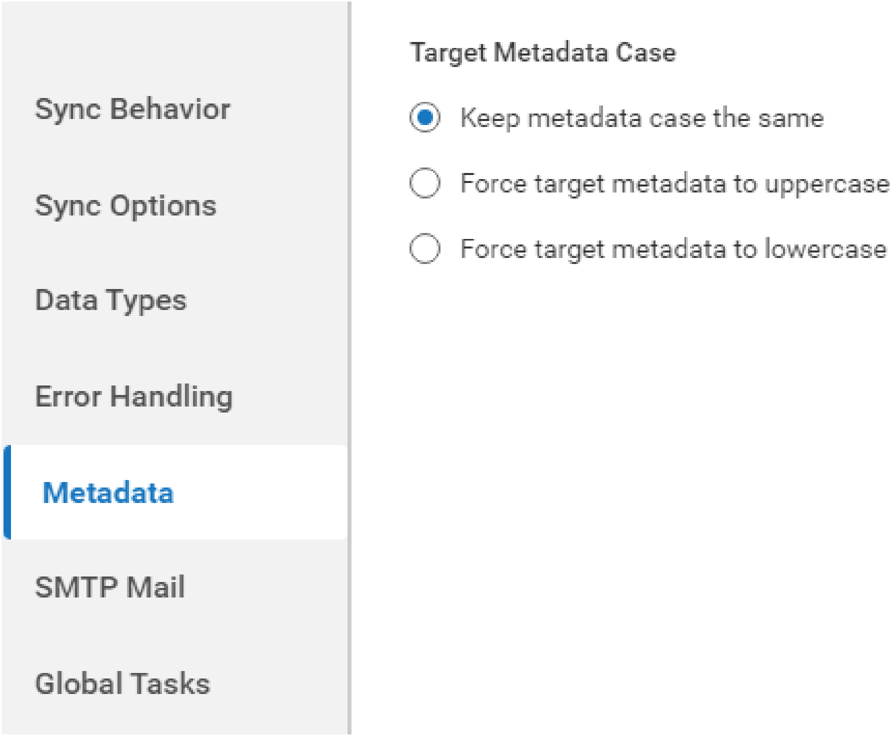
- **Retry failed syncs**
  To retry failed synchronizations, select Retry failed syncs and enter the number of times to retry the synchronization.
- **Create non-unique index if duplicate data detected**
  Selecting Create non-unique index if duplicate data detected instructs CONNX to create a non-unique primary key instead of a unique primary key during a full update if duplicate data is detected.

Handling Metadata Case

DataSync settings are accessed from the Gear menu



The Metadata page controls how DataSync handles the case of the target table's metadata



Target Metadata Case

● Keep metadata case the same

○ Force target metadata to uppercase

○ Force target metadata to lowercase

CONNX by default uses quoted identifiers when creating target tables. If column or table names are mixed-case, then double quotes must be used to refer to them in SQL statements.

However, it may make more sense for the target metadata to be in all uppercase, or all lowercase.

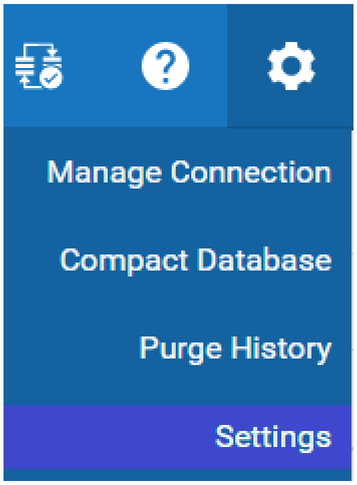
- If Oracle metadata is in all uppercase, then it is treated as a case-insensitive identifier and double quotes are not needed.
- If PostgreSQL metadata is in all lowercase, then it is treated as case insensitive.

CONNX Data Synchronization allows you to choose the case for the target metadata.

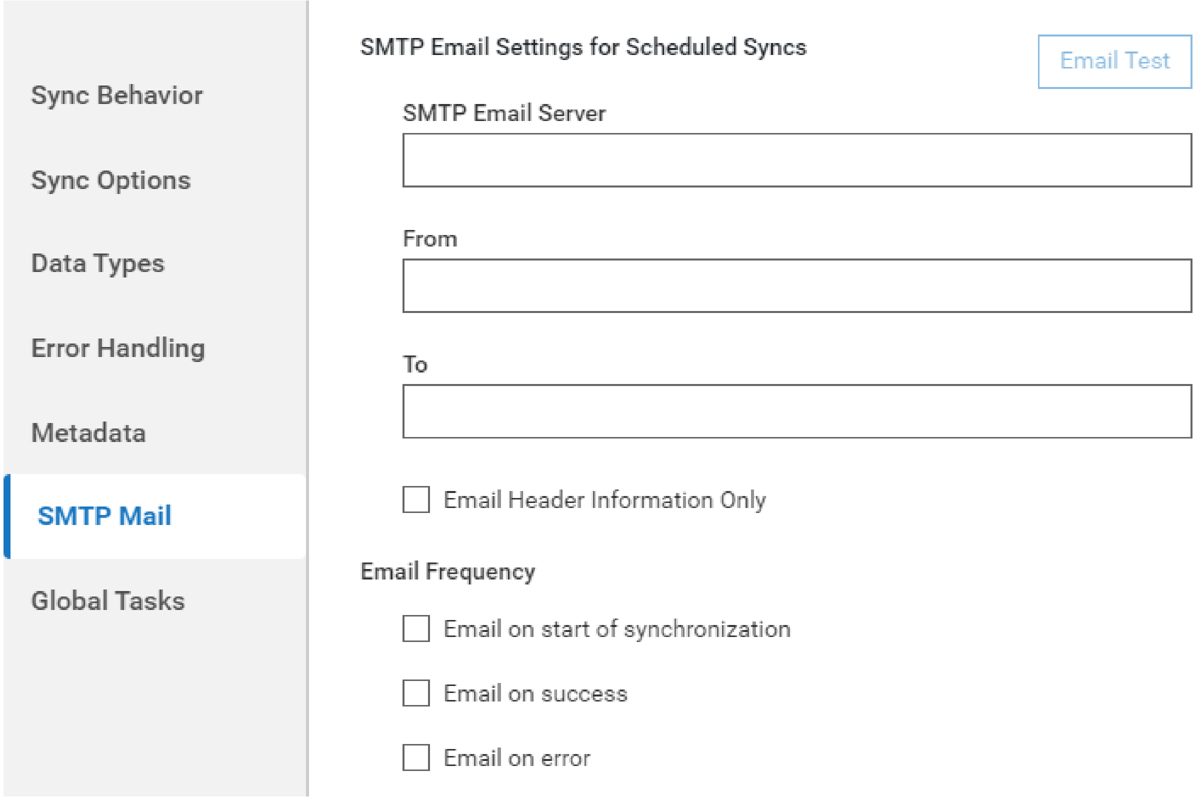
- **Target Metadata Case**
    - **Keep metadata case the same**
      Metadata on target will be unchanged
    - **Force target metadata to uppercase**
      Metadata on target will be in uppercase
    - **Force target metadata to lowercase**
      Metadata on target will be in lowercase

SMTP Mail from Scheduled Syncs

DataSync settings are accessed from the Gear menu



CONNX DataSync has the ability to have email notification sent to a group or to individuals after a scheduled sync has been performed. This enables administrators to have instant access to information relating to DataSync activity. In addition, it can also send email if problems result from a synchronization. To use this ability, the user should navigate to the SMTP Mail section in Settings



SMTP Email Settings for Scheduled Syncs

SMTP Email Server

From

To

□ Email Header Information Only

Email Frequency

□ Email on start of synchronization

□ Email on success

□ Email on error

The following values should be filled in:

- **SMTP Email Server**
  This is the server that will relay SMTP Mail.

- **From**
  This reflects the sender's email address. For this example, the sender and receiver are the same.
- **To**
  This indicates the recipient of the email.
- **Frequency**
  Emails can be sent on start, success, and error conditions.

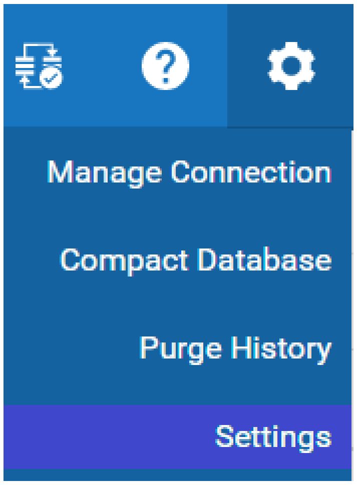
An **Email Test** button is provided to verify that the SMTP settings are correct. After pressing this button, an immediate response will be provided indicating that the settings are correct or incorrect.

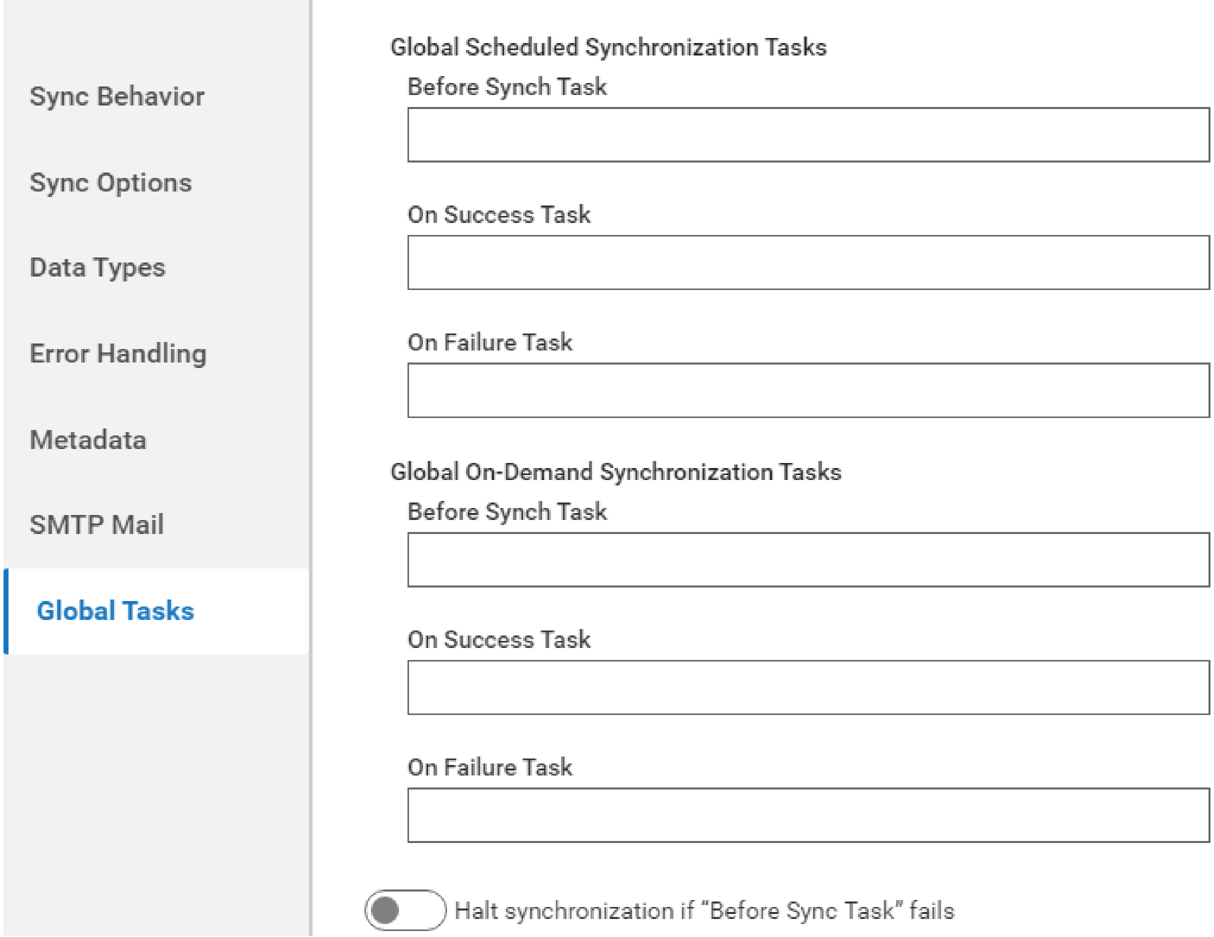
After a scheduled sync has been performed, email messages will be sent to the designated recipients and the content of the email will indicate detailed status relating to the synchronization that was performed.

The subject will always start with the word DataSync. Rules can be set up in most Email clients to have these messages placed to an appropriate folder.

Global Tasks

DataSync settings are accessed from the Gear menu





Global Scheduled Synchronization Tasks

Before Synch Task

On Success Task

On Failure Task

Global On-Demand Synchronization Tasks

Before Synch Task

On Success Task

On Failure Task

○ Halt synchronization if “Before Sync Task” fails

CONNX Data Synchronization enables the user to execute globally scheduled tasks that occur simultaneously with synchronization of data. These tasks can be any sort of operation, including batch files, executable files, backups, or data dumps. Execution of the command is automatic.

Tasks that should not be included in globally scheduled synchronization tasks include any task that requires input from a user. Such tasks cause the synchronization to stop abruptly as it waits for user input.

There are three types of tasks available:

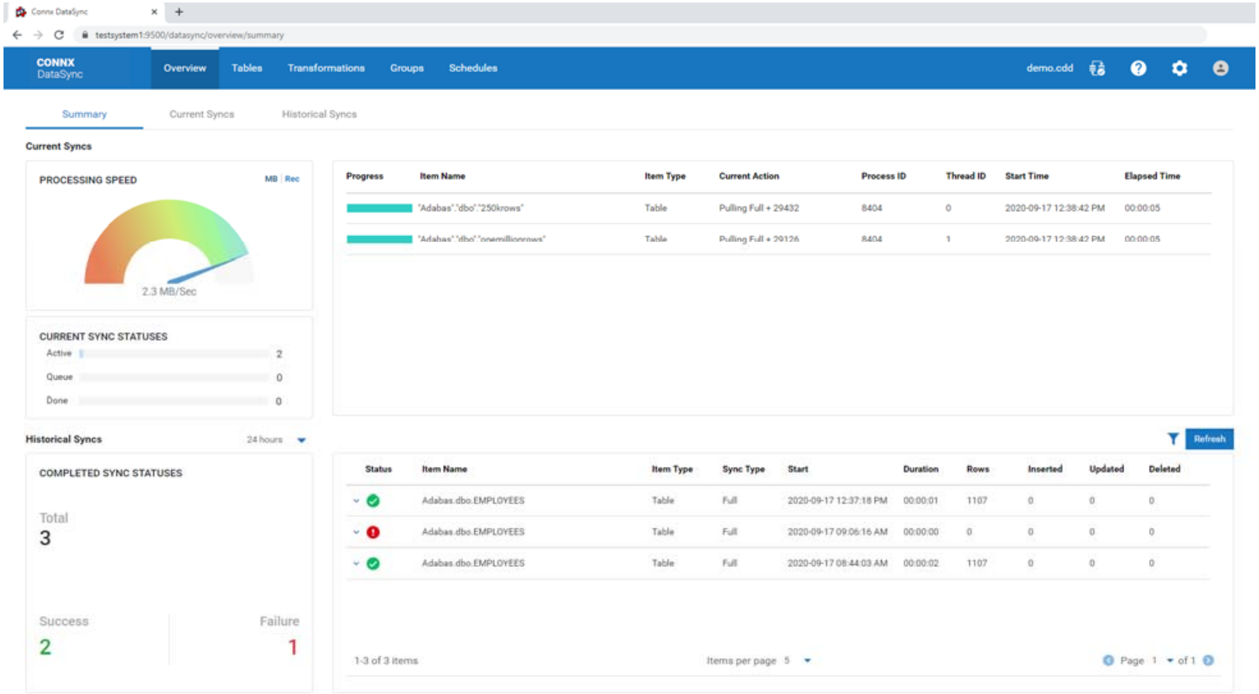
- **Global Scheduled Synchronization Tasks**
  Global Scheduled Synchronization Tasks can be performed for every scheduled synchronization.
    -
    - **Before Sync Task**
      Enter the name of the executable file or batch script to be executed before scheduled syncs start. Include the path to the executable along with any parameters.
    - **On Success Task**
      Enter the name of the executable file or batch script to be executed after syncs complete with no errors. Include the path to the executable along with any parameters.
    - **On Failure Task**
      Enter the name of the executable file or batch script to be executed after syncs complete with an error. Include the path to the executable along with any parameters.
    -
- **Global On Demand Synchronization Tasks**
  Global On-Demand Synchronization tasks can be performed for every interactive synchronization that is manually started.
    - **Before Sync Task**
      Enter the name of the executable file or batch script to be executed before scheduled syncs start. Include the path to the executable along with any parameters.
    - **On success Task**
      Enter the name of the executable file or batch script to be executed after syncs complete with no errors. Include the path to the executable along with any parameters.
    - **On Failure Task**
      Enter the name of the executable file or batch script to be executed after syncs complete with an error. Include the path to the executable along with any parameters.
- **Per Schedule Synchronization Tasks** (Located on the Schedule page)

Select **Halt synchronization if "Before Sync Task" fails** to prevent syncs from running if the Before Sync Task is not successful.

Overview page

Summary Page

The DataSync **Overview Summary Page** is the landing page for the DataSync administrator and provides real time information about synchronization jobs currently running in the system as well has historical summary information about completed jobs.



Progress	Item Name	Item Type	Current Action	Process ID	Thread ID	Start Time	Elapsed Time
	"Adabas"."dbo"."250krows"	Table	Pulling Full + 29432	8404	0	2020-09-17 12:38:42 PM	00:00:05
	"Adabas"."dbo"."onemillionrows"	Table	Pulling Full + 29126	8404	1	2020-09-17 12:38:42 PM	00:00:05

Status	Item Name	Item Type	Sync Type	Start	Duration	Rows	Inserted	Updated	Deleted
✓	Adabas.dbo.EMPLOYEES	Table	Full	2020-09-17 12:37:18 PM	00:00:01	1107	0	0	0
!	Adabas.dbo.EMPLOYEES	Table	Full	2020-09-17 09:06:16 AM	00:00:00	0	0	0	0
✓	Adabas.dbo.EMPLOYEES	Table	Full	2020-09-17 08:44:03 AM	00:00:02	1107	0	0	0

The Summary page is divided into two sections. The top section represents information about currently running Synchronization jobs and the bottom section provides a summary of historical syncs. Some functions on this screen use icons. The description of these icons will have a hyperlink to the Glossary of Icons which contains a picture of the icon and an explanation of its purpose.

The **Current Syncs** section provides real time information for all synchronization tasks currently running. This section auto refreshes and always shows current information for the system.

- **Processing Speed**
  Processing Speed is represented by the speedometer graph and shows the overall speed of all running jobs in either megabytes per second or records per second. The graph can be toggled between these two metrics by clicking either MB or Rec in the upper right corner of the speedometer's display.

- **Current Sync Statuses**
  This section provides information on how many jobs are currently running, how many are queued waiting to run and have many have finished.

- **Individual Progress of each running Job**
  The section to the right of Processing Speed and Current Sync Statuses shows statistics for each running job.
  - The **Progress** column provides a bar graph that indicates progress.

  - The **Item Name** column contains the name of the table, transform or group that is currently running
  - **Current Action** provides information about what DataSync is doing. For example, Pulling Full+29432 indicates that DataSync is performing a full reload of the target table and has retrieved 29432 records from the source. Other actions may be Analyzing, Inserting, Updating or Deleting.
  - **Process ID** displays the process ID or PID of the DataSync process that is performing the sync. The Summary screen shows information for all DataSync jobs in the system so if there are multiple scheduled syncs running along with on demand syncs, there may be several PIDs represented on the display.
  - **Thread ID** displays the thread ID within the process. Each table or transform will be synchronized in a separate thread. The maximum number of worker threads per process can be adjusted in Settings.
  - **Start Time** is a timestamp indicating when the synchronization task started
  - **Elapsed Time** displays how long the synchronization has been running. It is in the form hh:mm:ss

The **Historical Syncs** section provides a summary of the completed synchronization tasks as well as a table view showing the individual tasks and their statistics.

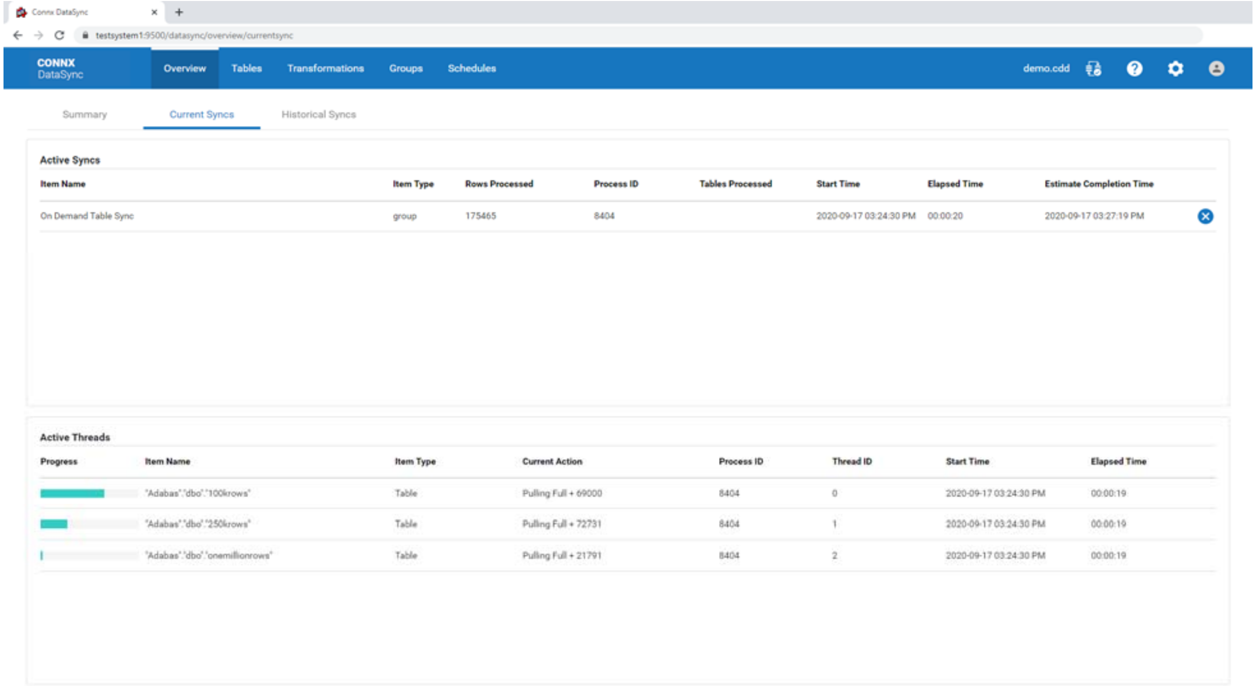
- **Control buttons**
  There are three control buttons above the information section in the lower half of the display. They are:
  - **Time frame specification**. This drop down box allows the time frame for historical data to be specified. Possible values are **24 hours**, **1 week**, **1 month**, **1 year** and **all time**
  - **Search Filter**
    Pressing this button activates search fields for the **Status**, **Item Name** and **Item Type** fields. To filter the results in the list of historical synchronization tasks, enter a value in one or more of these search fields. For example, entering the word EMPLOYEES in the *Item Name* search box will cause only those tasks that have the word EMPLOYEES in the Item Name to be displayed.
  - **Refresh** This button refreshes the lower portion of the display.

- **Completed Sync Statuses**
  Completed Sync Statuses are displayed in the lower left section of the display. This section shows the total number of sync tasks that have been run in the time frame specified in the Time Frame drop down. It also shows the total number of Successful tasks as well as the total number of tasks that failed with an error.

- **List of Historical Syncs**
  The lower right section contains a table list of the synchronization tasks that have run during the time frame specified in the time frame drop down. For details about the information presented in this table, please refer to the Historical Syncs topic later in this section.

Current Syncs

The Current Syncs page displays detailed information about synchronization jobs currently running on the DataSync Server. These can be jobs that are running from the scheduler or On Demand jobs running from multiple users connecting through the REST interface. There are two sections to this screen: **Active Syncs** and **Active Threads**



Active Syncs

Item Name	Item Type	Rows Processed	Process ID	Tables Processed	Start Time	Elapsed Time	Estimate Completion Time
On Demand Table Sync	group	175465	8404		2020-09-17 03:24:30 PM	00:00:20	2020-09-17 03:27:19 PM

Active Threads

Progress	Item Name	Item Type	Current Action	Process ID	Thread ID	Start Time	Elapsed Time
▬	"Adabas"."dbo"."100krows"	Table	Pulling Full + 69000	8404	0	2020-09-17 03:24:30 PM	00:00:19
▬	"Adabas"."dbo"."250krows"	Table	Pulling Full + 72731	8404	1	2020-09-17 03:24:30 PM	00:00:19
▏	"Adabas"."dbo"."onemillionrows"	Table	Pulling Full + 21791	8404	2	2020-09-17 03:24:30 PM	00:00:19

- **Active Syncs**
  The Active Syncs table displays a list of all active synchronization processes running in the system. Below is a description of each column in the table:
  - **Item Name**
    Identifies the name of a scheduled job or in the case of on demand syncs, a name is generated that describes the running job
  - **Item Type**
    Identifies the type of synchronization running. Possible values are Table, Transform and Group
  - **Rows Processed**
    A real time display of the number of rows processed. This information can be used to determine how far along a synchronization is
  - **Process ID**
    The Windows Process ID or PID for the synchronization task that is running
  - **Tables Processed**
    The number of tables processed by this task. Useful when the Item Type is Group
  - **Start Time**
    A timestamp indicating when the synchronization task started
  - **Elapsed Time**
    Displays how long the synchronization has been running. It is in the form hh:mm:ss

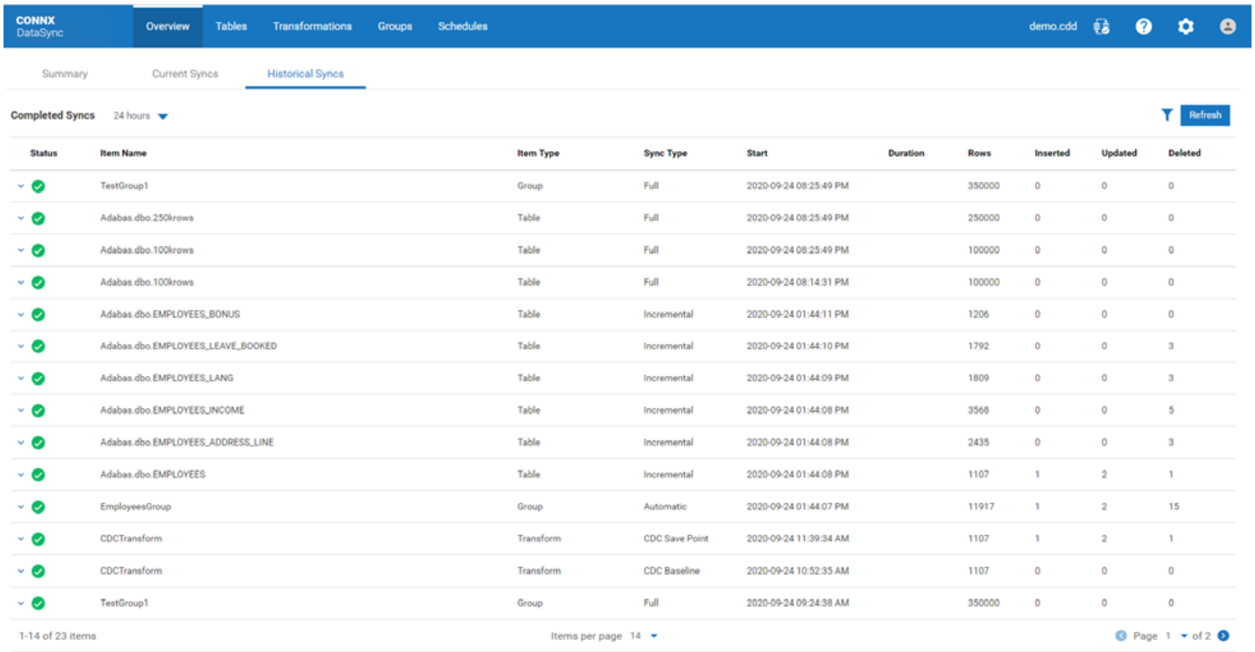
  - **Cancel Task button**
    The Cancel Task button can be used to cancel any running synchronization task regardless of whether it is a scheduled task, an on demand task running locally or and on demand task initiated from anther computer

- **Active Threads**
  The Active Threads table displays a list of all the active threads for each active task.
  - **Progress**
    A graphical horizontal bar showing the progress of the item being synchronized
  - **Item Name**
    The name of the table or transform being synchronized
  - **Item Type**
    The type (table or transform) of the item being synchronized
  - **Current Action**
    Describes what the part of the process the thread is currently executing
  - **Process ID**
    This identifies which process the thread is associated with. This is useful from cross referencing with the Active Syncs list for the purpose of identifying a Sync that you may wish to cancel
  - **Thread ID**
    Identifies which thread within the Sync process
  - **Start Time**
    A timestamp indicating when the synchronization task started
  - **Elapsed Time**
    Displays how long the synchronization has been running. It is in the form hh:mm:ss

Historical Syncs

The Historical Syncs page displays information for Sync tasks that have finished.

Some functions on this screen use icons. The description of these icons will have a hyperlink to the Glossary of Icons which contains a picture of the icon and an explanation of its purpose.



Completed Syncs 24 hours

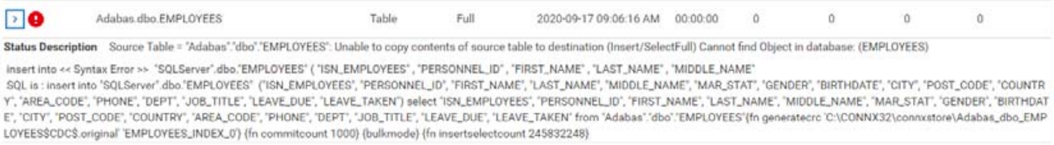
Status	Item Name	Item Type	Sync Type	Start	Duration	Rows	Inserted	Updated	Deleted
✓	TestGroup1	Group	Full	2020-09-24 08:25:49 PM		350000	0	0	0
✓	Adabas.dbo.250krows	Table	Full	2020-09-24 08:25:49 PM		250000	0	0	0
✓	Adabas.dbo.100krows	Table	Full	2020-09-24 08:25:49 PM		100000	0	0	0
✓	Adabas.dbo.100krows	Table	Full	2020-09-24 08:14:31 PM		100000	0	0	0
✓	Adabas.dbo.EMPLOYEES_BONUS	Table	Incremental	2020-09-24 01:44:11 PM		1206	0	0	0
✓	Adabas.dbo.EMPLOYEES_LEAVE_BOOKED	Table	Incremental	2020-09-24 01:44:10 PM		1792	0	0	3
✓	Adabas.dbo.EMPLOYEES_LANG	Table	Incremental	2020-09-24 01:44:09 PM		1809	0	0	3
✓	Adabas.dbo.EMPLOYEES_INCOME	Table	Incremental	2020-09-24 01:44:08 PM		3568	0	0	5
✓	Adabas.dbo.EMPLOYEES_ADDRESS_LINE	Table	Incremental	2020-09-24 01:44:08 PM		2435	0	0	3
✓	Adabas.dbo.EMPLOYEES	Table	Incremental	2020-09-24 01:44:08 PM		1107	1	2	1
✓	EmployeesGroup	Group	Automatic	2020-09-24 01:44:07 PM		11917	1	2	15
✓	CDCTransform	Transform	CDC Save Point	2020-09-24 11:39:34 AM		1107	1	2	1
✓	CDCTransform	Transform	CDC Baseline	2020-09-24 10:52:35 AM		1107	0	0	0
✓	TestGroup1	Group	Full	2020-09-24 09:24:38 AM		350000	0	0	0

1-14 of 23 items Items per page 14 Page 1 of 2

- **Control buttons**
  There are three control buttons above the information section in the lower half of the display. They are:
  - **Time frame specification**.
    This drop down box allows the time frame for historical data to be specified. Possible values are **24 hours**, **1 week**, **1 month**, **1 year** and **all time**
  - **Search Filter**
    Pressing this button activates search fields for the **Status**, **Item Name** and **Item Type** fields. To filter the results in the list of historical synchronization tasks, enter a value in one or more of these search fields. For example, entering the word EMPLOYEES in the *Item Name* search box will cause only those tasks that have the word EMPLOYEES in the Item Name to be displayed.
  - **Refresh** This button refreshes the display.

- **Completed Syncs List**
  The list displays information about all tasks that have completed in the time frame specified in the Time frame specification control. Below is a description of each column in the table:
  - **More|Less Chevron**
    If more information is present for a row than can be displayed, clicking the *more chevron* will expand the row. Clicking the *less chevron* will collapse the information. Below is an example of a row that contains an error message. The *more chevron* has been clicked to

display the additional information.



	Adabas.dbo.EMPLOYEES	Table	Full	2020-09-17 09:06:16 AM	00:00:00	0	0	0	0

Status Description Source Table = "Adabas"."dbo"."EMPLOYEES": Unable to copy contents of source table to destination (Insert/SelectFull) Cannot find Object in database: (EMPLOYEES)

```
insert into << Syntax Error >>  "SQLServer".dbo."EMPLOYEES" ( "ISN_EMPLOYEES" , "PERSONNEL_ID" , "FIRST_NAME" , "LAST_NAME" , "MIDDLE_NAME"
SQL is : insert into "SQLServer".dbo."EMPLOYEES"  ("ISN_EMPLOYEES", "PERSONNEL_ID", "FIRST_NAME", "LAST_NAME", "MIDDLE_NAME", "MAR_STAT", "GENDER", "BIRTHDATE", "CITY", "POST_CODE", "COUNTR
Y", "AREA_CODE", "PHONE", "DEPT", "JOB_TITLE", "LEAVE_DUE", "LEAVE_TAKEN") select "ISN_EMPLOYEES", "PERSONNEL_ID", "FIRST_NAME", "LAST_NAME", "MIDDLE_NAME", "MAR_STAT", "GENDER", "BIRTHDAT
E", "CITY", "POST_CODE", "COUNTRY", "AREA_CODE", "PHONE", "DEPT", "JOB_TITLE", "LEAVE_DUE", "LEAVE_TAKEN" from "Adabas"."dbo"."EMPLOYEES"{fn generatecrc 'C:\CONNX32\connxstore\Adabas_dbo_EMP
LOYEES$CDC$.original' 'EMPLOYEES_INDEX_0'} {fn commitcount 1000} {bulkmode} {fn insertselectcount 245832248}
```

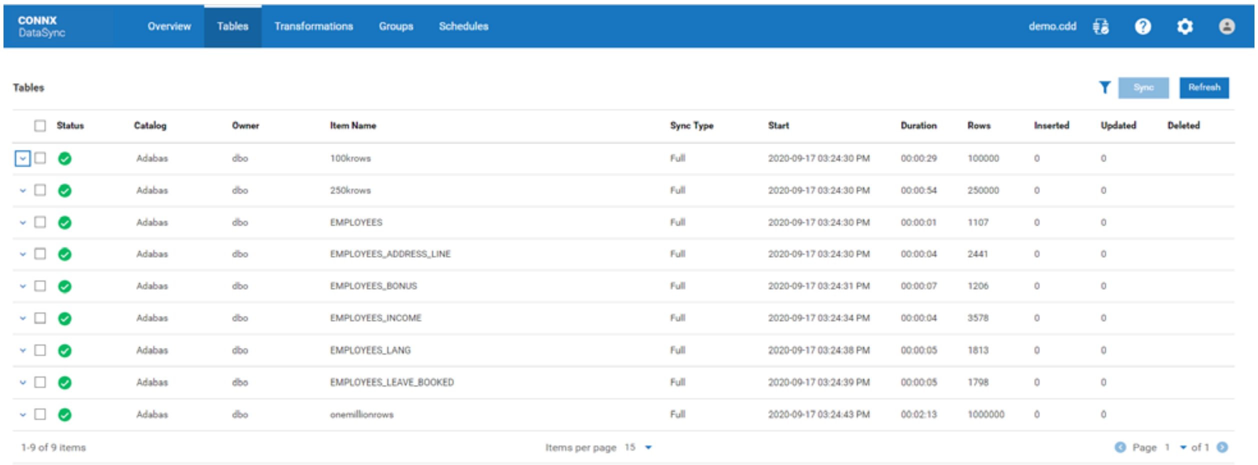
- **Status**
  The status column will contain either a green success icon or a red error icon depending on whether the last time the transform was run was successful or resulted in an error. If the status shows an error, clicking on the *more chevron* will expand the row and display the error text.
- **Item Name**
  This is the name of the table, transformation or group
- **Item Type**
  Possible types are Table, Transform and Group
- **Sync Type**
  Displays the type (Automatic, Full, Incremental, Incremental Only, Baseline or Savepoint) of synchronization
- **Start**
  Timestamp displaying the starting date and time
- **Duration**
  The duration in hours, minutes and seconds (hh:mm:ss)
- **Rows**
  Number of rows affected by the operation
- **Inserted**
  Number of rows inserted by the operation
- **Updated**
  Number of rows updated by the operation
- **Deleted**
  Number of rows deleted by the operation

Tables Tab

Tables Tab

The Tables Tab provides the ability to view the available source tables as well as see the status of the last run synchronizations.

Some functions on this screen use icons. The description of these icons will have a hyperlink to the Glossary of Icons which contains a picture of the icon and an explanation of its purpose.

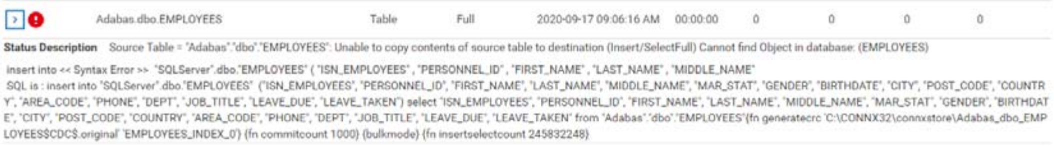


Tables

☐	Status	Catalog	Owner	Item Name	Sync Type	Start	Duration	Rows	Inserted	Updated	Deleted
☐	✓	Adabas	dbo	100krows	Full	2020-09-17 03:24:30 PM	00:00:29	100000	0	0	
☐	✓	Adabas	dbo	250krows	Full	2020-09-17 03:24:30 PM	00:00:54	250000	0	0	
☐	✓	Adabas	dbo	EMPLOYEES	Full	2020-09-17 03:24:30 PM	00:00:01	1107	0	0	
☐	✓	Adabas	dbo	EMPLOYEES_ADDRESS_LINE	Full	2020-09-17 03:24:30 PM	00:00:04	2441	0	0	
☐	✓	Adabas	dbo	EMPLOYEES_BONUS	Full	2020-09-17 03:24:31 PM	00:00:07	1206	0	0	
☐	✓	Adabas	dbo	EMPLOYEES_INCOME	Full	2020-09-17 03:24:34 PM	00:00:04	3578	0	0	
☐	✓	Adabas	dbo	EMPLOYEES_LANG	Full	2020-09-17 03:24:38 PM	00:00:05	1813	0	0	
☐	✓	Adabas	dbo	EMPLOYEES_LEAVE_BOOKED	Full	2020-09-17 03:24:39 PM	00:00:05	1798	0	0	
☐	✓	Adabas	dbo	onemillionrows	Full	2020-09-17 03:24:43 PM	00:02:13	1000000	0	0	

1-9 of 9 items   Items per page 15   Page 1 of 1

- **Tables List**
  The tables list displays information about all the tables available to be used as source tables. Tables in the database specified as the Default Target database will not be listed. Below is a description of each column in the table:
  - **More|Less Chevron**
    If more information is present for a row than can be displayed, clicking the *more chevron* will expand the row. Clicking the *less chevron* will collapse the information. Below is an example of a row that contains an error message. The *more chevron* has been clicked to display the additional information.



Adabas.dbo.EMPLOYEES	Table	Full	2020-09-17 09:06:16 AM	00:00:00	0	0	0	0

Status Description Source Table = "Adabas"."dbo"."EMPLOYEES": Unable to copy contents of source table to destination (Insert/SelectFull) Cannot find Object in database: (EMPLOYEES)

```
insert into << Syntax Error >> "SQLServer".dbo."EMPLOYEES" ( "ISN_EMPLOYEES" , "PERSONNEL_ID" , "FIRST_NAME" , "LAST_NAME" , "MIDDLE_NAME"
SQL is : insert into "SQLServer".dbo."EMPLOYEES" ("ISN_EMPLOYEES", "PERSONNEL_ID", "FIRST_NAME", "LAST_NAME", "MIDDLE_NAME", "MAR_STAT", "GENDER", "BIRTHDATE", "CITY", "POST_CODE", "COUNTR
Y", "AREA_CODE", "PHONE", "DEPT", "JOB_TITLE", "LEAVE_DUE", "LEAVE_TAKEN") select "ISN_EMPLOYEES", "PERSONNEL_ID", "FIRST_NAME", "LAST_NAME", "MIDDLE_NAME", "MAR_STAT", "GENDER", "BIRTHDAT
E", "CITY", "POST_CODE", "COUNTRY", "AREA_CODE", "PHONE", "DEPT", "JOB_TITLE", "LEAVE_DUE", "LEAVE_TAKEN" from "Adabas"."dbo"."EMPLOYEES"{fn generatecrc 'C:\CONNX32\connxstore\Adabas_dbo_EMP
LOYEES$CDC$.original' 'EMPLOYEES_INDEX_0'} {fn commitcount 1000} {bulkmode} {fn insertselectcount 245832248}
```

  - **Selection Checkbox**
    Click the checkbox next to each row you wish to do an operation against. The row selection checkbox applies to the Duplicate, Import, Export and Sync functions
  - **Status**
    The status column will contain either a green success icon or a red error icon depending on whether the last time the transform was run was successful or resulted in an error. If the table has never been synchronized, the column will be blank. If the status shows an error, clicking on the *more chevron* will expand the row and display the error text.
  - **Catalog**
    This is the name of the catalog the table is in.
  - **Owner**
    This is the owner name for the table.

  - **Item Name**
    This is the name of the table.
  - **Sync Type**
    Displays the type (Automatic, Full, Incremental or Incremental Only) of synchronization from the last time this table was synchronized.
  - **Start**
    Timestamp displaying the starting date and time from the most recent synchronization.
  - **Duration**
    The duration in hours, minutes and seconds (hh:mm:ss) from the most recent synchronization
  - **Rows**
    Number of rows affected by the operation
  - **Inserted**
    Number of rows inserted by the operation
  - **Updated**
    Number of rows updated by the operation
  - **Deleted**
    Number of rows deleted by the operation
- **Search Filter**
  Pressing this button activates search fields for the Status, Catalog, Owner and Item Name fields. To filter the results in the list of tables, enter a value in one or more of these search fields. For example, entering the word EMPLOYEES in the *Item Name* search box will cause only those tables that have the word EMPLOYEES in the *Item Name* to be displayed.
- **Sync** button
  To perform an **on demand sync**, select a table or tables by clicking the selection checkbox next to the table or tables you want to sync. Once your selection has been made, press the Sync button. This will start the synchronization process on the selected tables. The view will switch to the Overview Summary Page where you will be able to monitor the progress.

  To perform this synchronization on a regular basis, you can create a schedule. To create a schedule, go to the **Schedule** tab to establish the timing of the synchronization. See To schedule a synchronization task for more information.

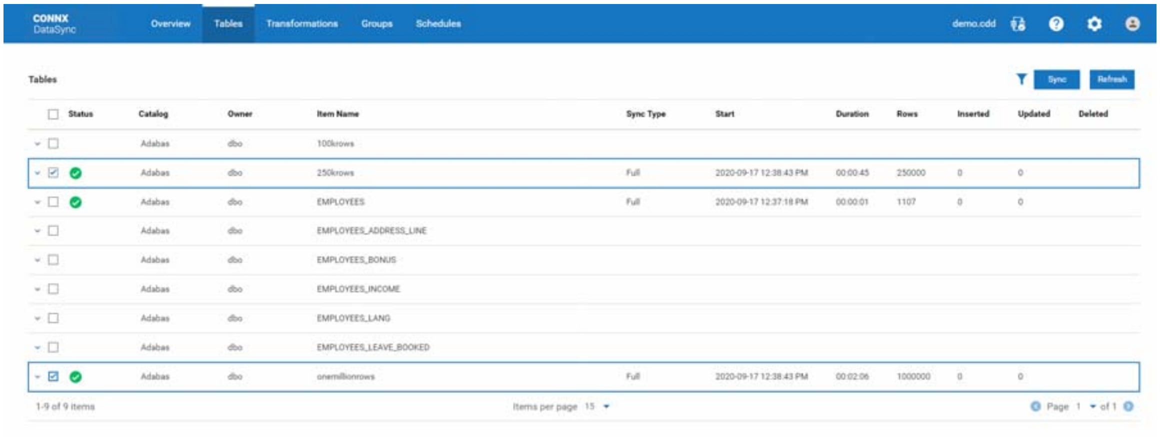
  For more information about On Demand Synchronization, see Synchronizing Tables to Target Destination.
- **Refresh** button
  The refresh button refreshes the list of tables.

Synchronizing Tables to Target Destination

Tables can be synchronized on demand or via the scheduler. This section describes how to synchronize a table on demand. For information on scheduling tables for synchronization, please refer to the section on scheduling a synchronization task.

To execute an on demand synchronization task for a table:

1. Select the **Tables** tab
2. Select a table or tables, and click the **Sync** button to start the synchronization process. The target database for Table Synchronizations is the database specified as the Default Target in the Sync Behavior page of Settings

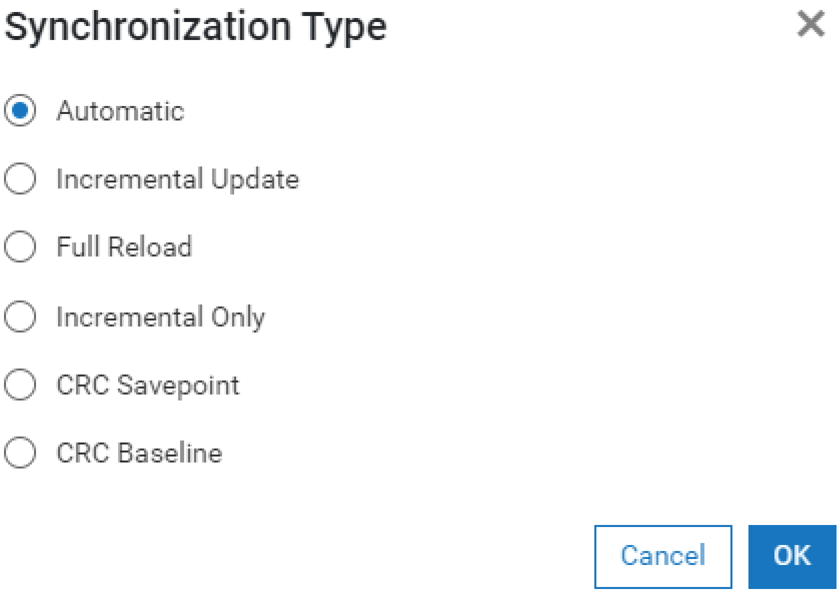


Tables

	□	Status	Catalog	Owner	Item Name	Sync Type	Start	Duration	Rows	Inserted	Updated	Deleted
⌄	□		Adabas	dbo	100krows							
⌄	☑	✓	Adabas	dbo	250krows	Full	2020-09-17 12:38:43 PM	00:00:45	250000	0	0	
⌄	□	✓	Adabas	dbo	EMPLOYEES	Full	2020-09-17 12:37:18 PM	00:00:01	1107	0	0	
⌄	□		Adabas	dbo	EMPLOYEES_ADDRESS_LINE							
⌄	□		Adabas	dbo	EMPLOYEES_BONUS							
⌄	□		Adabas	dbo	EMPLOYEES_INCOME							
⌄	□		Adabas	dbo	EMPLOYEES_LANG							
⌄	□		Adabas	dbo	EMPLOYEES_LEAVE_BOOKED							
⌄	☑	✓	Adabas	dbo	onemillionrows	Full	2020-09-17 12:38:43 PM	00:02:06	1000000	0	0	

1-9 of 9 items Items per page 15 Page 1 of 1

3. The **Synchronize Type** dialog box appears.



Synchronization Type ✕

● Automatic

○ Incremental Update

○ Full Reload

○ Incremental Only

○ CRC Savepoint

○ CRC Baseline

4. Select the synchronization type, **Automatic, Incremental Update, Full Reload,** or **Incremental Only**, depending on what is required. The recommended synchronization for the select database is displayed. Note that an **Incremental Update** is only available if a full reload has already been performed. Automatic will perform an Incremental Update unless the amount of changed records has exceeded the Full Reload Threshold, or a unique key is not available for the table. If Incremental Only has been selected, it will always incrementally update the table (the full reload threshold will be ignored). CRC SavePoint and CRC Baseline are not used with Table Synchronizations. For more information on the CRC types, see the section on Creating and Syncing Change Data Capture Transforms.
5. Click the **OK** button to start the synchronization process.
6. When the synchronization process starts, the application will switch to the Overview Summary Page and will display the current status and progress of all running synchronization jobs.

Transformations Tab

Before you start

Transformations provide support for ETL (Extract, Transform, Load) Projects. Datasync offers the ability to transform the source feeds and produce a target table on a selected target database. Some examples of the types of transformations supported are the ability to filter columns from the source table, add columns to the target table, join multiple source tables into a single target table, provide expressions to modify the data in a source column before sending it to the target table and more. Before a transformation is created, it is often a good idea to take advantage of CONNX Tools to help design the potential transformation. During the DataSync design process, the following CONNX Tools can offer valuable support:

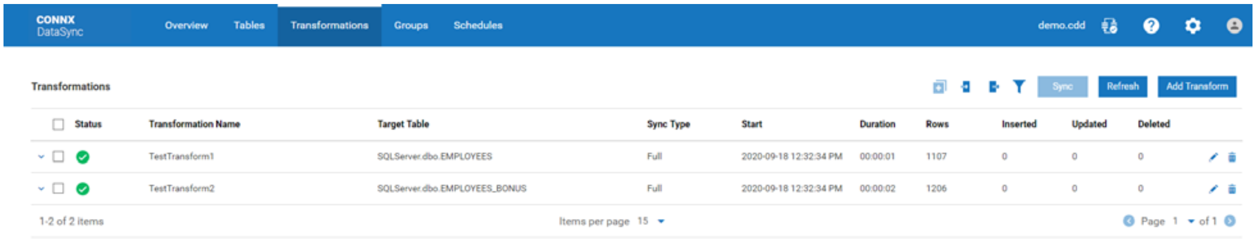
- CONNX Data Dictionary Tool.
- CONNX Infonaut Tool.

Both of these tools are available in the CONNX Driver section of the Start menu on your server

Transformations Tab

The Transformations Tab provides the ability to view, edit and manage transformations.

Some functions on this screen use icons. The description of these icons will have a hyperlink to the Glossary of Icons which contains a picture of the icon and an explanation of its purpose.

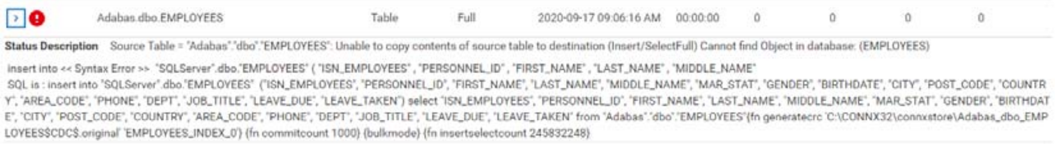


Transformations

	Status	Transformation Name	Target Table	Sync Type	Start	Duration	Rows	Inserted	Updated	Deleted
⌄ □	✓	TestTransform1	SQLServer.dbo.EMPLOYEES	Full	2020-09-18 12:32:34 PM	00:00:01	1107	0	0	0
⌄ □	✓	TestTransform2	SQLServer.dbo.EMPLOYEES_BONUS	Full	2020-09-18 12:32:34 PM	00:00:02	1206	0	0	0

1-2 of 2 items Items per page 15 Page 1 of 1

- **Transformations List**
  The transformations list displays summary information about all defined transformations. Below is a description of each column in the table:
  - **More|Less Chevron**
    If more information is present for a row than can be displayed, clicking the *more chevron* will expand the row. Clicking the *less chevron* will collapse the information. Below is an example of a row that contains an error message. The *more chevron* has been clicked to display the additional information.



›	!	Adabas.dbo.EMPLOYEES	Table	Full	2020-09-17 09:06:16 AM	00:00:00	0	0	0	0

Status Description Source Table = "Adabas"."dbo"."EMPLOYEES": Unable to copy contents of source table to destination (Insert/SelectFull) Cannot find Object in database: (EMPLOYEES)

```
insert into << Syntax Error >> "SQLServer".dbo."EMPLOYEES" ( "ISN_EMPLOYEES" , "PERSONNEL_ID" , "FIRST_NAME" , "LAST_NAME" , "MIDDLE_NAME"
SQL is : insert into "SQLServer".dbo."EMPLOYEES" ("ISN_EMPLOYEES", "PERSONNEL_ID", "FIRST_NAME", "LAST_NAME", "MIDDLE_NAME", "MAR_STAT", "GENDER", "BIRTHDATE", "CITY", "POST_CODE", "COUNTR
Y", "AREA_CODE", "PHONE", "DEPT", "JOB_TITLE", "LEAVE_DUE", "LEAVE_TAKEN") select "ISN_EMPLOYEES", "PERSONNEL_ID", "FIRST_NAME", "LAST_NAME", "MIDDLE_NAME", "MAR_STAT", "GENDER", "BIRTHDAT
E", "CITY", "POST_CODE", "COUNTRY", "AREA_CODE", "PHONE", "DEPT", "JOB_TITLE", "LEAVE_DUE", "LEAVE_TAKEN" from "Adabas"."dbo"."EMPLOYEES"{fn generatecrc 'C:\CONNX32\connxstore\Adabas_dbo_EMP
LOYEES$CDC$.original' 'EMPLOYEES_INDEX_0'} {fn commitcount 1000} {bulkmode} {fn insertselectcount 245832248}
```

  - **Selection Checkbox**
    Click the checkbox next to each row you wish to do an operation against. The row selection checkbox applies to the Duplicate, Import, Export and Sync functions
  - **Status**
    The status column will contain either a green success icon or a red error icon depending on whether the last time the transform was run was successful or resulted in an error. If the transform has never been synchronized, the column will be blank. If the status shows an error, clicking on the *more chevron* will expand the row and display the error text.
  - **Transformation Name**
    This is the name of the transformation that was specified when the transformation was created.
  - **Target Table**
    Displays the three part name consisting of:
    - Database
    - Owner
    - Table name
  - **Sync Type**
    Displays the type (Automatic, Full, Incremental, Incremental Only, Baseline or Savepoint) of synchronization from the last time this transformation was run.
  - **Start**
    Timestamp displaying the starting date and time of the most recent synchronization.

  - **Duration**
    The duration in hours, minutes and seconds (hh:mm:ss) of the most recent synchronization
  - **Rows**
    Number of rows affected by the operation
  - **Inserted**
    Number of rows inserted by the operation
  - **Updated**
    Number of rows updated by the operation
  - **Deleted**
    Number of rows deleted by the operation
  - **Edit**
    Clicking this button will take you to the stepper used when creating a transformation. When editing a transformation, all fields will be filled in with current values. Unlike the creation process where each step must be completed in order, when editing it is possible to jump to the specific step you wish to edit. To jump to a step, click on the step number that is displayed at the top of the stepper. Depending on which fields are edited, you may be required to complete all subsequent steps before saving changes. For example, if the source table is changed it will be necessary complete all subsequent steps. In this case, the stepper will not allow you to jump forward once this change as been made. Instead, you will need to use the Next button as you would when creating a new transformation.
  - **Delete**
    Clicking the Delete button will remove the transformation from the list of available transformations to sync. It will not delete the Destination Target Table (if it exists). If the transformation is part of a schedule, the schedule must be deleted before the transformation can be deleted.
- **Duplicate**
  To create a duplicate of a transformation, click the selection checkbox for the transformations to be duplicated and then click the *duplicate* button. This feature is useful in circumstances where you have a transformation with complex SQL as the source and you want to create a second transformation that does the same thing but writes the data to a different database. For each transformation selected, a copy will be created with _Copy_x appended to the name where x represents a unique number for the copy. For example, if the transform *TestTransform1* is duplicated, the new transform name will be *TestTransform1_Copy_1*. The duplicate transform will be created with a target table that has _Copy_x appended to the name. You may need to edit the new transform and adjust the target table name to the suit your needs.
- **Import** and **Export**
  Transformations can be moved from one environment (i.e. development), to another environment (i.e. production) with Export and Import. One or multiple transformations can be exported/imported to/from a single file. In the first environment, check the desired transformations to move and press Export, then specify a file name and that file will be downloaded to the machine logged into the DataSync Administrator. Then a connection can be made to the second environment, when Import is pressed, specify the file that was downloaded and those transformations will be put into DataSync.
- **Search Filter**
  Pressing this button activates search fields for the Status, Transformation Name and Target Table fields. To filter the results in the list of transformations, enter a value in one or more of these search fields. For example, entering the word EMPLOYEES in the *Transformation Name* search box will cause only those transformations that have the word EMPLOYEES in the *Transformation Name* to be displayed.
- **Sync** button
  To perform an **on demand sync**, select the transformation(s) by clicking the selection checkbox

next to the transformation(s) you want to sync. Once your selection has been made, press the Sync button. This will start the synchronization process on the selected transforms. The view will switch to the Overview Summary Page where you will be able to monitor the progress.

To perform this synchronization on a regular basis, you can create a schedule. To create a schedule, go to the **Schedule** tab to establish the timing of the synchronization. See To schedule a synchronization task for more information.

For more information about On Demand Synchronization, see Synchronizing Transformations to Target Destination.

- **Refresh** button
  The refresh button refreshes the list of transforms.
- **Add Transformation button**
  Adding a Transform will add a new Transform to the list of existing Transforms that can be synchronized. This button will take you to the Transformation Creating Stepper where you will be guided through each step of creating a transform. The steps are:
  -
  - Select a Transform Name
  - Choose a Transform Source
  - Choose a Destination Target
  - Choose a Sync Type
  - Set Column Mapping
  - Check Indexes

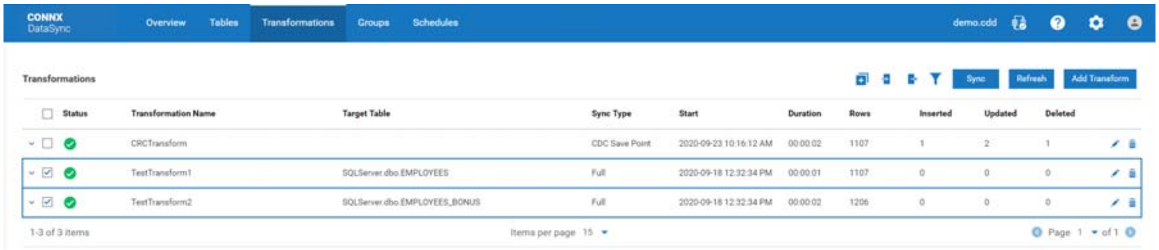
For more information on creating transformations, see Creating a Transformation in the "Designing a Transformation". section

Synchronizing Transformations to Target Destination

Transformations can be synchronized on demand or via the scheduler. This section describes how to synchronize a table on demand. For information on scheduling transformations for synchronization, please refer to the section on scheduling a synchronization task.

To execute an on demand synchronization task for a transformation:

1. Select the **Transformations** tab
2. Select a transformation or transformations, and click the **Sync** button to start the synchronization process.

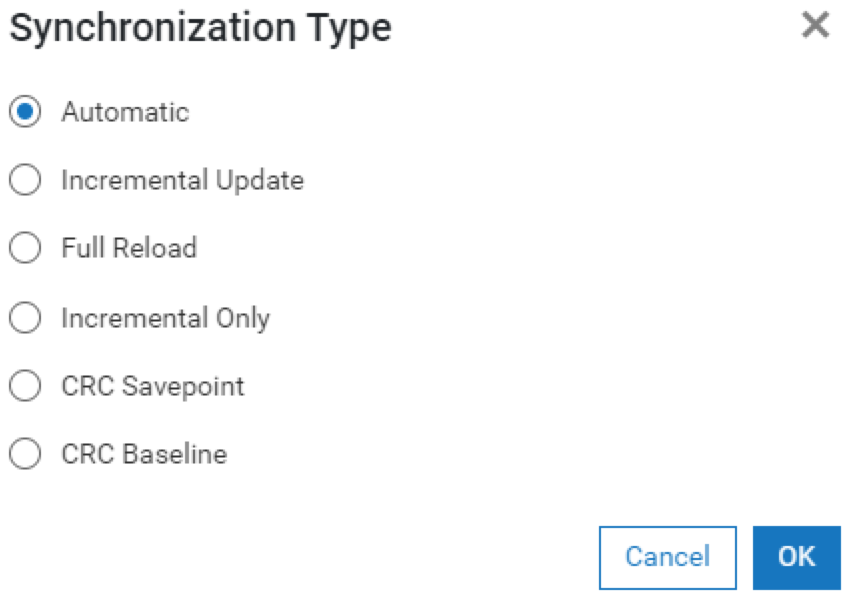


Transformations

	Status	Transformation Name	Target Table	Sync Type	Start	Duration	Rows	Inserted	Updated	Deleted
□	✓	CRCTransform		CDC Save Point	2020-09-23 10:16:12 AM	00:00:02	1107	1	2	1
☑	✓	TestTransform1	SQLServer.dbo.EMPLOYEES	Full	2020-09-18 12:32:34 PM	00:00:01	1107	0	0	0
☑	✓	TestTransform2	SQLServer.dbo.EMPLOYEES_BONUS	Full	2020-09-18 12:32:34 PM	00:00:02	1206	0	0	0

1-3 of 3 items Items per page 15 Page 1 of 1

3. The **Synchronize Type** dialog box appears.



● Automatic

○ Incremental Update

○ Full Reload

○ Incremental Only

○ CRC Savepoint

○ CRC Baseline

4. Select the synchronization type, **Automatic, Incremental Update, Full Reload,** or **Incremental Only**, depending on what is required. The recommended synchronization for the select database is displayed. Note that an **Incremental Update** is only available if a full reload has already been performed. Automatic will perform an Incremental Update unless the amount of changed records has exceeded the Full Reload Threshold, or a unique key is not available for the table. If Incremental Only has been selected, it will always incrementally update the table (the full reload threshold will be ignored). CRC SavePoint and CRC Baseline are used with Change Data

Capture Transformations. For more information on the CRC types, see the section on Creating and Syncing Change Data Capture Transforms.

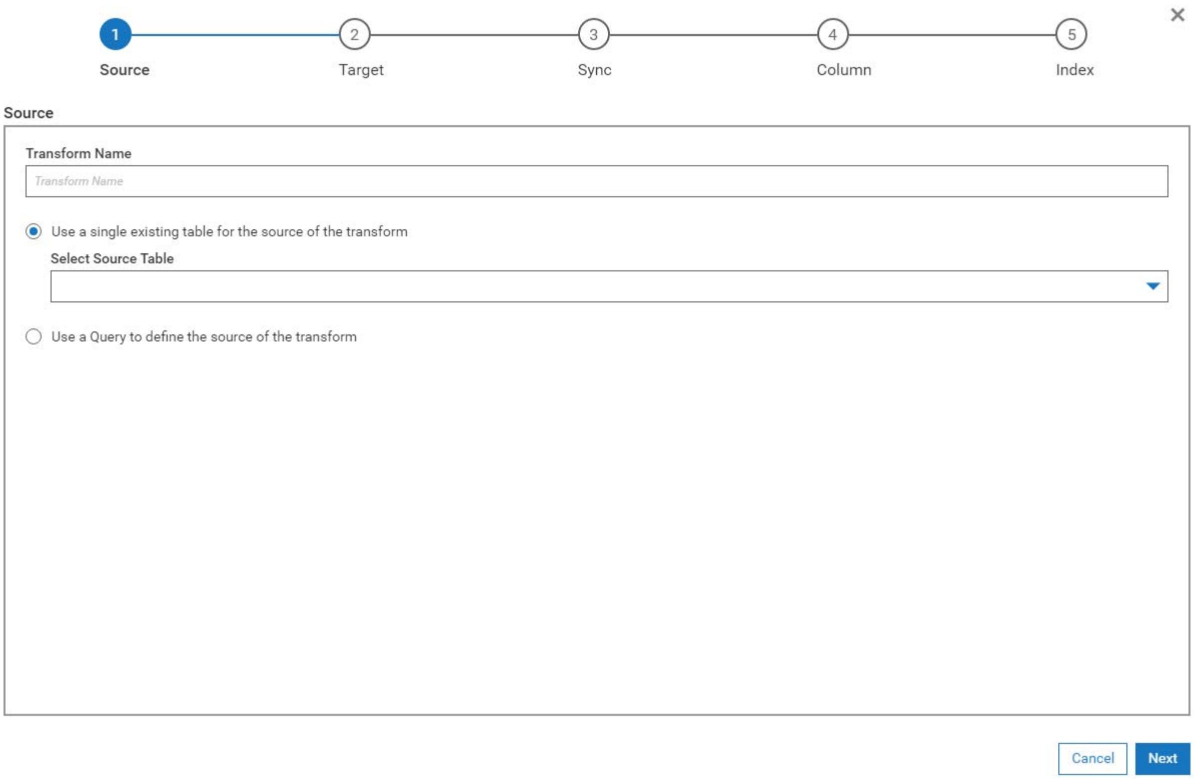
5. Click the **OK** button to start the synchronization process.
6. When the synchronization process starts, the application will switch to the Overview Summary Page and will display the current status and progress of all running synchronization jobs.

Designing a Transform

Source step

Creating a Transformation

Adding a Transform is accomplished by selecting the "Add Transform" button from the Main Transforms Page. A new window will be created that invokes the Transformation Stepper process.



Source

Transform Name

Transform Name

● Use a single existing table for the source of the transform

Select Source Table

○ Use a Query to define the source of the transform

The first part of the 5 step process is to define the Source. The Source step is responsible for the following:

- **Transform Name**
  Enter a descriptive name for the Transform of up to 32 characters.
- **Transform Source option**
  Select from Single Table Transform or Free-Form SQL

Selecting Transform Name

The Transform Name represents the name that will be applied to the Transform. The name should describe something that relates to the transform.

Choosing the source of the Transformation

The source can be defined with a single table selection or with a SQL Statement. The following table offers advantages of each source to help decide which source is appropriate to move the correct data.

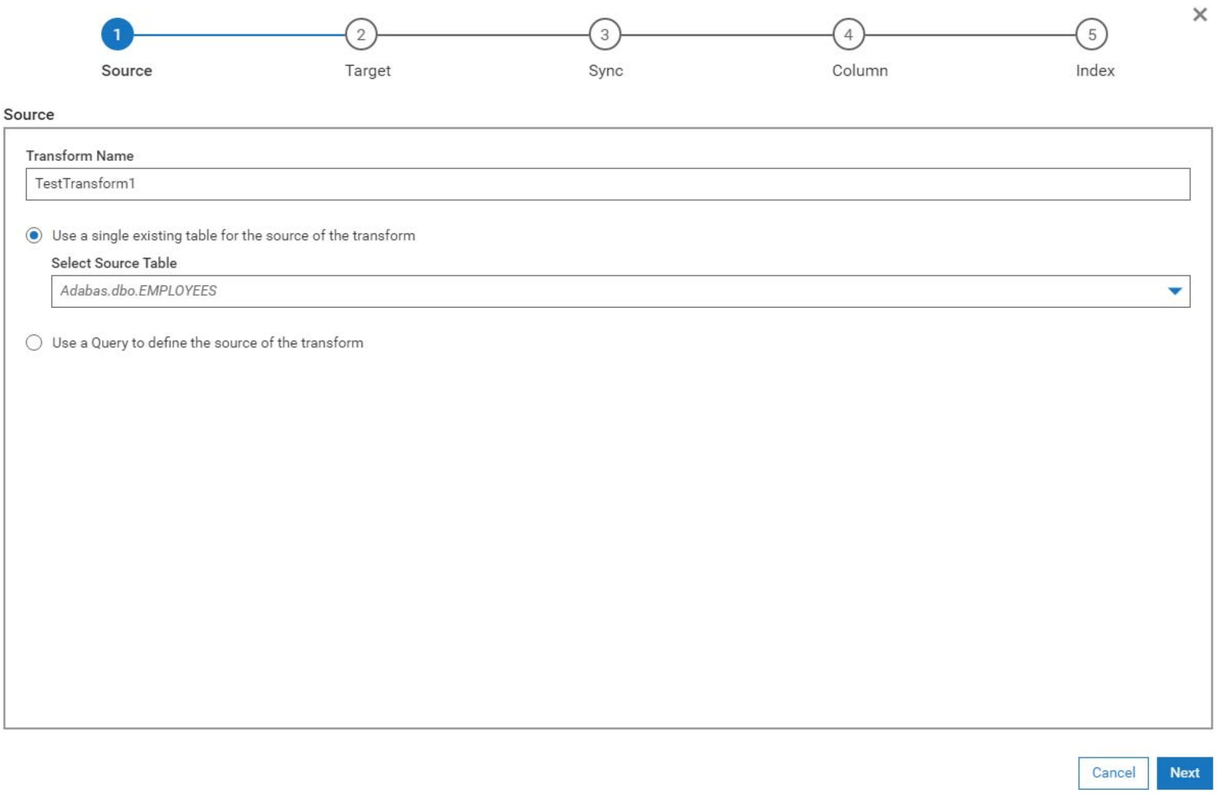
Transform Type	Simple Source Table	Free From SQL

	Transform	
Selecting Subset of Table or Tables	x	x
Translating coded values (e.g. Changing the Source column from a 1 to Male)	x	x
Calculated Values (e.g. OrderAmt * 1000)	x	x
Filtering		x
Sorting		x
Joining Multiple Tables (e.g. Lookups)		x
Aggregation		x
Unions		x

Designing a Single Table Transform

To create a Single Table transform, the radio button must be selected that says, "Use single existing table for the source of the transform". A drop down list box will be displayed under the Select Source Table caption. This list will contain all of the potential tables that included in the CONNX Data Dictionary File. The user must select a single table which will serve as the input for the target destination file.

One potential reason to choose this type of transformation might be that you only want to include certain fields of an existing table. It can also be used to map derived columns (e.g. orderamt * 100). When choosing this type of transformation, SQL is not required unless a derived field is created and requires SQL to create a transformation from source column(s).



Source

Transform Name: TestTransform1

● Use a single existing table for the source of the transform

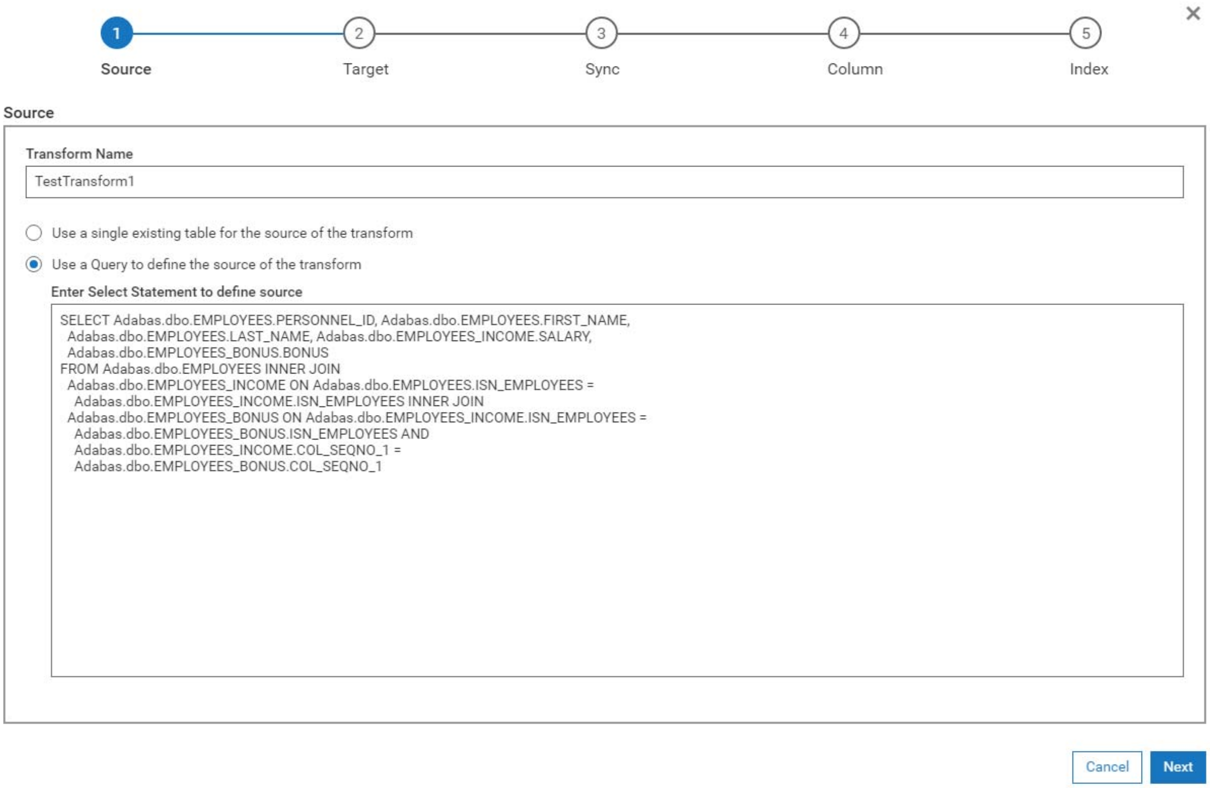
Select Source Table: Adabas.dbo.EMPLOYEES

○ Use a Query to define the source of the transform

Click the next button to continue to Step 2 Target

Designing a Free-Form SQL Transformation

If use has a specific SQL query as the Transformation source, select ***Use a Query to define the source of the transform*** and then either type or cut/paste from Infonaut the query that is to become the source of the transformation. If there is a syntax error, it will be caught when **Next** is selected.



Transform Name: TestTransform1

○ Use a single existing table for the source of the transform

● Use a Query to define the source of the transform

Enter Select Statement to define source

```
SELECT Adabas.dbo.EMPLOYEES.PERSONNEL_ID, Adabas.dbo.EMPLOYEES.FIRST_NAME,
 Adabas.dbo.EMPLOYEES.LAST_NAME, Adabas.dbo.EMPLOYEES_INCOME.SALARY,
 Adabas.dbo.EMPLOYEES_BONUS.BONUS
FROM Adabas.dbo.EMPLOYEES INNER JOIN
 Adabas.dbo.EMPLOYEES_INCOME ON Adabas.dbo.EMPLOYEES.ISN_EMPLOYEES =
  Adabas.dbo.EMPLOYEES_INCOME.ISN_EMPLOYEES INNER JOIN
 Adabas.dbo.EMPLOYEES_BONUS ON Adabas.dbo.EMPLOYEES_INCOME.ISN_EMPLOYEES =
  Adabas.dbo.EMPLOYEES_BONUS.ISN_EMPLOYEES AND
  Adabas.dbo.EMPLOYEES_INCOME.COL_SEQNO_1 =
  Adabas.dbo.EMPLOYEES_BONUS.COL_SEQNO_1
```

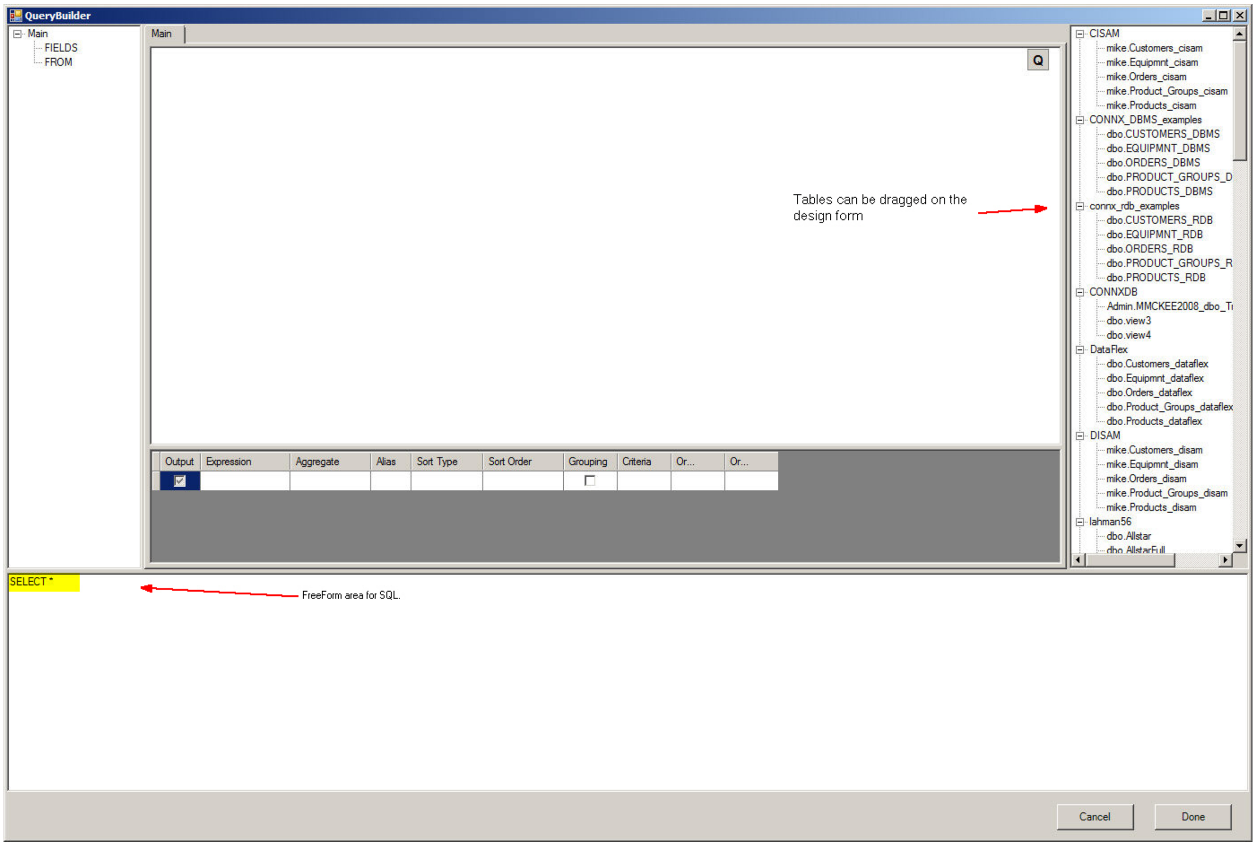
Click the next button to continue to Step 2 Target

Designing a Transform in Query Builder

Certain Transformation are easier to be created when a graphical tool can enable the necessary tables/views/columns to be seen. InfoNaut is query tool that ships with CONNX and has a graphical query tool that enables transformations to be easily created.

To use the Query Builder, Go to the CONNX Driver section of the Windows Start menu and select InfoNaut. Establish a DSN-less connection to the CDD that was uploaded to DataSync and select Build Query from the tool bar.

The QueryBuilder screen appears as follows:

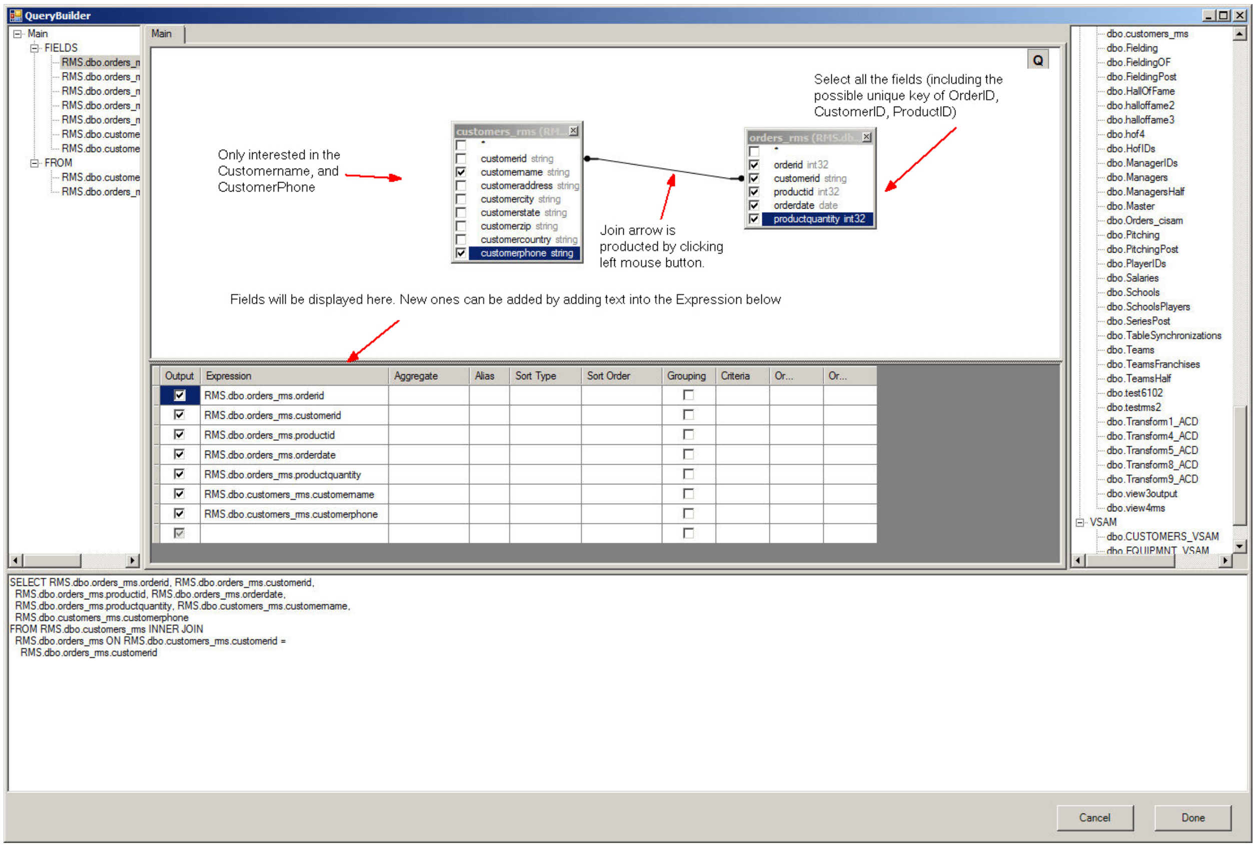


Tables can be dragged on the design form

```
SELECT *
```

FreeForm area for SQL.

In this example, the two tables (CUSTOMERS_RMS and ORDERS_RMS) will be used to generate a transform that produces a combined table that produces output only if the order occurred in 1997.



Only interested in the Customername, and CustomerPhone

Select all the fields (including the possible unique key of OrderID, CustomerID, ProductID)

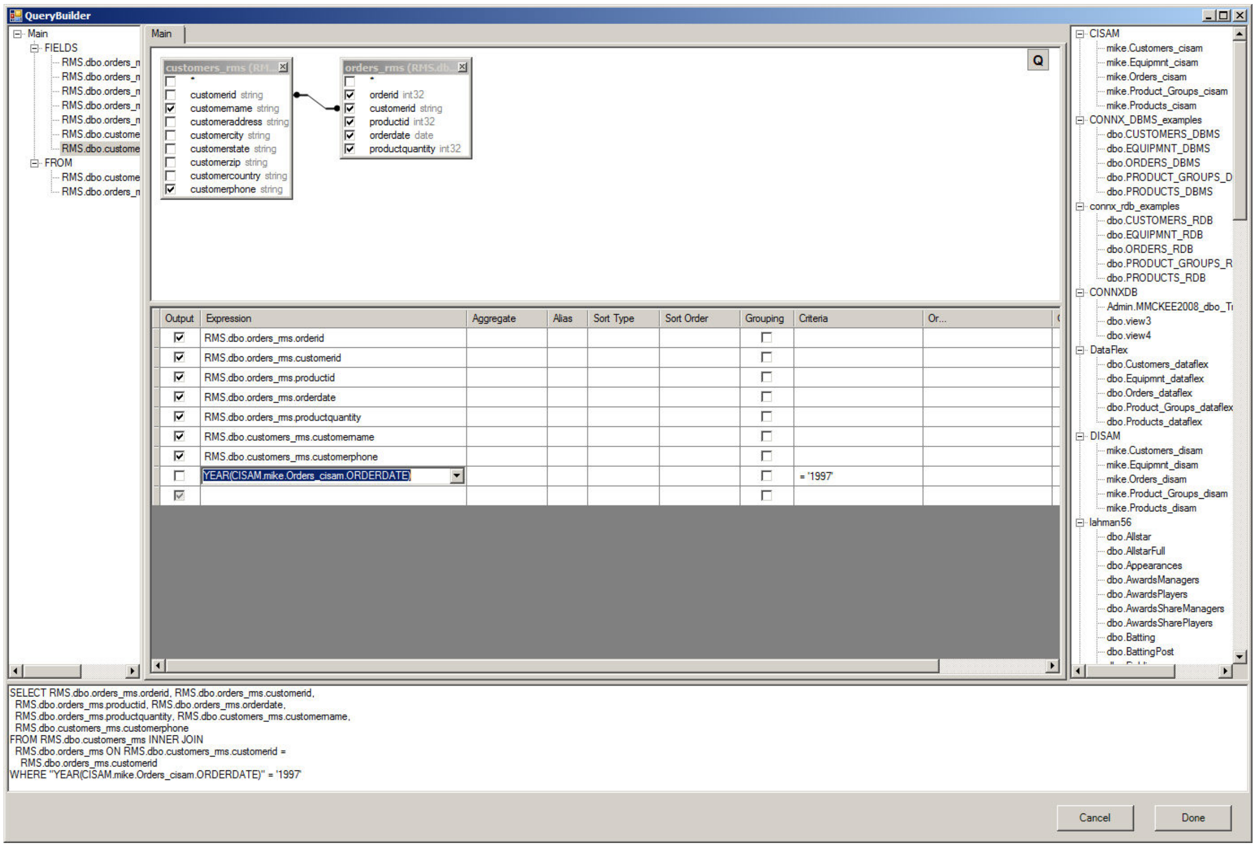
Join arrow is producted by clicking left mouse button.

Fields will be displayed here. New ones can be added by adding text into the Expression below

Output	Expression	Aggregate	Alias	Sort Type	Sort Order	Grouping	Criteria	Or...	Or...
☑	RMS.dbo.orders_rms.orderid					□			
☑	RMS.dbo.orders_rms.customerid					□			
☑	RMS.dbo.orders_rms.productid					□			
☑	RMS.dbo.orders_rms.orderdate					□			
☑	RMS.dbo.orders_rms.productquantity					□			
☑	RMS.dbo.customers_rms.customername					□			
☑	RMS.dbo.customers_rms.customerphone					□			
☑						□			

```
SELECT RMS.dbo.orders_rms.orderid, RMS.dbo.orders_rms.customerid,
 RMS.dbo.orders_rms.productid, RMS.dbo.orders_rms.orderdate,
 RMS.dbo.orders_rms.productquantity, RMS.dbo.customers_rms.customername,
 RMS.dbo.customers_rms.customerphone
FROM RMS.dbo.customers_rms INNER JOIN
 RMS.dbo.orders_rms ON RMS.dbo.customers_rms.customerid =
  RMS.dbo.orders_rms.customerid
```

The SQL Generated does not include the condition where the year must be 1997. The SQL to generate this is YEAR(CISAM.mike.Orders_cisam.ORDERDATE) = '1997' . This can be added in the grid below.



Output	Expression	Aggregate	Alias	Sort Type	Sort Order	Grouping	Criteria	Or...
☑	RMS.dbo.orders_rms.orderid					□		
☑	RMS.dbo.orders_rms.customerid					□		
☑	RMS.dbo.orders_rms.productid					□		
☑	RMS.dbo.orders_rms.orderdate					□		
☑	RMS.dbo.orders_rms.productquantity					□		
☑	RMS.dbo.customers_rms.customername					□		
☑	RMS.dbo.customers_rms.customerphone					□		
□	YEAR(CISAM.mike.Orders_cisam.ORDERDATE)					□	= '1997'	
☑						□		

```
SELECT RMS.dbo.orders_rms.orderid, RMS.dbo.orders_rms.customerid,
  RMS.dbo.orders_rms.productid, RMS.dbo.orders_rms.orderdate,
  RMS.dbo.orders_rms.productquantity, RMS.dbo.customers_rms.customername,
  RMS.dbo.customers_rms.customerphone
FROM RMS.dbo.customers_rms INNER JOIN
  RMS.dbo.orders_rms ON RMS.dbo.customers_rms.customerid =
    RMS.dbo.orders_rms.customerid
WHERE "YEAR(CISAM.mike.Orders_cisam.ORDERDATE)" = '1997'
```

More complex expressions can be created using the Expression Builder as described in the next section.

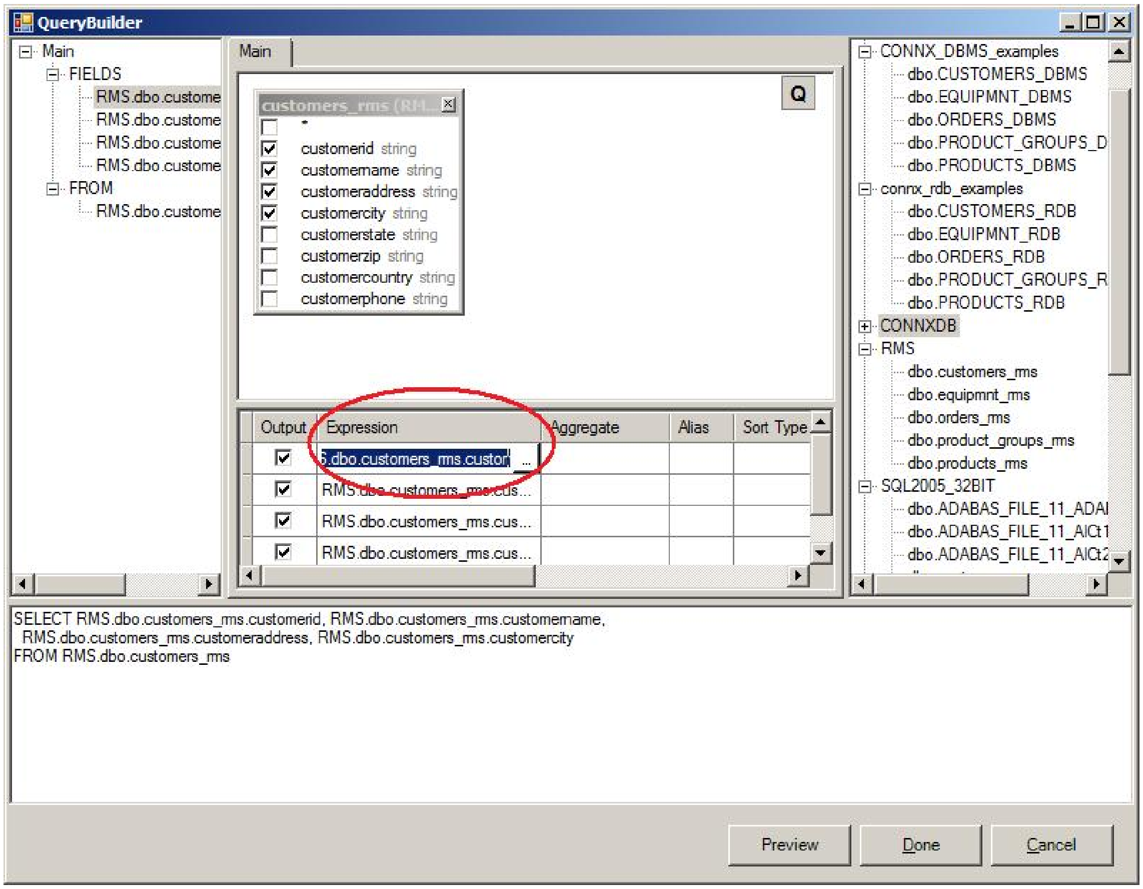
Click Done and the SQL Query will be copied to the Free-Form area of the Transformation.

After the text has been copied, click the next button to continue to Step 2 Target

Using the Expression Builder

How to enter:

The Expression Builder can be opened from the **Query Builder** in the **InfoNaut** tool which can be found in the Windows Start | CONNX Driver menu. To start the Expression Builder, connect to the CDD being used for DataSync with InfoNaut and click the **Query Builder** button. Right click on the **Expression field** and select **Expression Builder** from the menu. Another way to enter the Expression Builder is to select the box with the **"..."** at the right side of the **expression cell** in the **query builder**. For more information on starting the Query Builder, see the section on Designing a Transformation in Query Builder.



```
SELECT RMS.dbo.customers_rms.customerid, RMS.dbo.customers_rms.customername,
  RMS.dbo.customers_rms.customeraddress, RMS.dbo.customers_rms.customercity
FROM RMS.dbo.customers_rms
```

Why to use:

The Expression builder is designed to help the user create complex expressions easily. It lists functions supported by **CONNX**, gives information about the functions and the arguments it contains. This can be a helpful reminder of standard SQL functions and a good way to learn about **CONNX** specific functions.

The Layout:

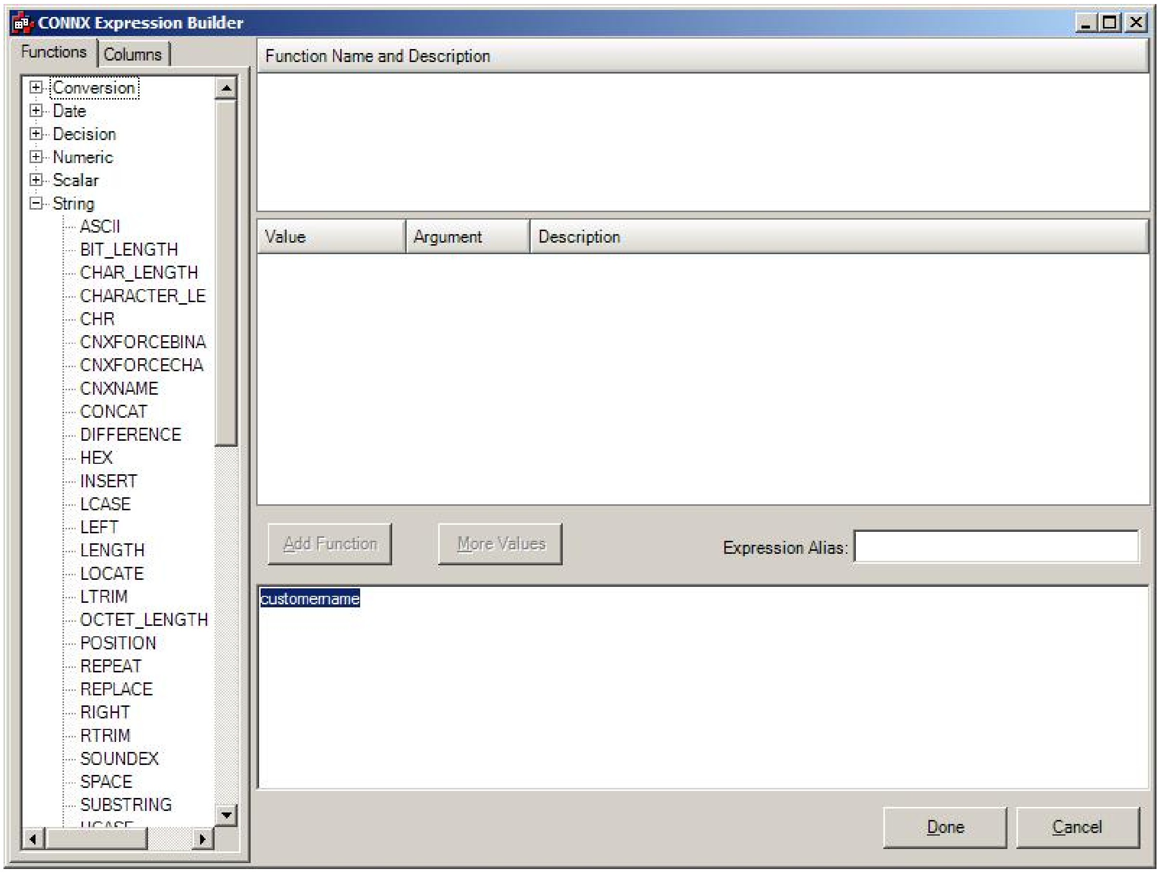
The **Expression builder** is composed of the following parts: **Functions available** list, **Columns available** list, **Function Name and Description**, **Function Argument Description** table, **Expression Alias** input field, and **Expression SQL** field. All functions supported by CONNX are listed in groups in the left panel, the groups can be expanded or collapsed to see all functions available. When a function is selected the name and the description of the function are shown in the table on the top right. Also after a function is selected the list of function arguments and their descriptions can be found in the **Argument**

Description table. The **Expression field** at the lower right can either be edited directly or it will show the built Expression. The **Expression Alias** entry field is for creating a simple name for the expression that can be used for the Target column name when mapping.

Example:

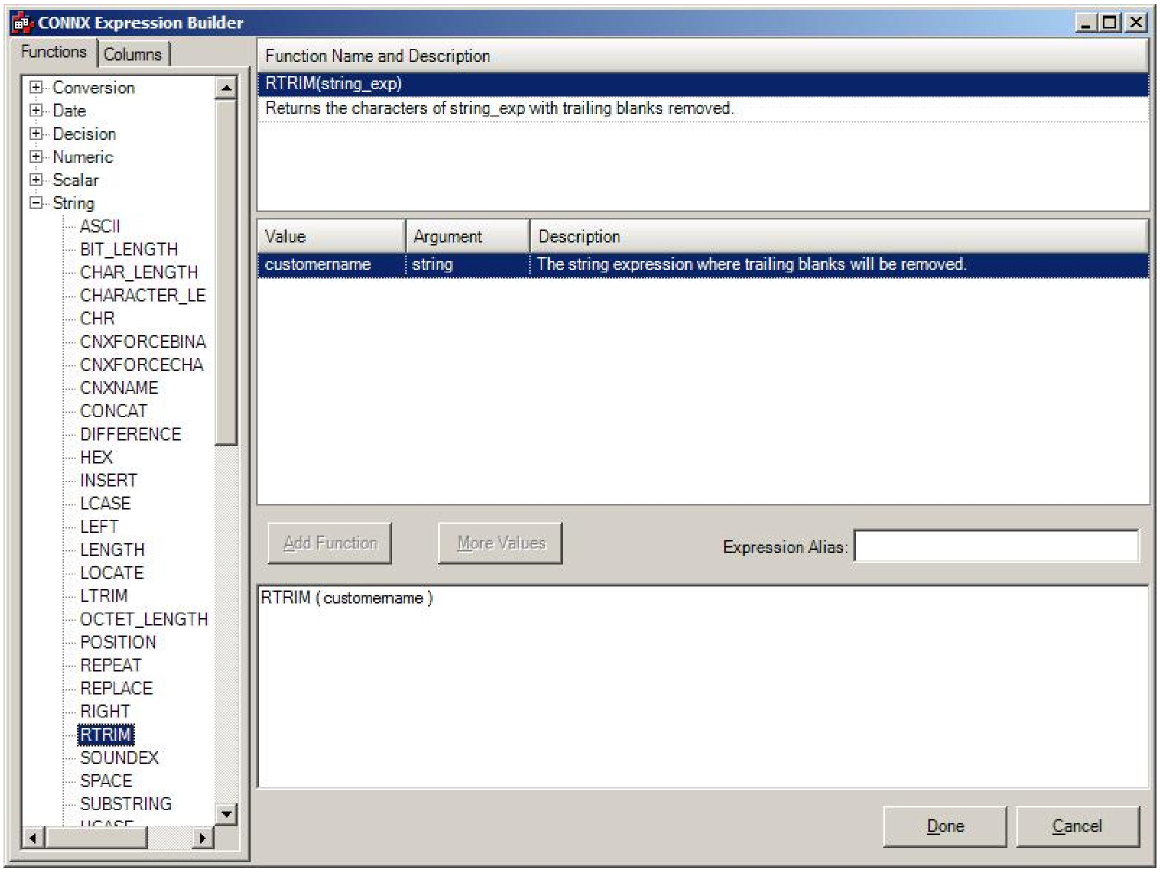
The following example goes step-by-step through an example of building a transform

1. Upon entry to the **Expression Builder** the contents of the column selected will appear in the **Expression SQL field** and be highlighted.



customername

2. Browse through the **function list** expanding or collapsing groups to find the appropriate function. Select functions that may be of interest to see their description to aid selection. Once the appropriate function is found double click or select the **Add Function** button and it will appear in the **Expression SQL** field.



Function Name and Description

RTRIM(string_exp)

Returns the characters of string_exp with trailing blanks removed.

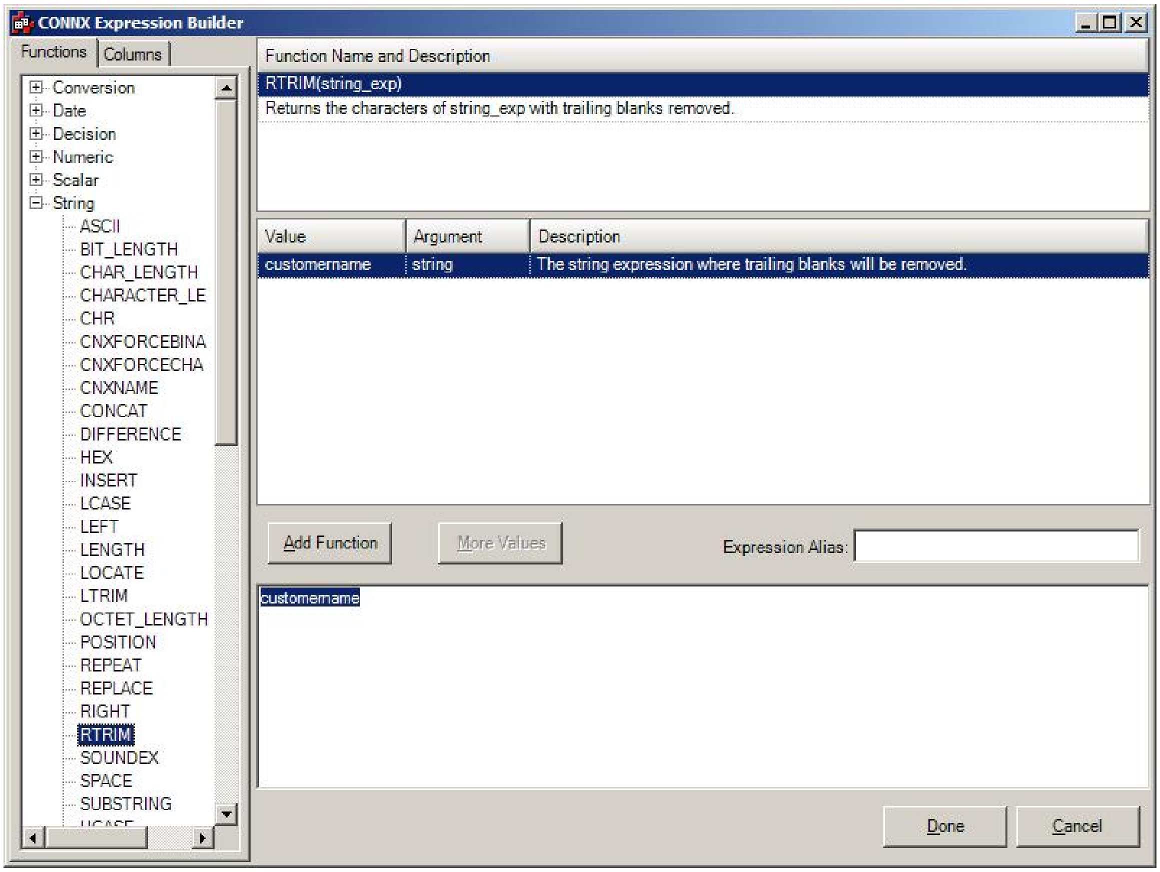
Value	Argument	Description
customername	string	The string expression where trailing blanks will be removed.

Expression Alias:

RTRIM (customername)

3. When the function appears in the **Expression SQL** field the highlighted text will then appear as the first argument of the expression. Other expression arguments can be filled out by either direct editing in the **Expression SQL** field, or by editing the value cell in the **Expression Arguments** table.

4. The **Expression Arguments** table can be used to enter the values for the expression. The first field can be edited and used to put values into the function. The second column in the table tells the datatype of argument the function is expecting. The third column has a description of what the purpose of the argument is in the function. In the for each argument in the table edit the Value field and enter the appropriate column name, characters or numbers for the function argument. Note, as values are entered here they also appear in the **Expression SQL** field.



Function Name and Description

RTRIM(string_exp)

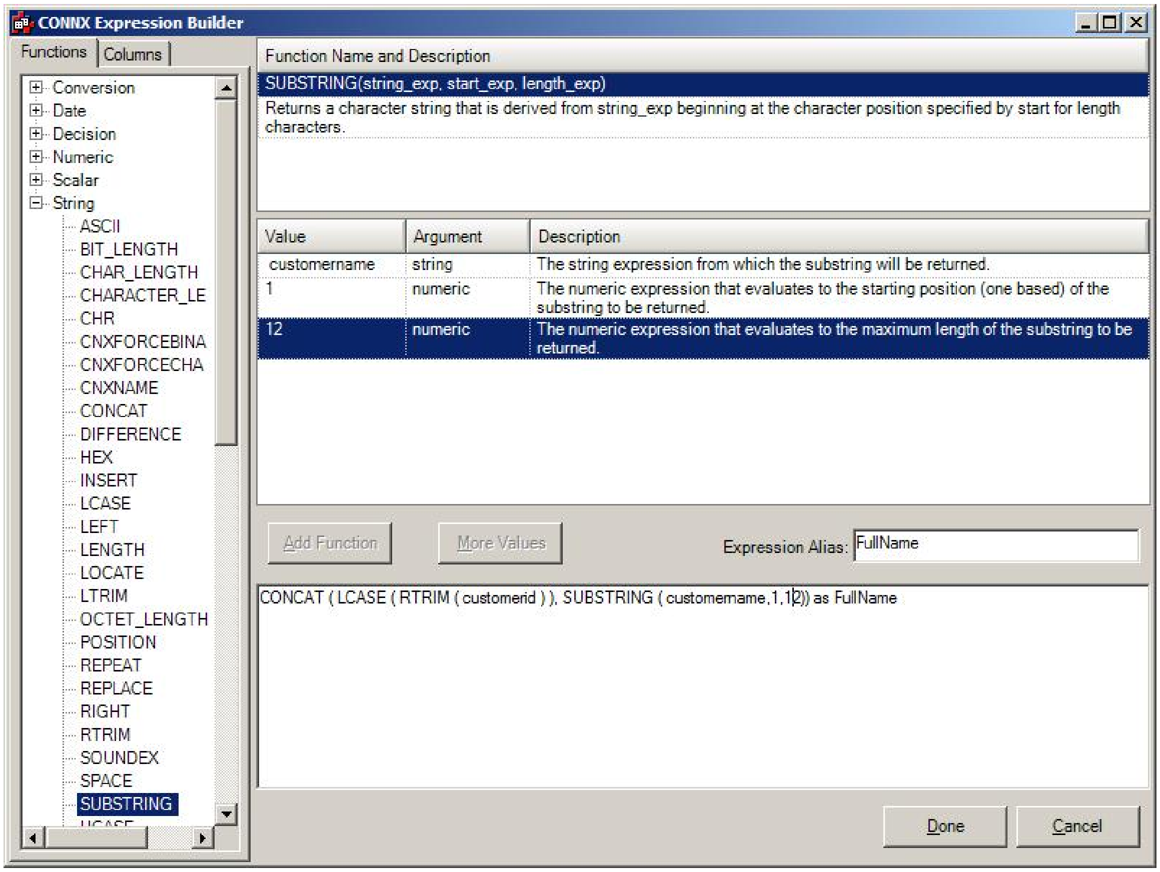
Returns the characters of string_exp with trailing blanks removed.

Value	Argument	Description
customername	string	The string expression where trailing blanks will be removed.

Expression Alias:

```
customername
```

5. Another way to enter the values for the function arguments is to edit them directly in the **Expression SQL** field. Any changes here will be reflected in the **Expression Arguments** table**.**
6. Add another function to the expression by placing the cursor to the desired location of the new function in the **Expression SQL** field and double clicking the function in the list. Dragging the cursor to make a selection in the **Expression SQL** field before double clicking the function will cause the selected text to appear in the first argument of the function.



Function Name and Description

SUBSTRING(string_exp, start_exp, length_exp)

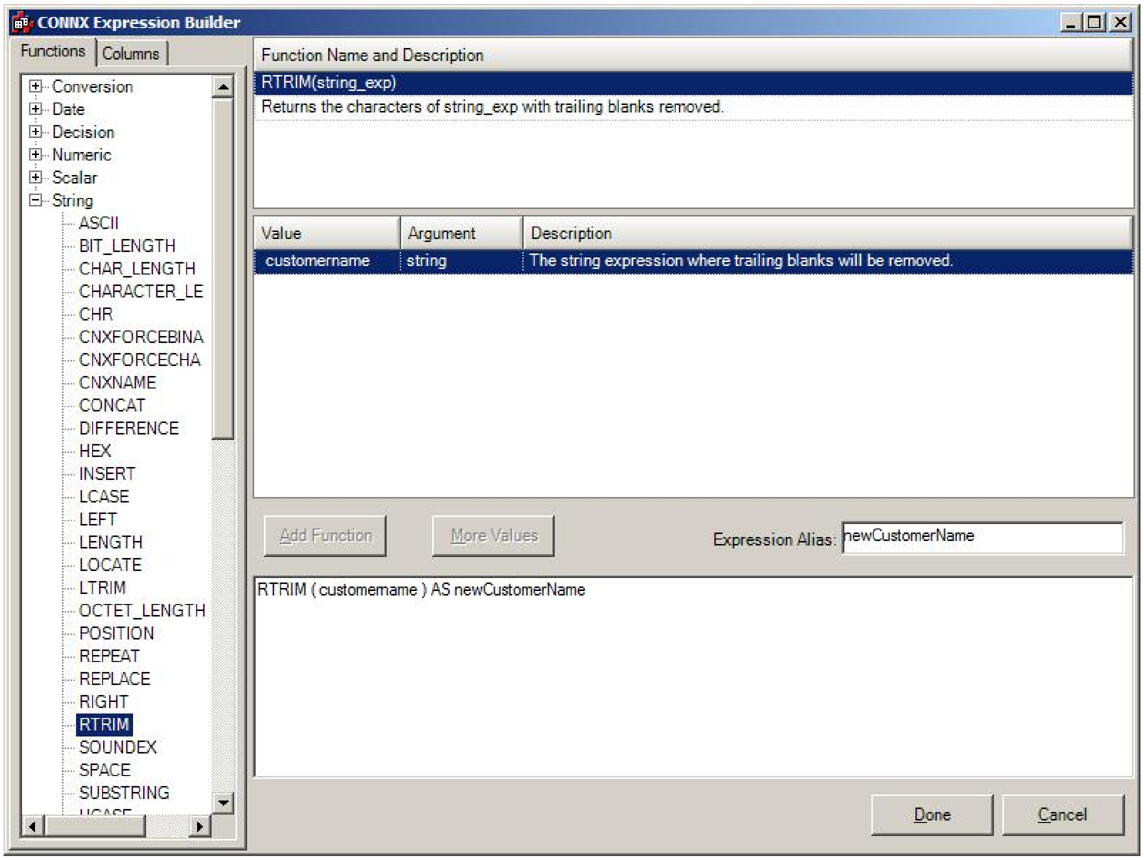
Returns a character string that is derived from string_exp beginning at the character position specified by start for length characters.

Value	Argument	Description
customername	string	The string expression from which the substring will be returned.
1	numeric	The numeric expression that evaluates to the starting position (one based) of the substring to be returned.
12	numeric	The numeric expression that evaluates to the maximum length of the substring to be returned.

Expression Alias: FullName

```
CONCAT ( LCASE ( RTRIM ( customerid ) ), SUBSTRING ( customername,1,12)) as FullName
```

7. Add an alias to be used for the corresponding target table column name. In the Expression alias field type a simple and descriptive name for the expression. It will automatically be added to the end of the expression.



Function Name and Description
RTRIM(string_exp)
Returns the characters of string_exp with trailing blanks removed.

Value	Argument	Description
customername	string	The string expression where trailing blanks will be removed.

Expression Alias: newCustomerName

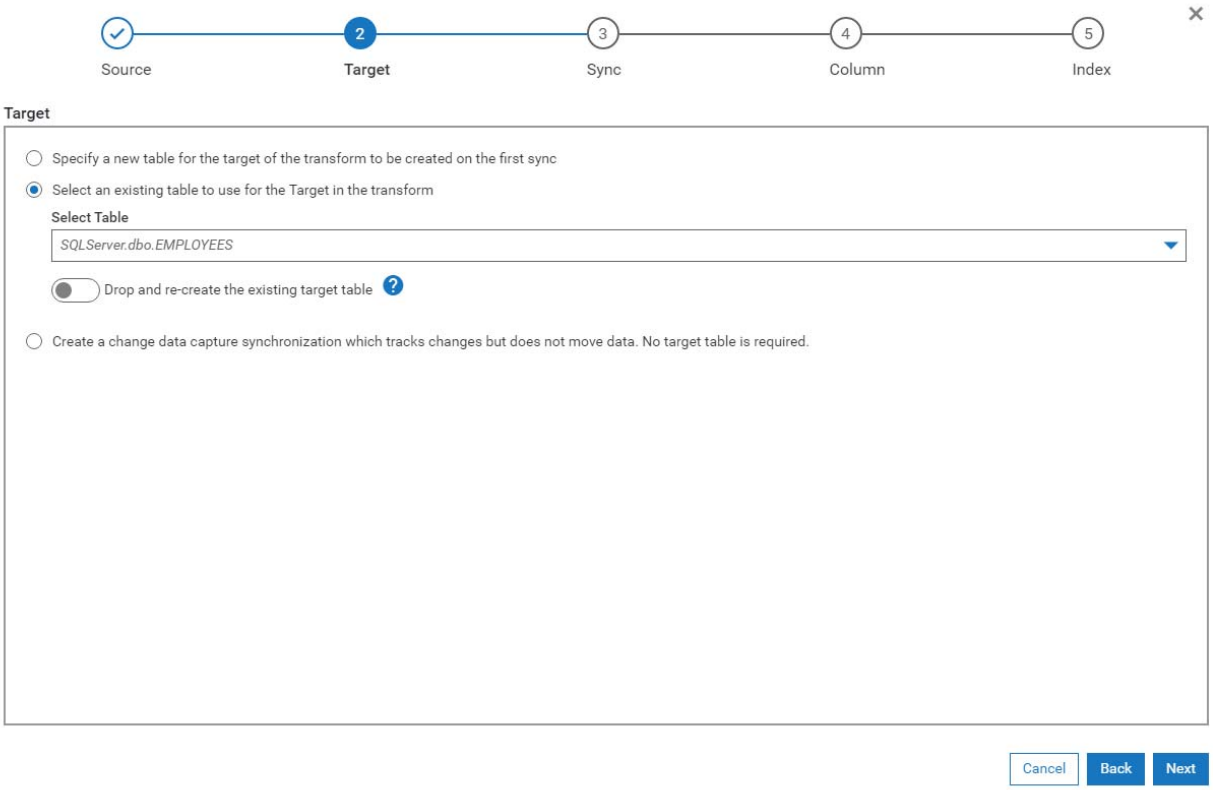
```
RTRIM ( customername ) AS newCustomerName
```

Target step

Selecting a Target Destination

Selecting a Target Destination involves selecting a target file for an appropriate database. The designer has a few choices to make in this section.

1. Create a new table as the destination for the data?
   If so, select the first radio button "Specify a new table for the target...", select the database and owner, then specify the new table name.
2. Use an existing table as the destination?
   If so, select the second radio button "Select an existing table to user for the Target..." and then use the drop down to choose the table.
3. If an existing table is selected, does the table structure need to change for the transform?
   If so, the target table can be dropped and recreated with the new structure by selecting "Drop and re-create the existing target table"
4. Track only changes to the source (For use with Microsoft SQL Server Integration Services)?
   In this case a target table is not necessary, select the third radio button "Create a change data capture synchronization..." and only changes in the source will be tracked. See the section on Change Data Capture transforms for more information.



Target

○ Specify a new table for the target of the transform to be created on the first sync

● Select an existing table to use for the Target in the transform

Select Table: SQLServer.dbo.EMPLOYEES

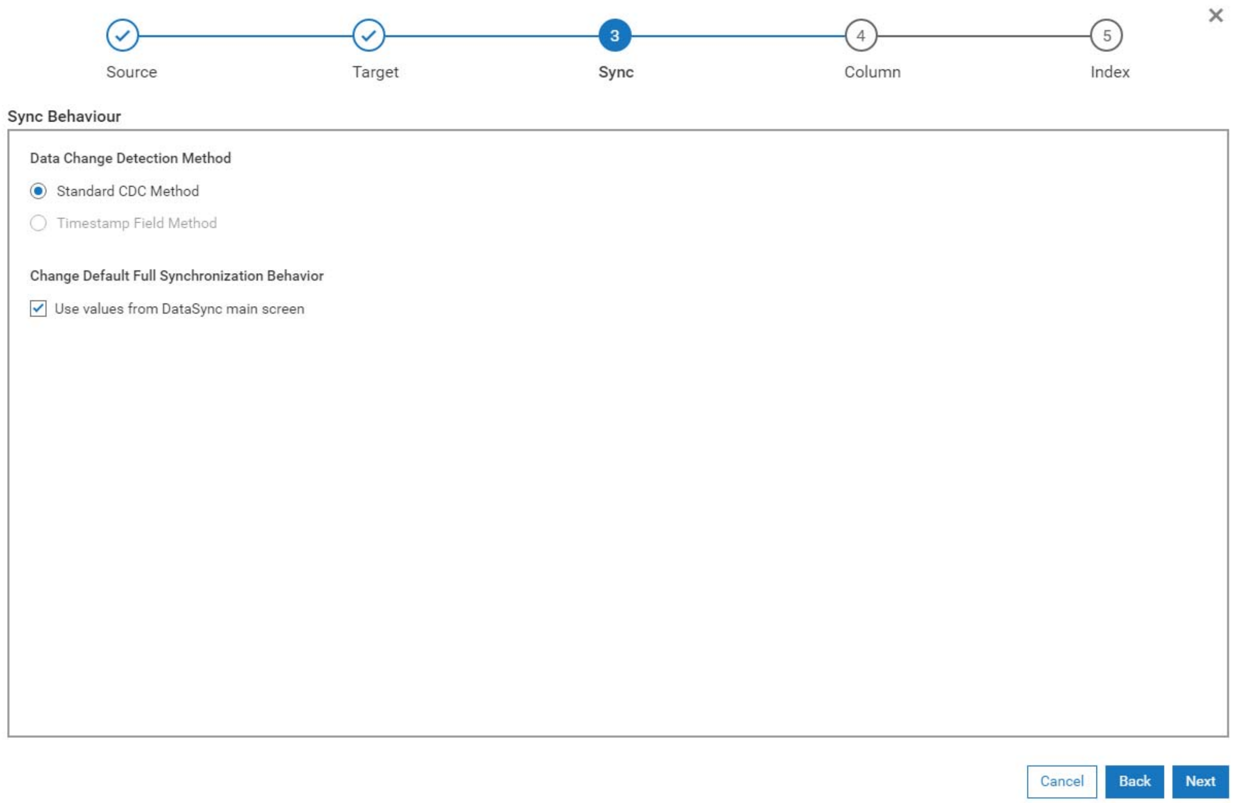
Drop and re-create the existing target table ?

○ Create a change data capture synchronization which tracks changes but does not move data. No target table is required.

Click Next to proceed to Step 3: Sync

Sync Step

Sync Behavior



Sync Behaviour

Data Change Detection Method

● Standard CDC Method

○ Timestamp Field Method

Change Default Full Synchronization Behavior

☑ Use values from DataSync main screen

Data Change Detection Method

The **Standard CDC Method** is the most commonly used setting.

The **Timestamp Field Method** can be used under special circumstances and requires two special conditions to operate correctly.

- First, there must be a column that contains timestamp information within the table that is updated with the current timestamp every time the record is updated.
- Second, there must not be any deletes performed against the data unless you do not care if the deletes get propagated to the target. The **Timestamp Field Method** only operates against inserts and updates of the data. Deletes are never performed. You can think of this method as an archival of any inserts or updates to the data.

When the **Timestamp Field Method** radio button is chosen, a drop-down list will show all available timestamp fields. Choose from this list the field updated with a new timestamp every time the record is modified.

The edit field **Drift Seconds** is used to handle time drift for the source database server. For example, if the source database server is synchronized on a daily basis using an NTP service, the apparent time of the machine may suddenly change by a small amount. To avoid the possibility of missing a crucial data update, a **drift seconds** field is supplied that allows for time corrections to the server's internal clock.   Due to the existence of leap seconds, this field should always be set to at least one second, even if the clock on the source database server is perfectly accurate.

Change Default Full Synchronization Behavior

By default the **Use values from DataSync main screen** is checked. The settings below this are set for all DataSync synchronizations in the main Settings form. However they can be overridden on an individual transformation basis. They can be overridden by unchecking **Use values from DataSync main screen** and selecting the desired behavior. Changing the settings here will only affect the current transformation and will not change any other transformations.

Click Next to proceed to Step 4: Column Mapping

Column Map step

Column Mapping

Column mapping involves the proper mapping of Source to Target Columns. This screen has many functions to allow changing of data before it is written to the target table. The functionality changes depending on what type of source and target has been selected previously. A chart below outlines these differences.

The functionality on the column map changes depending on the source and target selections made previously. The functionality can be described in 4 scenarios.

Scenario 1) Source table is SQL statement, target table is new (does not exist yet or if an Existing target table is selected with **Drop and recreate the existing target table** checked on the Target step.)

Scenario 2) Source table is SQL statement, target table already exists and **Drop and recreate the existing target table** is not checked on the Target step.

Scenario 3) Source is a single table, target table is new (does not exist yet or if an Existing target table is selected with **Drop and recreate the existing target table** checked on the Target step.)

Scenario 4) Source is a single table, target table already exists and **Drop and recreate the existing target table** is not checked on the Target step.

1. Source SQL statement, target table new
   - There is one entry for each column in the source SQL statement and all must be mapped to be valid.
   - The source columns are not editable, to change a source column, it must be done in the source SQL statement itself.
   - The Row up and Row down buttons are visible, they are used to control the order of columns in the new target table.
   - The target column names are defaulted to be the same as the source column names, however they can be changed along with their attributes (data type, length, precision, scale)
   - The Add column and Delete column buttons are not visible because they do not apply to a source SQL statement, all source columns must be mapped.
   - The "Populate Columns" with "Clear Target" and "Clear Source" buttons are not visible because they do not apply to a new target table, they are for changing the view of a mapping to an existing target table.
2. Source SQL statement, target table exists
   - There is one entry for each column in the source SQL statement and all must be mapped to be valid.
   - The source columns are not editable, to change a source column, it must be done in the source SQL statement itself.

- The "Populate Columns - Source Columns" with "Clear Target" buttons are visible and enabled. They can be used to clear the default mappings and map the target columns to the source columns.
- The "Populate Columns - Target Columns" with "Clear Source" buttons are not enabled. They do not apply when using a source SQL statement, all source columns must be mapped.
- The target column names are by default mapped to source columns with the same name. The target columns can be changed to specify other target columns. However the target names cannot be changed to a name that doesn't exist in the target table. To change an existing target table, "Drop and recreate the existing target table“ must be selected in step 2 Target.
- The Add column and Delete column buttons are not visible because they do not apply to a source SQL statement, all source columns must be mapped.
- The Row up and Row down buttons are not visible, they do not apply to an existing table since the order of the table cannot be changed.

3. Source table, target table new
    - The source columns can be changed to other columns in the table.
    - The source column field itself can be edited to create a SQL expression.
    - The Add column and Delete column buttons are visible and can be used to add more mappings using new sql expressions or duplicating a source column to many target columns.
    - The Row up and Row down buttons are visible, they are used to control the order of columns in the new target table.
    - The target column names are defaulted to be the same as the source column names, however they can be changed along with their attributes (data type, length, precision, scale)
    - The "Populate Columns" with "Clear Target" and "Clear Source" buttons are not visible because they do not apply to a new target table, they are for changing the view of a mapping to an existing target table.
4. Source table, target table exists
    - The source columns can be changed to other columns in the table.
    - The source column field itself can be edited to create a SQL expression.
    - The Add column and Delete column buttons are visible and can be used to add more mappings using new sql expressions or duplicating a source column to many target columns.
    - The "Populate Columns" with "Clear Target" and "Clear Source" buttons are visible and enabled. They can be used to clear the default mappings and map the target columns to the source columns or map the source columns to the target columns. This is helpful when the source and target table have many rows that are different and will not be mapped.
    - The target column names are by default mapped to source columns with the same name. The target columns can be changed to specify other target columns. However the target names cannot be changed to a name that doesn't exist in the target table. To change an existing target table, "Drop and recreate the existing target table“ must be selected in step 2 Target.

- The Row up and Row down buttons are not visible, they do not apply to an existing table since the order of the table cannot be changed.

Column Mapping to new Target Table

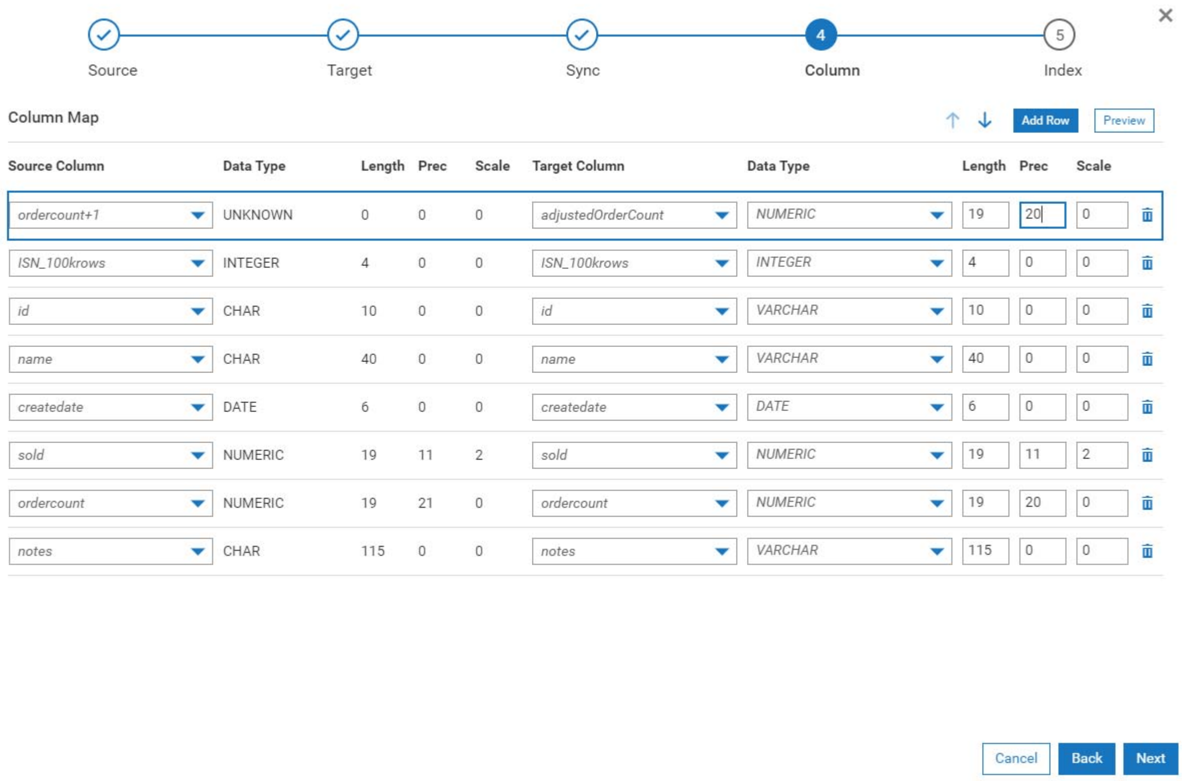
When a transform has a new table as a target, columns can be added or removed from the target table prior to the target table being created. The order of the columns can also be changed.

Note: When the source is a SQL Statement rather than a table, the column mapper does not allow columns to be added, deleted or moved in the order. In the case where the source is a SQL Statement, the columns and their order are determined by the SQL Statement.

Examples:

Adding a Derived Column

To add a column to the target table, a row needs to be added to the list of columns. Click the **Add Row** button. The column mapper will create an empty row at the top of the list. Fill in values for Source Column, Target Column, Data Type, Length, Precision and Scale as appropriate. In the below example, the source is an expression where 1 is added to the value in the ordercount field. The new column in the target table will be adjustedOrderCount. It will have a data type of NUMERIC with a length of 19 and a precision of 20. Once the new column has been defined, use the up and down arrows to position it in the desired location.

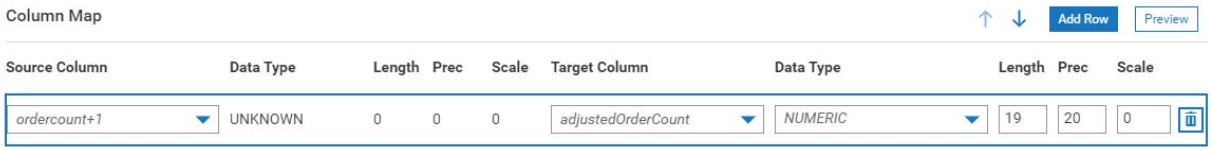


Column Map

Source Column	Data Type	Length	Prec	Scale	Target Column	Data Type	Length	Prec	Scale
ordercount+1	UNKNOWN	0	0	0	adjustedOrderCount	NUMERIC	19	20	0
ISN_100krows	INTEGER	4	0	0	ISN_100krows	INTEGER	4	0	0
id	CHAR	10	0	0	id	VARCHAR	10	0	0
name	CHAR	40	0	0	name	VARCHAR	40	0	0
createdate	DATE	6	0	0	createdate	DATE	6	0	0
sold	NUMERIC	19	11	2	sold	NUMERIC	19	11	2
ordercount	NUMERIC	19	21	0	ordercount	NUMERIC	19	20	0
notes	CHAR	115	0	0	notes	VARCHAR	115	0	0

Delete Row

To remove a column from the metadata of the target table, delete that row from the column mapper. To delete a row, click the delete button on the right edge of the row.



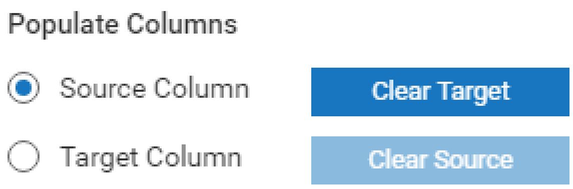
Source Column	Data Type	Length	Prec	Scale	Target Column	Data Type	Length	Prec	Scale
ordercount+1	UNKNOWN	0	0	0	adjustedOrderCount	NUMERIC	19	20	0

In this scenario, the highlighted row will be deleted. It is important to avoid deleting columns that might be a unique key, as Incremental synchronization depends on having a unique key.

Click Next to proceed to Step 5: Index

Column Mapping to Existing Target Table

In the situations where a target table exists, column mapping cannot alter the physical target table. New target columns cannot be added, but adjustments can be made to alter the source. The screen appears slightly different, as there are new **Populate Columns** controls.

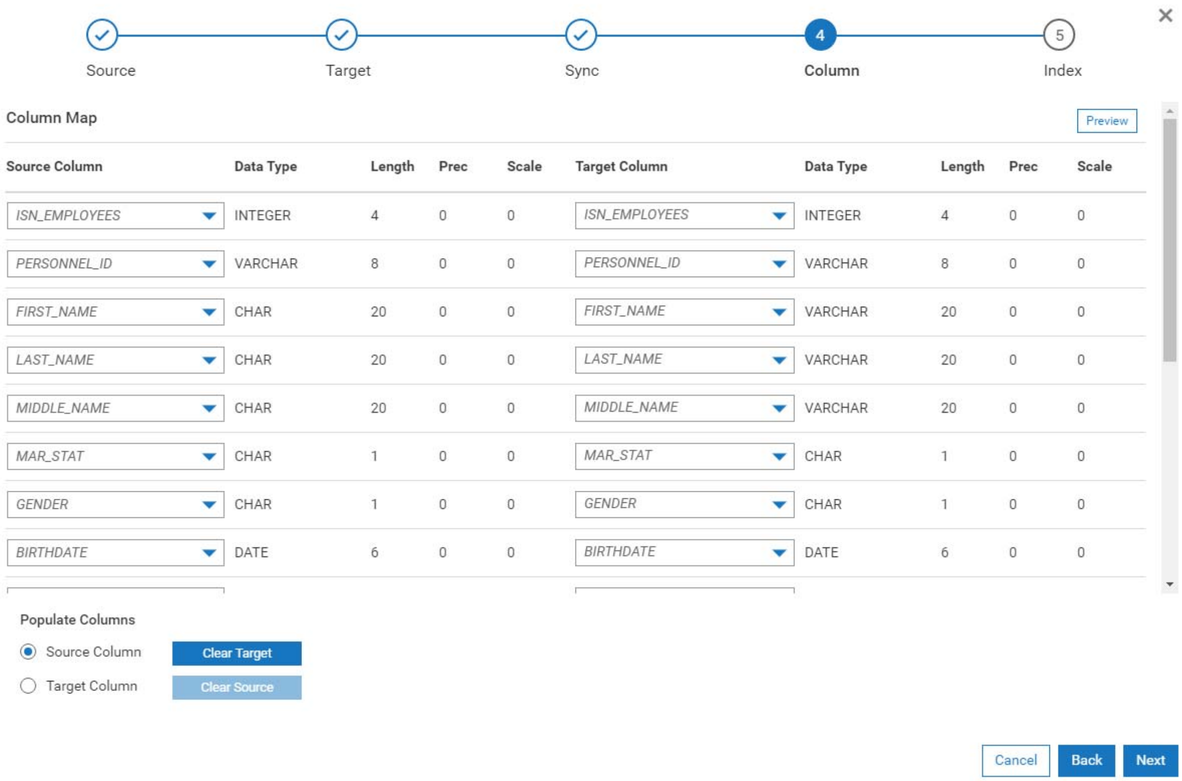


Populate Columns

● Source Column

○ Target Column

The Column Mapping to Existing Target Table screen appears as:



Column Map

Source Column	Data Type	Length	Prec	Scale	Target Column	Data Type	Length	Prec	Scale
ISN_EMPLOYEES	INTEGER	4	0	0	ISN_EMPLOYEES	INTEGER	4	0	0
PERSONNEL_ID	VARCHAR	8	0	0	PERSONNEL_ID	VARCHAR	8	0	0
FIRST_NAME	CHAR	20	0	0	FIRST_NAME	VARCHAR	20	0	0
LAST_NAME	CHAR	20	0	0	LAST_NAME	VARCHAR	20	0	0
MIDDLE_NAME	CHAR	20	0	0	MIDDLE_NAME	VARCHAR	20	0	0
MAR_STAT	CHAR	1	0	0	MAR_STAT	CHAR	1	0	0
GENDER	CHAR	1	0	0	GENDER	CHAR	1	0	0
BIRTHDATE	DATE	6	0	0	BIRTHDATE	DATE	6	0	0

Populate Columns

● Source Column

○ Target Column

Let suppose that *BIRTHDATE* was no longer desired. The user must navigate to the *BIRTHDATE* column on the Source Column, and blank it out. This will prevent *BIRTHDATE* from being used. Derived Fields can be changed, but the Target Table Column that the Derived Column feeds to CANNOT be altered.

For cases where the Designer wishes to have the flexibility of mapping each column individually, the "CLEAR TARGET/CLEAR SOURCE buttons can be used. This has the net effect of clearing either the

Source Mappings or Target Mappings. This is generally done in the case where there is a large discrepancy between the Source Feed, and the existing Target Table.

Example 1:

Target Table A had 1000 columns, but the source feed only represented 10 Columns.

The designer might do the following:

1. Select Source Columns radio button, and press the Clear Target. This will remove the Target Mappings from the screen.
2. Individually map each Target Column to the Source Column. This is accomplished by using the Drop-Down List box for the Target Column names. You will need to do this for the 10 columns desired.

Example 2:

Target Table A has 10 Columns, but the Source Feed has 10000 columns.

1. Select Target Columns radio column, and press the Clear Source. This will remove the Source mappings from the screen.
2. Individually map each Source Column by using the Drop-Down list box for the Column Names. You will need to do this for the 10 columns desired.

In certain scenarios, the Designer might want to Clear the Source or Target Columns. This might be done in the case where the Target Table has hundreds of columns, and the designer wants to map a source feed that is much smaller.

Click Next to proceed to Step 5: Index

Index step

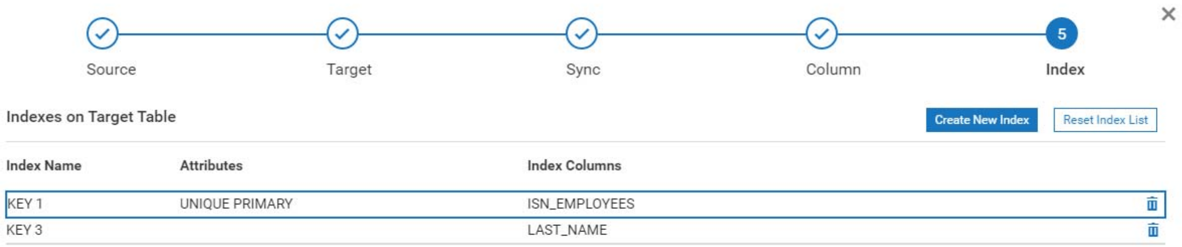
Indexes Used

The Index tab shows the indexes that are used on existing target tables, or the indexes that will be created for new target tables. For new target tables there are Index edit functions:

- Ability to propagate suggested indexes that it has harvested from the source tables to the target table
- Ability to create a new index
- Ability to delete an index

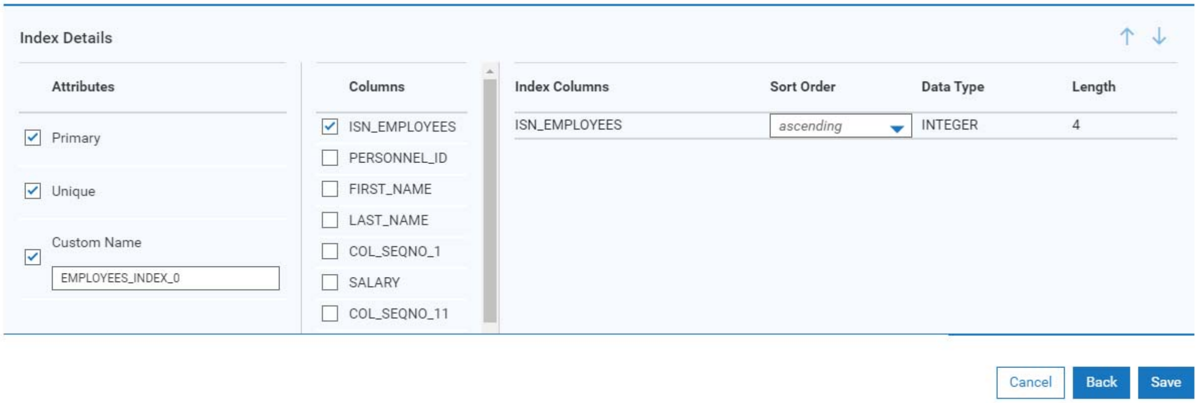
For Single Table Transforms, the defaults given might not require any user intervention. For SQL Statement transforms the unique indexes might not be able to be determined and the user will need to add them. The goal of index mapping is to select which indexes will be added to the target table when it is created. If the transform is going to an existing target table, the list of indexes is what is on the target table and cannot be changed. It is essential that a unique key be present in order for Incremental Synchronization to work. A unique key is not required in cases where Incremental Synchronization is not desired. When these transforms are synchronized, a full synchronization will be performed.

The screen may appear as:



Indexes on Target Table

Index Name	Attributes	Index Columns
KEY 1	UNIQUE PRIMARY	ISN_EMPLOYEES
KEY 3		LAST_NAME



Index Details

Attributes

☑ Primary

☑ Unique

☑ Custom Name: EMPLOYEES_INDEX_0

Columns

☑ ISN_EMPLOYEES
□ PERSONNEL_ID
□ FIRST_NAME
□ LAST_NAME
□ COL_SEQNO_1
□ SALARY
□ COL_SEQNO_11

Index Columns	Sort Order	Data Type	Length
ISN_EMPLOYEES	ascending	INTEGER	4

The available actions are:

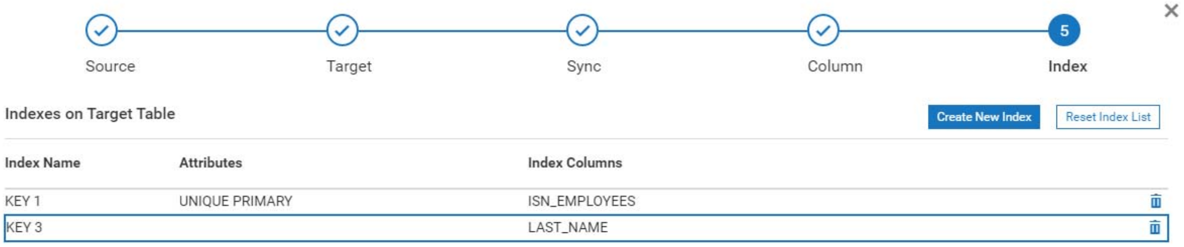
Create an Index	By doing this, the user will be prompted to select the appropriate

	columns that make up the index. They will also determine whether it is unique, and the order of the columns (in case it is a compound key). This is caused by selecting the "CREATE INDEX" button.
Delete an Index	This will delete the index from the target table. This is caused by selecting the "DELETE INDEX" button.
Reset Index List	This will reset the indexes so that they appear as they were before entering this screen.

Examples:

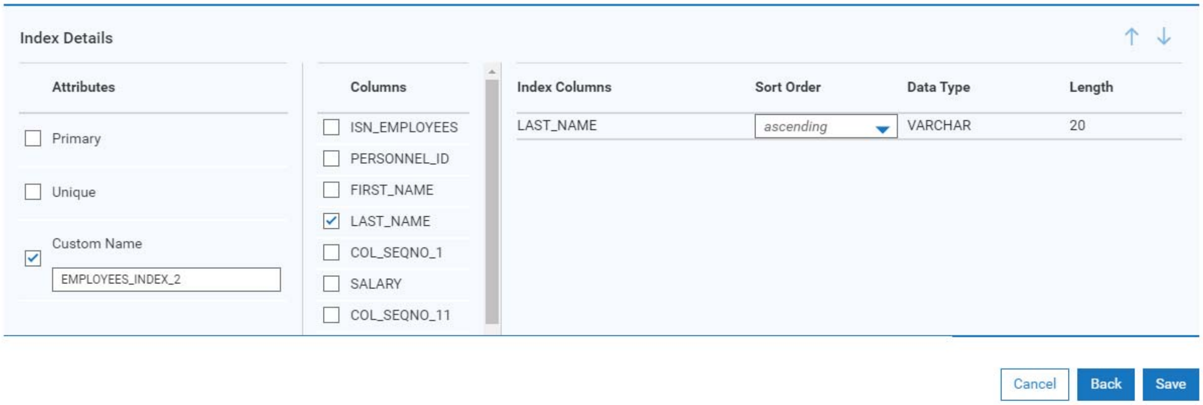
Deleting an Index

In this example, KEY3 is not needed in the target table so it will be deleted. Select the Index named KEY3 and click the delete button on the right side of the field. The index will be deleted immediately upon clicking the delete button.



Indexes on Target Table

Index Name	Attributes	Index Columns
KEY 1	UNIQUE PRIMARY	ISN_EMPLOYEES
KEY 3		LAST_NAME



Index Details

Attributes

□ Primary

□ Unique

☑ Custom Name: EMPLOYEES_INDEX_2

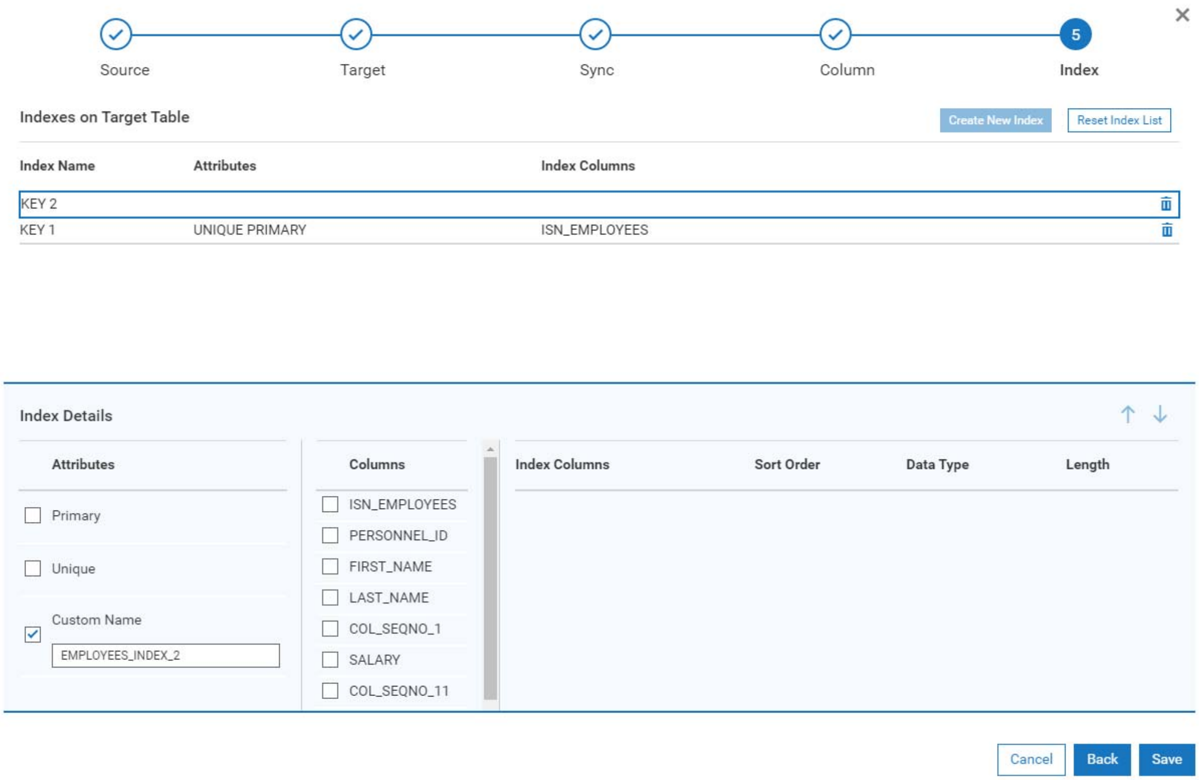
Columns

□ ISN_EMPLOYEES
□ PERSONNEL_ID
□ FIRST_NAME
☑ LAST_NAME
□ COL_SEQNO_1
□ SALARY
□ COL_SEQNO_11

Index Columns	Sort Order	Data Type	Length
LAST_NAME	ascending	VARCHAR	20

Adding an Index

By pressing the Create Index Button, an empty row will appear.



Indexes on Target Table

Index Name	Attributes	Index Columns
KEY 2		
KEY 1	UNIQUE PRIMARY	ISN_EMPLOYEES

Index Details

Attributes	Columns	Index Columns	Sort Order	Data Type	Length
□ Primary	□ ISN_EMPLOYEES				
□ Unique	□ PERSONNEL_ID				
☑ Custom Name: EMPLOYEES_INDEX_2	□ FIRST_NAME				
	□ LAST_NAME				
	□ COL_SEQNO_1				
	□ SALARY				
	□ COL_SEQNO_11				

The lower section of the screen will be highlighted. Select a column or columns that will make up the index as well as the Primary or Unique attribute. A customer name can also be specified.

Creating a Unique Index for a Transform that has multiple tables

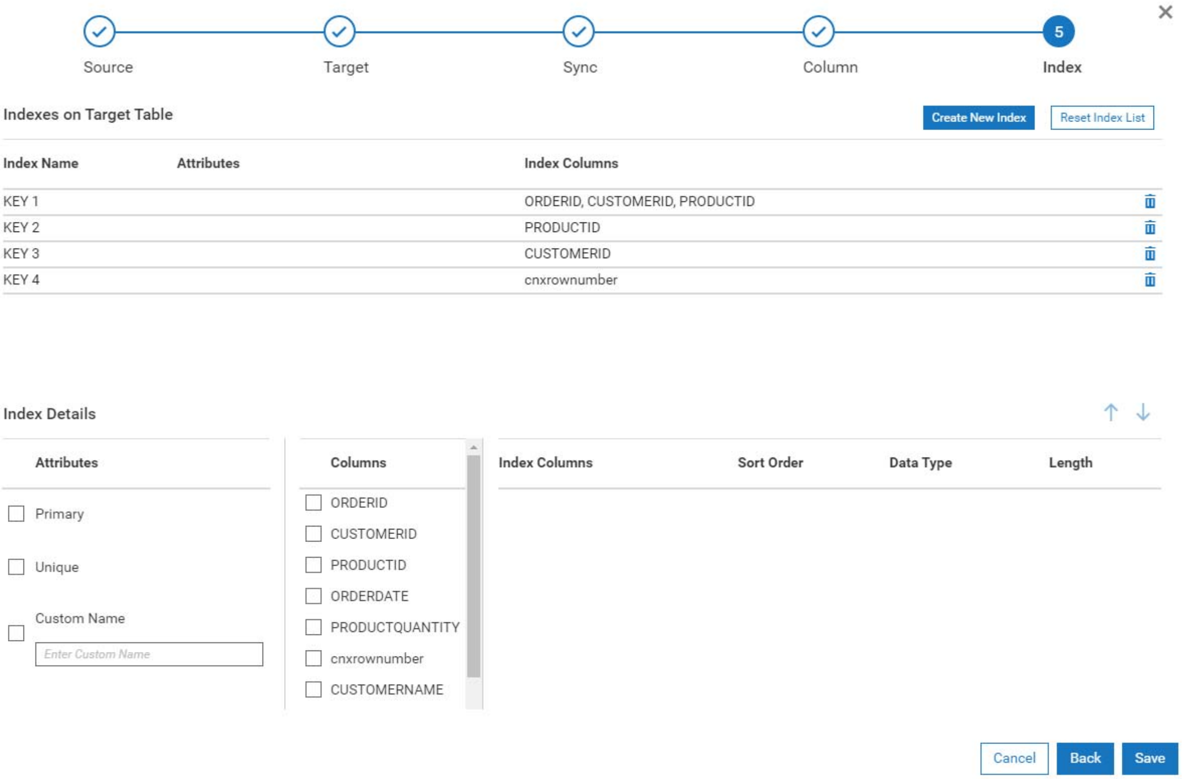
DataSync attempts to determine the possible keys that a transform might have. To do this, it looks at the source feed that makes up the transform and checks to see whether these column(s) are indexed. If they are, they are possible candidates for target table indexes. It is up to the designer to decide whether the default indexes that are shown are valid for the target feed.

When a transform is composed of more than a single table, all potential keys are extracted from the source table inputs and are listed as defaults. DataSync cannot determine whether the key is unique, because it is a joined table. A unique key for a single table might not be a unique key for the joined transform.

Consider this example:

When the Index Mapper screen is reached, it displays all of the indexes that make up this transformation. It has retrieved this index information by displaying all the indexes for:

- Orders_cisam
- Customers_cisam



Indexes on Target Table

Index Name	Attributes	Index Columns
KEY 1		ORDERID, CUSTOMERID, PRODUCTID
KEY 2		PRODUCTID
KEY 3		CUSTOMERID
KEY 4		cnxrownumber

Index Details

Attributes

□ Primary

□ Unique

□ Custom Name

Enter Custom Name

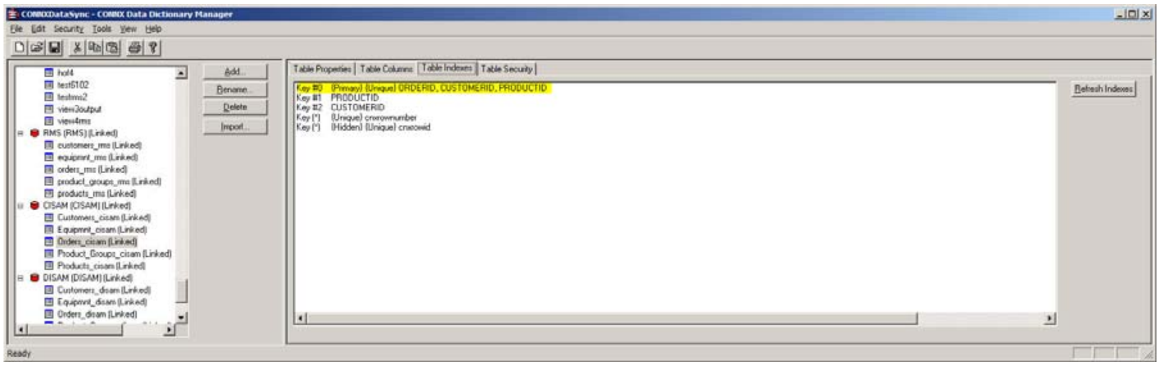
Columns

□ ORDERID
□ CUSTOMERID
□ PRODUCTID
□ ORDERDATE
□ PRODUCTQUANTITY
□ cnxrownumber
□ CUSTOMERNAME

Index Columns	Sort Order	Data Type	Length

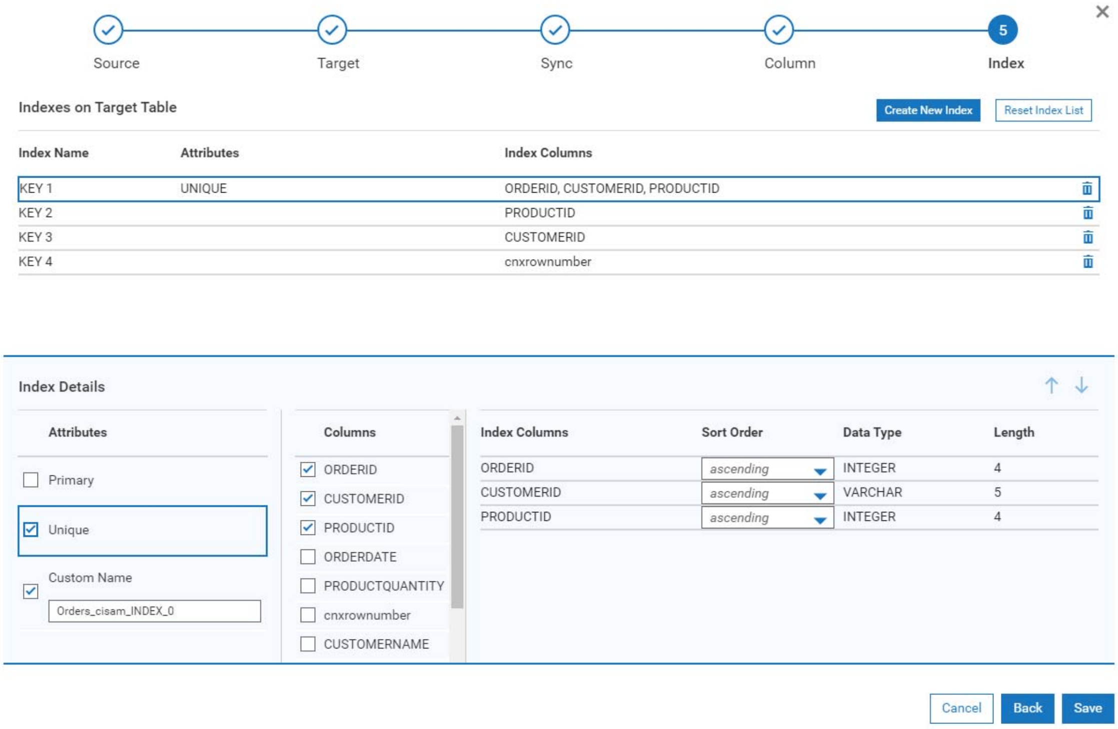
None of the Indexes show a unique key (If one existed, it would show up in the Attributes column for the specified key. Therefore, this transformation CANNOT be incrementally synchronized. The designer might intuitively know that a unique key could be composed of ORDERID, CUSTOMERID, PRODUCTID. In fact, the ORDERS_CISAM table recognizes that combination as a unique key. If the designer needs assistance on determining how to make an unique key, the following steps should be followed:

1. In the two table join, determine which table is considered the child table. This means when they are joined to each other, which table will produce more than 1 occurrence if there are matches. A good way to determine this is to examine the column(s) that join the tables. Determine which table represents the joined column(s) as a unique key. This is generally the Parent Table (Customers_cisam in this example). The other table will be the Child Table (Orders_cisam). In our example, Customers_cisam and Orders_cisam are joined on the CUSTOMERID column. This is a unique index in Customers_cisam but there can be multiple instances of a value in Orders_cisam.
2. Go to the Windows Start menu and Launch the CONNX Data Dictionary Tool. Look for the Orders_cisam table. When you have found this table, look at the indexes for this table.



```
Key #0   (Primary) (Unique) ORDERID, CUSTOMERID, PRODUCTID
Key #1   PRODUCTID
Key #2   CUSTOMERID
Key (*)  (Unique) cnxrownumber
Key (*)  (Hidden) (Unique) cnxrowid
```

3. The highlighted Key #0 is unique, which represents the ORDERID, CUSTOMERID, and PRODUCTID. This could be considered a valid unique key.
4. Go back to the Index Mapper screen, and complete the Index Transformation by clicking the *Unique* checkbox in the Attributes section. This action will result in the Target Table having a unique key of ORDERID, CUSTOMERID, and PRODUCTID. The only way to determine whether the key is unique is to run a manual synchronization of this transformation. If there is a duplicate, an error will result, and the Transformation should be edited to identify a proper unique key.



Indexes on Target Table

Index Name	Attributes	Index Columns
KEY 1	UNIQUE	ORDERID, CUSTOMERID, PRODUCTID
KEY 2		PRODUCTID
KEY 3		CUSTOMERID
KEY 4		cnxrownumber

Index Details

Attributes

□ Primary

☑ Unique

☑ Custom Name: Orders_cisam_INDEX_0

Columns

☑ ORDERID
☑ CUSTOMERID
☑ PRODUCTID
□ ORDERDATE
□ PRODUCTQUANTITY
□ cnxrownumber
□ CUSTOMERNAME

Index Columns	Sort Order	Data Type	Length
ORDERID	ascending	INTEGER	4
CUSTOMERID	ascending	VARCHAR	5
PRODUCTID	ascending	INTEGER	4

Creating a Unique Index for a Transform that does not show a potential unique or non-unique key in the default list

When the potential indexes are shown, the user might not see a potential key that could be used as an unique key. The user has two choices:

1. If the Index is not shown, this could be due to the fact that a column that potentially identifies the unique key was not part of the transformation. Consider the last example. If PRODUCTID was not part of the select list, the key (ORDERID, CUSTOMERID, PRODUCTID) would not be displayed. The designer might have to revise the transformation to include this column.
2. If the index is not shown, the user may have to create an unique key by selecting the **Create New Index** button. This button can be used in creating all types of keys (unique, composite, non-unique). Please see the example on adding an index earlier in this section

After the Index Mapping has been done, the main window will be displayed, and the transformation can be checked by running a manual synchronization on it.

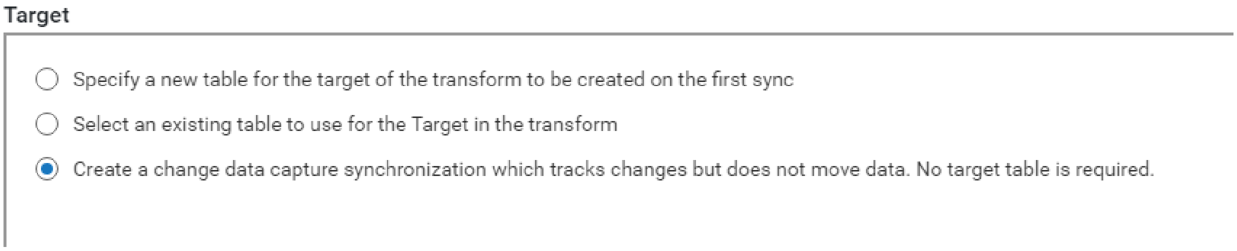
Change Data Capture transform

Creating and Syncing the Change Data Capture Transformation

A Change Data Capture Transform is a type of transformation that tracks the changes made in the source table but does not move any data. This type of transformation is designed to be used with Microsoft SQL Server Integration Services (SSIS).

For information on how to use the data from a Change Data Capture Transformation see the section Using change data capture data.

To create a change data capture transform, Click Add Transform from the Transformations Tab to enter the stepper. Select the source as before. On the Target step, check the *Create a change data capture*... radio button and press Next to proceed through the stepper. The column mapping step will be skipped because there is no target table, the transform is just tracking the changes on the source table. On the index tab, select or specify a unique index if there is one and press *Save*.



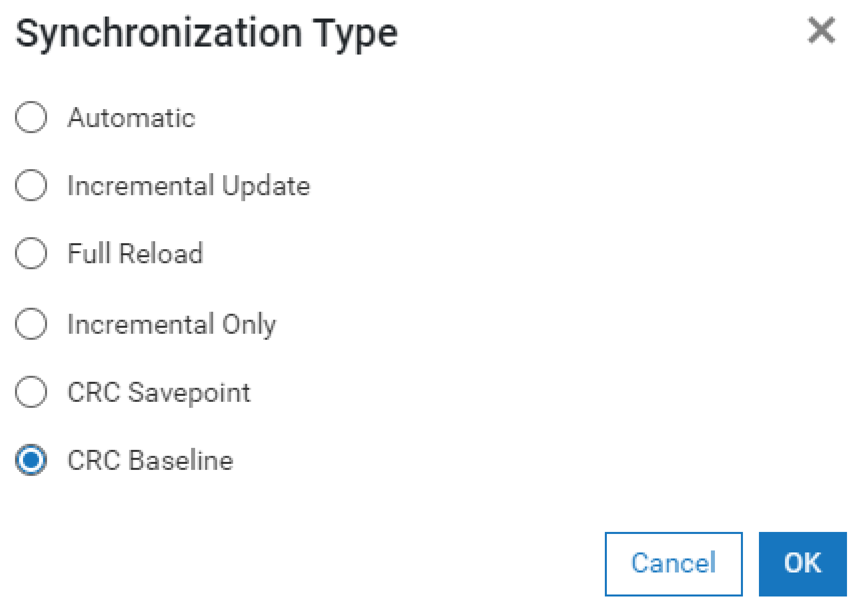
Target

○ Specify a new table for the target of the transform to be created on the first sync

○ Select an existing table to use for the Target in the transform

● Create a change data capture synchronization which tracks changes but does not move data. No target table is required.

After pressing save you are returned to the Transformations tab. Select the change data capture transform that was just created and press the *Sync* button. The Synchronize window will appear. The first time a change data capture transformation is synchronized it must create a baseline that future changes will be compared to. To do this select the *CRC Baseline* option, if there is no baseline and *CRC Savepoint* is checked the baseline will still be created. Press OK, the transform will be synced.



Synchronization Type

○ Automatic

○ Incremental Update

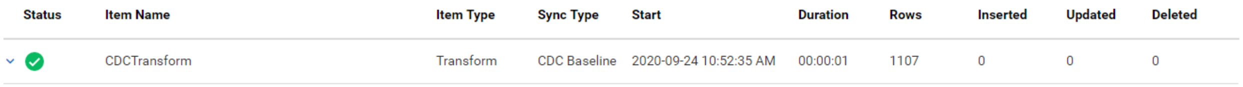
○ Full Reload

○ Incremental Only

○ CRC Savepoint

● CRC Baseline

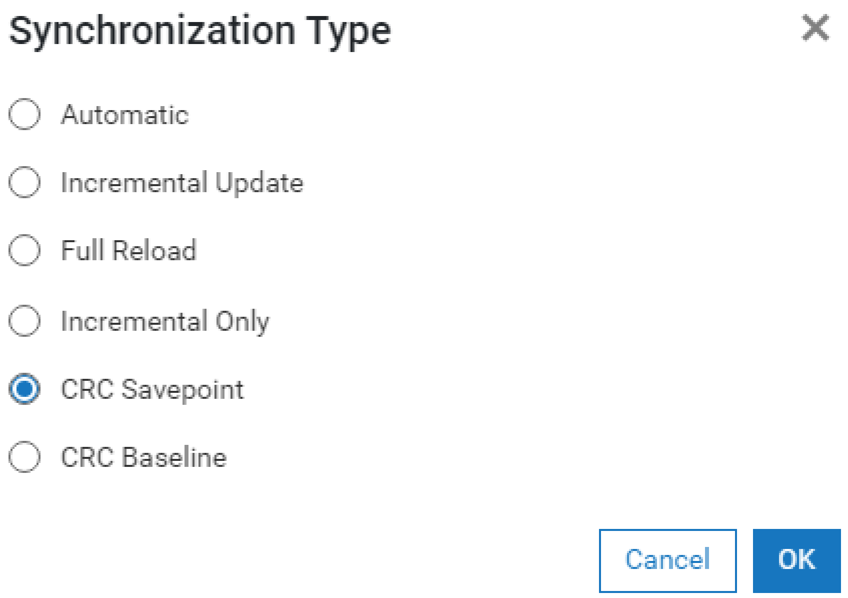
Looking at the statistics for the baseline sync, 1107 rows were synced. Since this is a baseline, there were no Inserts updates or deletes.



Status	Item Name	Item Type	Sync Type	Start	Duration	Rows	Inserted	Updated	Deleted
✓	CDCTransform	Transform	CDC Baseline	2020-09-24 10:52:35 AM	00:00:01	1107	0	0	0

InfoNaut was used to add, update and delete some rows in the source table. To see those changes, click the *Sync* button with the same cdc transform selected.

This time in the Synchronize window, select CRC Savepoint and then click OK. DataSync will now get the changes from the last time a baseline was run.



Synchronization Type

○ Automatic

○ Incremental Update

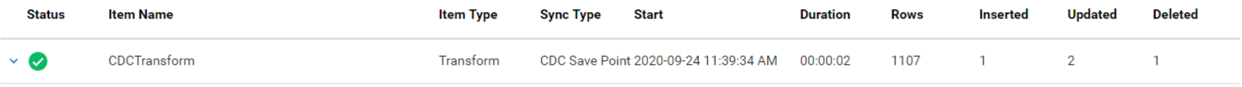
○ Full Reload

○ Incremental Only

● CRC Savepoint

○ CRC Baseline

Looking at the statistics on the transform, we can see that since the baseline was run, 1 row was inserted, 2 were updated and 1 deleted. Now we have the change data that can be used in SQL Server Integration Services.



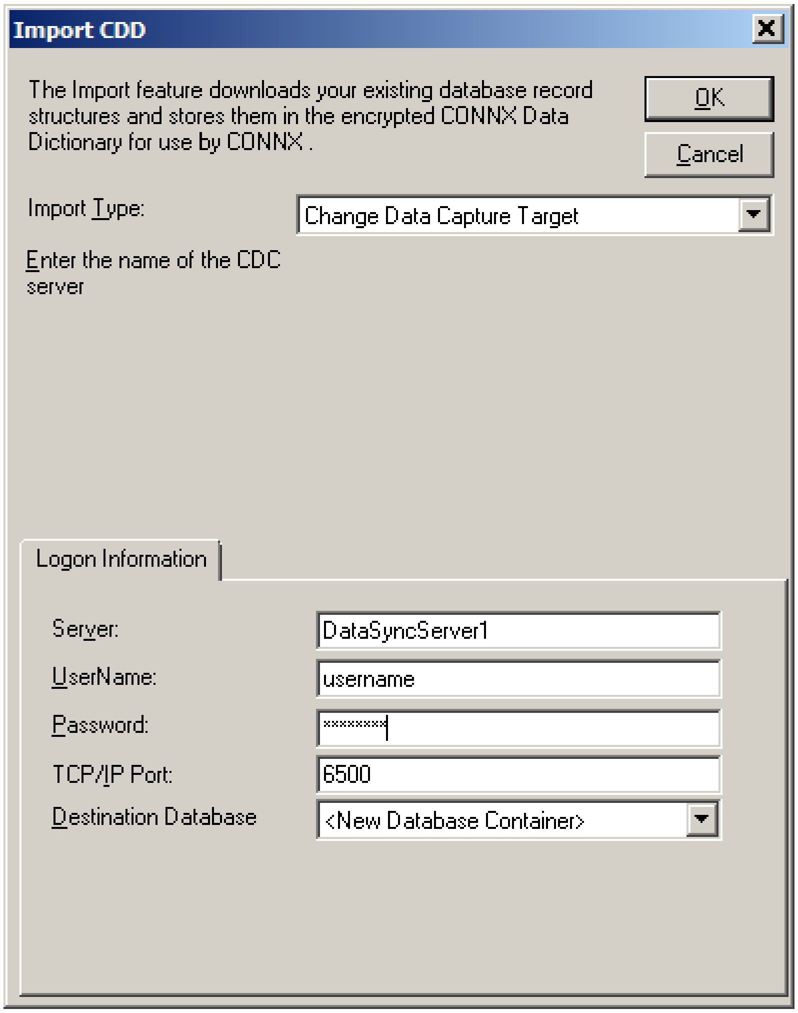
Status	Item Name	Item Type	Sync Type	Start	Duration	Rows	Inserted	Updated	Deleted
✓	CDCTransform	Transform	CDC Save Point	2020-09-24 11:39:34 AM	00:00:02	1107	1	2	1

Using the change data capture synchronization in SQL Server Integration Services

Change data capture transformations are designed to be used with Microsoft SQL Server Integration Services.

1) Create a new Data Dictionary, different than the one used for DataSync, to use for the transformation data tables. Click the Start button and then point to All Programs. Point to CONNX Driver and then click CONNX Data Dictionary. The Open dialog box appears, click cancel to get to a new empty Data Dictionary.

2) Import the new change data capture tables into the new Data Dictionary. Press the import button, then in the *Import Type* dropdown list, select *Change Data Capture Target*. Then in the *Server* field put the name of the server where DataSync is running. For the UserName and Password fields, enter the user name and password that was used to connect to the DataSync Data Dictionary. For example if user1 and pwd1 were used to connect to the Data Dictionary opened by DataSync, then user1 and pwd1 would be entered on this screen.



The Import feature downloads your existing database record structures and stores them in the encrypted CONNX Data Dictionary for use by CONNX .

Import Type: Change Data Capture Target

Enter the name of the CDC server

Server: DataSyncServer1

UserName: username

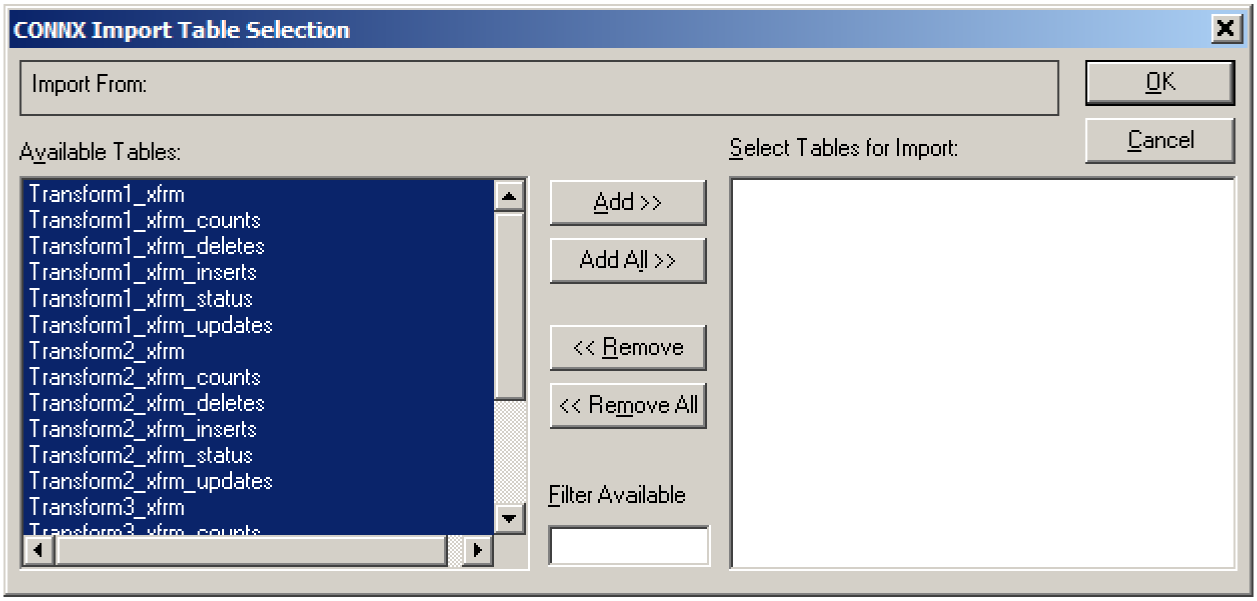
Password: ********

TCP/IP Port: 6500

Destination Database <New Database Container>

3) In the Import Table Selection screen, press the *Add All* button to get all the change data capture tables. These will be used later by the SQL Server Integration Services Work packages to get the rows that

have been inserted, updated or deleted. Press *OK* and the tables are added to the Data Dictionary. Save the Data Dictionary and close the Data Dictionary Manager tool. It can be in the same folder as the DataSync Data Dictionary, but it needs to have a different name.



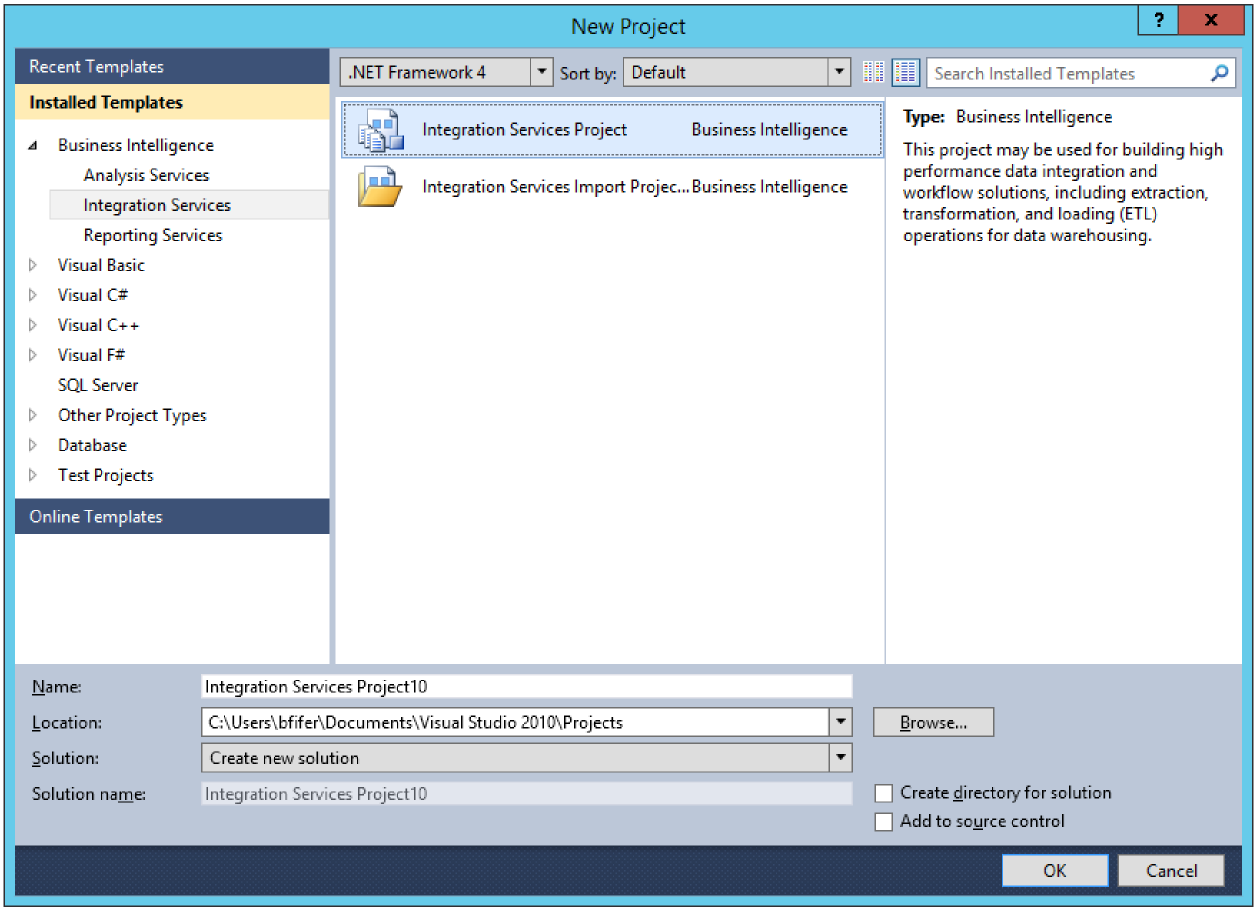
Import From:

Available Tables:

Transform1_xfrm
Transform1_xfrm_counts
Transform1_xfrm_deletes
Transform1_xfrm_inserts
Transform1_xfrm_status
Transform1_xfrm_updates
Transform2_xfrm
Transform2_xfrm_counts
Transform2_xfrm_deletes
Transform2_xfrm_inserts
Transform2_xfrm_status
Transform2_xfrm_updates
Transform3_xfrm
Transform3_xfrm_counts

Select Tables for Import:

4) Now the SQL Server Integration Services package can be built. This walkthrough will be with SQL Server 2012 Integration Services, however SQL Server 2005, 2008 and 2014 Integration Services works the same way. Click the Start button and then point to All Programs. Point to Microsoft SQL Server 2012, then click SQL Server Business Intelligence Development Studio. Click File --> New --> Project to open the New Project form. Select the "Business Intelligence Project" project type and then "Integration Services Project" enter a name and press the OK button.



.NET Framework 4 Sort by: Default

Type: Business Intelligence

This project may be used for building high performance data integration and workflow solutions, including extraction, transformation, and loading (ETL) operations for data warehousing.

Name: Integration Services Project10

Location: C:\Users\bfifer\Documents\Visual Studio 2010\Projects

Solution: Create new solution

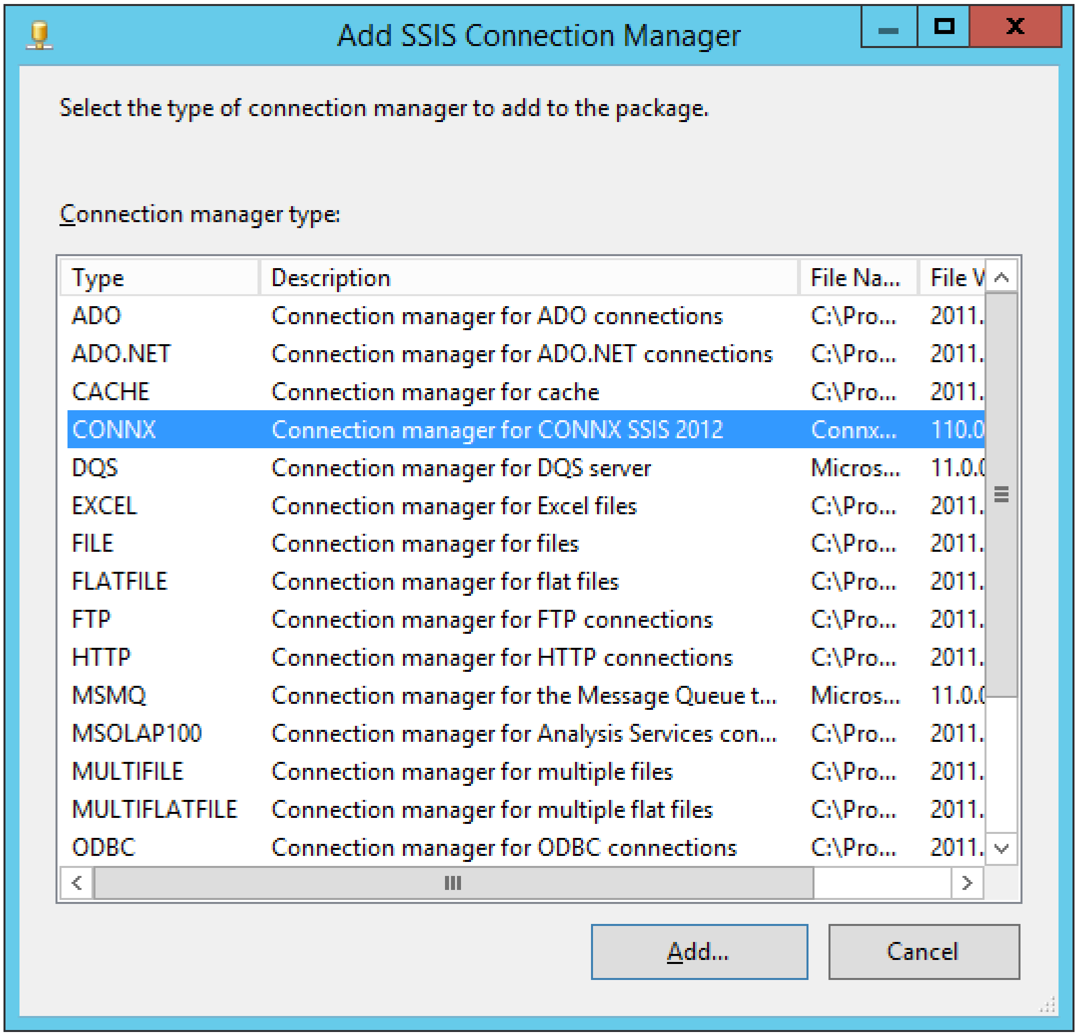
Solution name: Integration Services Project10

□ Create directory for solution

□ Add to source control

5) This should open an empty Package.dtsx environment, and the selected tab should be "Control Flow". The first thing that must be done is to add a connection to the transform table CONNX Data Dictionary. To add a connection, right click in the "Connection Managers" tab in the lower part of the "Control Flow" tab then select "New Connection.." from the pop up menu.

6) Select "CONNX" from the list of Connection manager Types and press the "Add..." button.



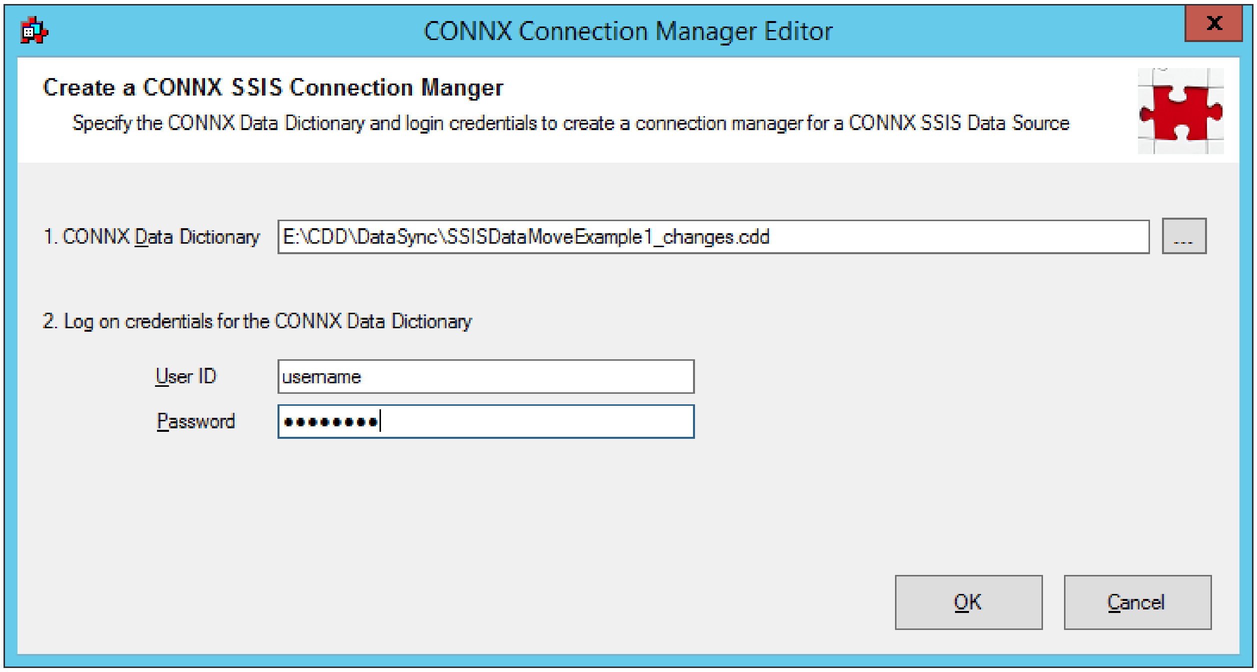
Add SSIS Connection Manager

Select the type of connection manager to add to the package.

Connection manager type:

Type	Description	File Na...	File V
ADO	Connection manager for ADO connections	C:\Pro...	2011.
ADO.NET	Connection manager for ADO.NET connections	C:\Pro...	2011.
CACHE	Connection manager for cache	C:\Pro...	2011.
CONNX	Connection manager for CONNX SSIS 2012	Connx...	110.0
DQS	Connection manager for DQS server	Micros...	11.0.(
EXCEL	Connection manager for Excel files	C:\Pro...	2011.
FILE	Connection manager for files	C:\Pro...	2011.
FLATFILE	Connection manager for flat files	C:\Pro...	2011.
FTP	Connection manager for FTP connections	C:\Pro...	2011.
HTTP	Connection manager for HTTP connections	C:\Pro...	2011.
MSMQ	Connection manager for the Message Queue t...	Micros...	11.0.(
MSOLAP100	Connection manager for Analysis Services con...	C:\Pro...	2011.
MULTIFILE	Connection manager for multiple files	C:\Pro...	2011.
MULTIFLATFILE	Connection manager for multiple flat files	C:\Pro...	2011.
ODBC	Connection manager for ODBC connections	C:\Pro...	2011.

7) Enter the name and location of the CONNX Data Dictionary that contains the transform tables. Also enter the username and password used to connect to that transform Data Dictionary. Press the OK button and it will connect to the data dictionary and verify the connection information.



Create a CONNX SSIS Connection Manger

Specify the CONNX Data Dictionary and login credentials to create a connection manager for a CONNX SSIS Data Source

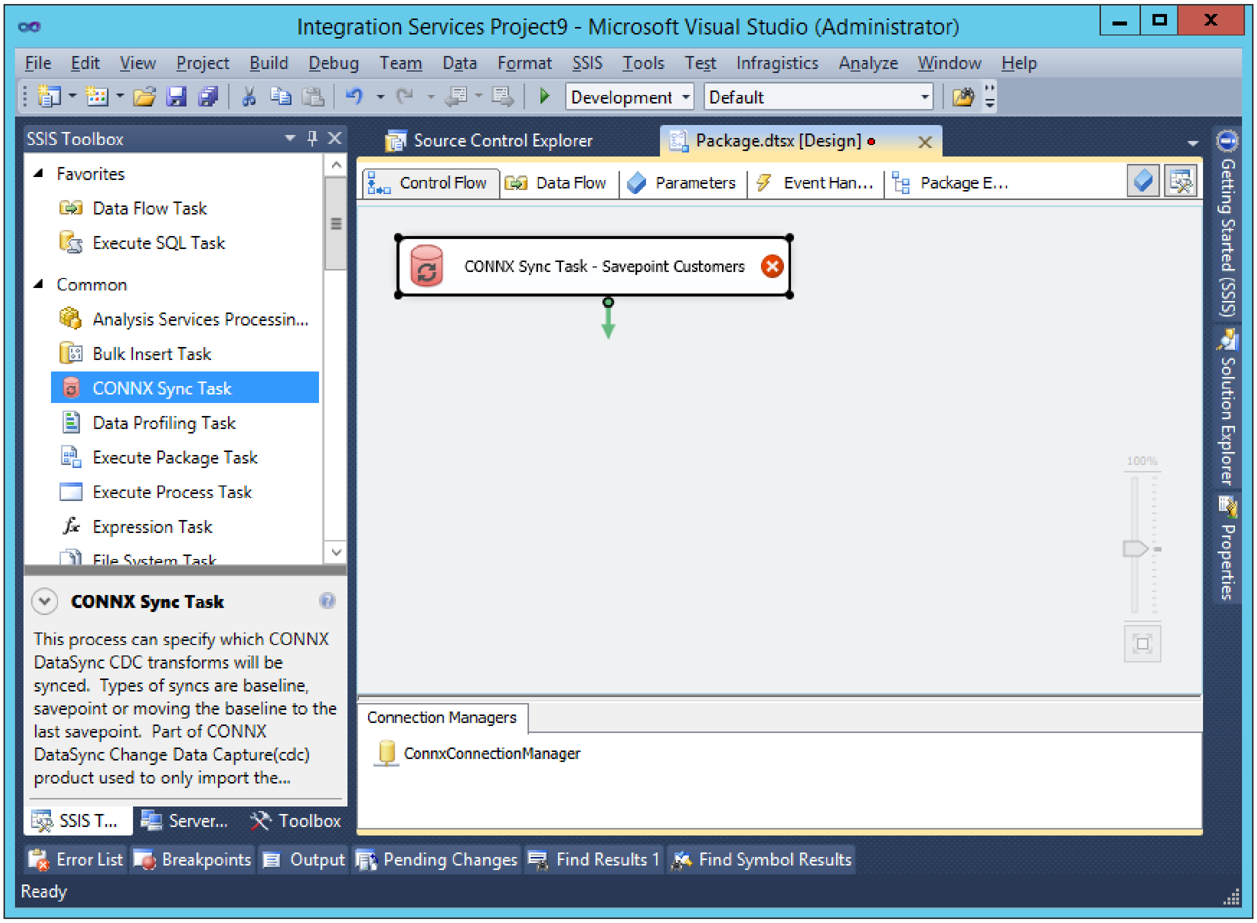
1. CONNX Data Dictionary: E:\CDD\DataSync\SSISDataMoveExample1_changes.cdd

2. Log on credentials for the CONNX Data Dictionary

User ID: username

Password: ●●●●●●●●●

8) Now that the connection is added in the "Connection Managers" tab, in the SSIS toolbox select drag the CONNX Sync Task to the design space. This will run the Savepoint Sync in the DataSync tool from the SSIS package. Rename it to - Savepoint Customers.



CONNX Sync Task

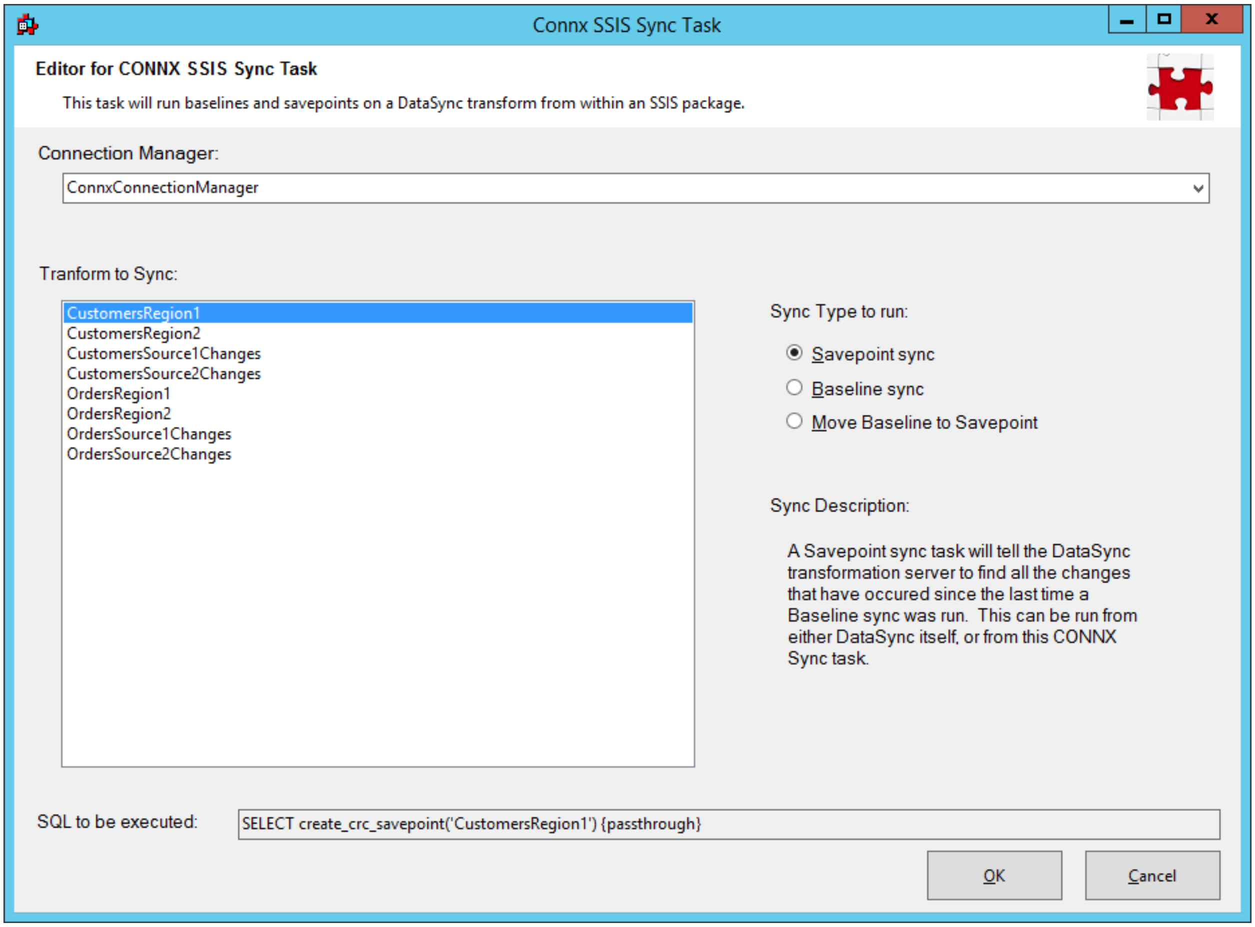
This process can specify which CONNX DataSync CDC transforms will be synced. Types of syncs are baseline, savepoint or moving the baseline to the last savepoint. Part of CONNX DataSync Change Data Capture(cdc) product used to only import the...

9) Double click on the CONNX Sync Task to configure it. If there is only one CONNX Connection manager, it will connect to that and list the Change Data Capture Transforms available. Select the transform to sync and then on the right select the type of Sync to run. Descriptions are:

> Savepoint sync: A Savepoint sync task will tell the DataSync transformation server to find all the changes that have occurred since the last time a Baseline sync was run. This can be run from either DataSync itself, or from this CONNX Sync task.
>
> Baseline sync: A Baseline sync task when run will tell the DataSync transformation server to read and save the data as it is. That way all future Savepoint syncs will use this as starting point to find changes in the data. Can be run from either DataSync,or this CONNX Sync task.
>
> Move Baseline to Savepoint sync: Making the last Savepoint sync the new change baseline is the perfect way to reset the starting point for tracking changes after the previous changes have been migrated to the targets. It doesn't actually run a Baseline sync, it simply sets the last Savepoint sync to be the new baseline.

Press OK to close the window and save the changes.

* The SQL to be executed is for reference only, The sql string is created by selecting the transform and type of sync, it is displayed so the admin knows what SQL is actually being executed if they wish to. No user editing of this line directly is necessary.



Editor for CONNX SSIS Sync Task

This task will run baselines and savepoints on a DataSync transform from within an SSIS package.

Connection Manager: ConnxConnectionManager

Tranform to Sync:

- CustomersRegion1
- CustomersRegion2
- CustomersSource1Changes
- CustomersSource2Changes
- OrdersRegion1
- OrdersRegion2
- OrdersSource1Changes
- OrdersSource2Changes

Sync Type to run:

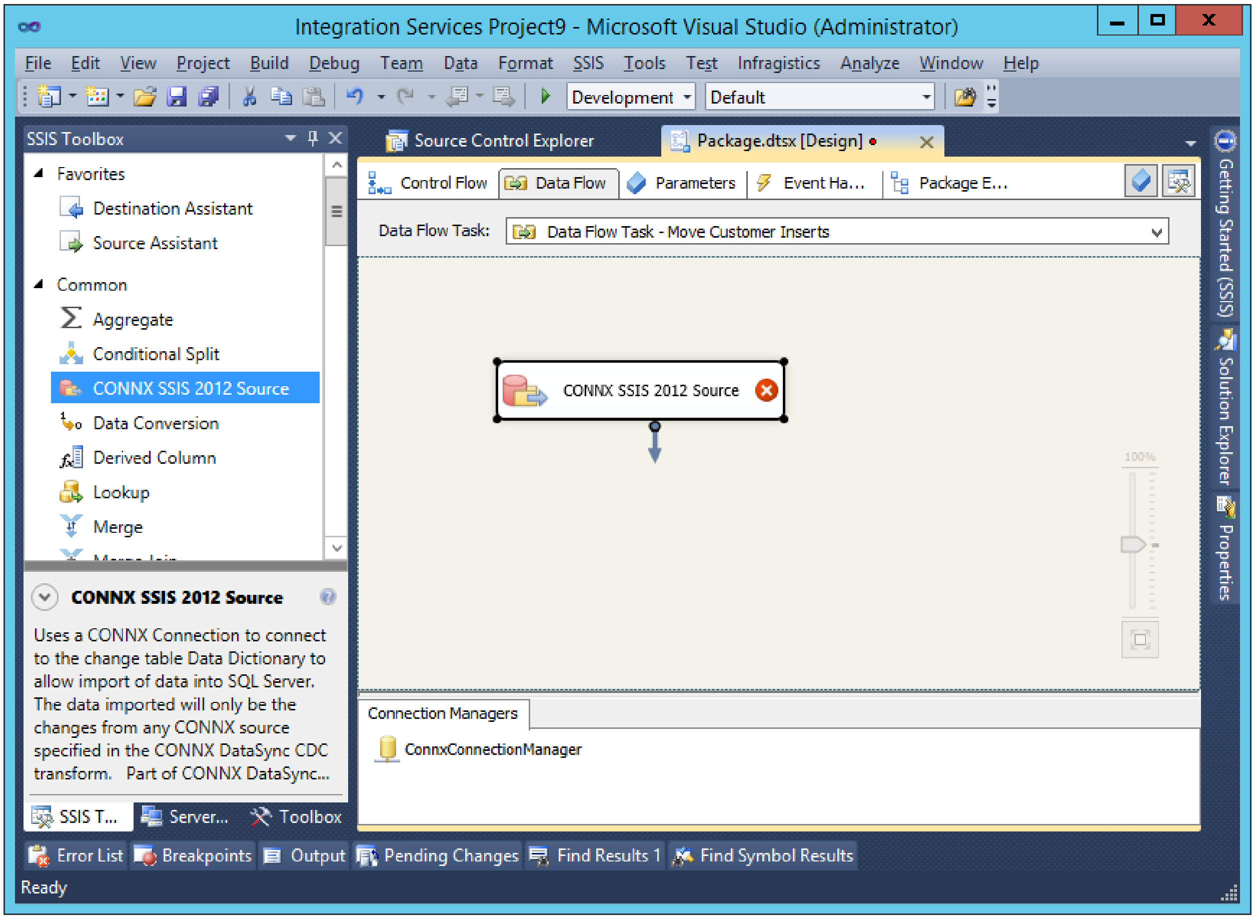
● Savepoint sync
○ Baseline sync
○ Move Baseline to Savepoint

Sync Description:

A Savepoint sync task will tell the DataSync transformation server to find all the changes that have occured since the last time a Baseline sync was run. This can be run from either DataSync itself, or from this CONNX Sync task.

SQL to be executed: SELECT create_crc_savepoint('CustomersRegion1') {passthrough}

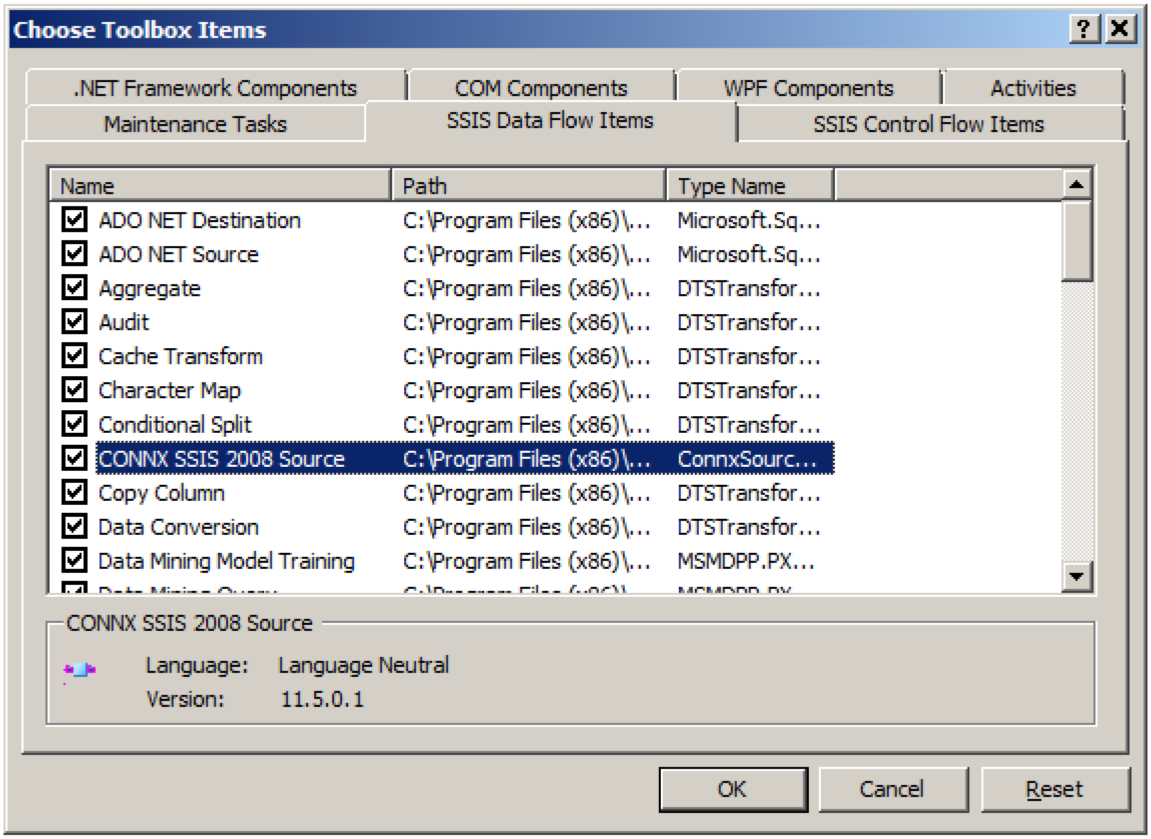
10) Now that the savepoint sync is run, we can get the data that has changed and move it into the target table. From the SSIS toolbox, drag a Data Flow Task to the designer and double click on it to bring up the Data Flow designer. Notice the SSIS toolbox has changed to only tools that work in a Data Flow Task. Drag a CONNX SSIS 2012 Source object to the design pane.



CONNX SSIS 2012 Source

Uses a CONNX Connection to connect to the change table Data Dictionary to allow import of data into SQL Server. The data imported will only be the changes from any CONNX source specified in the CONNX DataSync CDC transform. Part of CONNX DataSync...

* Note: In the Business Development Studios for SQL Server 2012 and 2014, the toolbox will automatically updated with the CONNX SSIS components. However in Business Development Studio for SQL Server 2005 and 2008 you will need to add them to the toolbox. If there is no "CONNX SSIS 2008 Source" object in the toolbox, right click on the toolbox and select "Choose Items..." from the popup menu. The list of available tools will take several seconds to appear, when it does, choose "SSIS Data Flow Items" and put a check in the box next to"CONNX SSIS 2008 Source".



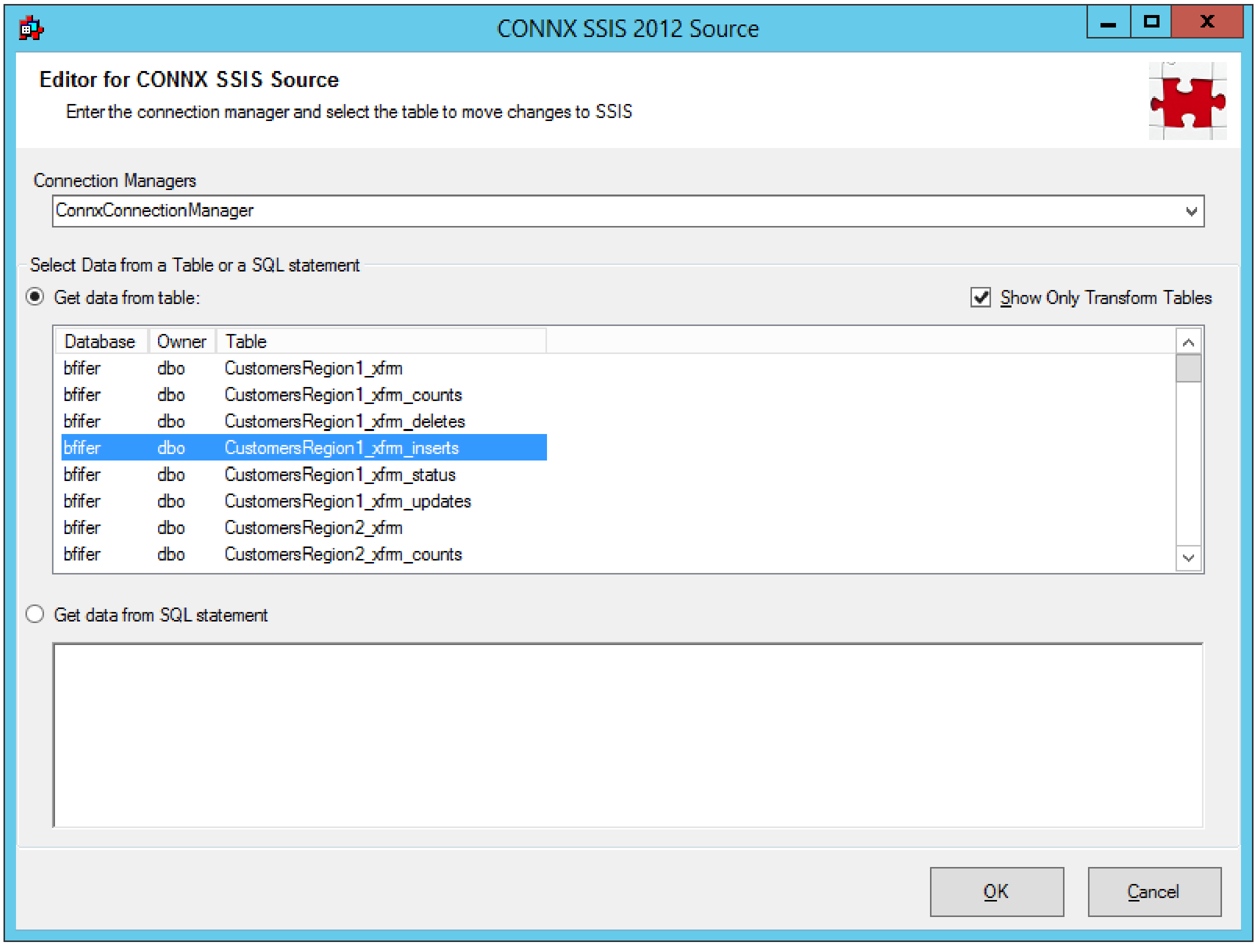
Name	Path	Type Name
☑ ADO NET Destination	C:\Program Files (x86)\...	Microsoft.Sq...
☑ ADO NET Source	C:\Program Files (x86)\...	Microsoft.Sq...
☑ Aggregate	C:\Program Files (x86)\...	DTSTransfor...
☑ Audit	C:\Program Files (x86)\...	DTSTransfor...
☑ Cache Transform	C:\Program Files (x86)\...	DTSTransfor...
☑ Character Map	C:\Program Files (x86)\...	DTSTransfor...
☑ Conditional Split	C:\Program Files (x86)\...	DTSTransfor...
☑ CONNX SSIS 2008 Source	C:\Program Files (x86)\...	ConnxSourc...
☑ Copy Column	C:\Program Files (x86)\...	DTSTransfor...
☑ Data Conversion	C:\Program Files (x86)\...	DTSTransfor...
☑ Data Mining Model Training	C:\Program Files (x86)\...	MSMDPP.PX...
☑ Data Mining Query	C:\Program Files (x86)\	MSMDPP.PX

CONNX SSIS 2008 Source

Language: Language Neutral

Version: 11.5.0.1

11) Double click on the CONNX SSIS 2012 Source box in the design window to configure the source data. If there is only one CONNX Connection manager then it will automatically connect to it and display the change data capture tables. The list of tables will have the name of the transform first, then be appended with what data is in the table. For this one we are going to use the _xfrm_inserts table to get all new rows that have been inserted into the table. Press the OK button to Save the new configuration and return to the Data Flow designer. Another option to specifying a table is to use a select SQL Statement using the change tables. This is there for customized data retrieval and is an option, however likely all the change data is needed and selecting a table will suffice.



CONNX SSIS 2012 Source

Editor for CONNX SSIS Source

Enter the connection manager and select the table to move changes to SSIS

Connection Managers: ConnxConnectionManager

Select Data from a Table or a SQL statement

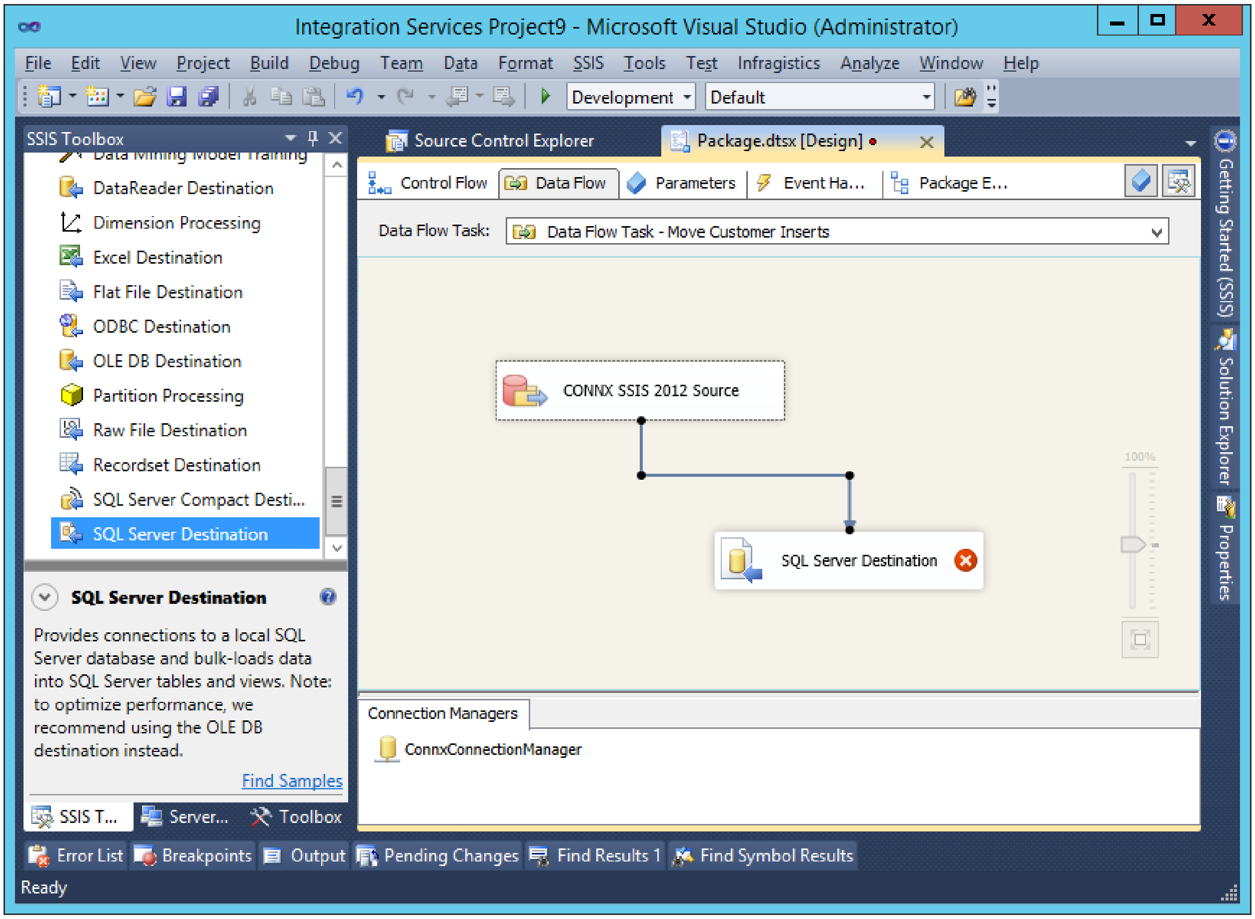
● Get data from table:

☑ Show Only Transform Tables

Database	Owner	Table
bfifer	dbo	CustomersRegion1_xfrm
bfifer	dbo	CustomersRegion1_xfrm_counts
bfifer	dbo	CustomersRegion1_xfrm_deletes
bfifer	dbo	CustomersRegion1_xfrm_inserts
bfifer	dbo	CustomersRegion1_xfrm_status
bfifer	dbo	CustomersRegion1_xfrm_updates
bfifer	dbo	CustomersRegion2_xfrm
bfifer	dbo	CustomersRegion2_xfrm_counts

○ Get data from SQL statement

12) The Data Flow is now ready for a target to move the change data to. Go to the toolbox, in the "Data Flow Destinations" section and drag "SQL Server Destination" to the design pane and drop it. Then drag the green arrow from the "CONNX SSIS 2012 Source" object to the "SQL Server Destination" object and the designer will create a green line between the two objects.



Data Flow Task: Data Flow Task - Move Customer Inserts

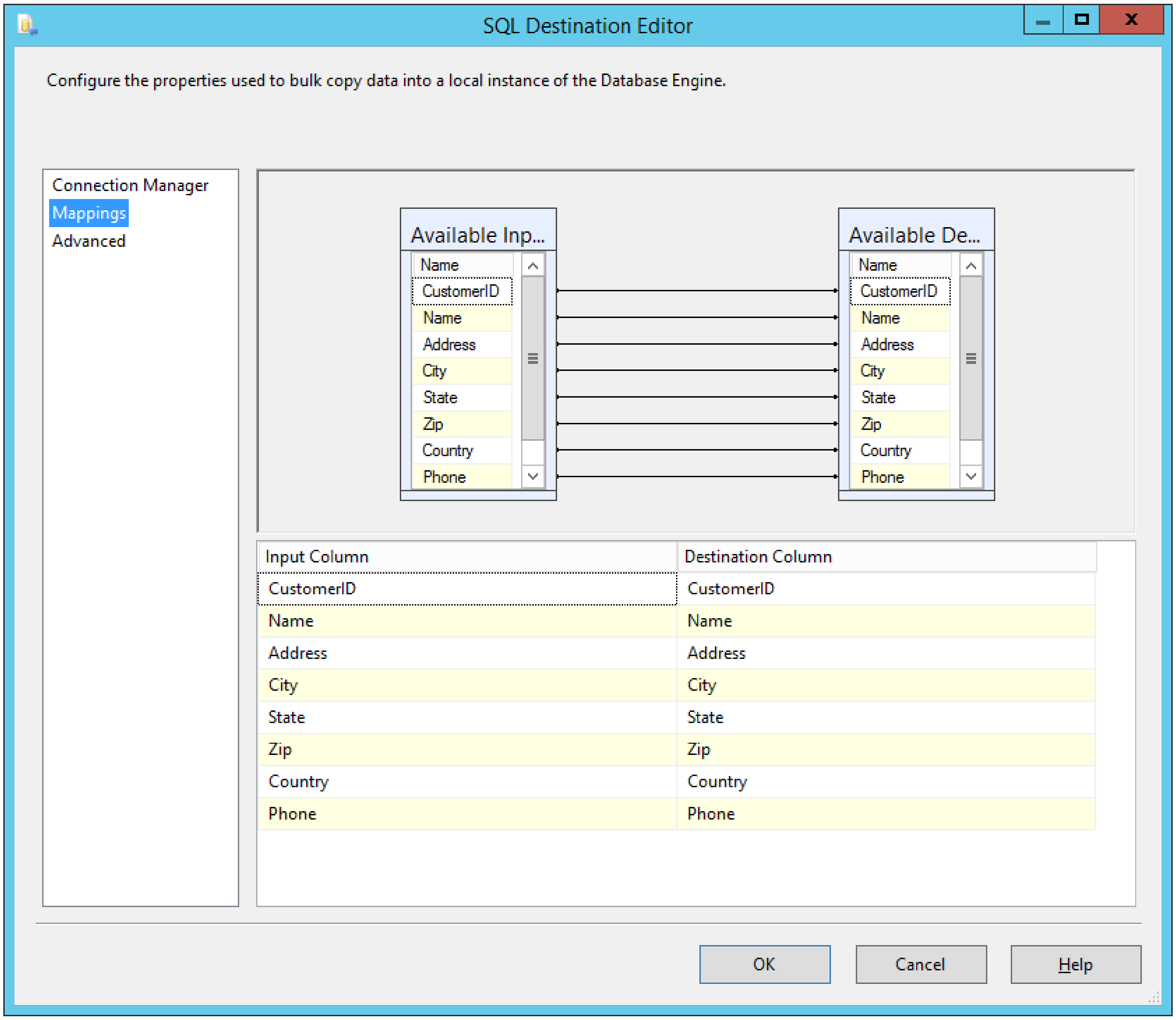
SQL Server Destination

Provides connections to a local SQL Server database and bulk-loads data into SQL Server tables and views. Note: to optimize performance, we recommend using the OLE DB destination instead.

13) Now we need to edit where the change data is moved to, right click on the SQL Server Destination object and select "Edit". That will open the SQL Destination Editor on the Connection manager screen. Next to the "Connection Manager" dropdown list select the "New..." button. That will bring up the Connection Configure screen, if you have a SQL Server connection it will be displayed in the list. If not, Select the "New" button and input the connection information to the SQL Database you wish to be the target. Select "OK" to finish and the connection specified will now be in the "Connection Manager" dropdown.

14) Select a table to move the data to from the "Use a table or view:" dropdown menu. If this is going to a new table, press the "New..." button and specify the name in the create table statement then hit "OK" and it will be selected in the dropdown menu.

15) Select "Mappings" in the list on the left, the default mappings will be displayed. Change if necessary and when finished press "OK" to return to the Data Flow designer.

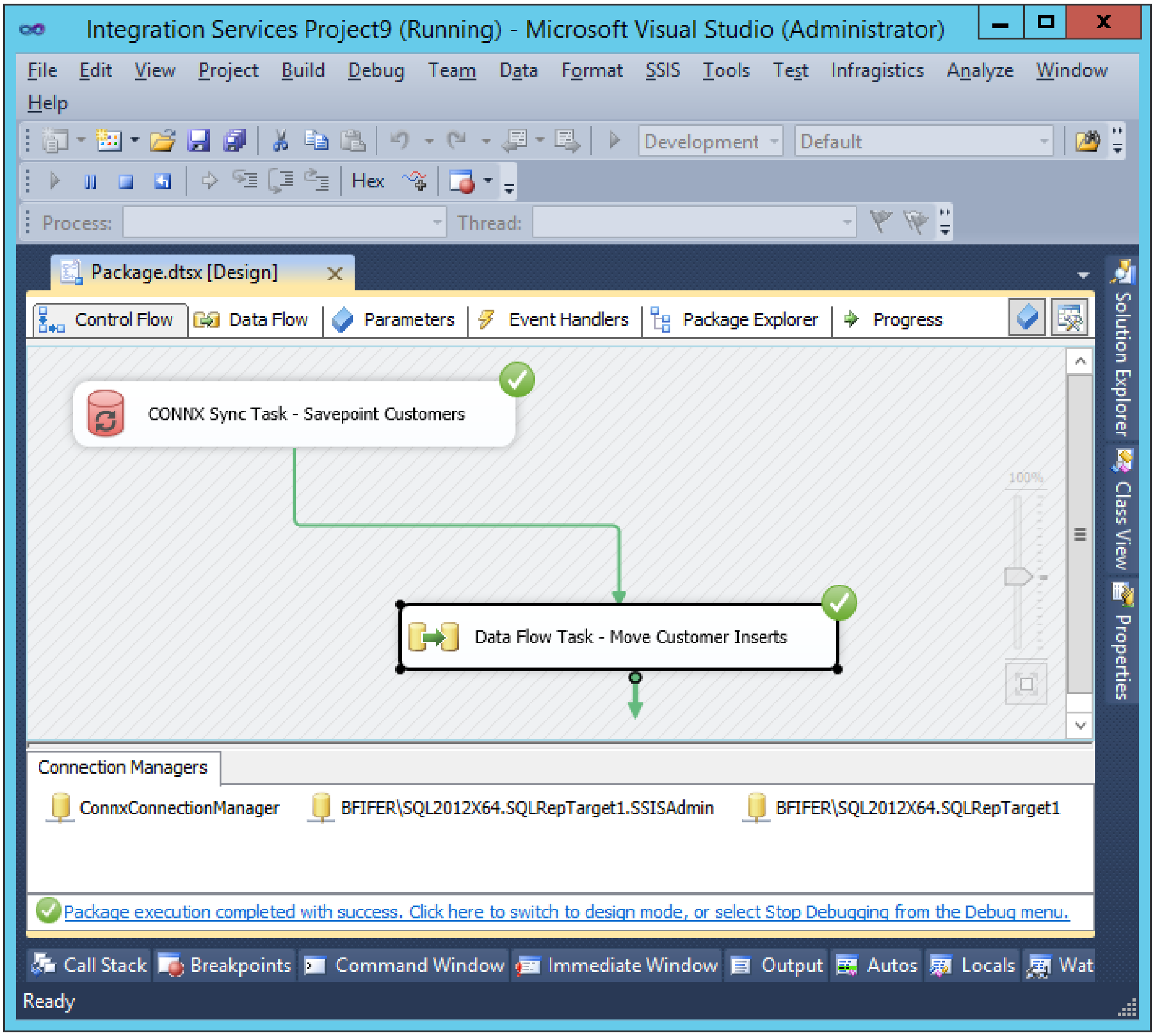


SQL Destination Editor

Configure the properties used to bulk copy data into a local instance of the Database Engine.

Input Column	Destination Column
CustomerID	CustomerID
Name	Name
Address	Address
City	City
State	State
Zip	Zip
Country	Country
Phone	Phone

16) The package is now ready to move change data. The package can be run by pressing F5, green indicates a successful run, red indicates an error.   Note, if there has not been any changes on the source data the sync will be green for success, however no entries will be moved because there were no changes.

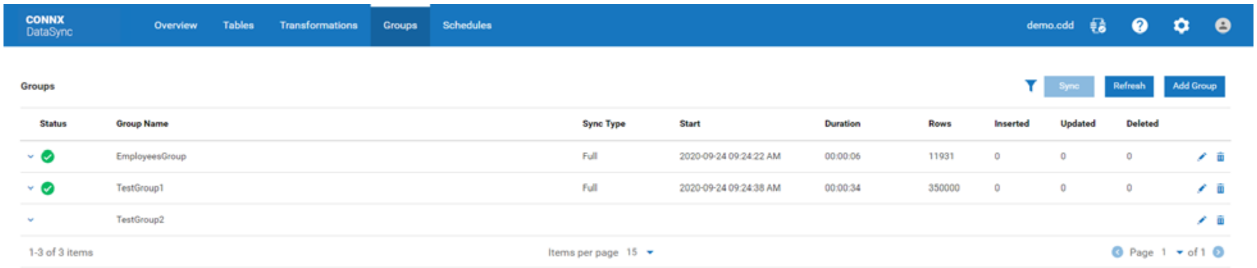


Package execution completed with success. Click here to switch to design mode, or select Stop Debugging from the Debug menu.

Groups Tab

The Groups Tab provides the ability to view, edit and manage groups. Groups provides the ability to combine related tables and transformations into a single unit called a group. When a group is synchronized, all the tables and transformations within it will be synchronized as part of the same job.

Some functions on this screen use icons. The description of these icons will have a hyperlink to the Glossary of Icons which contains a picture of the icon and an explanation of its purpose.

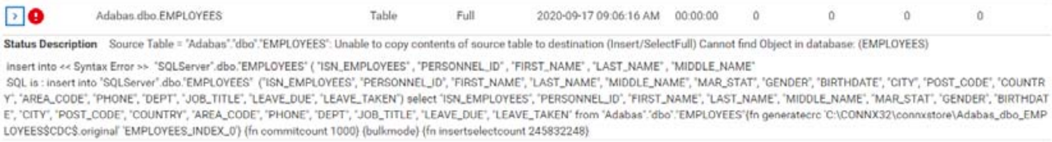


Groups

Status	Group Name	Sync Type	Start	Duration	Rows	Inserted	Updated	Deleted
✓	EmployeesGroup	Full	2020-09-24 09:24:22 AM	00:00:06	11931	0	0	0
✓	TestGroup1	Full	2020-09-24 09:24:38 AM	00:00:34	350000	0	0	0
	TestGroup2							

1-3 of 3 items   Items per page 15   Page 1 of 1

- **Groups List**
  The groups list displays information about all the groups available. Below is a description of each column in the table:
  - **More|Less Chevron**
    If more information is present for a row than can be displayed, clicking the *more chevron* will expand the row. Clicking the *less chevron* will collapse the information. Below is an example of a row that contains an error message. The *more chevron* has been clicked to display the additional information.



› !	Adabas.dbo.EMPLOYEES	Table	Full	2020-09-17 09:06:16 AM	00:00:00	0	0	0	0

```
Status Description   Source Table = 'Adabas'.'dbo'.'EMPLOYEES': Unable to copy contents of source table to destination (Insert/SelectFull) Cannot find Object in database: (EMPLOYEES)
insert into << Syntax Error >> 'SQLServer'.dbo.'EMPLOYEES' ( 'ISN_EMPLOYEES' , 'PERSONNEL_ID' , 'FIRST_NAME' , 'LAST_NAME' , 'MIDDLE_NAME'
SQL is : insert into 'SQLServer'.dbo.'EMPLOYEES' ('ISN_EMPLOYEES', 'PERSONNEL_ID', 'FIRST_NAME', 'LAST_NAME', 'MIDDLE_NAME', 'MAR_STAT', 'GENDER', 'BIRTHDATE', 'CITY', 'POST_CODE', 'COUNTR
Y', 'AREA_CODE', 'PHONE', 'DEPT', 'JOB_TITLE', 'LEAVE_DUE', 'LEAVE_TAKEN') select 'ISN_EMPLOYEES', 'PERSONNEL_ID', 'FIRST_NAME', 'LAST_NAME', 'MIDDLE_NAME', 'MAR_STAT', 'GENDER', 'BIRTHDAT
E', 'CITY', 'POST_CODE', 'COUNTRY', 'AREA_CODE', 'PHONE', 'DEPT', 'JOB_TITLE', 'LEAVE_DUE', 'LEAVE_TAKEN' from 'Adabas'.'dbo'.'EMPLOYEES'{fn generatecrc 'C:\CONNX32\connxstore\Adabas_dbo_EMP
LOYEES$CDC$.original' 'EMPLOYEES_INDEX_0'} {fn commitcount 1000} {bulkmode} {fn insertselectcount 245832248}
```

  - **Status**
    The status column will contain either a green success icon or a red error icon depending on whether the last time the transform was run was successful or resulted in an error. If the group has never been synchronized, the column will be blank. If the status shows an error, clicking on the *more chevron* will expand the row and display the error text.
  - **Group Name**
    This is the name of the group that was specified when the group was created.
  - **Sync Type**
    Displays the type (Automatic, Full, Incremental or Incremental Only) of synchronization from the last time this table was synchronized.
  - **Start**
    Timestamp displaying the starting date and time from the most recent synchronization.
  - **Duration**
    The duration in hours, minutes and seconds (hh:mm:ss) from the most recent synchronization
  - **Rows**
    Number of rows affected by the operation
  - **Inserted**
    Number of rows inserted by the operation

    - **Updated**
      Number of rows updated by the operation
    - **Deleted**
      Number of rows deleted by the operation
  - **Edit**
    Clicking this button will take you to the Group Properties dialog where tables and transformations can be added to or removed from an existing group. See Adding and Editing Groups for more information.
  - **Delete**
    Clicking the Delete button will remove the group from the list of available groups to sync. It will not delete the tables or transformations that are in the group. If the group is part of a schedule, the schedule must be deleted before the group can be deleted.
- **Search Filter**
  Pressing this button activates search fields for the Status and Group Name fields. To filter the results in the list of groups, enter a value in one or more of these search fields. For example, entering the word Employees in the *Group Name* search box will cause only those groups that have the word EMPLOYEES in the *Group Name* to be displayed.
- **Sync** button
  To perform an **on demand sync**, select a table or tables by clicking the selection checkbox next to the table or tables you want to sync. Once your selection has been made, press the Sync button. This will start the synchronization process on the selected tables. The view will switch to the Overview Summary Page where you will be able to monitor the progress.

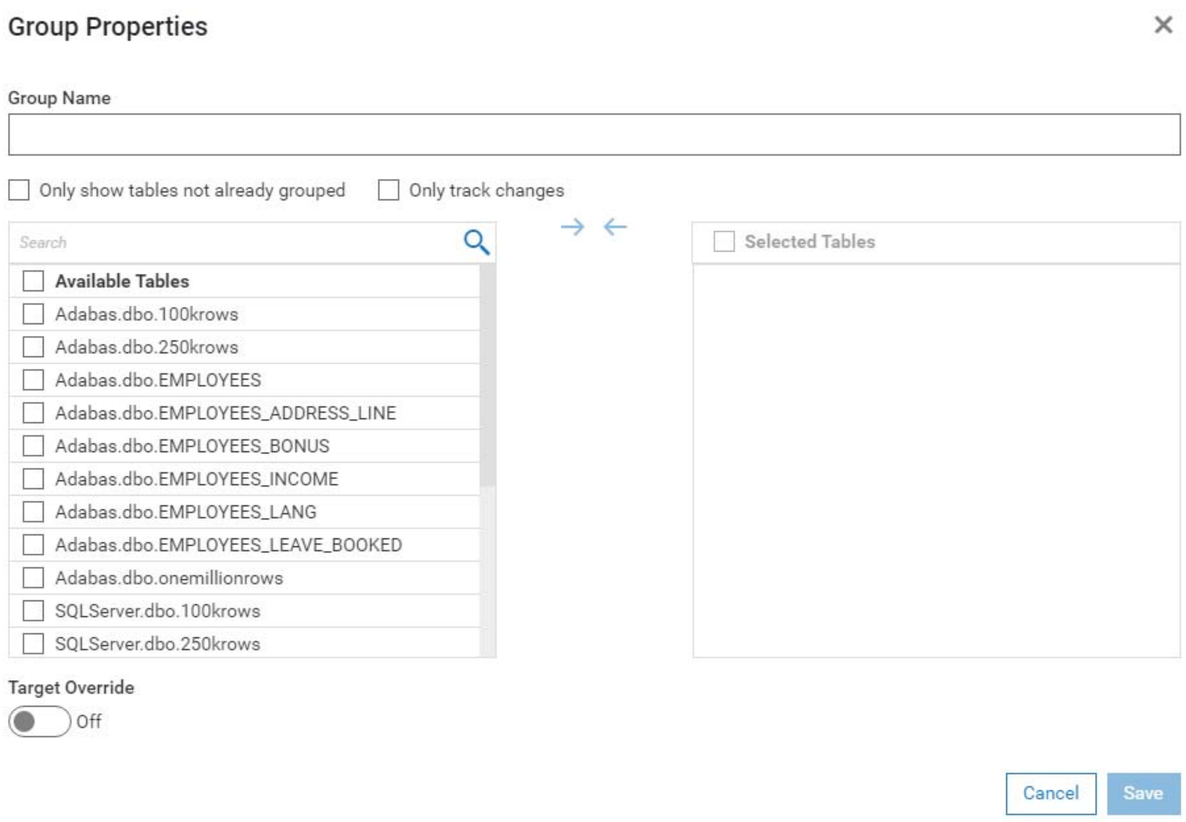
  To perform this synchronization on a regular basis, you can create a schedule. To create a schedule, go to the **Schedule** tab to establish the timing of the synchronization. See To schedule a synchronization task for more information.

  For more information about On Demand Synchronization, see Synchronizing Groups to Target Destination.
- **Refresh** button
  The refresh button refreshes the list of groups.
- **Add Group** button
  Adding a Group will add a new group to the list of existing groups that can be synchronized. This button will take you to the Group Properties dialog where you will be able to specify which tables and/or transformations to add to the group. See Adding and Editing Groups for more information.

Adding and Editing Groups

Adding and editing groups are both accomplished in the Group Properties dialog. To access the Group Properties dialog, go to the Groups Tab and click the **Add Group** button to add a new group, or click the edit button on the group to be edited.

When **Adding a new group**, the Group Properties fields will be empty:



Group Properties

Group Name

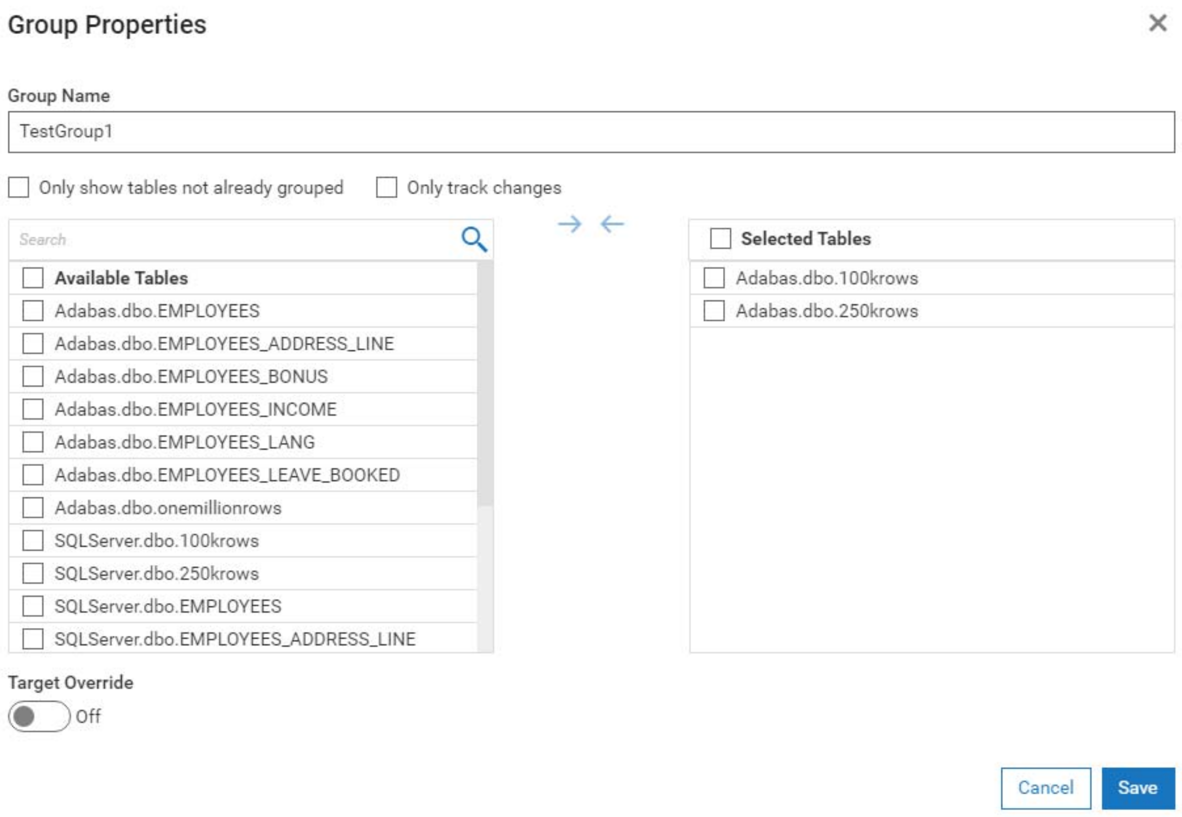
□ Only show tables not already grouped □ Only track changes

□ Available Tables
□ Adabas.dbo.100krows
□ Adabas.dbo.250krows
□ Adabas.dbo.EMPLOYEES
□ Adabas.dbo.EMPLOYEES_ADDRESS_LINE
□ Adabas.dbo.EMPLOYEES_BONUS
□ Adabas.dbo.EMPLOYEES_INCOME
□ Adabas.dbo.EMPLOYEES_LANG
□ Adabas.dbo.EMPLOYEES_LEAVE_BOOKED
□ Adabas.dbo.onemillionrows
□ SQLServer.dbo.100krows
□ SQLServer.dbo.250krows

□ Selected Tables

Target Override
Off

When **Editing an existing group**, the Group Properties fields will contain the current values for the group:



Group Properties

Group Name: TestGroup1

□ Only show tables not already grouped □ Only track changes

□ Available Tables
□ Adabas.dbo.EMPLOYEES
□ Adabas.dbo.EMPLOYEES_ADDRESS_LINE
□ Adabas.dbo.EMPLOYEES_BONUS
□ Adabas.dbo.EMPLOYEES_INCOME
□ Adabas.dbo.EMPLOYEES_LANG
□ Adabas.dbo.EMPLOYEES_LEAVE_BOOKED
□ Adabas.dbo.onemillionrows
□ SQLServer.dbo.100krows
□ SQLServer.dbo.250krows
□ SQLServer.dbo.EMPLOYEES
□ SQLServer.dbo.EMPLOYEES_ADDRESS_LINE

□ Selected Tables
□ Adabas.dbo.100krows
□ Adabas.dbo.250krows

Target Override: Off

The Group Properties Dialog contains the following fields:

- **Group Name**
  This is the name the group will be known as. Spaces and non printable characters are not allowed as part of the name.

- **Only show tables not already grouped** checkbox
  Click the Only show tables not already grouped check box to show ungrouped tables. Displaying and viewing ungrouped tables may help to avoid accidentally scheduling resynchronization of identical tables or transformations.

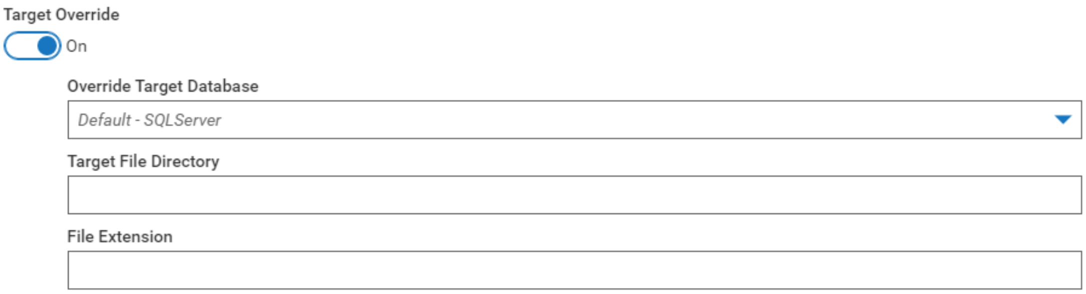
- **Only track changes** checkbox
  Checking this checkbox will filter the list of available tables to only show transformations. This includes Change Data Capture Transformations.
  Note: When unchecked, CDC Transformations will not be displayed in the Available Tables list.

- **Available Tables** listbox
  This is the list of tables and transformations that are available to be added to the group. There is a Search field at the top of the list that will filter the list based on what is put in the field. For example, entering "employees" in the search field will filter the list to only show entries that contain the string "employees".

- **Selected Tables** listbox
  This is the list of tables and transformations that are in the group.

- **Right and Left Arrow** buttons
  These arrow buttons move items to and from the Available Tables listbox and the Selected Tables listbox. To move an item, check the checkbox next to the item and then click the arrow button for the direction the item is to be moved. Items can also be moved by dragging and dropping from one listbox to the other. Multiple selections can be made by clicking individual check boxes or by selecting a range using the shift key.

- **Target Override**
  Optionally, you may override the default target database by selecting **Target Override.**



Target Override
On

Override Target Database
Default - SQLServer

Target File Directory

File Extension

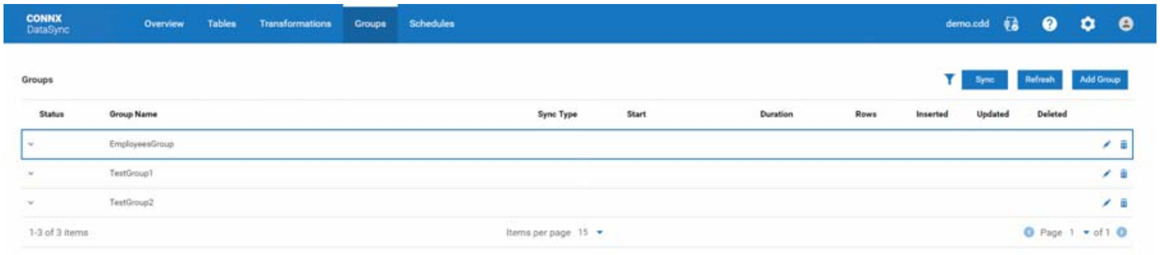
Selecting a new target from the Override Target Database dropdown. Enter the target file name directory, if applicable. (This function is only available for databases which use target file names.) To change the file name extension, for example, from .dat to .bak, enter the new file name extension in the **File Extension** text box

Synchronizing Groups to Target Destination

Groups can be synchronized on demand or via the scheduler. This section describes how to synchronize a group on demand. For information on scheduling groups for synchronization, please refer to the section on scheduling a synchronization task.

To execute an on demand synchronization task for a group:

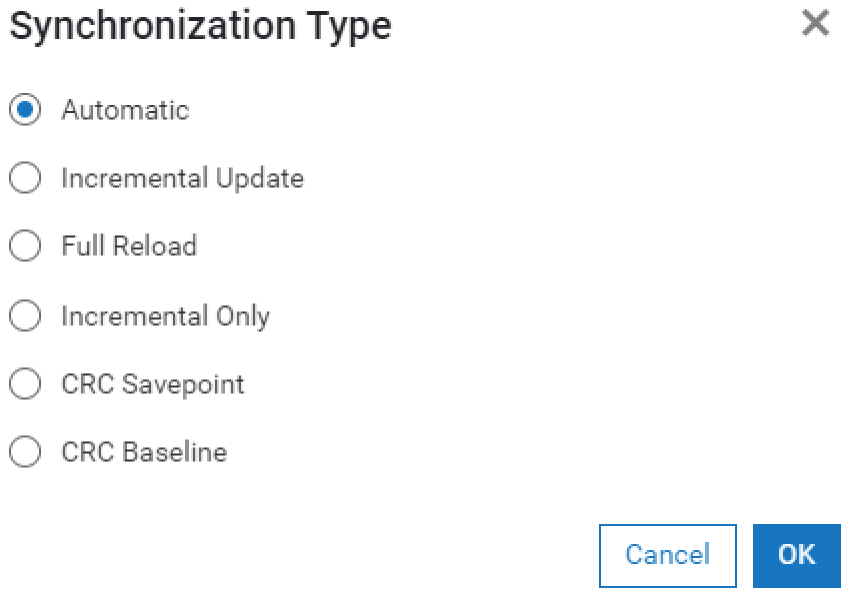
1. Select the **Groups** tab
2. Select a group by clicking on the row the group is in and click the **Sync** button to start the synchronization process.



Status	Group Name	Sync Type	Start	Duration	Rows	Inserted	Updated	Deleted
	EmployeesGroup							
	TestGroup1							
	TestGroup2							

1-3 of 3 items   Items per page 15   Page 1 of 1

3. The **Synchronize Type** dialog box appears.


Synchronization Type

● Automatic
○ Incremental Update
○ Full Reload
○ Incremental Only
○ CRC Savepoint
○ CRC Baseline

4. Select the synchronization type, **Automatic, Incremental Update, Full Reload,** or **Incremental Only**, depending on what is required. The recommended synchronization for the selected database is displayed. Note that an **Incremental Update** is only available if a full reload has already been performed. Automatic will perform an Incremental Update unless the amount of changed records has exceeded the Full Reload Threshold, or a unique key is not available for the table. If Incremental Only has been selected, it will always incrementally update the table (the full reload threshold will be ignored). CRC SavePoint and CRC Baseline are not used with Group

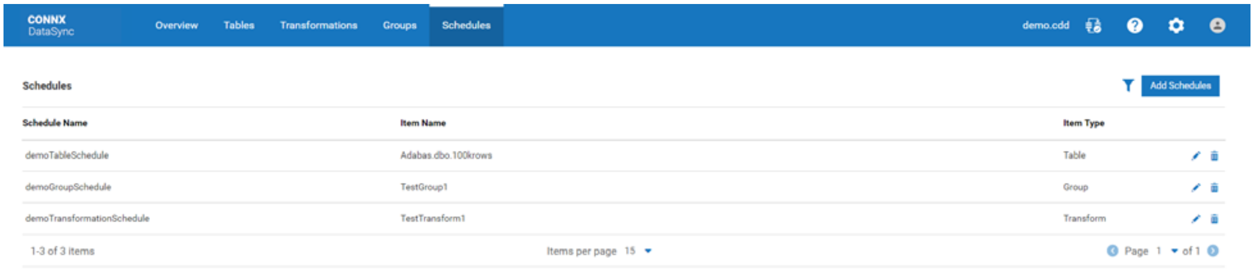
Synchronizations. For more information on the CRC types, see the section on Creating and Syncing Change Data Capture Transforms.

5. Click the **OK** button to start the synchronization process.
6. When the synchronization process starts, the application will switch to the Overview Summary Page and will display the current status and progress of all running synchronization jobs.

Schedules Tab

The Schedules Tab provides the ability to view, edit and manage schedules. Schedules provides the ability to run synchronization tasks on a schedule. Schedules can be defined to synchronize a single table, a single transformation or groups which are comprised of one or more tables and/or transformations.

Some functions on this screen use icons. The description of these icons will have a hyperlink to the Glossary of Icons which contains a picture of the icon and an explanation of its purpose.



Schedules

Schedule Name	Item Name	Item Type
demoTableSchedule	Adabas.dbo.100krows	Table
demoGroupSchedule	TestGroup1	Group
demoTransformationSchedule	TestTransform1	Transform

1-3 of 3 items   Items per page 15   Page 1 of 1

- **Schedules List**
  The schedules list displays information about all the schedules available. Below is a description of each column in the table:
  - **Schedule Name**
    This is the name of the schedule that was specified when the schedule was created.
  - **Item Name**
    This is the name of the table, transformation or group that is scheduled.
  - **Item Type**
    Specifies if the item being scheduled is a Table, Transformation or Group.

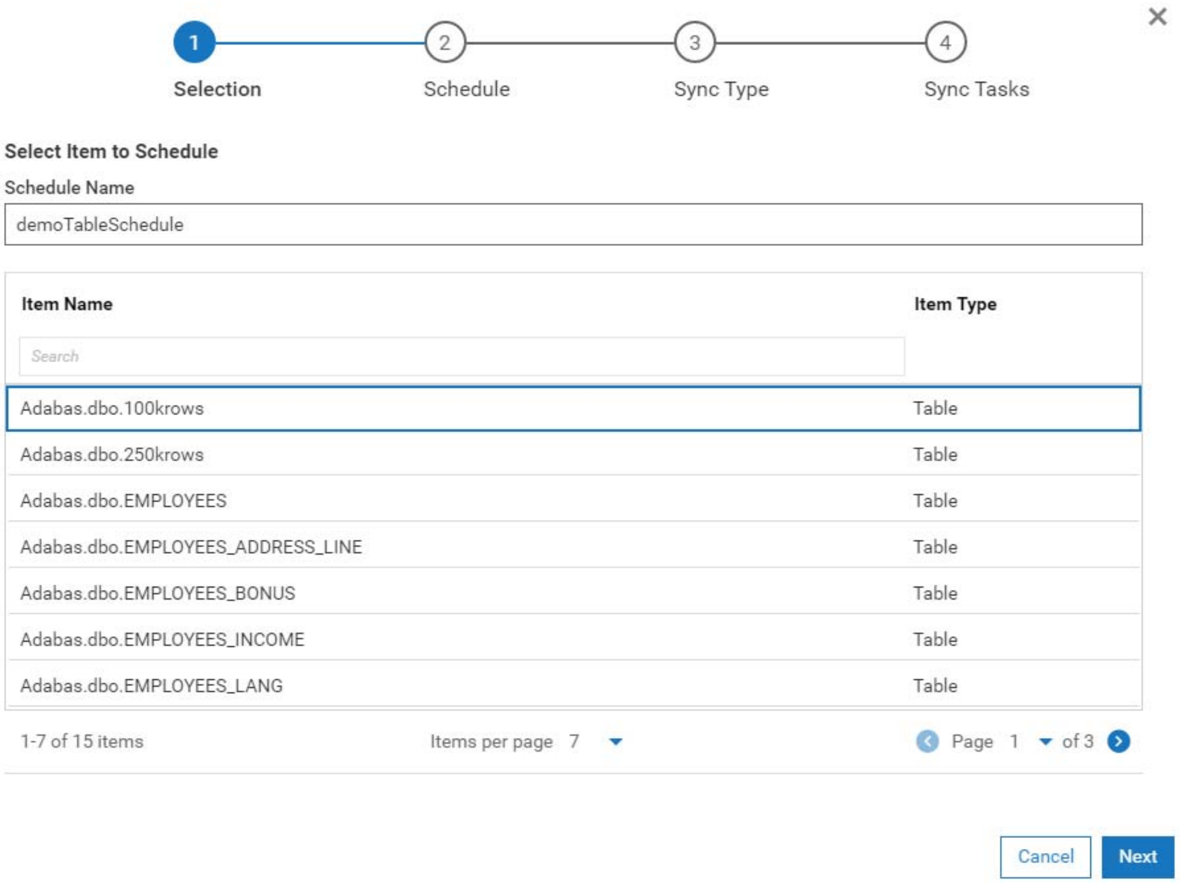
  - **Edit**
    Clicking this button will take you to the Schedule stepper. This stepper takes you through all the steps of editing an existing schedule. When accessed from the Edit button, the each step will be populated with the values for the current schedule. See Designing a Schedule for more information.
  - **Delete**
    Clicking the Delete button will remove the schedule from the list of available schedules. Once removed, the scheduled job will no longer run. It will not delete the tables, transformations or groups that were in the schedule. If a group is part of a schedule, the schedule must be deleted before the group can be deleted.
- **Search Filter**
  Pressing this button activates search fields for the Schedule Name and Item Name fields. To filter the results in the list of schedules, enter a value in one or more of these search fields. For example, entering the word "test" in the *Item Name* search box will cause only those schedules that have the word "test" in the *Item Name* to be displayed.
- **Add Schedules** button
  Add Schedules will add a new schedule to the list of existing schedules that can be synchronized. This button will take you to the Schedule Stepper. This stepper takes you through all the steps of creating a new schedule. See Designing a Schedule for more information.

Designing a Schedule

Select Item to Schedule

***Note:** Do not attempt to schedule a new task while a Data Sync scheduling task is currently running.*

From the Schedules Tab, either click the **Add Schedules** button to create a new schedule or click the **Edit** button on an existing schedule. The Schedules stepper will be displayed.



Select Item to Schedule

Schedule Name: demoTableSchedule

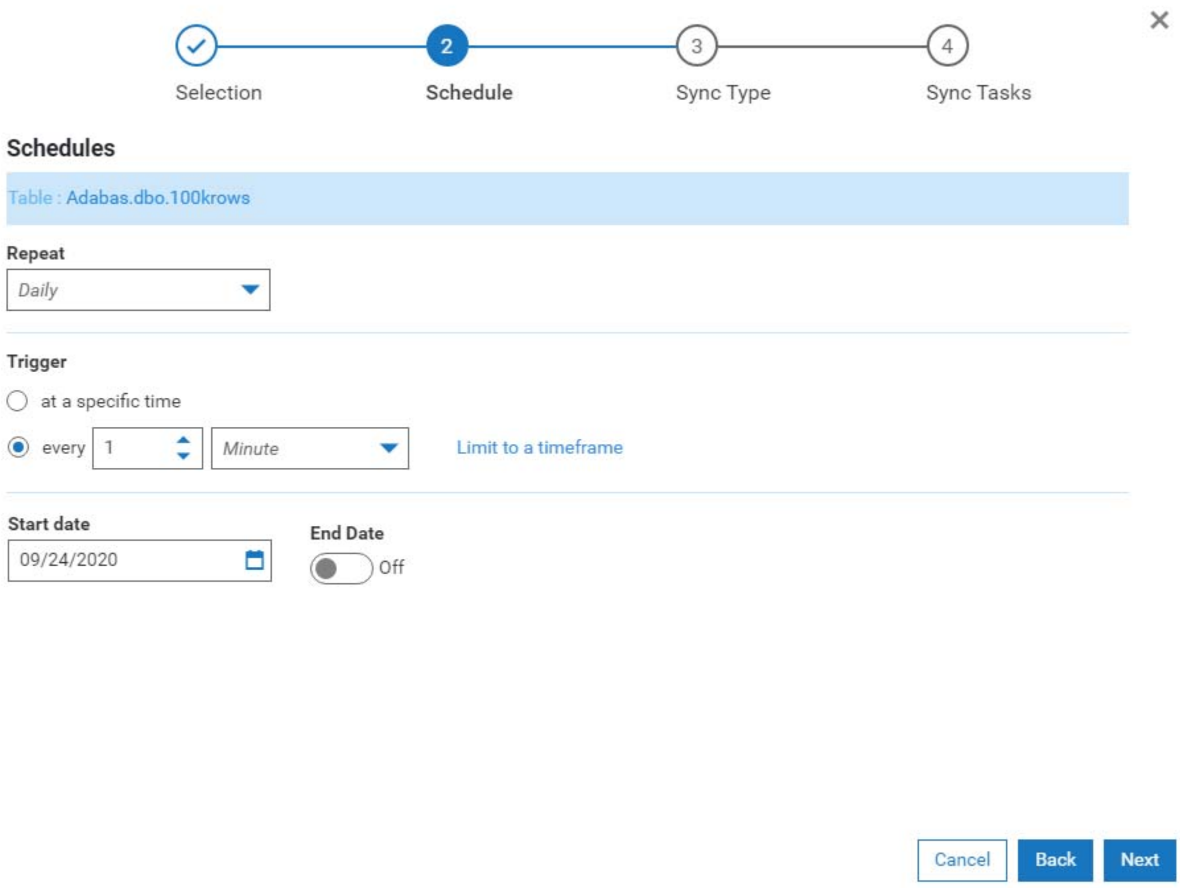
Item Name	Item Type
Adabas.dbo.100krows	Table
Adabas.dbo.250krows	Table
Adabas.dbo.EMPLOYEES	Table
Adabas.dbo.EMPLOYEES_ADDRESS_LINE	Table
Adabas.dbo.EMPLOYEES_BONUS	Table
Adabas.dbo.EMPLOYEES_INCOME	Table
Adabas.dbo.EMPLOYEES_LANG	Table

1-7 of 15 items   Items per page 7   Page 1 of 3

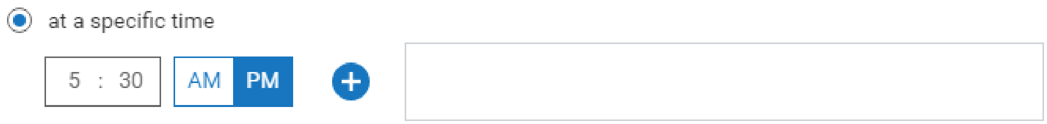
1. Enter the name for the schedule in the **Schedule Name** field.
2. Select an item to schedule from the **Item Name** list. This list contains a list of all the source tables (tables from the default target database are excluded from this list), transformations and groups. There is a search box at the top of the list that can be used to filter the list. For example, entering "employees" in the search field will filter the list so it only displays entries with the text "employees" in it. Note that the list supports pagination. You may need to move forward to another page to find the desired item.
3. Click **Next** to proceed to Step 2: Schedules

Schedules

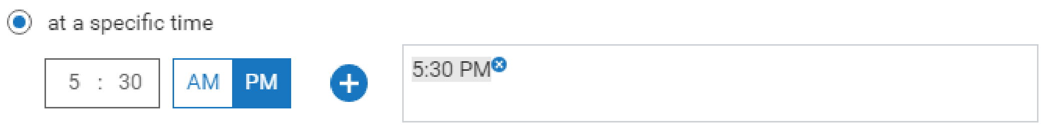
The Schedules step defines how often a scheduled synchronization will run.



- **Repeat**
  The Repeat dropdown specifies how often the task will repeat. Possible values are Daily, Weekly, Monthly and Yearly
- **Trigger**
  A schedule can trigger at a specific time or can be set to trigger every *x* Minutes or Hours
  - **Trigger at a specific time**
    To trigger a specific times, select the at a specific time radio button.



Enter a time in the time field. In this example, we have used 5:30 PM. Click the plus button to add the time to the list of specific times:



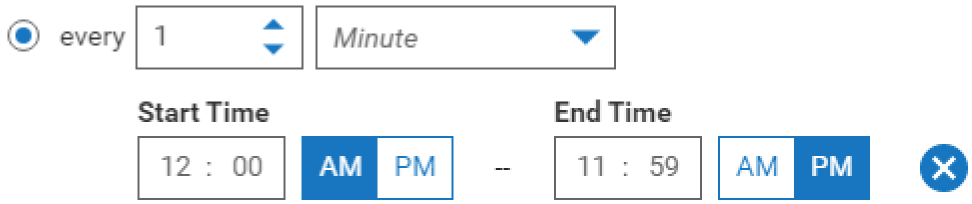
● at a specific time

5 : 30 AM PM

5:30 PM

Up to 10 specific times can be added. Specific times can be removed by clicking the blue x to the right of the time entry in the list.

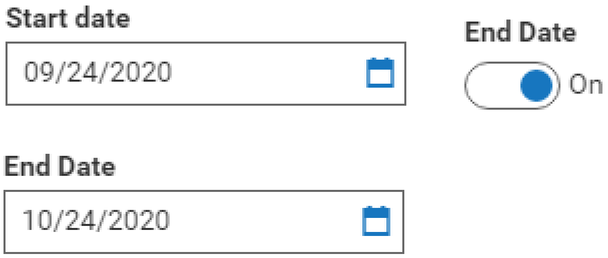
- **Trigger every x Minute/Hour**
  To trigger on a per minute or per hour bases, select the **every...** radio button. Select the desired number of minutes and the desired value of Minute or Hour from the dropdown box. Click Limit to a timeframe to specify a start and end time.



● every 1 Minute

Start Time 12 : 00 AM PM -- End Time 11 : 59 AM PM

- **Start Date**
  Use the Calendar Picker to select a Start Date for this schedule
- **End Date**
  The End Date is optional. To specify an End Date click the selection button to set it to On and select an End Date from the calendar picker.



Start date 09/24/2020

End Date On

End Date 10/24/2020

Click **Next** to proceed to Step 3: Synchronization Type

Synchronization Type

This step determines the type of synchronization that will be run with this schedule.

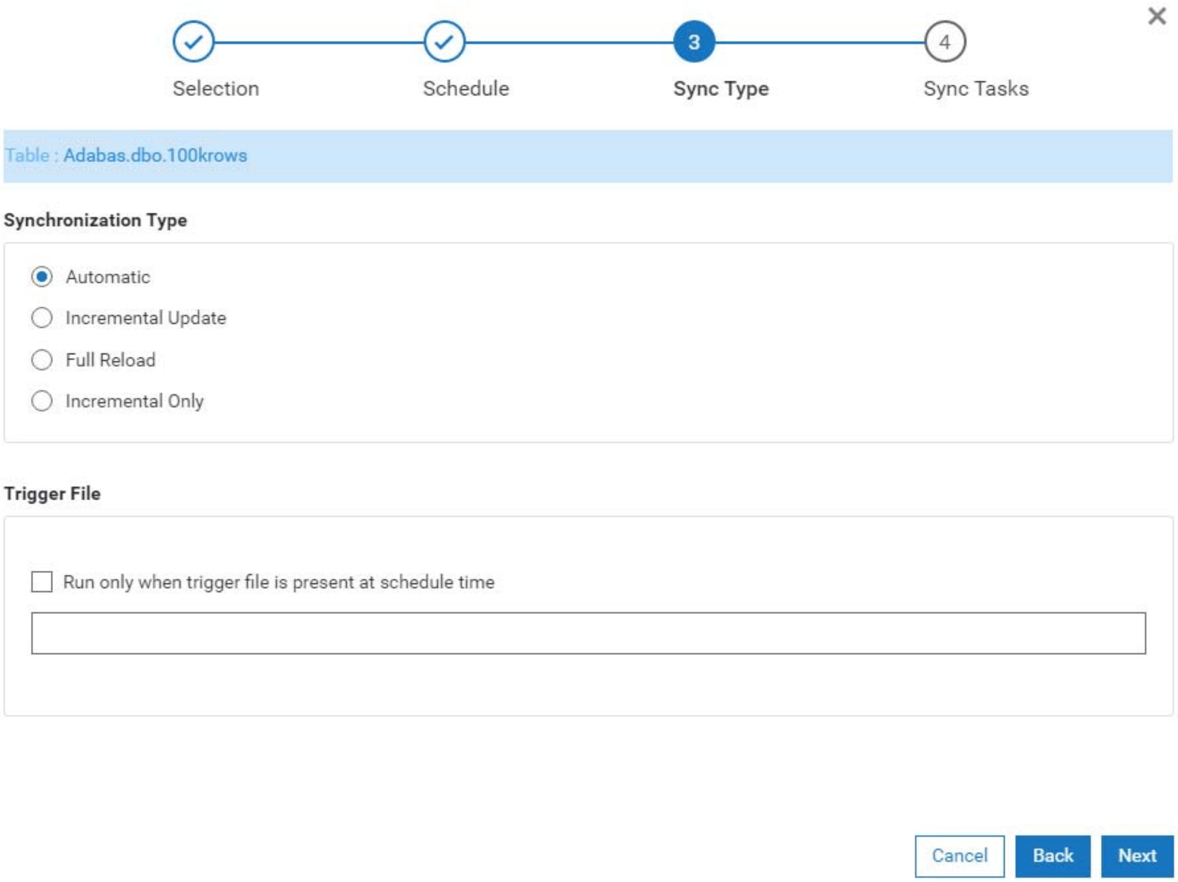


Table : Adabas.dbo.100krows

Synchronization Type

● Automatic
○ Incremental Update
○ Full Reload
○ Incremental Only

Trigger File

□ Run only when trigger file is present at schedule time

- **Synchronization Type**
  Specify Automatic, Incremental Update, Fill Reload or Incremental Only. See Methods of Synchronization for a description of each option.

- **Trigger File**
  Click **Run only when trigger file is present at schedule time** and supply the fully qualified name (i.e. c:\connx32\datasync\trigger.dat) of a trigger file in the edit box. Trigger files are created by external database processes and are designed to signal the appropriateness and timing of a synchronization. For example, a user might not want to run a scheduled synchronization until an important database backup is complete. The job that performs the database backup can delete the trigger file, perform the backup, and then create the trigger file. If the backup is still being run, the synchronization is further delayed until the trigger file is found. Once the backup is complete, the trigger file appears, and the synchronization proceeds as expected.

Click **Next** to proceed to Step 4: Synchronization Tasks

Synchronization Tasks

Tasks can be associated with individual scheduled synchronizations. The tasks defined in this step run on an individual schedule, unlike Global Scheduled tasks which run for every scheduled synchronization. Synch Tasks can be any executable file or batch script that can be called within Windows.

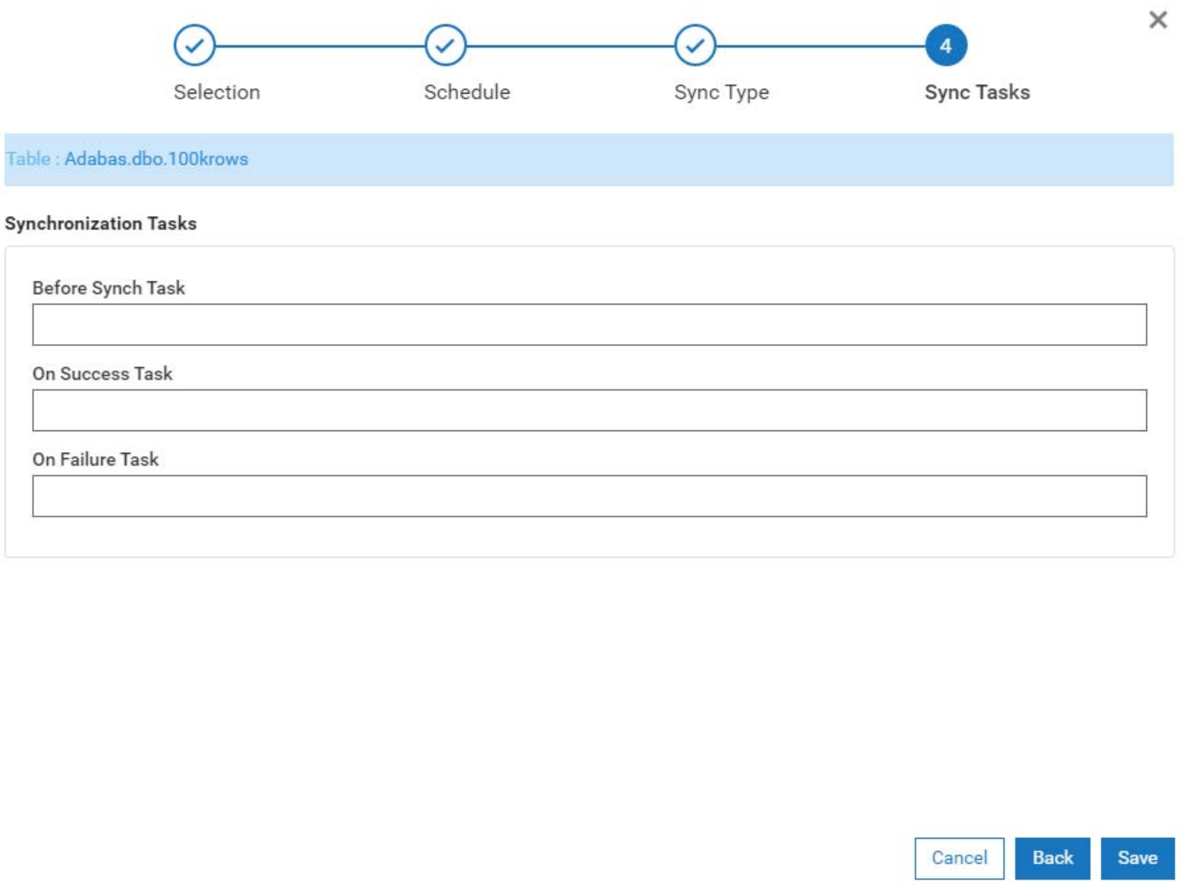


Table : Adabas.dbo.100krows

Synchronization Tasks

Before Synch Task

On Success Task

On Failure Task

- **Before Sync Task**
  To define a task to run before a synchronization is performed on the table, transformation or group specified in the schedule definition, use the Before Synch Task text box to enter a valid executable filename or batch file. For example, to synchronize a particular set of database files and then inform the administrator of that particular database of the scheduled synchronization, a command line may resemble the following: (**Note:** This is a fictitious example)

  c:\utils\epsendmail user@domain.com, databaseadmin-ops@domain.com, "DB Sync", "DB Sync Task started", 255.255.255.255

  The command can be tested with the Global On-Demand Synchronization Before Sync Task and then cut-and-paste into the Before Synch Task text box once it is working properly.

- **On Success Task**
  To define a task to run upon the successful completion of a synchronization, use the On Success Task text box to enter a valid executable filename or batch file. For example, to synchronize a particular set of database files and then inform the administrator of that particular database that the scheduled synchronization successfully completely, a command line such as the following

could be used: (**Note:** This is a fictitious example)

c:\utils\epsendmail user@domain.com, databaseadmin-ops@domain.com, "DB Sync", "DB Sync Task Completed Successfully", 255.255.255.255

The command can be tested with the Global On-Demand Synchronization On Success Task and then cut-and-paste into the On Success Task text box once it is working properly.

- **On Failure Task**
  To define a task to run when a synchronization fails with an error, use the On Failure Task text box to enter a valid executable filename or batch file. For example, to synchronize a particular set of database files and then inform the administrator of that particular database that the scheduled synchronization failed to successfully complete, a command line such as the following could be used: (**Note:** This is a fictitious example)

  c:\utils\epsendmail user@domain.com, databaseadmin-ops@domain.com, "DB Sync", "DB Sync Task Failed", 255.255.255.255

  The command can be tested with the Global On-Demand Synchronization On Failure Task and then cut-and-paste into the On Failure Task text box once it is working properly.

Click the **Save** button to save the Schedule and return to the Schedules Tab

Troubleshooting

Duplicate keys

If a target index does not build, a message about duplicate keys may appear. This may be due to case sensitivity of the source and target servers.

For example, an RMS server may read "JOHN DOE" and "John Doe" and "john doe" differently, but SQL Server may read them equally.

Although a unique index may exist on a source database, indicating that the data is unique, the server may not accept the data and declare that it is not unique.

To analyze this problem, first perform a query consisting of concatenating the key columns together and checking for counts greater than one. Such a query might show 4 duplicate rows for a table with unique key columns Name and Address, for example.

To solve the problem, correct the data on the source server. If (for instance) "John Doe", "JOHN DOE" and "john doe" are really the same person, merge the information in the three records together.

In cases where part number "aaa-2756" is, in fact, distinct from part number "AAA-2756", build SQL Server (or whatever other target server we are using) in case-sensitive mode.

If an incremental synchronization gives an error about duplicate keys, the error message provides a sample query that should return the duplicate data to aid in resolving the issue. Insert the query directly into the InfoNaut query tool to locate the duplicate data.

Error messages

Error message	Resolution
The CONNX Listener process (CNXRUN##_MAIN) is not running on the system.	The listener is not running or otherwise unavailable on the database server.
Colliding Schedules. This schedule will not be run.	You have overlapping schedules. Perhaps you have run the synchronization task on a basis that is too frequent for your Win32 synchronization server to keep up. You can tell how long (in seconds) table synchronizations take to perform by performing the following query: SELECT b.TableName, timestampdiff(SQL_TSI_SECOND, a.LastSynchStart, a.LastSynchEnd) AS "seconds", a.SynchType, a.InsertedRowCount, a.UpdatedRowCount, a.DeletedRowCount , a.Cardinality FROM CONNXDataSync..TableSynchronizations a, CONNXDataSync..TableList b WHERE a.TableID = b.TableID ORDER BY timestampdiff(SQL_TSI_SECOND, a.LastSynchStart, a.LastSynchEnd) desc You can tell how long (in seconds) groups take to synchronize with the following query: SELECT b.GroupName, timestampdiff(SQL_TSI_SECOND, a.LastSynchStart, a.LastSynchEnd) AS "seconds", a.SynchType, a.InsertedRowCount, a.UpdatedRowCount, a.DeletedRowCount , a.Cardinality FROM CONNXDataSync..GroupSynchronizations a, CONNXDataSync..Groups b WHERE a.GroupID = b.GroupID ORDER BY timestampdiff(SQL_TSI_SECOND, a.LastSynchStart, a.LastSynchEnd) desc If you do not require a complete history, you can simply look at the Tables and Groups tabs to see about how long the last synchronization took. By taking a look at how long the tasks take, you can decide how frequently you might schedule them. If a file takes 300 seconds to synchronize, then you would not want to schedule synchronizations more frequently than perhaps 10 minutes (twice the sync rate). If a group takes 4000 seconds to synchronize then one hour is too frequent, since that is only 3600 seconds.

Probably out of disk space.	Perform some disk housekeeping or purchase a larger disk drive.
Unable to start task (the paging file is too small for this operation to complete).	Create a larger page file on your system.

Index