

# Sage CRM Connector for Sage X3 2.2

## Installation Guide

**Revision:** CON-INS-ENG-2-10.0  
**Updated:** February 2017

© 2016, The Sage Group plc or its licensors. Sage, Sage logos, and Sage product and service names mentioned herein are the trademarks of The Sage Group plc or its licensors. All other trademarks are the property of their respective owners.

# Contents

<b>Contents</b>	<b>3</b>
<b>About this Guide</b>	<b>5</b>
<b>Introduction to the Sage CRM Connector</b>	<b>6</b>
About the Connector	6
Connector Framework architecture	7
Web Server and Gateway	7
Gateway connections to Sage X3 objects	8
Sage X3 objects and web services	8
<b>Installing the Connector</b>	<b>9</b>
Installation prerequisites	9
Configuring a web service pool in Sage X3 Product Update 9 and above	9
Configuring a web service pool in Sage X3 Product Update 8 and prior	10
Installation steps	11
Installing the Gateway on a different server	11
<b>Configuring the Connector</b>	<b>12</b>
Configuring web services	12
Specifying web services access credentials	13
Testing the Gateway	14
Defining web services	14
Adding a timestamp field	16
Testing the web services connection	17
<b>Mapping Data</b>	<b>18</b>
Data entry considerations	18
Country and Category Code	18
Contacts	18
Person address	19
Sales representatives	19
Phone numbers	20
Date format for quotes and orders	20
Configuration files	21
PostToX3.asp	21
FieldMappings.inc	22
<b>Synchronizing Data</b>	<b>23</b>
Account synchronization process	23

Accelerated performance mode	23
Superior performance mode	24
Standard performance mode	24
Running a synchronization	25
Synchronizing from the UI	25
Synchronizing from the command line	25
<b>Sage X3 data in Sage CRM</b>	<b>27</b>
Synchronized data	27
Real-time data	27
Sage CRM Company record	27
<b>Customizing Connections and Entities</b>	<b>29</b>
Customizing connections	29
X3Data.asp	29
Customizing entities	29
<b>Uninstalling the Connector</b>	<b>30</b>
Uninstall script	30
<b>Troubleshooting</b>	<b>32</b>
Troubleshooting resources	32
Timestamp and daylight savings time	32
Update time conversion to UTC for Sage X3 version 6.5	32
WsOrder web service	33
Display X3 Accounts button	34
TSC code	34
Sage X3 version number	35
<b>Appendix: Sage X3 Definitions</b>	<b>36</b>
Screen definitions	36
List definitions	39

# About this Guide

This guide is written for system administrators who know how to install and configure both Sage CRM and Sage X3.

It includes step by step instructions for configuring Sage CRM and Sage X3 to work with the Sage CRM Connector. The instructions for Sage CRM assume that your system is using the Contemporary theme.

For additional information about configuring and working with these systems, see the following resources.

- [Sage CRM Knowledgebase](#)
- [Sage CRM System Administrator Help](#)
- [Sage CRM User Help](#)
- [Sage X3 Knowledgebase](#) (you must be logged in to the Sage X3 Partner Portal to access Knowledgebase articles)
- [Sage X3 Product Help](#) (you must be logged in to Sage X3 to access Help)

**Note:** Read the latest Sage CRM and Sage X3 updates and release notes before installing the Connector

# Introduction to the Sage CRM Connector

## About the Connector

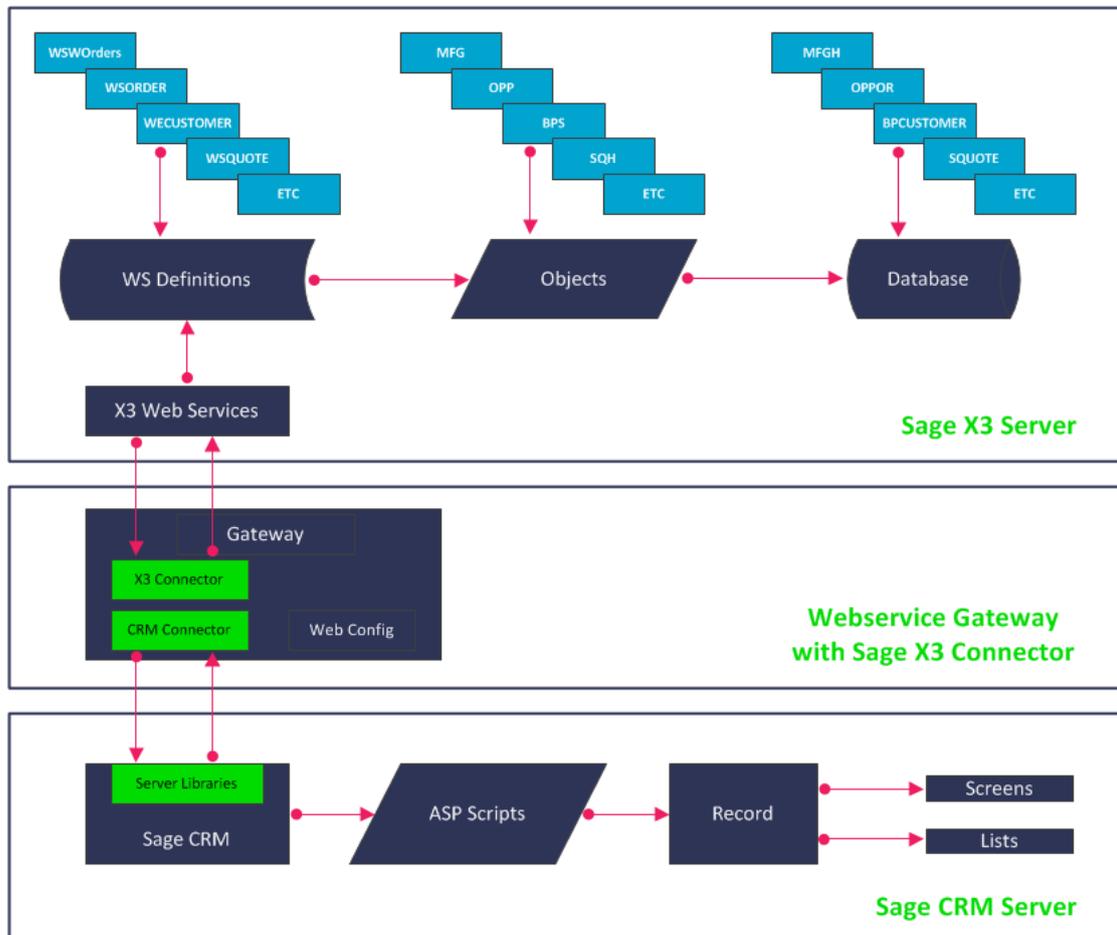
Sage provides scalable Accounting and Business Management solutions that deliver lasting benefits and reduced cost of ownership for every type of business.

- Sage CRM software helps businesses improve customer relationships by organizing and automating communications and activities across all customer-facing departments including sales, marketing and customer service. For more information about Sage CRM, visit [www.sagecrm.com](http://www.sagecrm.com).
- Sage X3 is a global accounting solution that manages domestic and international businesses across distribution, manufacturing, and services industries with built-in functionality, while adjusting to your unique company rules and processes. For more information about Sage X3, visit [www.sagex3.com](http://www.sagex3.com).

The Sage CRM Connector is a flexible, open-ended infrastructure applied on top of Sage CRM that makes a full read and write connection between Sage CRM and Sage X3. Synchronizing the data in these systems provides a holistic view across your business. Because everyone in your business can access complete information for each customer, they can make better decisions and identify the next best actions.

The Connector includes ready-built libraries and functions that enable access to Sage X3 data through all Sage CRM interfaces, including multiple browsers and mobile options. It updates information between Sage CRM and Sage X3 while ensuring validation rules are maintained. Sage X3 object metadata is added to Sage CRM so you can customize fields and screens in Sage CRM.

## Connector Framework architecture



The Connector Framework consists of

- The Gateway that includes a series of connectors.
- Server libraries that are installed on Sage CRM.
- The Sage X3 web services layers.

### Web Server and Gateway

In a standard installation, the Connector is installed on the Sage CRM server. Two virtual directories are created on the Web Server during installation of the Connector.

- The standard Gateway service routes data between Sage CRM and Sage X3 and transforms messages so that each system can understand them.
- The Demo web service runs iterative loops of requests on the Gateway to test Sage CRM based requests.

These services are set as applications and use a common application pool called the EnbuGatewayAppPool. So although it's installed on the same server, the Gateway uses different memory than Sage CRM. This helps to spread resources and CPU so users aren't affected by high-end bulk operations. However, if you require even greater load balancing, you can move the Gateway to another server during installation. For more information, see [Installing the Gateway on a different server](#).

## Gateway connections to Sage X3 objects

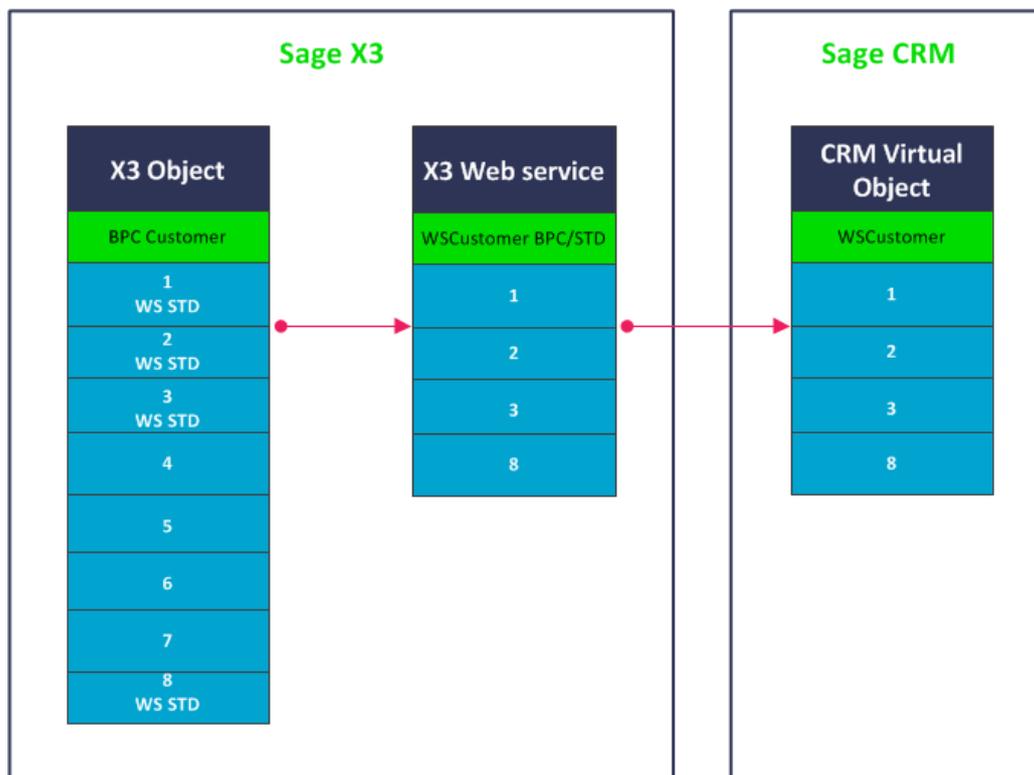
The Connector creates pre-configured connections on the Gateway that connect to Sage X3 Customer Account, Invoice, Quote, and Order objects. You can set up additional connections if required. For more information, see [Defining web services](#) and [Testing the web services connection](#). All data sent between Sage X3 and Sage CRM flows through the web services and business object layer, ensuring all rules are captured and any errors reported back to the user.

Sage X3 Customer Account records are mapped to Sage CRM Company records. During an initial data migration, new fields are added to Company screens to display Customer Account data. The data is saved in Company tables in the Sage CRM database.

Subsequent synchronizations ensure that the data values in both the Sage X3 and the Sage CRM databases are the same.

The Connector Framework also copies metadata from other Sage X3 objects and saves it in the Sage CRM database as virtual tables. The metadata enables you to customize how the Sage X3 objects are displayed in Sage CRM. However, the Sage X3 data remains on the Sage X3 database and a real-time view is displayed in Sage CRM. The Connector Framework enables you to edit this real-time data and post it back to Sage X3 to store it.

## Sage X3 objects and web services



You must manually [define Sage X3 web services](#) that correspond with the Gateway connections. The web services pull data from Sage X3 objects depending on the object fields marked for inclusion and the transaction code used. This code specifies what object data is displayed, how it's displayed, and how it behaves.

A corresponding virtual object is created in Sage CRM for each Sage X3 object that's referenced by a web service. A physical table is created in the Sage CRM database so you can customize its fields, screens, and lists in Sage CRM but the data remains in the Sage X3 database and is pulled into Sage CRM in real-time.

# Installing the Connector

## Installation prerequisites

To install the Connector, you need the following.

- Sage CRM version 7.3, 7.3 SP1, 7.3 SP2, or 7.3 SP3. For software requirements, see the [Sage CRM Software Support Matrix](#).
- Sage X3 version 6.5, version 7, Product Update 8, or Product Update 9.
- ASP.NET 4.5 settings configured and registered on the CRM server. For more information, see the [Sage CRM Software Support Matrix](#).
- One available web service folder for Sage X3. You may need more depending on performance requirements.
- Sage X3 Web Server version 131 or later.
- The following information.
  - Sage CRM solution name (install name)
  - Sage CRM server name
  - Sage CRM database username and password
  - Sage X3 Web Server name or IP address and port
  - Sage X3 web service pool name
  - Sage X3 username and password to connect using web services

## Configuring a web service pool in Sage X3 Product Update 9 and above

If a web services pool is not set up or configured on Sage X3, you must configure it before installing the connector.

1. Open the **Sage X3 Classic SOAP pools configuration** task which appears on the Administration menu in X3.
2. Select **Create soapClassicPool**.
3. Enter a value for the following fields.
  - **Alias**: Name of the pool.
  - **Initialization Size**: The default value is 1.
  - **Maximum Size**: The maximum number of web services license used with the pool.
  - **Endpoint**: Click the lookup to see a list of defined endpoints.
  - **Locale**: Click the lookup to choose the correct locale.
  - **User**: The global user defined in **Administration | Users**. It is not recommended to use the default ADMIN user. **Note**: the password for this global user (entered in

Administration | Users) must be entered in the web config file. See [Specifying web services access credentials](#) for details.

### Configuring a web service pool in Sage X3 Product Update 8 and prior

If a web services pool is not set up or configured on Sage X3, you must configure it before installing the connector.

1. Open the Sage Safe X3 management console and click **web servers**.
2. Click **Webservices** at the top of the screen, expand the application folder (for example, SEED), and confirm there are no defined pools.
3. Click the cog icon  and click **Add**.
4. Enter a value for the following fields.
  - **Alias:** Name or code of pool.
  - **Language:** Specify the default language.
  - **Userid:** The X3 user defined in Sage X3 in **Setup | Users**.
  - **Password:** The X3 user's password.
  - **Ouserid:** The Windows OS username used to login in to Sage X3.
  - **Ospassword:** The Windows OS user's password.
  - **Maxsize:** The maximum number of web services licenses used with the pool.
  - **Initsize:** The default is 1.
  - **Autostart:** Recommended to be set to On.
  - **Reserved:** Defaults to On.

## Installation steps

1. Navigate to where you've saved **SageCRMConnector.exe** and double click the file to start the installation process.
2. Click **Next** to proceed through the installation wizard.
3. Choose **CRM** from **Install Action**.
4. Choose the Sage CRM installation that the Connector will be installed on.
  - Ensure the server name that hosts the Sage CRM database is correct. This value is populated by default and you should change it only for specific reasons.
  - Ensure the Sage CRM database name used by the existing Sage CRM installation is correct. This value is populated by default and you should change it only for specific reasons.
5. Use Windows Authentication to connect to the database if you're logged in as an authenticated user. Alternatively, enter a specific SQL administrator user ID and password.
6. Perform an IIS restart on the server. Make sure you notify everyone that uses software and services on the server that you are going to restart. Files, data, and configurations are copied to the Sage CRM server.
7. Click **Yes** to confirm the creation of the [virtual directories on the Web Server](#) and then click **Finish**. The **Sage CRM Connector** option is available in the Administration menu in Sage CRM.

## Installing the Gateway on a different server

You can move the Gateway to a different server during installation. You might do this to increase load balancing.

1. Ensure IIS and ASP.NET are installed on the server on which you'll install the Gateway. For more information, see [Installation prerequisites](#).
2. Install the Connector as a Gateway on the server. On the **Install Options** window, select **Sage X3** from **Finance Solution** and select **Gateway** from **Install Action**. For more information, see [Installation steps](#).
3. On the Gateway server, click **Enbu | Framework | Server | EnbuGateway | web.config**.
4. Set **CRM Service Broker URL** to the IP address of the CRM server.
5. Install the Connector on the server running Sage CRM. For more information, see [Installation steps](#). On the **Install Options** window, select **Sage X3** from **Finance Solution** and select **CRM** from **Install Action**.
6. When the Connector is successfully installed, go to Sage CRM and click **<My Profile> | Administration | Enbu Products Configuration**.
7. Click **Change**.
8. Enter the IP address of the Gateway server in **Gateway URL** and click **Save**.

# Configuring the Connector

## Configuring web services

To import Sage X3 data into Sage CRM and ensure subsequent data posts and synchronizations work correctly, you must configure web services.

1. Specify access credentials to Sage X3 web services.
2. In Sage X3, manually define web services that correspond with the connections on the Gateway. You must define a web service for each of the Customer, ITS, Quote, and Order objects that specifies which object fields to include and the transaction code to use. Not all objects require a transaction code; Quote and Order do require a code. We recommend you use standard entry transaction (STD) as it works well out-of-the-box. However, if you configure STD, be aware that it might not work as expected. If necessary, create a new web service connection. This allows you to extend the Framework in order to access other Sage X3 objects.
3. In Sage X3, add a timestamp field to the BPC object to enable the recommended synchronization process mode.
4. In Sage CRM, connect to the Sage X3 web services and test each connection using a sample record. The sample record that's returned should include every field available from Sage X3 (depending on the configuration of the object that the web service points to, the transaction code used, and the web service) and Sage X3 hierarchical data. If a field that you were expecting isn't returned, you may need to change the sample record you are using, reconfigure the object, or perhaps change the transaction code used. For more information, see the Sage X3 Product Help (you must be logged in to Sage X3 to access Help). When you're happy with the returned data, create an entity in Sage CRM that reflects this Sage X3 object schema.

## Specifying web services access credentials

1. Ensure web services are enabled on Sage X3 and that the web services URL is exposed.
2. Open the Sage Safe X3 Management Console and click **web servers**.
3. Click **Webservices** at the top of the screen, expand the Sage X3 application folder and click each web service in turn. Note the values for the Sage X3 pool, language, user ID, and password. The settings are described in the table below.
4. Click **CRM | WWWRoot | CustomPages | Enbu | Framework | Server | EnbuGateway**.
5. Open the web.config file and specify the values for the Sage X3 pool, language, user ID, and password. For Sage X3 Product Update 9 and above, the user credentials are the global user and password, which are defined in the **Sage X3 Administration | Users** task. For Sage X3 Product Update 8 and prior, specify the X3 user's credentials entered in **Setup | Users**.
6. Configure the Gateway to point to the correct Sage X3 web services URL. Ensure you can access the URL from the server that hosts the connector.
  - For Sage X3 Product Update 9 and later:  
<http://<servername>:8124/soap-generic/syracuse/collaboration/syracuse/CAdxWebServiceXmlCC>
  - For Sage X3 Product Update 8 and prior:  
<http://<servername>:28880/adxwsvc/services/CAdxWebServiceXmlCC>

**Note:** To test the URL, paste it into your browser. The U9 version of the URL should present an X3 login prompt, while the U8 version of the URL should return a message in the browser window that reads, "Hi there, this is an AXIS service!"

*Servername* is the name of the Sage X3 server. If Sage X3 is on the same server as Sage CRM, you can use *localhost* as the server name. Test that you can open the URL using a browser on the Sage CRM server.

Setting	Description
Pool	The name of the web service pool.
Language	The language in which the web service pool delivers data and messaging to a web service client.
Userld	The Sage X3 username that the web service pool uses to log in to Sage X3. To ensure this user is web service enabled in Sage X3, click <b>Setup   Users   Users   &lt;user&gt;</b> and select <b>Web services connection</b> .
Password	The Sage X3 user's password.

## Testing the Gateway

1. Click **<My Profile> | Administration | Sage CRM Connector | Tools**.
2. Click **Check Gateway WebService** to access the Gateway directly and ensure that the service is running.
3. Click each service method hyperlink to invoke the method and analyze the returned values.
4. There are several other utilities on this screen but they are not used for the Sage CRM Connector.
  - The Demo web service illustrates how to integrate a third-party web service into the Connector Framework.
  - Demonstration tests use the test harness.
  - The Gateway Monitor displays real-time Gateway transactional information.
  - The CRM Broker tests how Sage CRM communicates with an external system.

## Defining web services

1. In Sage X3, click **Development | Script dictionary | Scripts | Web services**. Alternatively, call the function directly using GESAWWE. For more information, see the Sage X3 User Guide.
2. Define the required web services using the information in the table below. These are WSCUSTOMER, WSITS, WSQUOTE, WSORDER, WSFCY and WSREP.
3. Click **Create** and then **Publication**. A publication date is displayed.
4. If changes are made to a Sage X3 table that's used by the connector, you must delete the WS Connection in Sage CRM and then recreate it for that table. To do this:
  - Navigate to  
C:\Program Files  
(x86)\Sage\CRM\CRM\WWWRoot\CustomPages\Enbu\Framework\Server\EnbuGateway\Schema and delete the applicable schema. For example, if the BPCustomer table is modified in Sage X3, delete the WSCUSTOMER.XML file.
  - Recreate the applicable schema file by following the steps in [Testing the web service connection](#).

Web service name	Type	Linked X3 object	Transaction code
WSCUSTOMER (required)	Object	BPC	
WSITS (required)	Object	ITS Sales Products	
WSQUOTE (required)	Object	SQH	STD (default)
WSORDER (required)	Object	SOH	STD (default)
WSFCY (required)	Object	FCY Site lookups	
WSPROJECT (demo use only)	Object	OPP	
WSWORDER (demo use only)	Object	MFG	
WSREP (required)	Object	REP (sync Sage X3 rep codes with Sage CRM users)	

You can create additional web service connections as needed. In Sage X3, click **new** and define the web service. Then create and test the web service connection in Sage CRM. For more information, see [Testing the web services connection](#).

## Adding a timestamp field

1. In Sage X3, click **Development | Data and parameters | Tables | Tables**, open the BPCustomer table and click the **Columns** tab.
  - Verify that the UPDDATTIM field has not been renamed and that the data type is ADATIM.
2. Click **Development | Data and Parameters | Tables | Tables**, open the **BPCUSTOMER** table and click the **Columns** tab.
  - Add the ZUPDTIMTXT field with Type = A, Length = 30, and click **Save**.
  - Click **Validation** and then click **OK**.
3. Click **Development | Scripting | Screens**, open the BPC0 screen and click the **Fields** tab.
  - Add a field called ZUPDTIMTXT with Block = 1, Position = 6, Type = A, Length = 30, Display the field, and click **Save**.
  - Click **Validation** and then click **OK**.
4. Click **Development | Script dictionary | Scripts | Script Editor**.
  - Choose **X3** from **Folder**, enter SUBBPCA in **File name** and press **Enter** on the keyboard.
  - Use **CTRL+F** to navigate to the \$LIENS section and add the following line:  
[M:BPC0]ZUPDTIMTXT = num\$([F:BPC]UPDDATTIM)  
to the section below the line:  
Local Decimal WMONTDES (0..10)
  - Click **Compile** and then click **Close Page**.
5. Click **Development | Script dictionary | Scripts | Objects**, open the **BPC** object and click the **Selection** tab.
  - Add the ZUPDTIMTXT field and click **Save**.
  - Click **Validation** and then click **OK**.
6. Modify or recreate the WsCustomer web service and then republish it. See [Defining web services](#).
7. Go to **WWWRoot | CustomPages | Enbu | Framework | X3** and open AccountMigration.asp and AccountMigrationService.asp in Notepad.
8. Set **sX3SyncDateField** to the timestamp field name for the WsCustomer connection in both files. To check the field name in Sage CRM, click **<My Profile> | Administration | Customization | WsCustomer**.

## Testing the web services connection

1. In Sage CRM, click **<My Profile> | Administration | Sage CRM Connector | WS Connections**. Sage X3 web services are prefixed with X3. The MasterAccount connection is a sample connection that you can ignore. To add a new connection, click **New** and name the new connection *X3<new web service definition>*.
2. Click a connection hyperlink.
3. Click **Change**.
4. Specify a sample record for the connection. To do this, enter an existing Sage X3 ID value in **Gateway Method Test Params**. The values are case sensitive.
5. Click **Save**.
6. Click **Test Service** to test the connection to Sage X3. It might take a few minutes to return the data.
7. Verify that the returned data is correct.
8. When using the Test Service task, a schema file for the current web service is created if it does not already exist. If the Test Service fails, the schema file can contain errors. If this occurs, you must correct the condition that caused the Test Service to fail and delete the applicable schema file (for example, WSCUSTOMER.XML). When the file is deleted, the Test Service and Create Table functions can be performed. This step is also required to pull in changes made to Sage X3 tables and screens.
9. Click **Create Table** to create an entity in Sage CRM that reflects the Sage X3 object schema.
  - Use the default value in **Metadata table name**.
  - Click **Confirm Create Table**. The new entity is created in Sage CRM with field captions, drop-down lists, screens, and lists.

Follow these steps for each web service connection to ensure schema changes in Sage X3 are copied to Sage CRM. Rerunning the Create Table operation doesn't disrupt data, screens or lists. However, the operation overwrites any field captions that you've manually configured.

# Mapping Data

## Data entry considerations

To ensure the data can be synchronized seamlessly between Sage CRM and Sage X3, you should configure data entry settings in each system. How you configure the settings depends on the implementation and which system already contains data.

Sage CRM Company records map to Sage X3 Customer Account records.

### Country and Category Code

When you create a Customer Account record directly in Sage X3, you must enter a value for Country.

When you create a Company record in Sage CRM, **Country** is not a required field but it's recommended that you populate it. If you leave the value of Country blank in a Company record and then post it to Sage X3, a Customer Account is created with Country set to the default country that's set for Sage X3. The value of Country impacts how address and phone numbers are formatted.

When you create a Company record, you must enter a Category Code in order to post the record to Sage X3. The Company Category Code maps to the Customer Account Category Code.

**Note:** You can also hard code the default Category in PostToX3.ASP. For more information, see [Configuration files](#).

In a Company record, the value you set in Category Code overwrites the value in Country.

### Contacts

When you create a Company record in Sage CRM, contact information is required. When you create a Customer Account record in Sage X3, contact information is not required. However, Sage X3 Customer Account records without contacts are skipped in the synchronization and a notice of the skipped records is written to the migration log. For this reason, we recommend that Sage X3 Customer Account records contain at least one Contact.

## Person address

There are two ways to add an address when creating a new Person record in Sage CRM. The user selects a radio button in the Address section to specify how the address is saved.

- If the user selects **Person address only**, a new address is created for the Person. This is true even if the address is the same as the associated Company record address.
- If the user selects **Person and company address**, a new address is created for the Person if the address differs from the associated Company record address. If the address is the same as the associated Company record address, no new address is created.

## Sales representatives

You can map the sales representative on a Customer Account record to the assigned user on associated Company records.

1. In Sage X3, open a Customer Account record and click the **Commercial** tab. Note the code in **Sales rep**.
2. In Sage CRM, click **<My Profile> | Administration | Users | Users**.
3. Enter the name of the user you want to map to and click **Find**.
4. Click the name hyperlink.
5. Click **Change**.
6. Scroll to the **More User Details** panel and enter the Sales rep code from Sage X3 in **X3 User ID**.
7. Click **Save**. The user appears as the Account Manager on all associated Company records.

## Phone numbers

In Sage CRM, phone numbers are free text fields that allow a variety of formats and values, but in Sage X3, phone numbers follow a strict structure and format through regionalization settings and customer categories. As a result, phone mapping between Sage CRM and Sage X3 is disabled by default. To enable phone mapping, you must change the web.config and system behavior as described below. Additionally, to maintain data integrity, Sage CRM phone numbers should consist of numbers only and meet the length requirements set up in Sage X3 country configuration.

1. Click **CRM | WWWRoot | CustomPages | Enbu | Framework | Server | EnbuGateway | web.config**.
  - Set X3 Phone Updates to **True**.
2. In Sage CRM, click **<My Profile> | Administration | System | System Behavior**.
  - Click **Change**.
  - Set **Use country code** to **No**.
  - Set **Use area code** to **No**.
  - Click **Save**.

## Date format for quotes and orders

The date format for Sage X3 quote and order line items is set independently to Sage CRM. If you change the date format of Sage CRM, you need to change the date format of the line items.

1. In Sage CRM, click **<My Profile> | Administration | Customization | Secondary Entities | WsOrder** or **WsQuote**.
2. Click the **Lists** tab.
3. Click **X3OrderItemList** or **X3QuoteItemList** under **List Name**.
4. Choose a date field from **Desktop HTML List Contents**. For example, **WsOrder: Requested delivery date**.
5. In **CreateScript** add the new format. For example:  
DefaultValue = (new Date()).format("mm/dd/yyyy "); // X3 specific
6. Click **Update** and then click **Save**.
7. Repeat these steps for other line item date fields, such as **WsOrder: Ship Date** and **WsOrder: Expected Delivery Date**.

## Configuration files

Files that define data mapping between Sage CRM to Sage X3 are located in **CRM | WWWRoot | CustomPages | Enbu | Framework | X3**.

### PostToX3.asp

```
// Mandatory fields which do not map to CRM
// Please revise all values below to suit reasonable defaults for X3 accounts or
// alternatively program accordingly
// NB : X3 will give back various errors as it will validate the required fields
// based on the specific required configurations on the specific X3 environment.

X3Company.Records("bpc0_1_bpcsta") = 2;
var sX3Category = compRecord.comp_x3category;

// if the customer category is specified, allow it to automate the defaults
if (Defined(sX3Category) && sX3Category!=''){
    .....
    X3Company.Records("bpc0_1_bcgcod") = sX3Category;           // customer category
    // depending on the customer category used, you may or may not need to specify defaults for
    // the following fields. If not, then simply comment out the field listed here and
    // allow the customer category manage these settings
    X3Company.Records("bpc3_2_vacbpr") = "TAX";                 // tax rule
    X3Company.Records("bpc3_3_pte") = "CASH";                   // payment terms
} else {
    // in this case, no category specified, so create defaults for all required fields
    X3Company.Records("bpc0_1_bcgcod") = "US";                 // customer category
    X3Company.Records("bpc3_2_vacbpr") = "TAX";                 // tax rule
    X3Company.Records("bpc3_3_pte") = "CASH";                   // payment terms
    X3Company.Records("bprc_2_fcy") = "";                       // shipment site
}
```

PostToX3 is a script that posts a Company record to Sage X3 and creates a new Customer Account record. It lists all fields that must be mapped to the Customer Account record. It uses the **X3 Category** on the Company summary screen to default several fields. You should edit the Tax Code and Payment Terms fields for the specific Sage X3 installation. See comments in the file for more information. You should review this file before you post companies to Sage X3.

## FieldMappings.inc

```
function RunFieldMappings(compRecord, persRecord, addrRecord, phonRecord, emaiRecord, customerRec, X3isM
// two way updates based on isMaster flag

if (X3isMaster) {
    compRecord.comp_type = 'Customer';
    updateByMaster(compRecord, "comp_x3_creditlevel", customerRec, "bpc2_2_xost", X3isMaster);
    // alternative fields to look for value (can be found when in list only mode)
    if (isEmpty(compRecord.comp_x3_creditlevel))
        updateByMaster(compRecord, "comp_x3_creditlevel", customerRec, "BPC0_1_ostauz", X3isMaster);

    updateByMaster(compRecord, "comp_name", customerRec, "bpc0_1_bpcnam", X3isMaster);
} else {
    var sCompLong = (new String(compRecord.comp_name)).substring(0,34); // 35 char max
    updateByValue(customerRec, "bpc0_1_bpcnam", sCompLong);
    updateByValue(customerRec, "bpc4_1_bpdnam0", sCompLong);
    updateByValue(customerRec, "bprc_1_bprnam", sCompLong);

    // needed for v7 patch 5 - contact full name not being filled in automatically
    var sFullName = (new String(persRecord.pers_lastname + "\\ " + persRecord.pers_firstname)).toUpper
    // note : this should be disabled for X3 v6, only needed for v7
    updateByValue(customerRec, "cntc_1_cntfulnam", sFullName);
}

updateByMaster(compRecord, "comp_website", customerRec, "bpac_1_fcyweb", X3isMaster);
updateByMaster(compRecord, "comp_primaryUserId", customerRec, "bpc1_3_rep", X3isMaster);
updateByMaster(persRecord, "pers_firstname", customerRec, "cntc_1_cntfna", X3isMaster);
updateByMaster(persRecord, "pers_lastname", customerRec, "cntc_1_cntlna", X3isMaster);
persTitleCode(customerRec, persRecord, X3isMaster);
persSalutation(customerRec, persRecord, X3isMaster);

// company phone and email - note phone numbers only updated on the CRM side as X3 has format rules
if (X3isMaster) updateByMaster(phonRecord, "phon_number", customerRec, "bpac_1_tell", X3isMaster);
updateByMaster(emaiRecord, "emai_emailAddress", customerRec, "bpac_1_web1", X3isMaster);
```

FieldMappings is a function definition that specifies field mappings between Sage CRM and Sage X3 records for posting, migration and synchronization. You can hardcode values for the records. The X3isMaster flag uses timestamps to determine whether the Sage CRM or Sage X3 record is the most up to date. The most up to date record becomes the record that the other record is updated from.

# Synchronizing Data

## Account synchronization process

The Account Synchronization process can run in three modes depending on the level of configuration on Sage X3.

### Accelerated performance mode

This is the recommended way to synchronize data.

In this mode, a number of calls to the potentially heavy Account Retrieval process can be pre-empted by [adding the Sage X3 timestamp field](#) of the Customer Account to the BPC object in Sage X3. This mode automatically runs when the timestamp field is detected in the Account Listing.

In this mode, a number of Customer Account calls can be removed when the Sage X3 Account Updated timestamp is older than the last Synchronization Date of the last Account Synchronization call.

Assuming there are **D** Accounts which are already up to date from the last synchronization, the duration of a full Account Synchronization could be calculated on a worst case basis as:

$((C-D)*R) + (38 \times A)/60 =$  Number of minutes for complete Account Migration

**C** number of Accounts

**A** seconds to retrieve an Account Listing

**R** seconds to make a full Account Retrieval call

```
// UPDSTP is the full timestamp and is most reliable, "UPDDAT" will have a date only string format which  
// has no time segment, less accurate. However it seems that UPDSTP is often not available on the versio  
sX3SyncDateField = "bpc4_1_upddat";
```

## Superior performance mode

This performs a one-way migration of data from Sage X3 to Sage CRM.

In this mode, the efficiency of an Account Migration can be dramatically increased. No Account Retrieval calls are made because all data required for an Account Synchronization is kept directly on the BPC object in Sage X3. New fields must be added to the BPC object based on what's required in the FieldMapping library for the Account Synchronization process. The following are the default set of required fields.

XOST	CNTLNA	CTY
BPCNAM	CNTNAM	POSCOD
BPRNAM	ADDLIG1	CRY
FCYWEB	BPAADDLIG	TEL1
REP	ADDLIG2	WEB1
CNTFNA	ADDLIG3	

When you add each of these fields to the BPC object, a full listing is available in the Sage X3 Account Listing. For information about adding fields, contact your Sage X3 partner.

In this mode, the duration of a full Account Migration could be calculated on a worst case basis as:

$(38 \times A)/60 =$  Number of minutes for complete Account Migration

**A** seconds to retrieve an Account Listing

You must ensure the process runs as a List Mode, one-way option from Sage X3 to Sage CRM only. Go to **WWWRoot | CustomPages | Enbu | Framework | X3**, open AccountMigration.asp in Notepad and set the listMode flag to true.

## Standard performance mode

This is the standard process mode when no changes have been made to Sage X3. For each loop of the synchronization process, a call is made to the Account Listing request on Sage X3 and for each Customer Account found, a full Account Retrieval is made from Sage X3. Depending on the general scalability of the Sage CRM server, the Sage X3 server and the volume of data involved, this process can be lengthy when run in a high volume environment.

The duration of a full Account Synchronization could be calculated on a worst case basis as:

$((C \times R) + (38 \times A))/60 =$  Number of minutes for complete Account Migration

**C** number of Accounts

**A** seconds to retrieve an Account Listing

**R** seconds to make a full Account Retrieval call

## Running a synchronization

You can run a [synchronization from the Sage CRM UI](#). This is recommended when you first integrate your systems and want to validate the synchronized data. The synchronization adds accounting data fields to standard Sage CRM entities and creates a set of virtual tables, screens and lists in the Sage CRM metadata.

When you are happy with the data, you can schedule subsequent [synchronizations from the command line](#). The script doesn't need to refresh information on screen, so it makes a single full Customer Account Listing call which speeds up the synchronization. The duration of a synchronization from the command line is

- $((C \times R) + A)/60$  = Number of minutes for synchronizing all records
- $((C-D)*R) + A)/60$  = Number of minutes for synchronizing updated records
- $A/60$  = Number of minutes for migrating Customer Account data from Sage X3 to Sage CRM only

### Synchronizing from the UI

1. In Sage CRM, click **<My Profile> | Administration | Sage CRM Connector | X3 Account Migration**.
2. Click **Display X3 Accounts**. Sage X3 Customer Accounts are listed.
3. Click **Import & Sync** to start the migration. All Sage X3 Customer Accounts are migrated into Sage CRM as Companies, People, Addresses, Phones, and Emails.
4. To view the log file when all Customer Accounts are processed, go to **WWWRoot | CustomPages | Enbu | Framework | X3 | Logs**. It contains details of migrations and synchronizations that were run that day.

### Synchronizing from the command line

1. Go to **WWWRoot | CustomPages | Enbu | Framework | X3**.
2. Open CallService.js using Notepad. CallService is a command line JS script that runs migration and synchronization on a scheduled service. You must specify the correct Sage CRM URL in the script. You can set a showMessages flag to test the script manually on the command line and receive interactive messages from the service. CallService is located in **CRM | WWWRoot | CustomPages | Enbu | Framework | X3**.

```

//----- SETTINGS -----
// this is the url of the page you want to call
var sInstallName = "CRM";
var Url="http://localhost/" + sInstallName + "/custompages/enbu/framework/x3/AccountMigrationService.asp
var showMessages = true; // set to false when ready to run this script by scheduled task - prevents the

//----- END OF SETTINGS -----

try
{
    // you need XML Parser installed on whatever machine you are using to run this
    // script - if its the eWare machine it will be there already
    //var http=new ActiveXObject("msxml2.xmlhttp.3.0");
    var _lmin = 1000*60;
    var _lhour = _lmin*60;
    var http=new ActiveXObject("msxml2.ServerXMLHTTP.3.0");
    http.setTimeouts(_lmin, _lmin, _lhour*30); // resolve, connect, send, receive
    http.open("GET",Url,false);
    Msg(
        "Migration will begin once you confirm here, please allow process to finished completely.\n" +
        "Log files can be monitored from the Logs folder during sync process."
    );
    http.send();

    // the response text is everything that the page Response.Writes
    // therefore you can use it to provide some logging info
    strResult = http.responseText;
    Msg(strResult + "\n\nSee Log Files for further details.");
}

```

3. Specify the correct Sage CRM install name and URL.
4. Save and close the script.
5. Run the JS shell script from the command line to confirm it's working correctly.
6. You can schedule the script to run at a particular time, using the Windows Task Scheduler.
  - Click **Start | Control Panel | System and Security | Administrative Tools | Task Scheduler**.
  - Sign in and click **Action**.
  - Click **Create a Basic Task** and enter a name. For example, *Sage X3 and CRM integration service*. Click **Next**.
  - Choose the frequency of the task, the date and time when you want the task to start, and how often you want the task to run. For example, running daily, starting from November 1 at 15:12:00 and recurring every two days. Click **Next**.
  - Choose the action that the task performs and fill in the details of the action. For example, choose **Start a program** and specify the path to the script that the task runs. Click **Next**.
  - Click **Finish**.

To use AccountMigrationService.asp to schedule automatic updates, you need a Self Service license. Please contact your Sage partner to obtain this.

# Sage X3 data in Sage CRM

## Synchronized data

The following data is synchronized between Sage CRM and Sage X3. The data is added to existing Sage CRM tables.

- Company Name
- Sage X3 Account Number
- Sage X3 Credit Limit
- Data that's associated with a Company including Person, Address, Email, and Phone

## Real-time data

The following data is retrieved from Sage X3 and displayed in Sage CRM in real time. Virtual tables are created in Sage CRM to store the associated metadata, but the data is stored in the Sage X3 database. You can edit this data and send the new values to Sage X3.

- Orders
- Quotes
- Customer information
- Work Orders (demo only)
- Projects (demo only)
- Any other Sage X3 data that you've exposed by adding additional web services

## Sage CRM Company record

After synchronization, a Sage CRM Company record includes new fields that display Sage X3 data.

- **Account number** links the Company record to a Customer Account record. If you change this value, the link between the data records changes. To link a Company record to a Customer Account record, click **Post to X3**. A new Customer Account record is created in Sage X3 and linked to the Company record with a new Account number. You can post a Company record of any type or with any status to Sage X3 if **Account Number** is blank.

```

// if the customer category is specified, allow it to automate the defaults
if (Defined(sX3Category) && sX3Category!=''){
    .....
    X3Company.Records("bpc0_1_bcgcod") = sX3Category;           // customer category
    // depending on the customer category used, you may or may not need to specify defaults for
    // the following fields. If not, then simply comment out the field listed here and
    // allow the customer category manage these settings
    X3Company.Records("bpc3_2_vacbpr") = "TAX";                 // tax rule
    X3Company.Records("bpc3_3_pte")   = "CASH";                 // payment terms
} else {
    // in this case, no category specified, so create defaults for all required fields
    X3Company.Records("bpc0_1_bcgcod") = "US";                 // customer category
    X3Company.Records("bpc3_2_vacbpr") = "TAX";                 // tax rule
    X3Company.Records("bpc3_3_pte")   = "CASH";                 // payment terms
    X3Company.Records("bpcr_2_fcy")    = "";                    // shipment site
}

```

**Note:** You must update Tax Code (default is TAX) and Sage X3 Payment Terms (default is CASH) in PostToX3 before making a successful Post call.

- **X3 Category** prepopulates certain values required to successfully post a Customer Account record to Sage X3. It provides a consistent way of creating Customer Account records from Sage CRM. You can change the default field value (US) using the Sage CRM UI. See [Modifying the default field properties](#) in the System Administrator Help. Alternatively, update the [PostToX3](#) script.
- When a valid Sage X3 Account Number is linked to a Sage CRM Company record, an **X3** tab is displayed in the context of the Company. It displays the following real-time Sage X3 information which you can update.
  - Customer account and contact information.
  - Quotes and quote line items.
  - Orders and order line items.
  - Work orders and work order line items (read only).
  - Any other Sage X3 data that you've exposed by adding additional web services.

# Customizing Connections and Entities

## Customizing connections

Files used to connect Sage CRM to Sage X3 are located in **CRM | WWWRoot | CustomPages | Enbu | Framework | X3**.

### X3Data.asp

```
//----- SET UP OF X3 LISTS AND SCREENS-----  
var X3MenuChoices = new Array();  
X3MenuChoices["customer"] = new menuOption();  
X3MenuChoices["customer"].sTitle = "X3 Customer Info";  
X3MenuChoices["customer"].sX3WebService = "X3WSCustomer";  
X3MenuChoices["customer"].sListObject = "X3CustomerList";  
X3MenuChoices["customer"].sScreenObject = "X3CustomerScreen";  
X3MenuChoices["customer"].sCRMIdField = "bpc0_1_bpcnum";  
X3MenuChoices["customer"].sX3IdField = "BPCNUM";  
X3MenuChoices["customer"].sItemListObject = "X3CustomerContactList";  
X3MenuChoices["customer"].sItemRecordRef = "CNTC_1";  
X3MenuChoices["customer"].sImg = "../icons/account.gif";
```

X3Data.asp is a generic script that renders the real-time interfaces to Sage X3 data that are included in the Sage CRM Connector. It allows full read and write access to these established connections. By adding new entities to the X3MenuChoices object in the script, you can access other areas of Sage X3. See comments in the file for more information.

## Customizing entities

A Sage CRM entity is created for each Sage X3 object that the Gateway connects to. Sage X3 definitions and values are migrated to the Sage CRM entity's metadata and can be configured in Sage CRM.

- **Field definitions.** You can add new fields or configure captions, sizes, and drop-down values of existing fields. For more information, see [Field Customization](#) in the Sage CRM System Administrator Help.
- **Screens.** You can add new fields, rearrange existing fields, and run field-level scripts. For more information, see [Screen Customization](#) in the Sage CRM System Administrator Help.
- **Lists.** You can add columns, rearrange lists, and run field-level scripts. You can run CreateScript using Required, ReadOnly and DefaultValue variables, and the server-side eWare/CRM object. For configuration information, see [List Customization](#) in the Sage CRM System Administrator Help.
- **Values.** You can run table-level scripts from **<My Profile> | Administration | Customization | Secondary Entities | <Sage X3 object name>** to read and write to a Sage X3 record. For more information, see [Table-level scripts](#) in the Sage CRM Developer Help.

# Uninstalling the Connector

## Uninstall script

The following script removes changes made by the Connector and any references to the Connector that are displayed to an end user in Sage CRM. Additional fields may remain in the Administration panel.

1. Back up the Sage CRM database before running the script.
2. Open SQL Enterprise Manager and run the script against the Sage CRM database. You can enclose the script in a transaction as the script modifies data only. It does not modify the underlying table structure.
3. Restart IIS when the script has executed.

```
-- Delete Sage CRM Connector button from Admin screen
-- Removes the Sage CRM Connector button from Admin screen
delete from Custom_Tabs
where Tabs_Entity like 'Admin%'
and Tabs_Caption = 'Enbu WS Framework';

-- Delete X3 tab and post to X3 from Company
-- Removes the X3 tab
-- Removes the Post to X3 button from Company summary screen
delete from Custom_Tabs
where Tabs_Entity like 'Company%'
and Tabs_Action = 'customfile'
and Tabs_Caption in ( 'X3', 'Post to X3');

-- Delete Custom Screen entries
-- removes connector specific fields from screens
delete from Custom_Screens
where SeaP_ColName in (
'comp_x3_bpcord',
'comp_x3_creditlevel',
'comp_x3category',
'comp_x3_bpcord',
```

```
'pers_int_code',
'pers_int_addrcode',
'addr_int_code',
'user_int_userid'
);

-- Update the custom content of the address screen
-- Removes the Custom content from the Address - AddressBoxLong screen
update Custom_ScreenObjects
set Cobj_CustomContent = ''
where CObj_Name = 'AddressBoxLong'
and Cobj_CustomContent like '%if(Act == "1202" || Act == "1201" || Act == "141"
|| Act == "241" || Act == "221")%';
```

# Troubleshooting

## Troubleshooting resources

For a complete list of integration videos and troubleshooting articles, see the [Connector landing page](#) on the Sage CRM Partner Community.

## Timestamp and daylight savings time

Issue: The timestamp used when synchronizing data from Sage X3 to Sage CRM does not consider daylight savings time.

1. On the server where Sage X3 is installed, unselect **Automatically adjust clock for Daylight Savings Time**.
2. On the server where Sage CRM is installed, unselect **Automatically adjust clock for Daylight Savings Time**.

## Update time conversion to UTC for Sage X3 version 6.5

Issue: In Sage X3 version 6.5, update times are stored as local server time on the database. This was changed in Sage X3 version 7 onwards to store the update times as UTC time.

Sage CRM stores its update time in local server time. The Sage CRM Connector version 2.0 automatically converts Sage CRM update time from local server time to UTC time so a comparison can be made with Sage X3 when performing an account migration.

You can turn off this automatic conversion when running against Sage X3 version 6.5. To do this, add the following line to `accountMigration.asp` and `accountMigrationService.asp`.

```
AutoUTCAdjustment = false;
```

```
//----- SETTINGS -----  
// list mode does not make any per account calls and is much  
// sync must on the account listing (BPC) call and this is on  
listMode = false;           // false : is full 2 way sync  
migrateMode = false;       // true : ignore 2 way sync a  
listTimestampOnly = false; // even when the list based t  
                             // trying to get it on a full  
AutoUTCAdjustment = false;  
  
var totalNum = Request.QueryString("totalnum");
```

## WsOrder web service

Issue: When you publish Connector web services using Sage X3 PU9, the following error occurs.

*SOH WEB: Too many published fields.*

1. In Sage X3, click **Development | Data and parameters | Development setup | Activity codes**.
2. Click the activity code that's referenced in the error message.
3. Change the value of **Screen size** from **1002** to **300**.
4. Click **Parameters | Sales | Entry Transactions | Orders**.
5. From the list on the left side of the screen, click a sales order transaction and then click **Validation** on the right of the screen. Validate each sales order transaction in the list.
6. Click **Development | Script dictionary | Scripts | Web Services**.
7. Filter the list of Publications to find the WsOrder publication.
8. Click **Publication** on the right side of the screen.

## Display X3 Accounts button

Issue: When you click **Display X3 Accounts** during account migration using the Connector, no Sage X3 customer accounts are displayed.

1. Navigate to `C:\Program Files (x86)\Sage\CRM\CRM\WWWRoot\CustomPages\Enbu\Framework\Server\EnbuGateway\Schema` and delete the `WSCUSTOMER.xml` file.
2. Log in to Sage CRM.
3. Select the **WSCconnections** tab from the Sage CRM Connector task.
4. Select the `WSCUSTOMER` gateway method.
5. Click **Change**.
6. In **Gateway Method Test Parameters**, enter a valid X3 customer number for the `BPCNUM=` value.
7. Click **Save**.
8. Click **Test Service**.
9. When the valid customer record is returned, click **Create Table**, and then click **Confirm Create Table**.
10. Attempt the account migration again.

## TSC code

Issue: When you attempt to update the **TSC code** field on the X3 Customer Info screen in Sage CRM, a sync error occurs. This is because validation is required to update this field in Sage X3. The solution is to make the **TSC code** field on the X3 Customer Info screen in Sage CRM read-only.

1. Log in to Sage CRM.
2. Navigate to **<My Profile> | Administration | Customisation | Secondary Entities | WsCustomer**.
3. Click the **Screens** tab on the `WsCustomer` screen.
4. Click the `X3CustomerScreen` link.
5. Click **WsCustomer: TSC Code (bpc2\_6\_tsccod)** in the **Desktop HTML Screen Contents** section.
6. Enter `ReadOnly = true;` in **CreateScript**.
7. Click **Update** and then click **Save**.

## Sage X3 version number

Issue: On a Spanish environment where commas (,) are used instead of periods (.), the Gateway incorrectly reads the Sage X3 version number as 70 instead of 7. As a result, the Gateway assumes that the version is Sage X3 PU9 or later and tries to connect with the authentication requirements of X3 PU9 or later.

Solution: In web.config, change the value of **X3 Version** from *7.0* to *7*.

# Appendix: Sage X3 Definitions

## Screen definitions

Screen	Field Name	Caption
X3CustomerScreen	bprc_1_bprnam	Company name
X3CustomerScreen	bpc0_1_bpcnum	Customer
X3CustomerScreen	bpc2_1_bpctyp	Customer type
X3CustomerScreen	bpc2_1_chgtyp	Rate type
X3CustomerScreen	bpc1_2_freinv	Freight invoicing
X3CustomerScreen	bpc1_1_abcccls	ABC Class
X3CustomerScreen	bpc2_2_ostctl	Credit control
X3CustomerScreen	bpc1_2_ime	Invoicing mode
X3CustomerScreen	bprc_3_prvflg	Service supplier
X3CustomerScreen	bpc3_5_flgsttax	Collection agent
X3CustomerScreen	bpac_1_bpacry	Country
X3CustomerScreen	bpac_1_crynam	Country name
X3CustomerScreen	bpc2_2_xost	Credit level total
X3CustomerScreen	bpc2_2_wostauz	Authorized credit
X3CustomerScreen	bpc4_1_ffwnum	Freight agent
X3CustomerScreen	bpc2_2_bpccdtisr	Ins. Company
X3CustomerScreen	bpc4_1_lan	Language
X3CustomerScreen	bpc1_1_wordminamt	Minimum order
X3CustomerScreen	bpc3_3_pte	Payment terms
X3CustomerScreen	bpac_1_poscod	Postal code
X3CustomerScreen	bprc_3_repflg	Sales rep
X3CustomerScreen	bprc_1_crn	Site tax ID no.
X3CustomerScreen	bpac_1_fcyweb	Website
X3CustomerScreen	bpc2_6_tscod	TSC Code
X3CustomerScreen	bpc4_1_creat	Date created
X3CustomerScreen	bpc4_1_upddat	Change date
X3CustomerScreen	bpc4_1_flgmod	Change flag
X3CustomerScreen	bpc4_1_updusr	Change user
X3CustomerScreen	bpc4_1_enaflg	Active
X3CustomerScreen	bpac_1_cty	City
X3CustomerScreen	bidc_1_bidnum	Bank ID number

Screen	Field Name	Caption
X3CustomerScreen	bpc3_1_bpcinv	Bill-to customer
X3OrderScreen	soh0_1_sohnum	Number
X3OrderScreen	soh0_1_cusordref	Reference
X3OrderScreen	soh0_1_bpcord	Sold-to
X3OrderScreen	soh0_1_sohtyp	Type
X3OrderScreen	soh0_1_orddat	Date
X3OrderScreen	adb1_1_bprnam	Company name
X3OrderScreen	adb1_1_cry	Country
X3OrderScreen	soh0_1_cur	Currency
X3OrderScreen	soh4_4_ordinvnot	soh4_4_ordinvnot
X3OrderScreen	soh4_4_ordinvati	Tax included
X3OrderScreen	soh4_1_sau	SAL
X3OrderScreen	soh2_1_stofcy	Shipment site
X3OrderScreen	soh0_1_salfcy	Sales site
X3ProjectScreen	opp0_1_cur	Currency
X3ProjectScreen	opp0_1_oppcmp	BP
X3ProjectScreen	opp0_1_oppcmpcla	opp0_1_oppcmpcla
X3ProjectScreen	opp0_1_oppdes	Description
X3ProjectScreen	opp0_1_oppextnum	External ident.
X3ProjectScreen	opp0_1_oppmcn	Contact (relationship)
X3ProjectScreen	opp0_1_oppmcncla	opp0_1_oppmcncla
X3ProjectScreen	opp0_1_oppnum	Sequence no.
X3ProjectScreen	opp0_1_salfcy	Sales site
X3ProjectScreen	opp0_1_zcur	opp0_1_zcur
X3ProjectScreen	opp0_1_zsalfcy	opp0_1_zsalfcy
X3ProjectScreen	opp1_1_oppcda	Conclusion date
X3ProjectScreen	opp1_1_oppdatopn	Opening date
X3ProjectScreen	opp1_1_oppnbq	Number of quotes
X3ProjectScreen	opp1_1_opprep	Sales rep
X3ProjectScreen	opp1_1_oppste	Last completed stage
X3ProjectScreen	opp1_1_oppptyp	Category
X3ProjectScreen	opp1_1_strste	Since
X3ProjectScreen	opp1_1_zopprep	opp1_1_zopprep
X3ProjectScreen	opp1_1_zoppste	opp1_1_zoppste
X3ProjectScreen	opp1_1_zoppptyp	opp1_1_zoppptyp
X3ProjectScreen	opp1_2_oppaam	Weighted amount
X3QuoteScreen	sqh0_1_sqhnum	Quote no.
X3QuoteScreen	sqh0_1_bpcord	Customer

Screen	Field Name	Caption
X3QuoteScreen	sqh0_1_quodat	Date
X3QuoteScreen	sqh0_1_cusquoref	Reference
X3QuoteScreen	sqh2_4_quoinvnot	-Tax
X3QuoteScreen	sqh2_4_quoinvati	Tax included
X3QuoteScreen	sqh1_2_stofcy	Shipment site
X3QuoteScreen	sqh0_1_salfcy	Sales site
X3QuoteScreen	sqh1_4_vacbpr	Tax rule
X3WorkOrderScreen	mfg2_2_itmdes1	Description 1
X3WorkOrderScreen	mfg0_1_mfgnum	Work order
X3WorkOrderScreen	mfg1_1_mfgdes	Work order description
X3WorkOrderScreen	mfg0_1_plnfcy	Planning site
X3WorkOrderScreen	mfg0_1_mfgfcy	Production site
X3WorkOrderScreen	mfg1_1_tbosho	Title
X3WorkOrderScreen	mfg3_2_xwsttyp	Type
X3WorkOrderScreen	mfg1_2_strdat	Start date
X3WorkOrderScreen	mfg2_2_retdat	Required date
X3WorkOrderScreen	mfg1_2_enddat	End date
X3WorkOrderScreen	mfg4_1_labtikflg	Job ticket
X3WorkOrderScreen	mfg0_2_mfgsta	Order status
X3WorkOrderScreen	mfg1_3_mfgpio	Priority
X3WorkOrderScreen	mfg3_1_scdflg	Scheduling status
X3WorkOrderScreen	mfg2_2_cumflg	Total
X3WorkOrderScreen	mfg3_2_labwcr	Labor center
X3WorkOrderScreen	mfg4_1_piclisflg	Material issue slip
X3WorkOrderScreen	mfg3_2_extwstnbr	Number of resources
X3WorkOrderScreen	mfg2_2_allqty	Qty allocated
X3WorkOrderScreen	mfg1_1_itmtyp	Type

## List definitions

Screen	Field Name	Caption
X3CustomerContactList	cntc_1_cntttl	Title
X3CustomerContactList	cntc_1_cntfna	First name
X3CustomerContactList	cntc_1_cntlna	Name
X3CustomerContactList	cntc_1_cntweb	Email
X3CustomerContactList	cntc_1_cntfnc	Function
X3CustomerContactList	cntc_1_cntlan	Language
X3CustomerContactList	cntc_1_cntmob	Mobile phone
X3CustomerContactList	cntc_1_cnttel	Telephone
X3CustomerContactList	cntc_1_cntcry	Country
X3CustomerContactList	cntc_1_cntfbdmag	Mailing prohibited
X3CustomerList	bpc0_1_bpcnam	Customer
X3CustomerList	bpac_1_addlig1	Address
X3CustomerList	bpc2_2_wostauz	Authorized credit
X3CustomerList	bpac_1_cty	City
X3CustomerList	bpc0_1_bpcnum	Customer
X3CustomerList	bpc2_1_bpctyp	Customer type
X3CustomerList	bpc1_1_abccls	ABC Class
X3OrderItemList	soh4_1_itmdes	Description
X3OrderItemList	soh4_1_itmref	Product
X3OrderItemList	soh4_1_qty	Ordered qty
X3OrderItemList	soh4_1_gropri	Gross price
X3OrderItemList	soh4_1_linordnot	Line amt -tax
X3OrderItemList	soh4_1_ddemdlvdat	Last del date
X3OrderItemList	soh4_1_dshidat	Ship date
X3OrderItemList	soh4_1_extdlvdat	Exp delivery date
X3OrderItemList	soh4_1_fm	Source for delivery
X3OrderItemList	soh4_1_vacitm1	Tax level 1
X3OrderList	soh0_1_sohnum	Number
X3OrderList	soh0_1_cusordref	Reference
X3OrderList	soh0_1_orddat	Date
X3OrderList	soh0_1_bpcord	Sold-to
X3ProductList	its_1_itmref	its_1_itmref
X3ProductList	its_1_c2	its_1_c2
X3ProjectItemList	opp3_1_zste	opp3_1_zste
X3ProjectItemList	opp3_1_don	Phase end
X3ProjectItemList	opp3_1_ren	Reason
X3ProjectItemList	opp3_1_ste	Step

Screen	Field Name	Caption
X3ProjectList	opp0_1_oppnum	Sequence no.
X3ProjectList	opp0_1_oppdes	Description
X3ProjectList	opp1_1_opptyp	Category
X3ProjectList	opp0_1_oppcmp	BP
X3QuoteItem	sqh2_1_itmdes	Description
X3QuoteItem	sqh2_1_itmref	Product
X3QuoteItem	sqh2_1_qty	Quote quantity
X3QuoteItem	sqh2_1_gropri	Gross price
X3QuoteItem	sqh2_1_cprpri	Cost price
X3QuoteItem	sqh2_1_linpfm	Line margin
X3QuoteItem	sqh2_1_pfm	Margin
X3QuoteItem	sqh2_1_netpri	Net price
X3QuoteItem	sqh2_1_linquonot	Line amt -tax
X3QuoteItem	sqh2_1_linquoati	Line amt +tax
X3QuoteList	sqh0_1_sqhnum	Quote no.
X3QuoteList	sqh0_1_cusquoref	Reference
X3QuoteList	sqh0_1_cur	Currency
X3QuoteList	sqh0_1_bpcord	Customer
X3QuoteList	sqh0_1_quodat	Date
X3SummaryListing	bpc0_1_bpcnam	Customer
X3SummaryListing	bpc0_1_bpcnum	Customer
X3WorkOrderItem	mfg1_1_itmdes1	Description
X3WorkOrderItem	mfg1_1_itmref	Product
X3WorkOrderItem	mfg1_1_extqty	Expected STK qty
X3WorkOrderItem	mfg1_1_bpcnum	Destination
X3WorkOrderItem	mfg1_1_lot	Lot
X3WorkOrderItem	mfg1_1_bomalt	BOM code
X3WorkOrderItem	mfg1_1_itmsta	Line status
X3WorkOrderItem	mfg1_1_shr	% loss
X3WorkOrderList	mfg0_1_mfgnum	Work order
X3WorkOrderList	mfg1_4_rounum	Number
X3WorkOrderList	mfg0_1_mfgfcy	Production site
X3WorkOrderList	mfg1_2_strdat	Start date