Teamcenter Integration for SolidWorks®
Installation Guide
Version 10.1.0.3

International TechneGroup Incorporated
DuPont Circle
Milford, Ohio 45150

Contents

Introduction

This guide describes the prerequisites and installation procedure for the Teamcenter Integration for SolidWorks®.

Prerequisites

The following software must be installed before installing the Teamcenter Integration for SolidWorks.

Supported operating systems:

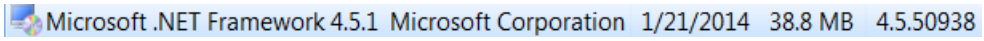
Supported client operating systems

- Windows 7 and Windows 8

  For Windows machines, the Microsoft Visual C++ Redistributable Package must be installed. Make sure to pick the appropriate (x86 or x64) version for you version of Windows. The latest Package can be found at the following URL:

  http://www.microsoft.com/en-us/download/search.aspx?q=redistributable%20package&p=0&r=10&t=&s=Relevancy~Descending

New requirement for version 10.1.0: The integration requires Microsoft .NET V4.5.1. Open the **Programs and Features** Control Panel and look for this entry:



The .NET installer can be found at the following URL:

http://www.microsoft.com/en-us/download/details.aspx?id=17718

Supported Teamcenter server operating systems:

- HP Itanium
- Sun Solaris
- IBM AIX
- Linux
- Microsoft Windows

Supported operating system version numbers are available in the Siemens certification matrix:

http://support.industrysoftware.automation.siemens.com/certification/teamcenter.shtml

Teamcenter:

- Teamcenter 9.1.3.2 and later MPs and patches
- Teamcenter 10.1.2.2 and later MPs and patches

Write permission to the Rich Client installation directory is needed to install files and configure the Rich Client to start SolidWorks.

Teamcenter Rich Client Portal:

Write permission to the Rich Client installation directory is needed to install files and configure the Rich Client to launch SolidWorks.

Minimum support SolidWorks versions:

- 2013 SP5
- 2014 SP0
- 2015 SP2.1
- 2016 SP0

Installation procedure

Stop all active sessions

Before performing a server installation, all users must be logged out of Teamcenter, all server processes must be halted, and all locks must be cleared from the database. The required steps are these:

1. Advise all users to log out of Teamcenter.
2. In a four-tier environment, shut down the pool manager using the control panel at http://serverhost:8082.
3. From a Teamcenter command line on a server host, run the clearlocks command:

```
clearlocks –assert_all_dead infodba infodba dba[1]
```

Failure to perform these steps will result in an incomplete server installation, and will prevent the integration from functioning properly. These steps are not necessary for client-only installations.

Starting the Installer on Windows

The installer for Windows is **swimsetup.exe**. Double-click on this file to start the installer. Make sure that user has access to environment variable %TMP% (usually redefined in most Windows installations). This is required by the installer and for running the Teamcenter Integration for SolidWorks clients.

[1] Your Teamcenter administrative user and password may be different; substitute the appropriate values

Starting the Installer on UNIX

Although the Teamcenter Integration for SolidWorks client can be installed only on Windows, a UNIX host may be used for the Teamcenter server. The installer for UNIX is **swimsetup.bin**. Type the following command in a shell to start the installer and then follow the steps for a server installation:

```
sh ./swimsetup.bin
```

Client/Server Installation Walkthrough

The following instructions will walk you through a typical client/server installation. A client/server installation contains all of the steps found in the other installations.

Choosing an Install Set

The Teamcenter Integration for SolidWorks offers several alternative installations called *install sets*. Shown in Figure 1 is the install set selection dialog. Before selecting an install set, you will need to determine some information about your site. This section will help you determine which type of installation should be performed on each workstation.

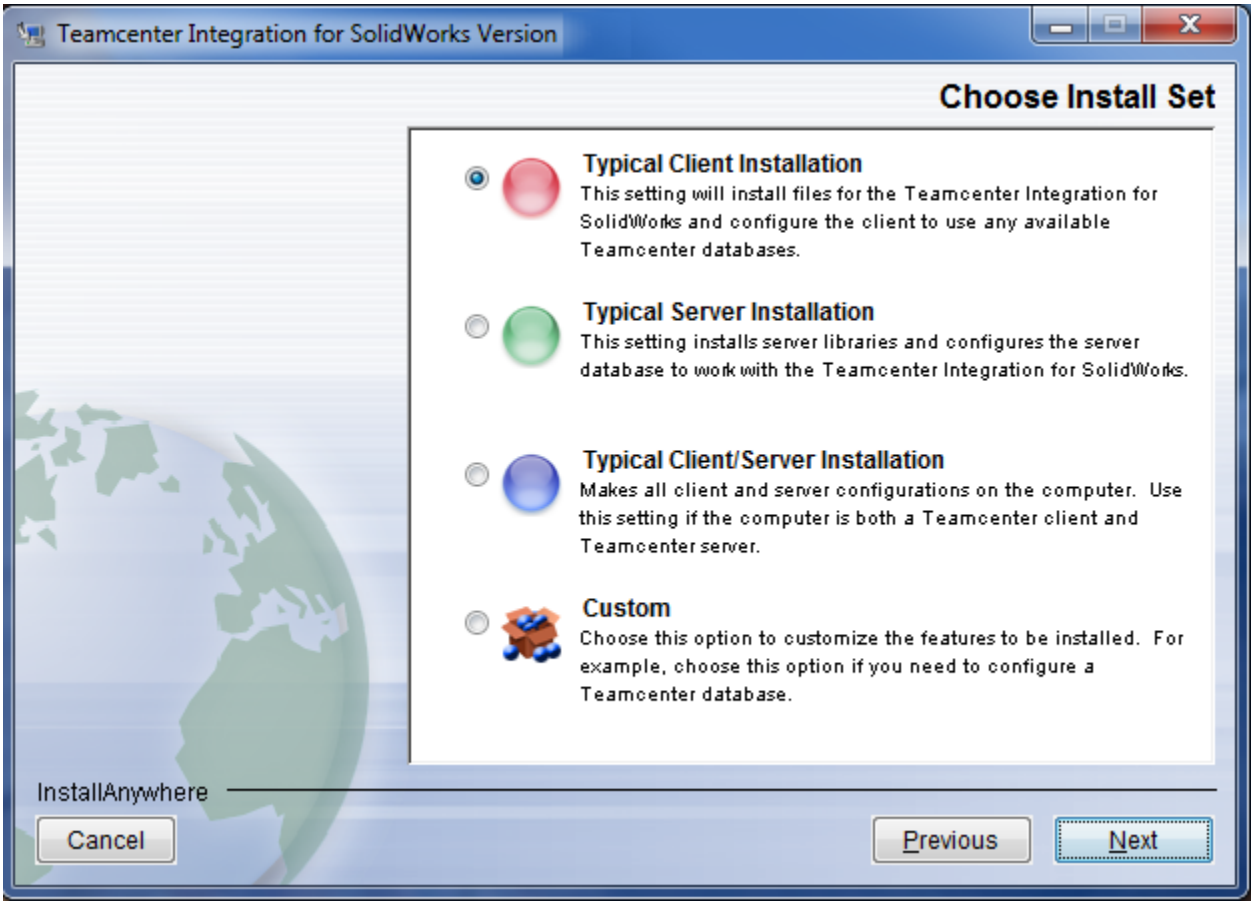


Figure 1 Select an install set

The location of the Teamcenter server relative to the client depends upon the configuration of Teamcenter itself. Each client will have a server process, and that process may run locally (in

the case of a two-tier configuration) or remotely (four-tier). In addition, at each Teamcenter site there is a single database that is updated as part of the integration installation. The server part of the installation ("Typical Server" or "Typical Client/Server") updates files on the server and also updates the database schema. The database update only has to be run once per site (although there is no harm in running it multiple times, and this may be convenient in some cases), but the server file update must be done on each server machine. Here are the appropriate installation choices for the most commonly used Teamcenter configurations:

- Two-tier: Select **Typical Client/Server installation** on each client. This will run the database update multiple times, but that does not cause a problem. You may also reply "No" to the prompt asking "Would you like to install the SolidWorks queries and preferences into the Teamcenter database?" on all but the first Client/Server installation, to bypass redundant database updates.
- Four-tier: Select **Typical server installation** on the Enterprise tier (where the tcserver pool runs), and select **Typical Client installation** on each client workstation.

Typical Client Installation

Select Typical Client Installation if you need to install the client portion of the Teamcenter Integration for SolidWorks on a client-only workstation. This installation is appropriate for a workstation that runs the Teamcenter Rich Client and SolidWorks, but not a Teamcenter server. This setting installs the files needed by the Teamcenter Integration for SolidWorks, and configures the client to access available Teamcenter databases.

Typical Server Installation

Select Typical Server Installation if you need to install the server portion of the Teamcenter Integration for SolidWorks on a Teamcenter server-only workstation. This setting installs server software needed by the Integration and configures the Teamcenter database. Before configuring the database, all users must log out of Teamcenter, because this step requires regenerating the database schema file.

Typical Client/Server Installation

Select Typical Client/Server Installation if your computer is both a Teamcenter client and server. You will need Teamcenter, the Teamcenter Rich Client, and SolidWorks installed. A server is a host that has a TC_ROOT directory. This setting performs all the operations of the **Typical Server Installation** as well as the **Typical Client Installation**[2].

Custom Installation

A custom installation allows you to customize the actions taken by the installer. This selection can be especially useful if you have already installed the Teamcenter Integration for SolidWorks, but need to reconfigure the database. See the appendix for details on custom installation options.

[2] Although all the steps are available, those which involve one-time operations such as database updates may be skipped on all but the first execution of the Typical Client/Server path.

Select the Teamcenter version

The Teamcenter Integration for SolidWorks needs to know the version of Teamcenter installed at the site. The list of available versions will vary by integration release level. Here is an example from an integration release that supports Teamcenter 9.1 and 10.1:

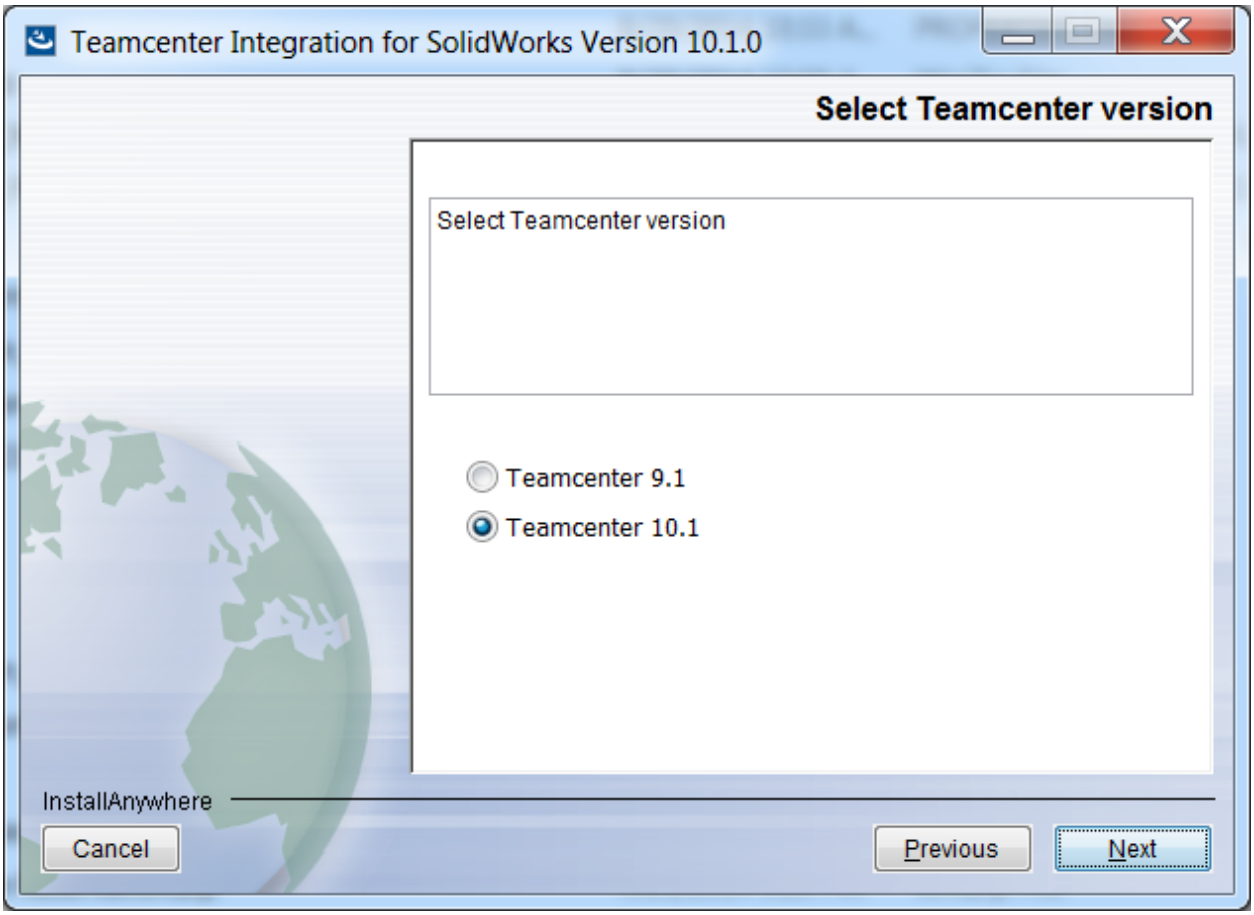


Figure 2 Select the appropriate Teamcenter version

Select an Install Folder

After choosing **Typical Client/Server Installation** as your install set, you must select the directory where the Teamcenter Integration for SolidWorks files will be installed. The interface for selecting the directory is shown in Figure 3. Please note that we recommend you do not install into a directory with spaces in its full path. If you have an existing installation, you may install into that directory. Installing into the same directory will preserve data from your existing **swim.properties** and **swim.xml** files, although it may replace the original files. The **Choose** button will display a file selection dialog to help you select a directory. **Restore Default Folder** will restore the default selection.

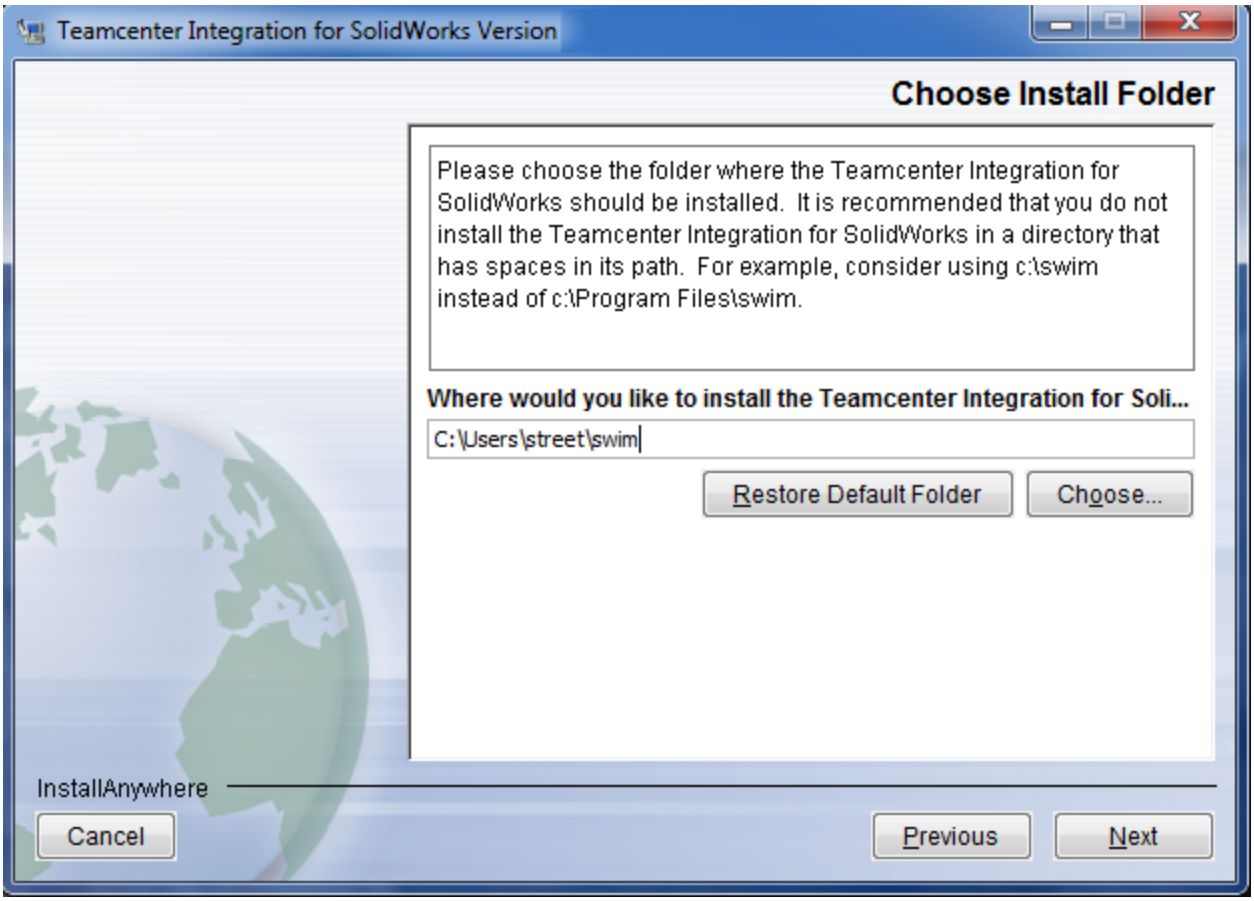


Figure 3. Select the install location

Select the Shortcut Folder

After selecting an installation folder, the installer will prompt for a location to create shortcuts as shown in Figure 4. You have the option of creating shortcuts in the Program Menu, in the Start Menu, on the desktop, or in the quick launch bar. **Create Icons for All Users** will install the shortcuts in a location common to all users. If this box is left unchecked, shortcuts will only be created for the current user.

Figure 4. Select the Shortcut Folder

Select the SolidWorks Start location

If you chose to create a shortcut in the previous screen, the installer will also prompt you to choose a folder for SolidWorks to start in. This will be SolidWorks' initial working directory each time you start the Teamcenter Integration for SolidWorks using the shortcut icon. It is recommended to use the same folder that will be used for exporting SolidWorks files from Teamcenter, as part of your SolidWorks design activity.

You will not be asked to choose a start folder if you chose **don't create icons** in the previous screen.

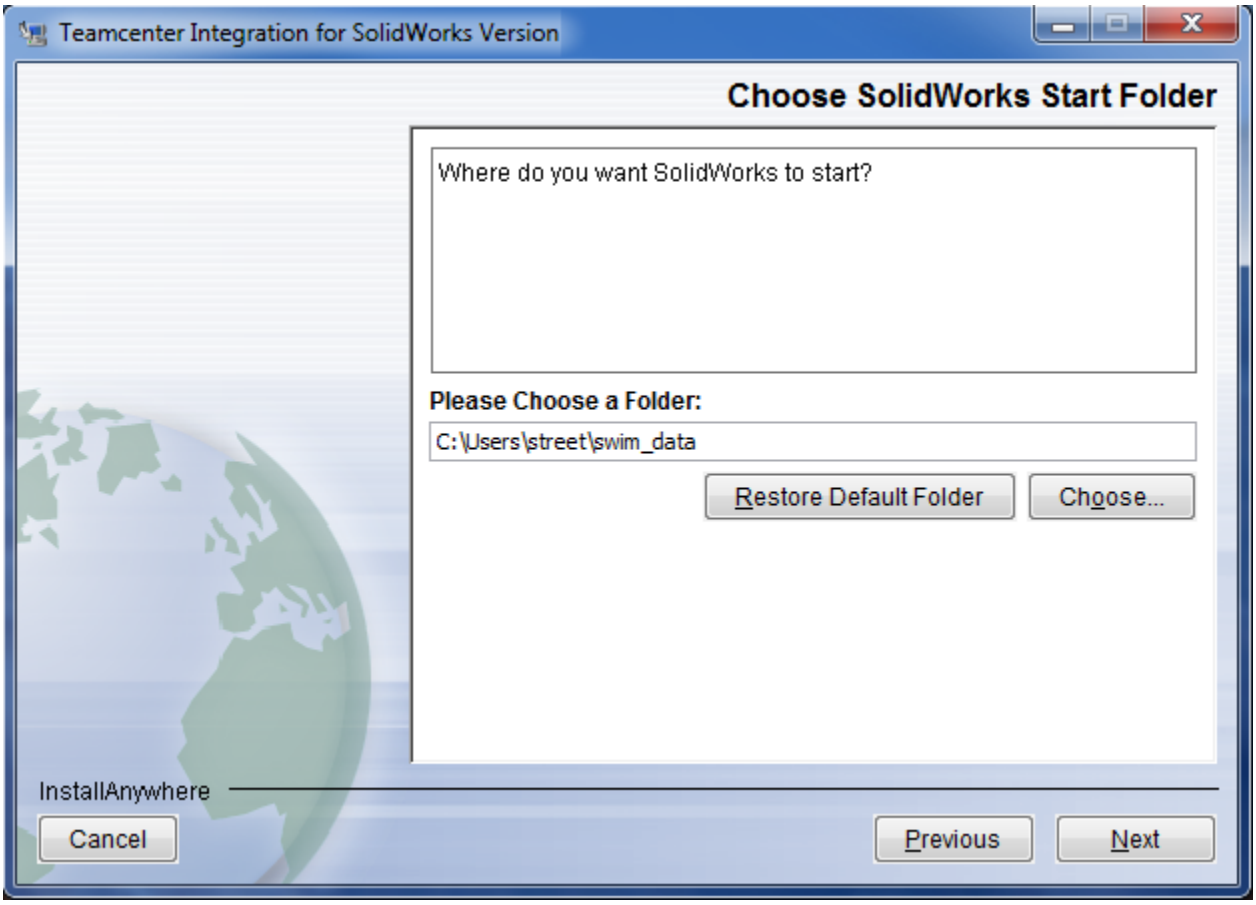


Figure 5 Select the SolidWorks startup folder

Select the Teamcenter Root location

After making your shortcut selections, find the directory where the Teamcenter server is installed. This is your TC_ROOT directory. If you have more than one Teamcenter server you should run a **Typical Server Installation** for each.

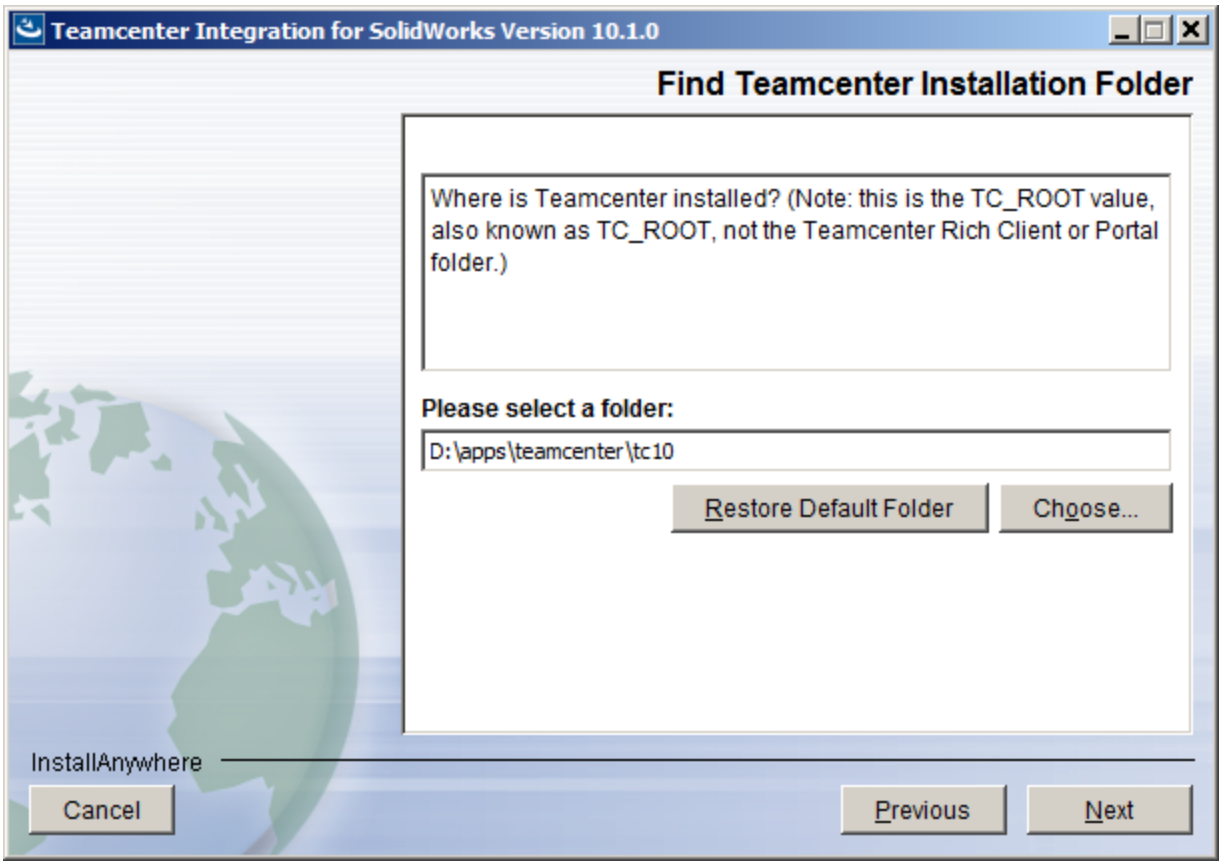


Figure 6 Select the TC_ROOT location

If you don't have a path to TC_ROOT on your workstation[3], then by definition it is a client-only workstation, and you should return to the initial dialog and choose the **Typical Client Installation** option.

Select the Teamcenter Data Directory

Find the data directory for your Teamcenter database. This is your TC_DATA directory. If you have more than one database, you will need to run a **Custom | Configure Server Database** installation for each of them.

[3] Bear in mind that the path to TC_ROOT may lead to a different workstation via a mounted drive letter, or a UNC path. As long as the TC_ROOT location is reachable from your operating system, and you have write permission to its subdirectories, you may proceed.

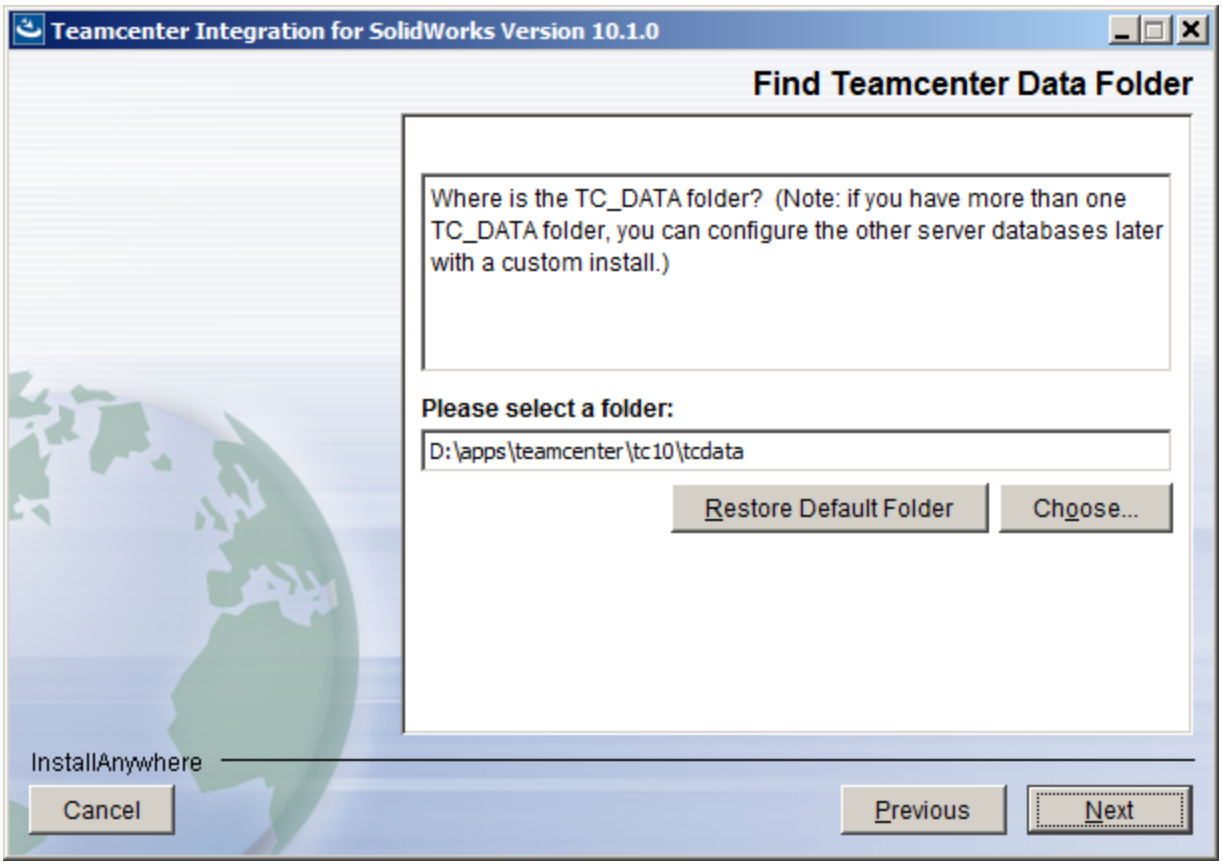


Figure 7 Select the TC_DATA directory

At some sites, the path to TC_DATA will lead to a network location which is accessed via a mounted drive letter or a UNC path.

Install Integration queries and preferences

Several queries and preferences must be installed in the Teamcenter database. The dialog for selecting this option is shown in Figure 8. Select "**Yes**" to have the queries and preferences installed into the Teamcenter database. You will be prompted for the Teamcenter administrator name and password. If you select "**No**," you may install the queries and preferences later by following the custom installation instructions in the Appendix.

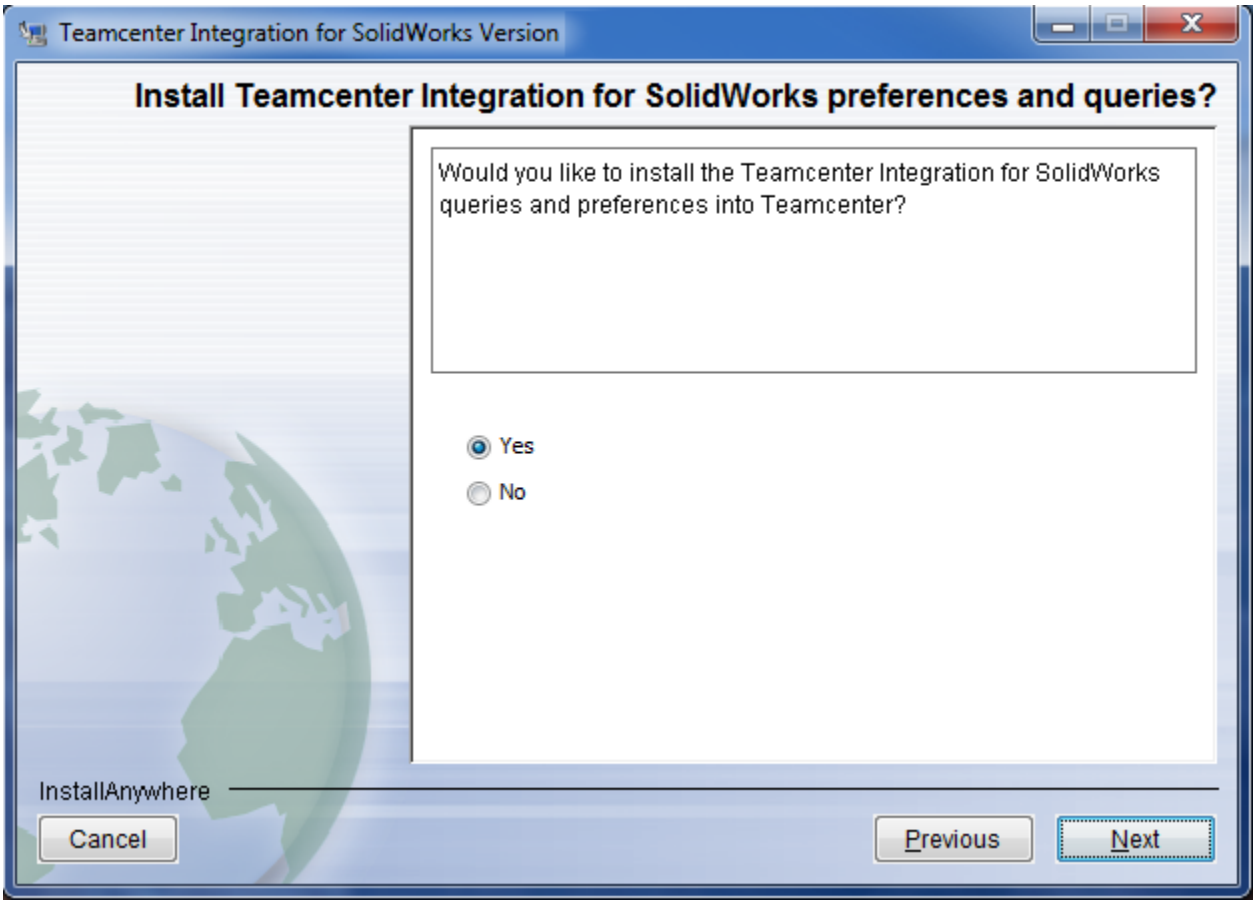


Figure 8. Install Integration queries and preferences

Enter the Teamcenter user name and password

If you chose “yes” in the previous dialog, you will be prompted for the Teamcenter administrative user name and password:

Figure 9 Enter the Teamcenter credentials

Install libtxd

A library named libtxd must be installed on your Teamcenter server, and a Teamcenter site preference must be defined to register that library. You must have write access to the *TC_ROOT* and *TC_DATA* directories. You should choose "**Yes**" when asked whether to install this library.

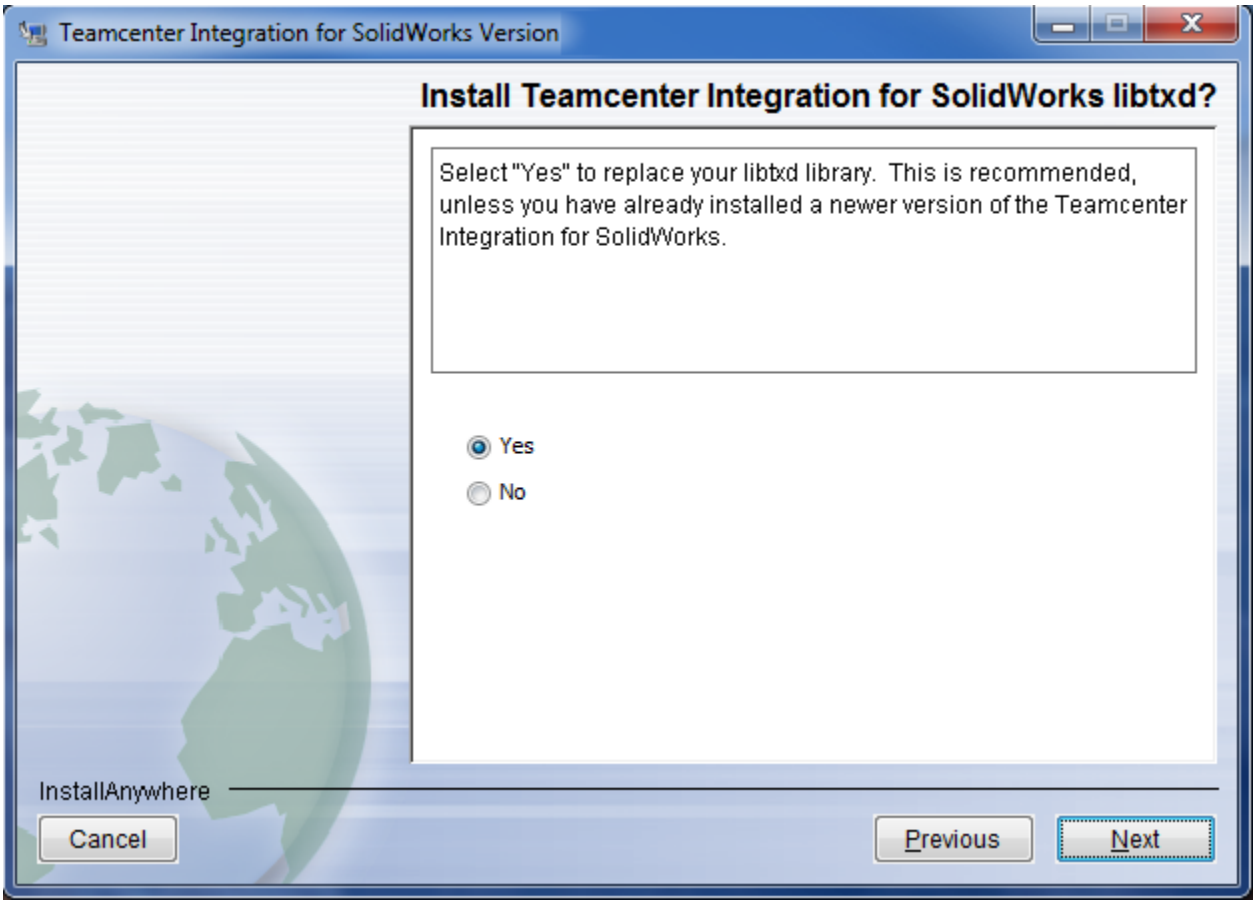


Figure 10 Install libtxd

If you have several Teamcenter databases, the libtxd library must be registered in the site preferences for each database. You may repeat this step by using a custom install for each database, or follow the custom installation instructions in the Appendix.

Select the Teamcenter Rich Client Folder

Choose the directory where the Teamcenter Rich Client is installed.

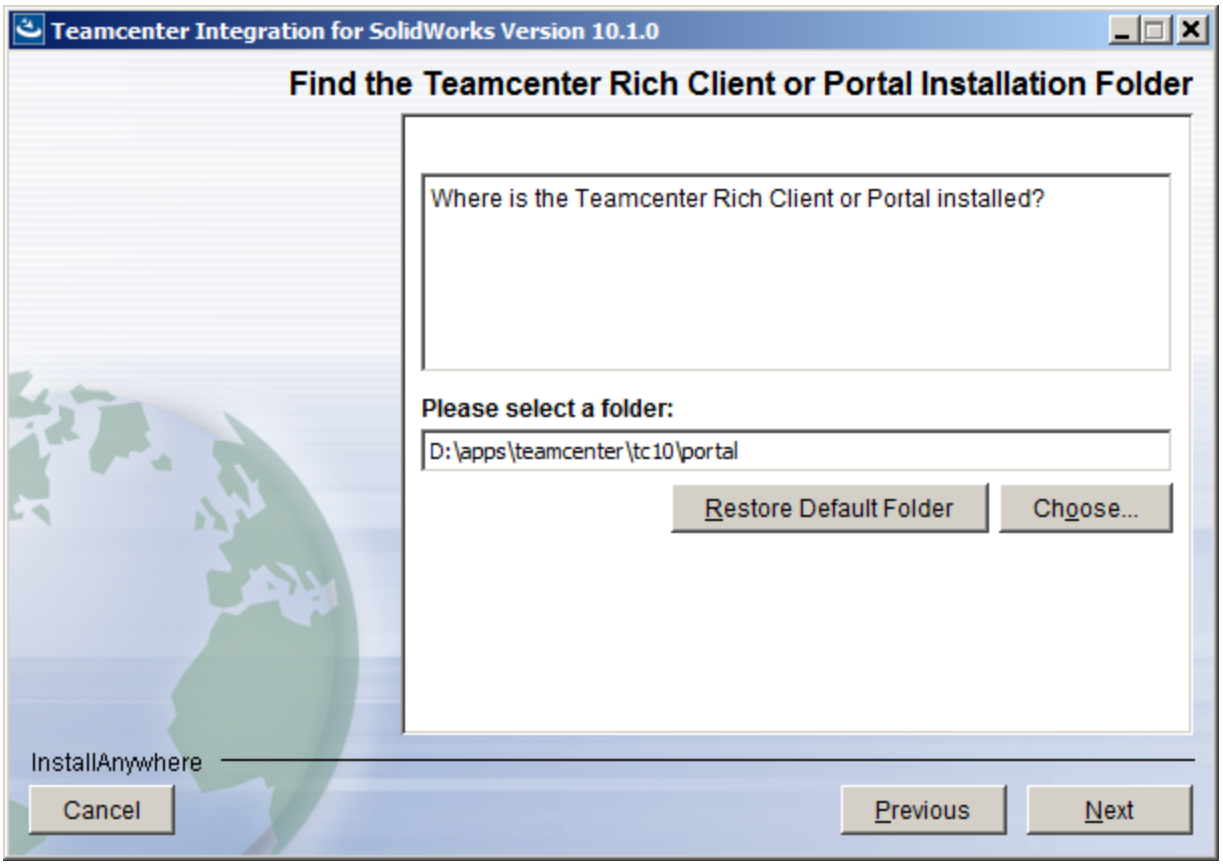


Figure 11 Select the Teamcenter Rich Client folder

Select the Teamcenter Rich Client Temporary Folder

Specify the directory where the Teamcenter Rich Client temporary folder is found. This location must exist in order for the integration to work properly.

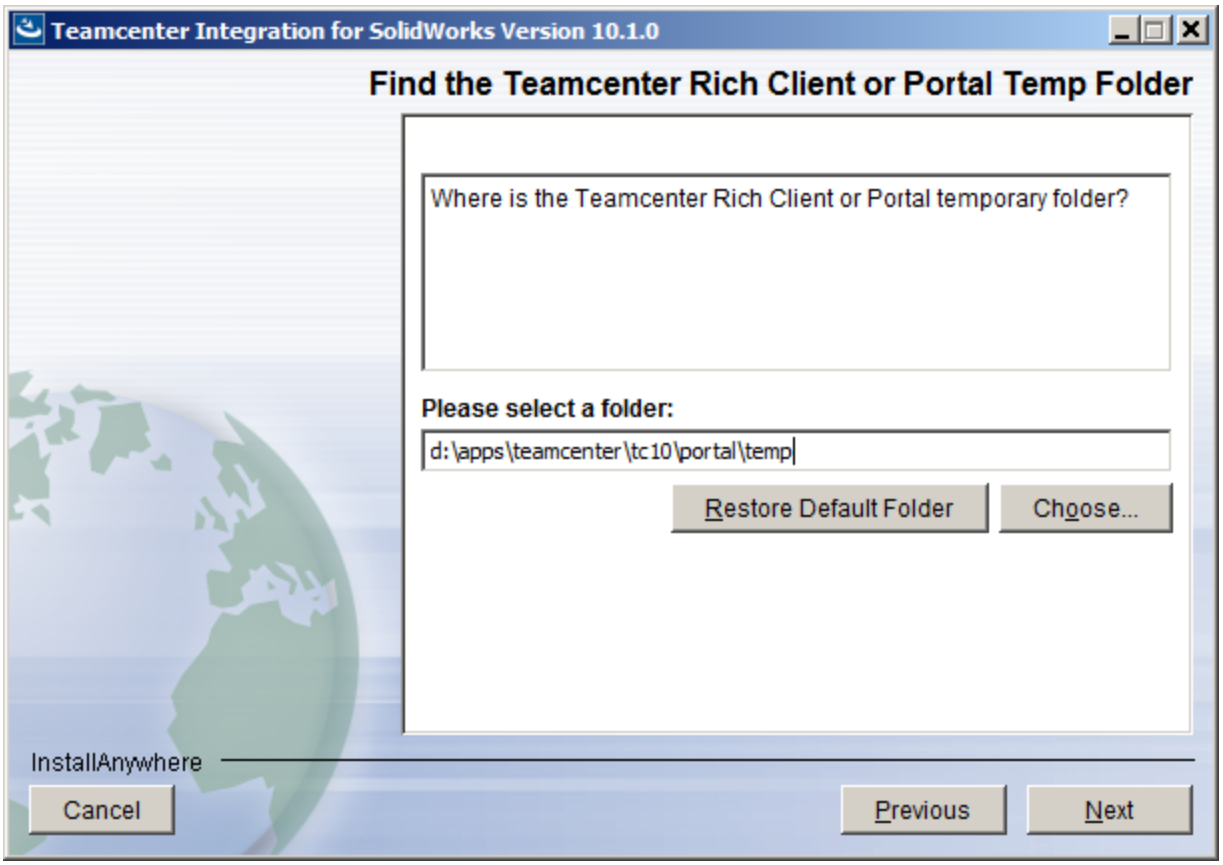


Figure 12 Select the Rich Client temporary directory

Select the Teamcenter FCC or TCCS directory

Specify the directory where the FMS or TCCS Client Cache is installed. This is usually found directly under TC_ROOT. The name and relative location of the directory may vary, depending upon the type of installation and the Teamcenter version.

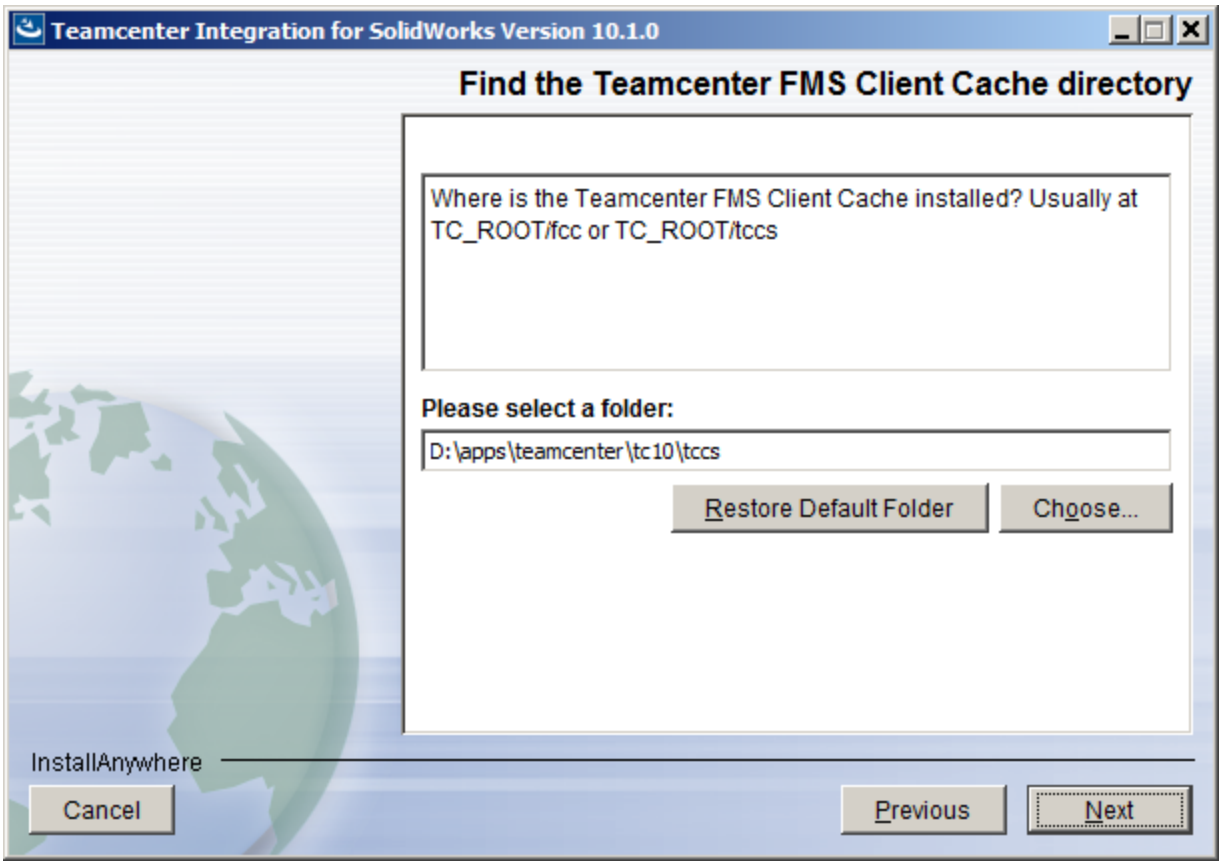


Figure 13 Select the client FCC or TCCS directory

Select the Teamcenter IIOP directory

Specify the directory where the Teamcenter IIOP Server is installed. This will usually be directly under the TC_ROOT location.

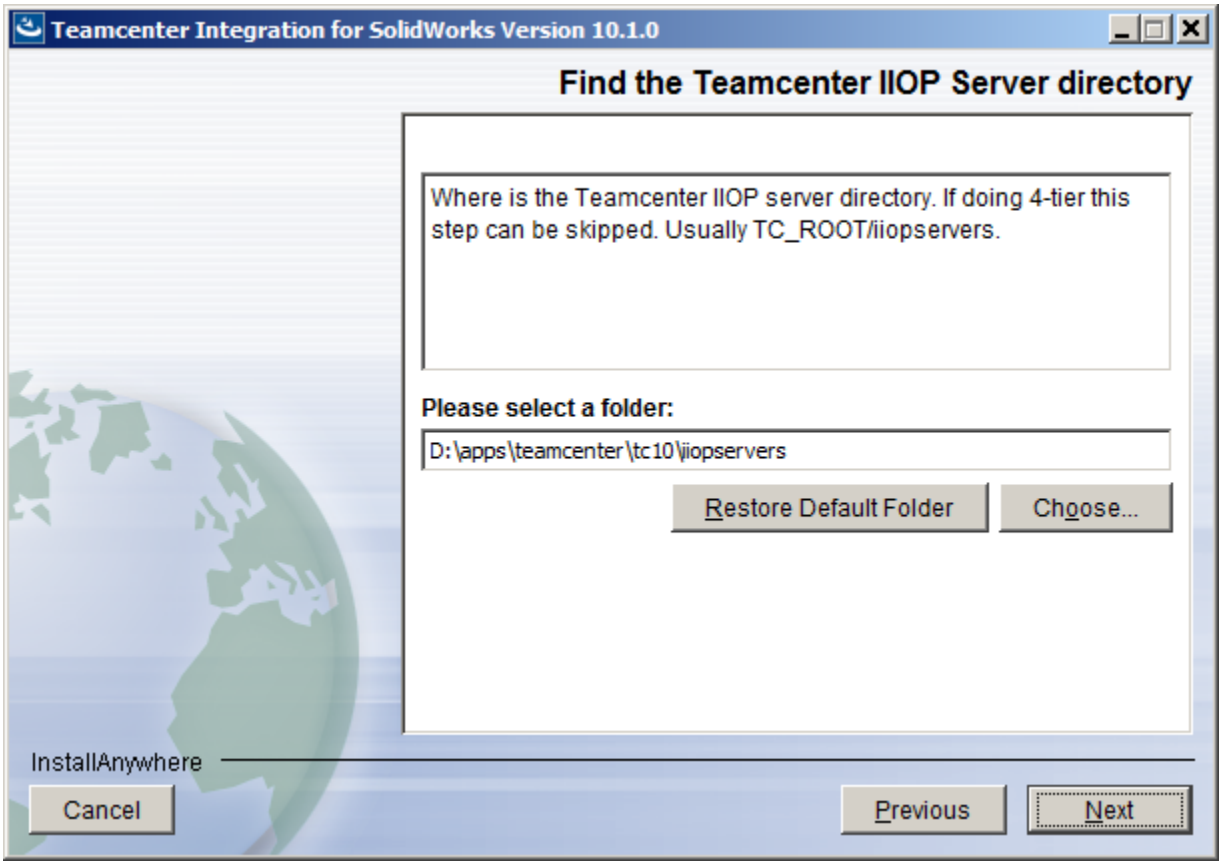


Figure 14 Select the IIOP server directory

Note that you will be prompted for the IIOP directory for both a **Typical Client/Server Install**, and a **Typical Client Install**. When the latter is performed on a four-tier client, there will be no valid IIOP directory. You should accept the default proposed location, and press "Continue" in the resulting warning dialog:

Figure 15 Warning message on a four-tier client

Select the Java Runtime Environment Folder

Specify the Java Runtime Environment (JRE) you wish to use to run the Teamcenter Integration for SolidWorks. The JRE located in the Rich Client or Portal directory is usually the best choice[4].

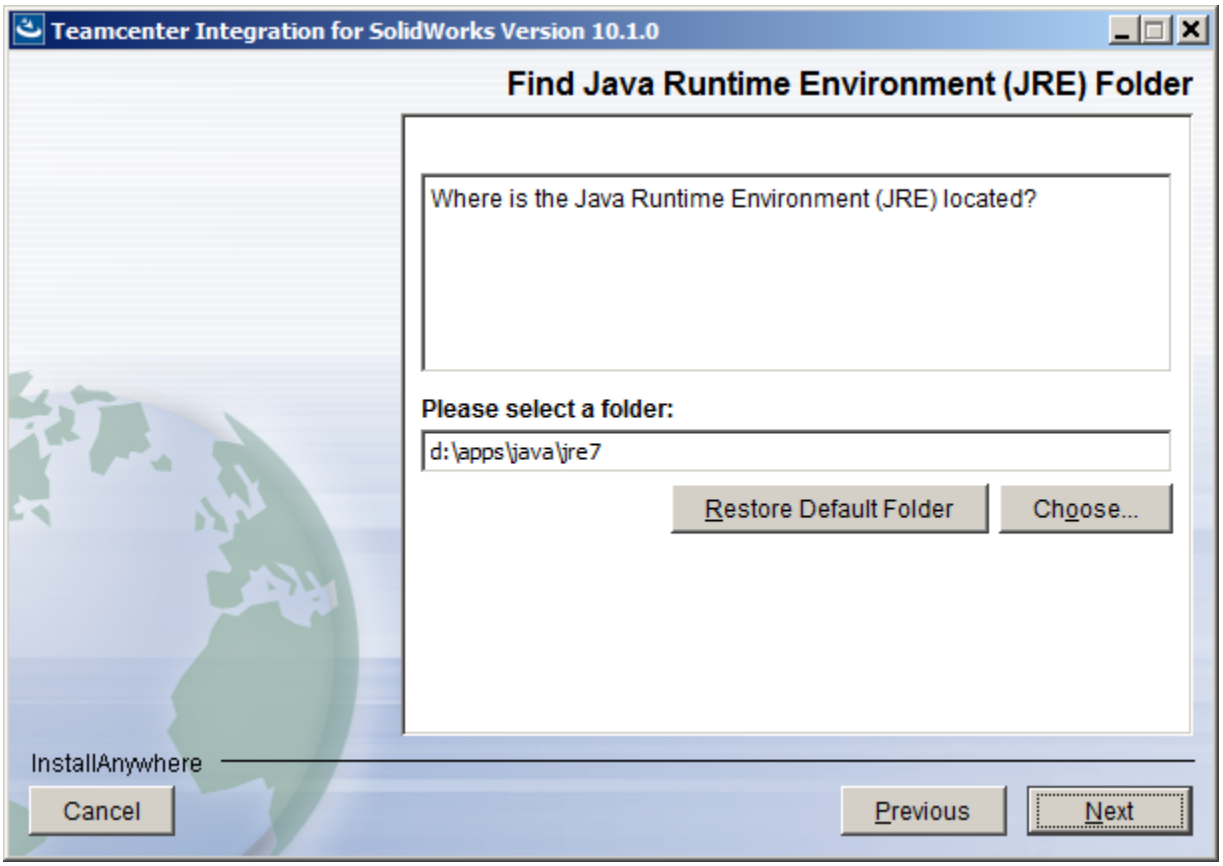


Figure 16 Select the JRE location

[4] Starting with version 10, Teamcenter no longer installs a JRE. Customers must provide their own compatible JRE installation.

Configure JT translation

The installer can configure the Teamcenter Integration for SolidWorks to save DirectModel (JT) files for viewing in the Teamcenter Rich Client. Figure 17 shows the dialog with the available options.

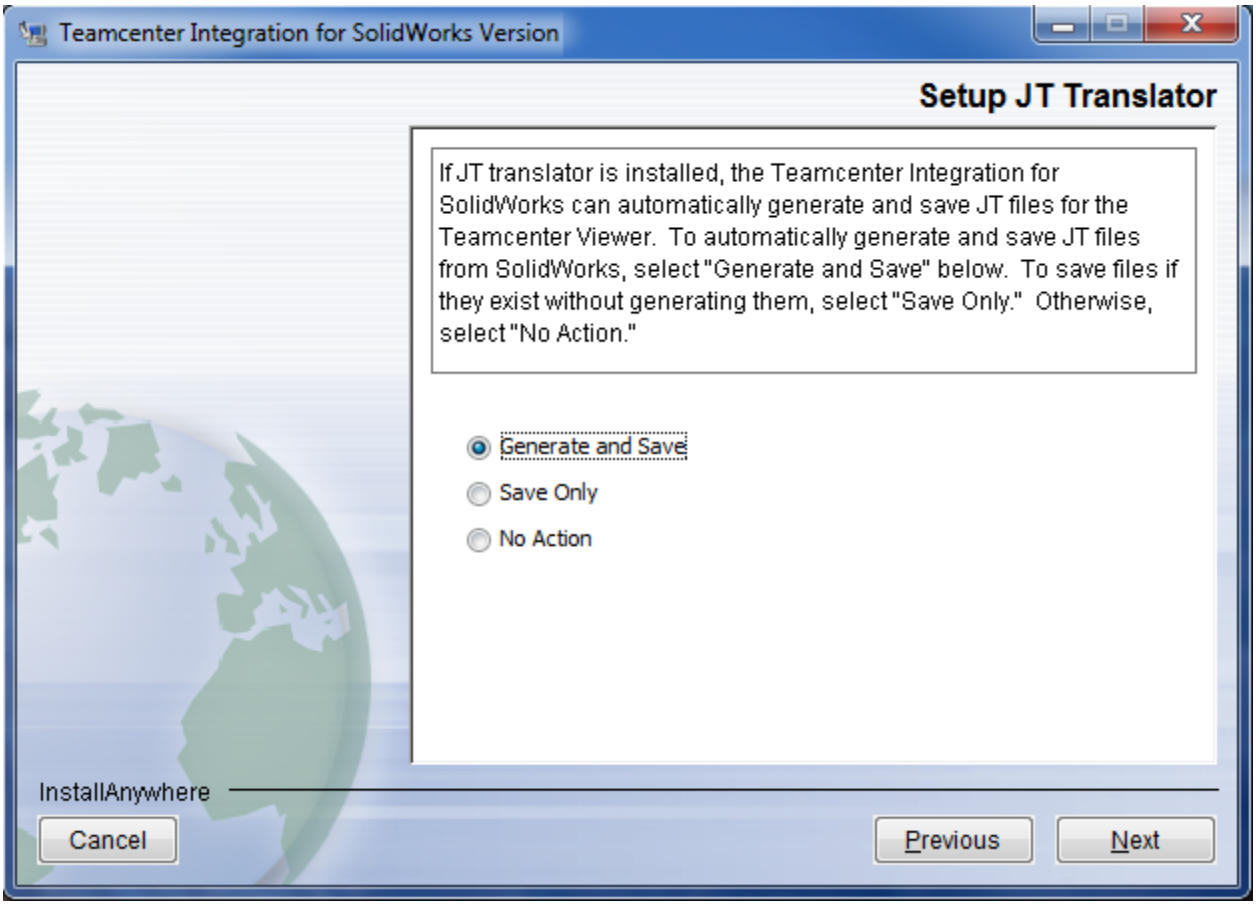


Figure 17 Configure JT translation

- The **Generate and Save** option generates and saves the JT file for a SolidWorks model while the model itself is saved to Teamcenter.
- The **Save Only** option saves the JT file for a SolidWorks model, while the model itself is saved to Teamcenter. This assumes that the user has previously generated the JT files and in the same directory which contains the SolidWorks files.
- The **No Action** option causes JT files to be ignored when models are saved to Teamcenter.

For more information on configuring the Teamcenter Integration for SolidWorks for saving JT files, see the section on Auxiliary files, in the integration's Administration guide.

Select the SolidWorks to JT translator location

If you selected “Generate and Save” in the previous dialog, you will be prompted for the location of the SolidWorksToJT translator installation. Enter or navigate to the appropriate directory location on your client workstation.

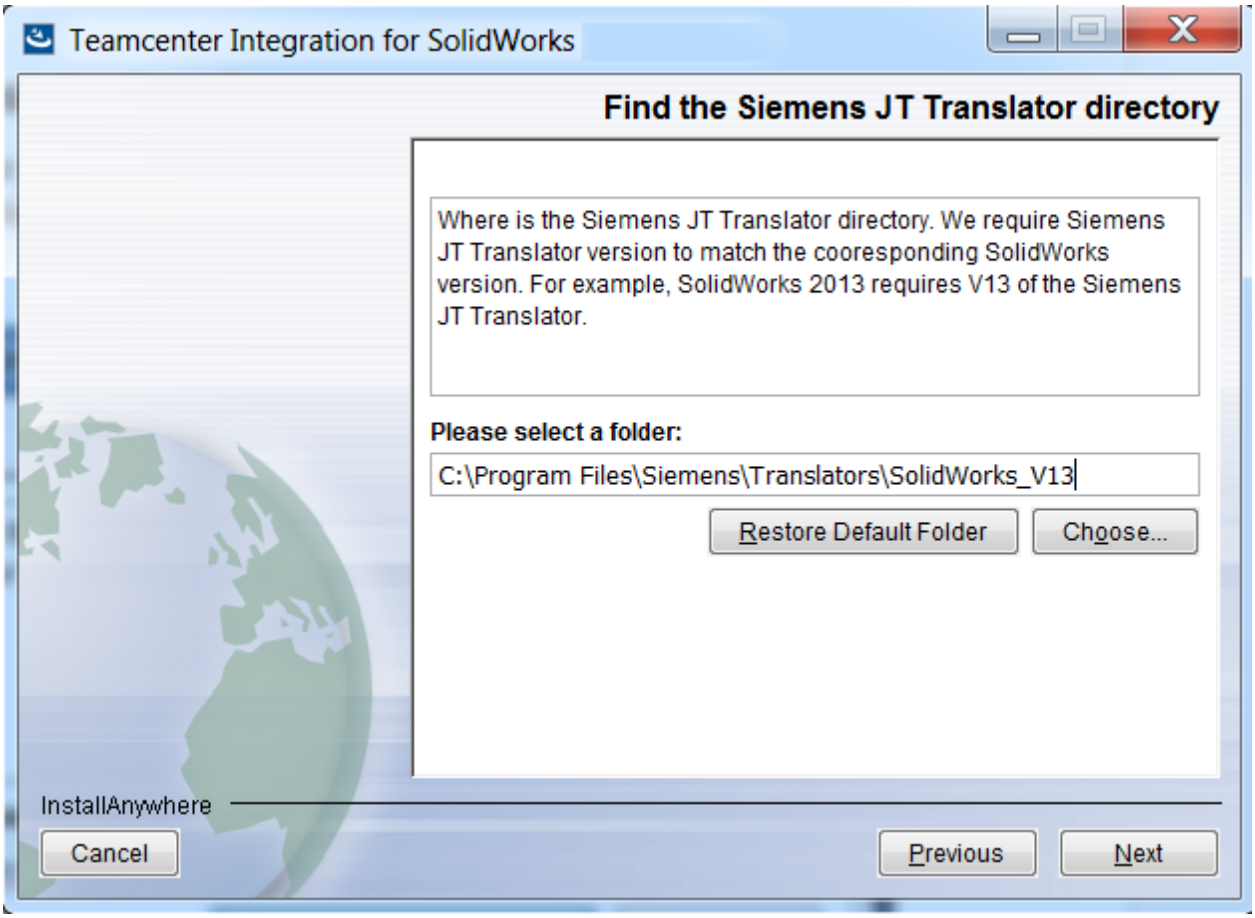


Figure 18 Select the JT translator location

Configure SolidWorks Toolbox interoperability

The installer can configure the Teamcenter Integration for SolidWorks to recognize and process SolidWorks Toolbox parts with special handling. Figure 19 shows the dialog with the available options. The **No Action** option will cause the integration to ignore the special characteristics and requirements of Toolbox parts, and to manage them like any other non-Toolbox data.

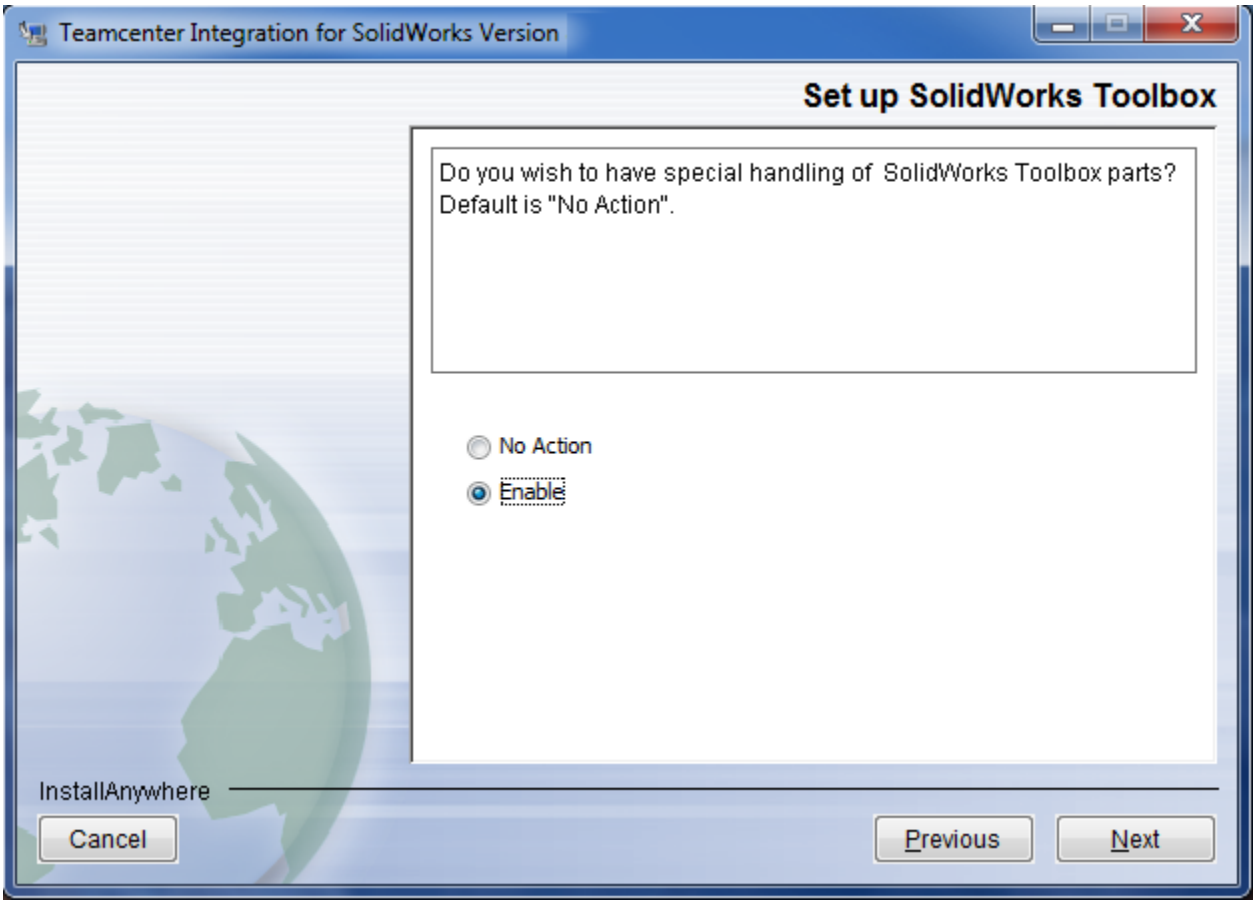


Figure 19 Configure SolidWorks Toolbox Interoperability

The **Enable** option sets sw.toolbox.dir in the swim.cfg, found in Integration installation directory, with the directory or directories where Toolbox parts are managed. See the Administration guide for more details on this and other configuration settings.

Select SolidWorks Toolbox folder locations

If **Enabled** is selected, you must enter at least one SolidWorks Toolbox path. The Windows registry is read for the default SolidWorks Toolbox location, but this can be changed and/or added to as necessary. If multiple directory locations are used by SolidWorks toolbox, separate each full path with a '**;**' character.

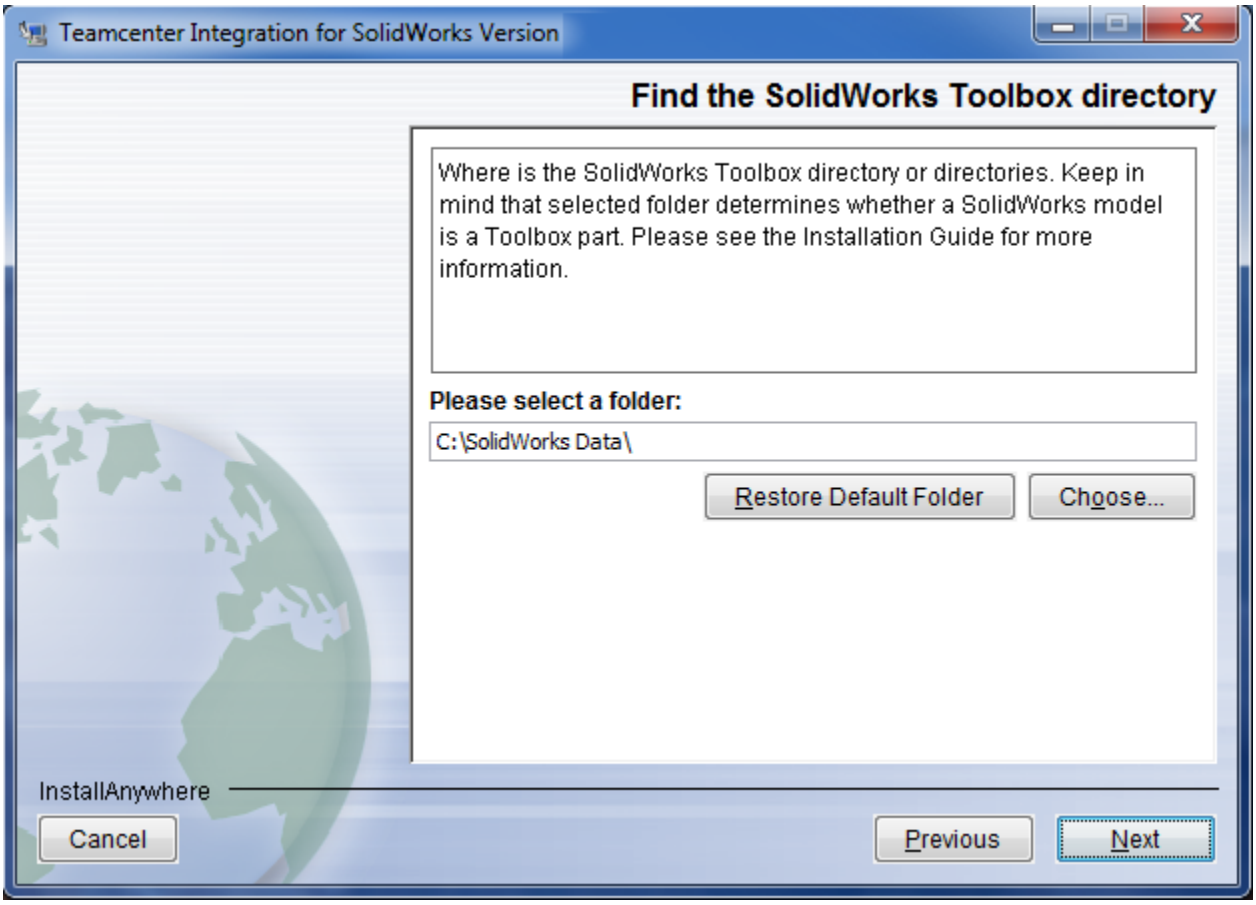


Figure 20 Select SolidWorks Toolbox locations

If the specified location is not recognized as a Toolbox-managed folder, you will see the following warning:

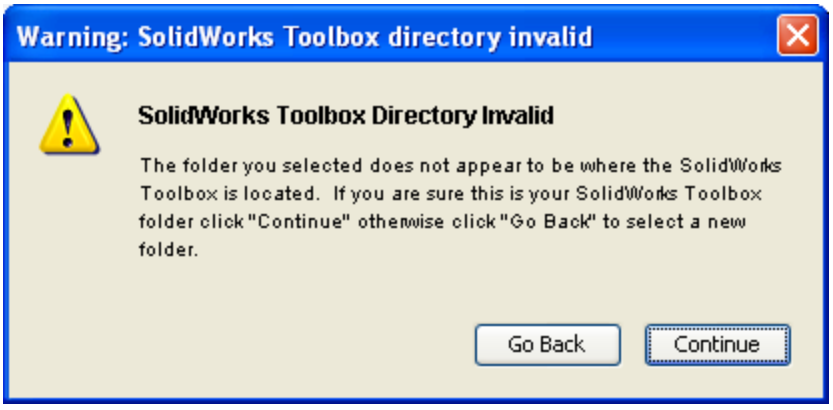


Figure 21 Warning message when selecting a non-Toolbox folder

If you proceed with this configuration, the Integration will manage all models in the specified folder, and its subdirectories, as if they were Toolbox parts. This may be useful for customers who do not use SolidWorks Toolbox, but who wish to manage standard parts in a fixed set of local directories.

Enable Teamcenter Multi-Site

The installer can configure the Teamcenter Integration for SolidWorks to allow users to transfer ownership of items from remote sites to their own site, making it possible to check out, modify, and check in the SolidWorks models that those items contain. The installer gives you the choice of whether the Transfer Ownership command should be available to users, as shown in Figure 22.

If you choose **Yes**, the Transfer Ownership command will appear in the Integration's shortcut menu when users right-click on a SolidWorks model in one of the Integration's dialogs. Please read the section on Multisite Configuration, in the Administration guide, for additional information on tasks that must be completed at your site and at remote sites before using the Transfer Ownership command in SolidWorks.

The Transfer Ownership command is configured by an entry in the swim.xml map file. Choosing **No** will leave the file's current contents unchanged. It does not remove any previous configuration that may have made the Transfer Ownership command available.

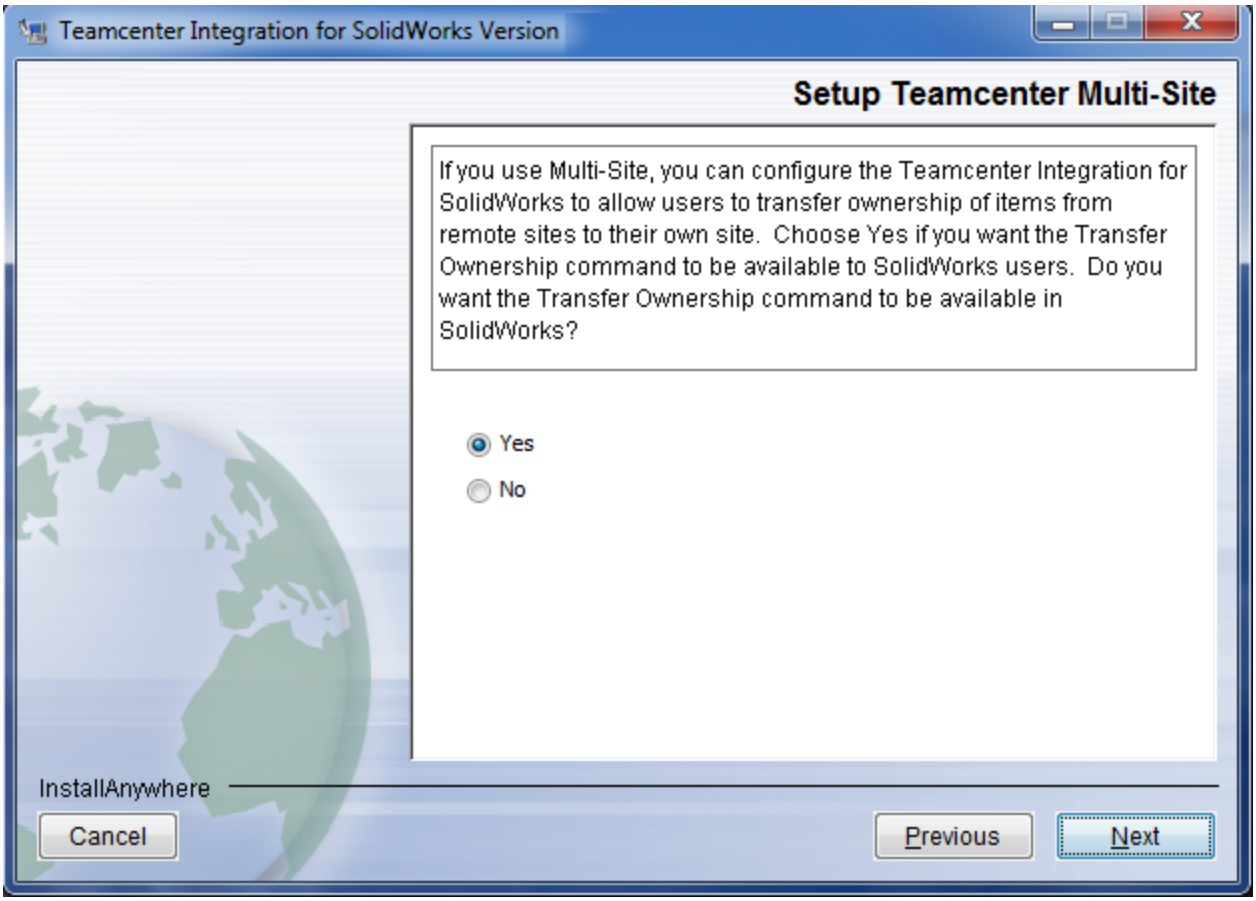


Figure 22 Configure Teamcenter Multi-site

Extract the BMIDE Template

Specify any accessible location for the BMIDE template files as shown in Figure 23. This prompt will only appear in a **Typical Server installation** install set. The template files will not be installed in a **Typical Client Installation** install set. For the **Typical Client/Server Installation** install set, the template files will be automatically installed in the %SWiM_DIR%\SWiM_BMIDE_Package{TC_version} directory This step simply extracts the template files into the specified directory location, so they can be loaded as directed in the template installation instructions, later in this document under **Teamcenter Environment Manager (TEM) procedures**.

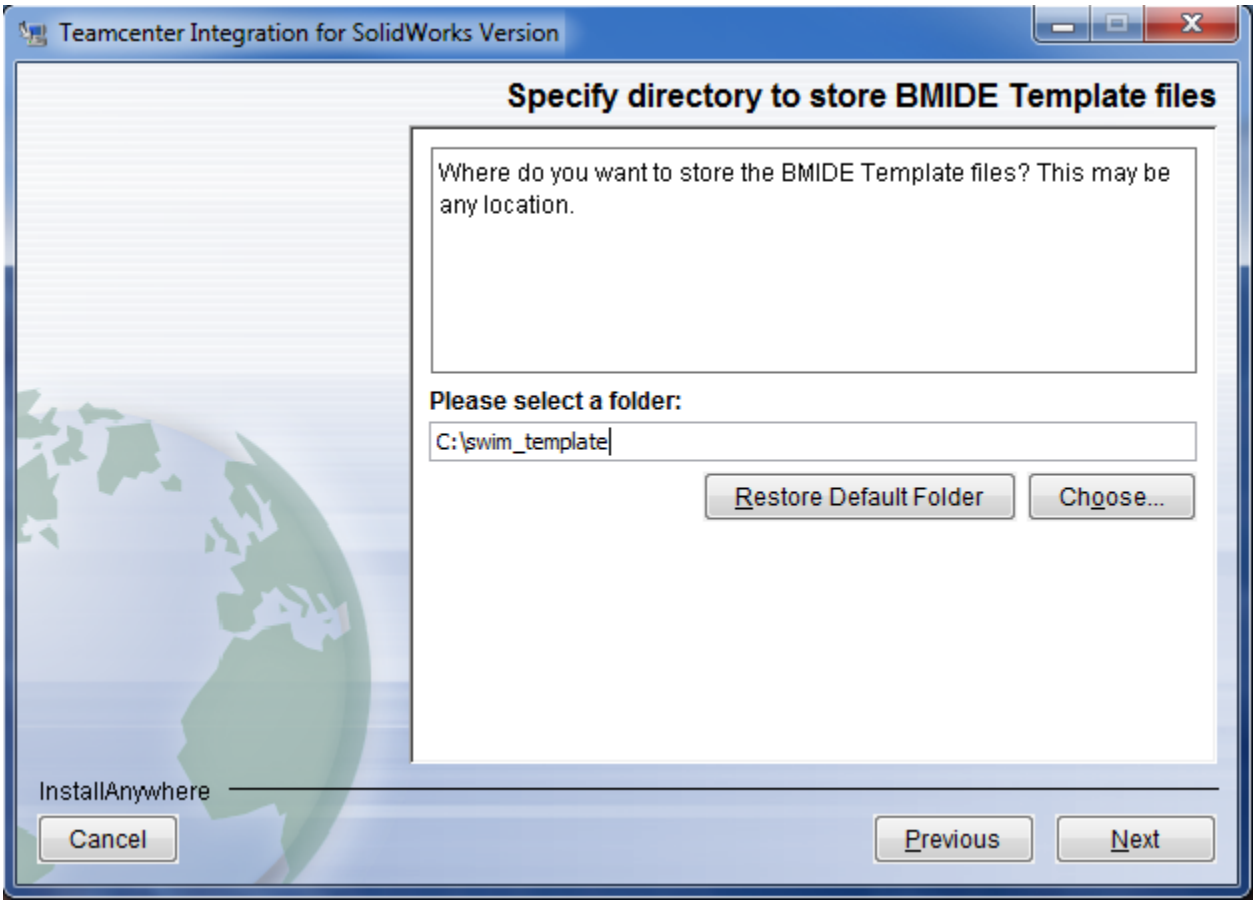


Figure 23 Extract the BMDE template

Pre-Installation Summary

Before beginning the installation process, the installer will display a pre-installation summary of all the options you have selected, as shown in Figure 24. If you are satisfied with all the settings, click **Install** to begin the installation. You can go back and make changes to these settings by using the **Previous** button.

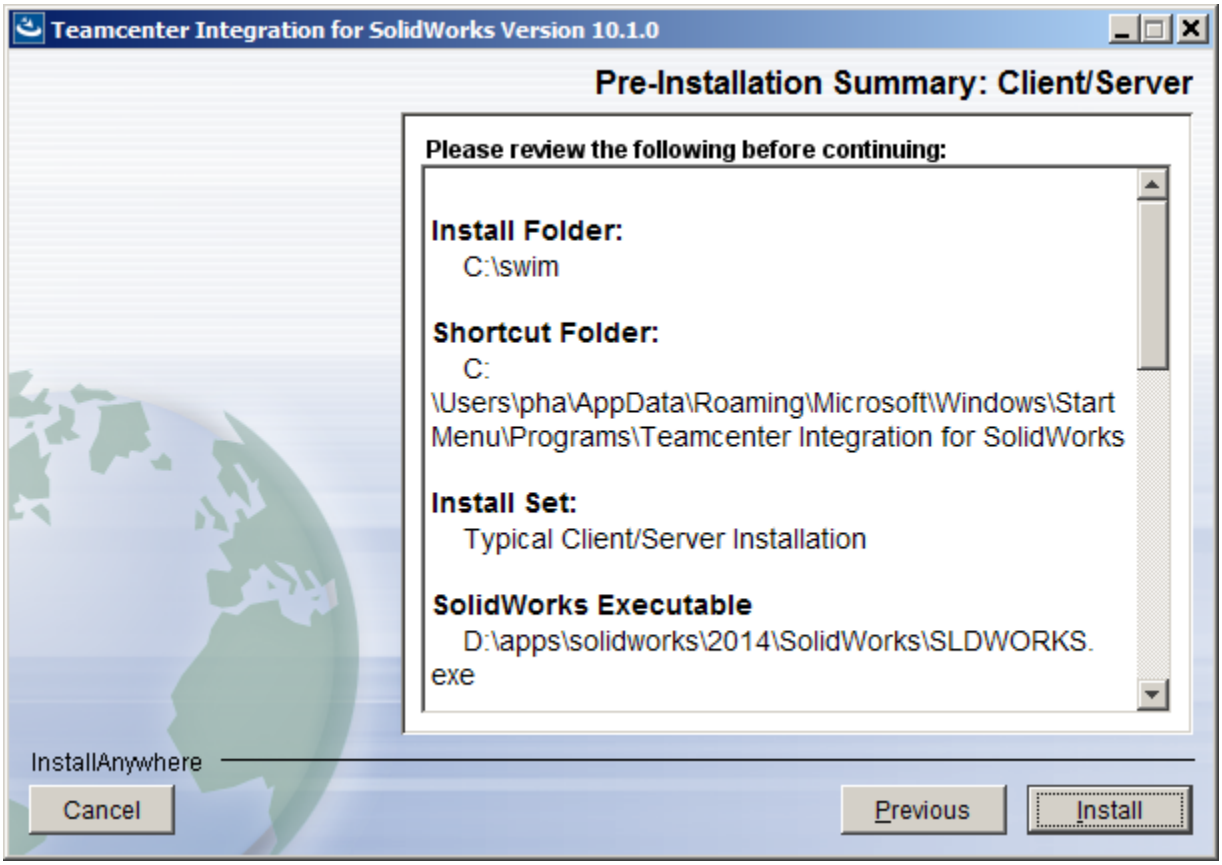


Figure 24 The Pre-installation summary

Installing

The installer will begin copying files and making any configuration changes you have selected, as shown in Figure 25.

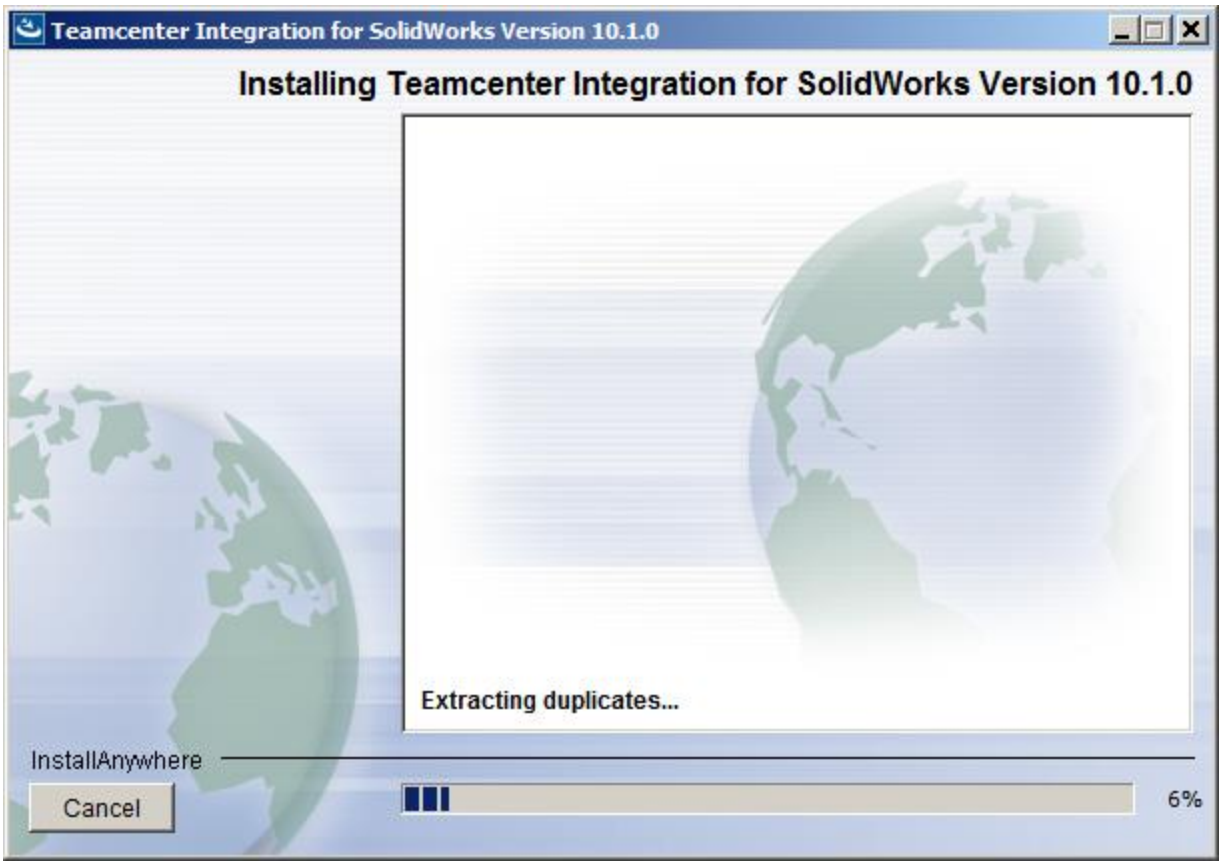


Figure 25 Installation progress

Post-Installation Summary

When the installer is finished, you will be notified of any steps that must be completed manually. Figure 26 shows the message that will be displayed if no steps were skipped, and all steps executed successfully.

Several additional tasks, not handled by the installer, may be necessary at some sites. Please read the section on **Teamcenter Environment Manager (TEM) procedures** for more information.

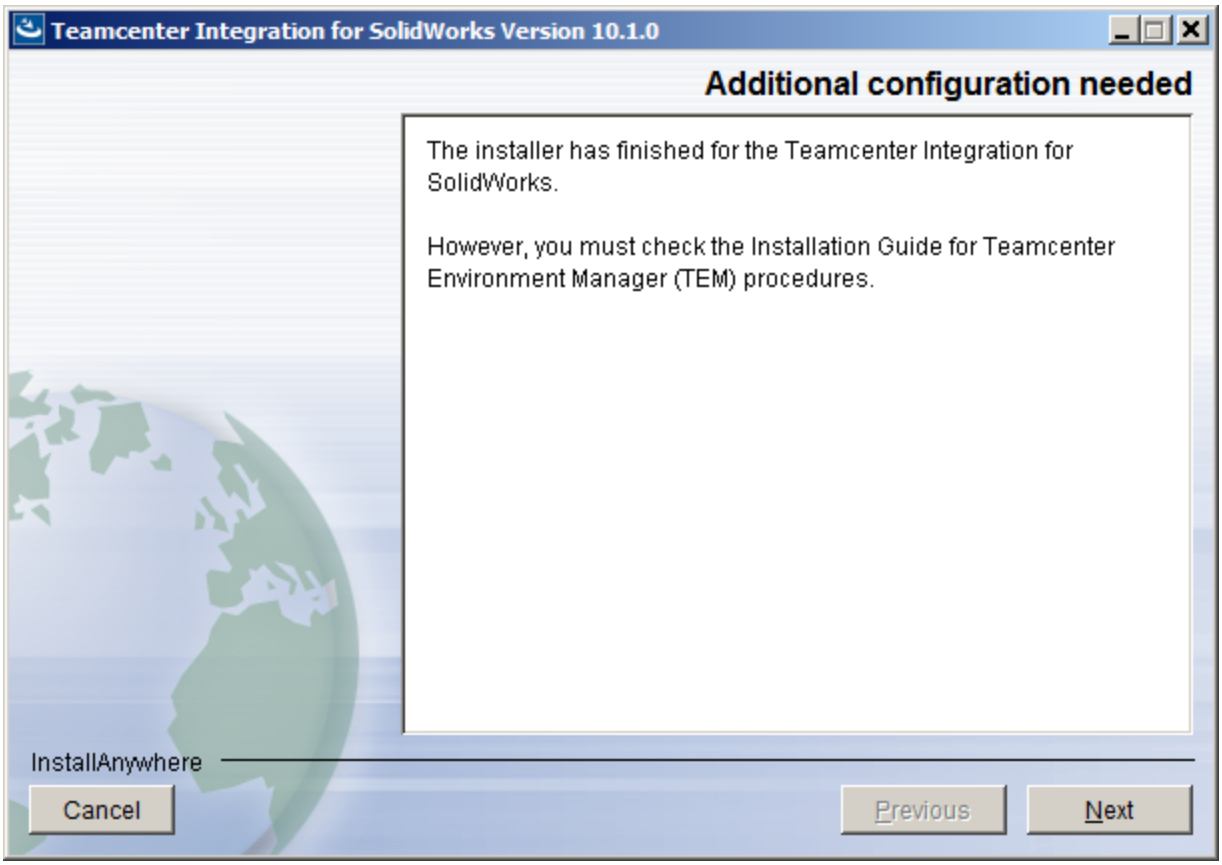


Figure 26 Post-installation summary

Server Installation

The server installation is a subset of the Client/Server Installation. A server installation uses the following steps from the Client/Server installation

- Select the Teamcenter version
- Select the Teamcenter Root Location
- Select the Teamcenter Data Directory
- Install SolidWorks Queries and preferences
- Enter the Teamcenter user name and password
- Install libtxd
- Select the location for the BMIDE template

Client Installation

The client installation is a subset of the Client/Server Installation. The same client installation is used regardless of whether the Teamcenter Rich Client installation is two-tier or four-tier. A client installation uses the following steps from the Client/Server installation.

- Select the Teamcenter version
- Select an Install Folder
- Select the Shortcut Folder

- Select the SolidWorks Start location
- Select the Teamcenter Rich Client Folder
- Select the Teamcenter Rich Client Temporary Folder
- Select the Teamcenter FCC directory
- Select the Teamcenter IIOP directory
- Select the Java Runtime Environment folder
- Configure JT translation
- Select the SolidWorks to JT translator location
- Configure SolidWorks Toolbox interoperability
- Select SolidWorks Toolbox folder locations
- Enable Teamcenter Multi-site

Silent Installation

The Teamcenter Integration for SolidWorks installer has a silent mode that can run without user interaction. This is only to be used for client installations; please use the interactive installer for any server installations. To use silent mode, create a swimsetup.properties file that defines the installation options for your site. To create a swimsetup.properties file, edit the template file samplesetup.properties, which can be found in any existing client installation directory. Documentation for the supported installation options is available in the template file. **In a Windows environment, use double-backslashes for any paths in the swimsetup.properties file**. For example, instead of the path "`c:\swim`" use the path "`c:\\swim`". Once you have finished editing the template file, either place the file in the same directory as the swimsetup.exe program or use the "-f" switch to swimsetup.exe to specify the path. For example, the following command could be used to install from a properties file located on a mounted network drive.

```
swimsetup.exe -f u:\defaultinstallfiles\swimsetup.properties
```

Teamcenter Environment Manager (TEM) procedures

To complete the installation for Teamcenter, one of the TEM procedures below must be performed, on each server database. Start the TEM and advance to the **Feature Maintenance** screen.

The steps from this point onward are divided into two sections. The first section applies if you are installing the Teamcenter Integration for SolidWorks for the first time. The second section applies if you have already installed the integration, and are now updating it to a newer version. Be sure to locate and follow the steps in the appropriate section.

New Installation

Follow the steps in this section if the Teamcenter Integration for SolidWorks has never been previously installed in the selected Teamcenter configuration.

Loading the Integration templates

Select **Add/Remove Features**, then **Next**

Figure 27 - Installing the integration for the first time

Press the **Browse** button, and then navigate to the location where the BMIDE templates were saved during the Integration's server installation process.

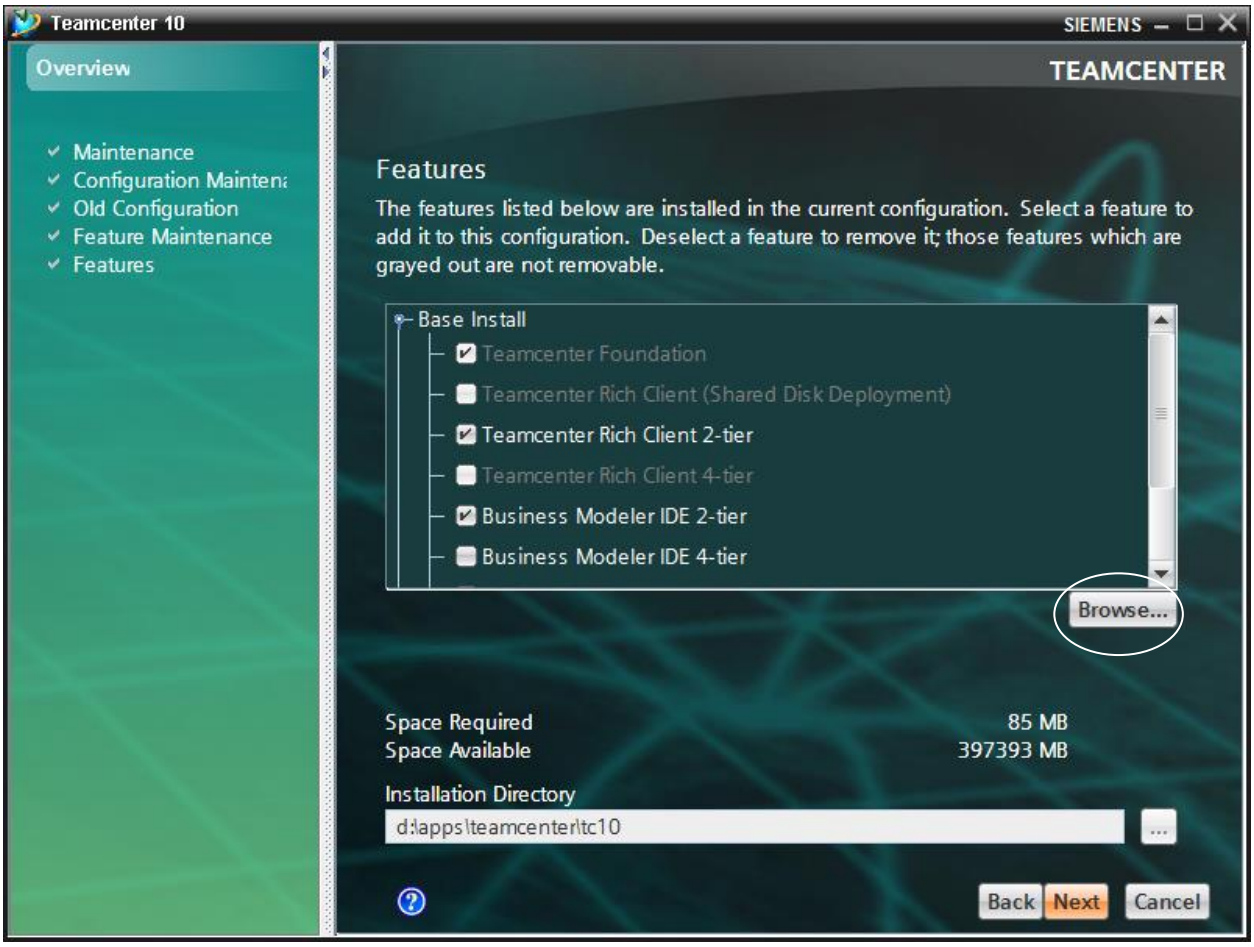


Figure 28 Browse to find the integration template

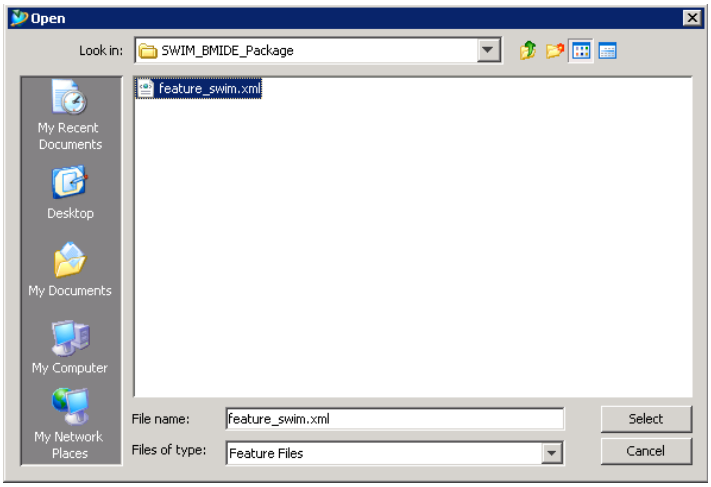


Figure 29 Select the feature_swim.xml file

Select the **feature_swim.xml** file from this location, then return to the "Select Features" screen and check the box next to the **Integration for SolidWorks** option[5]

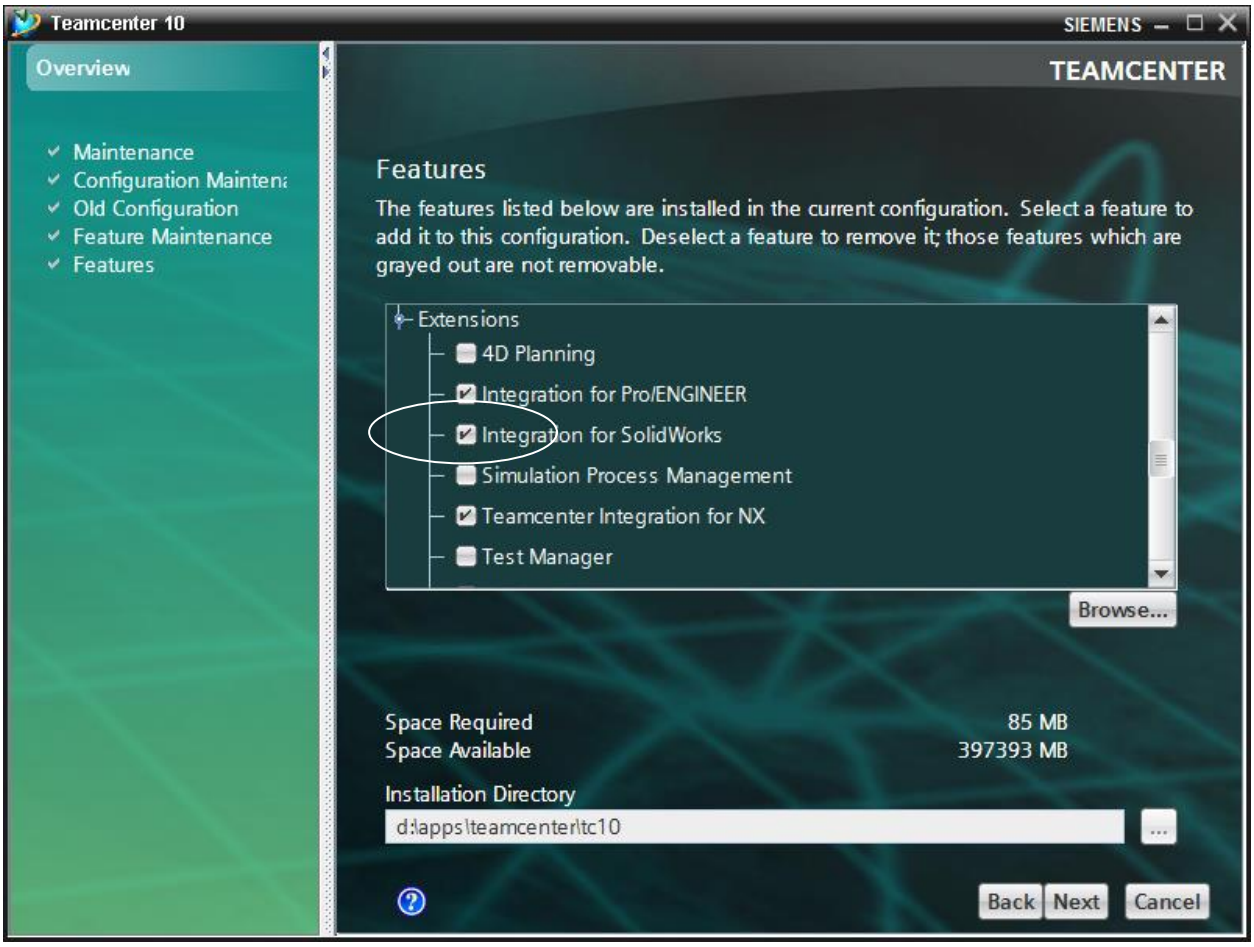


Figure 30 Select the integration for installation

Then press **Next**

[5] Different versions of Teamcenter will display the Integration's entry in different locations within the product tree in this dialog. You may have to expand the tree structure to see the Integration's check box.

Enter the Teamcenter Administrative User password:

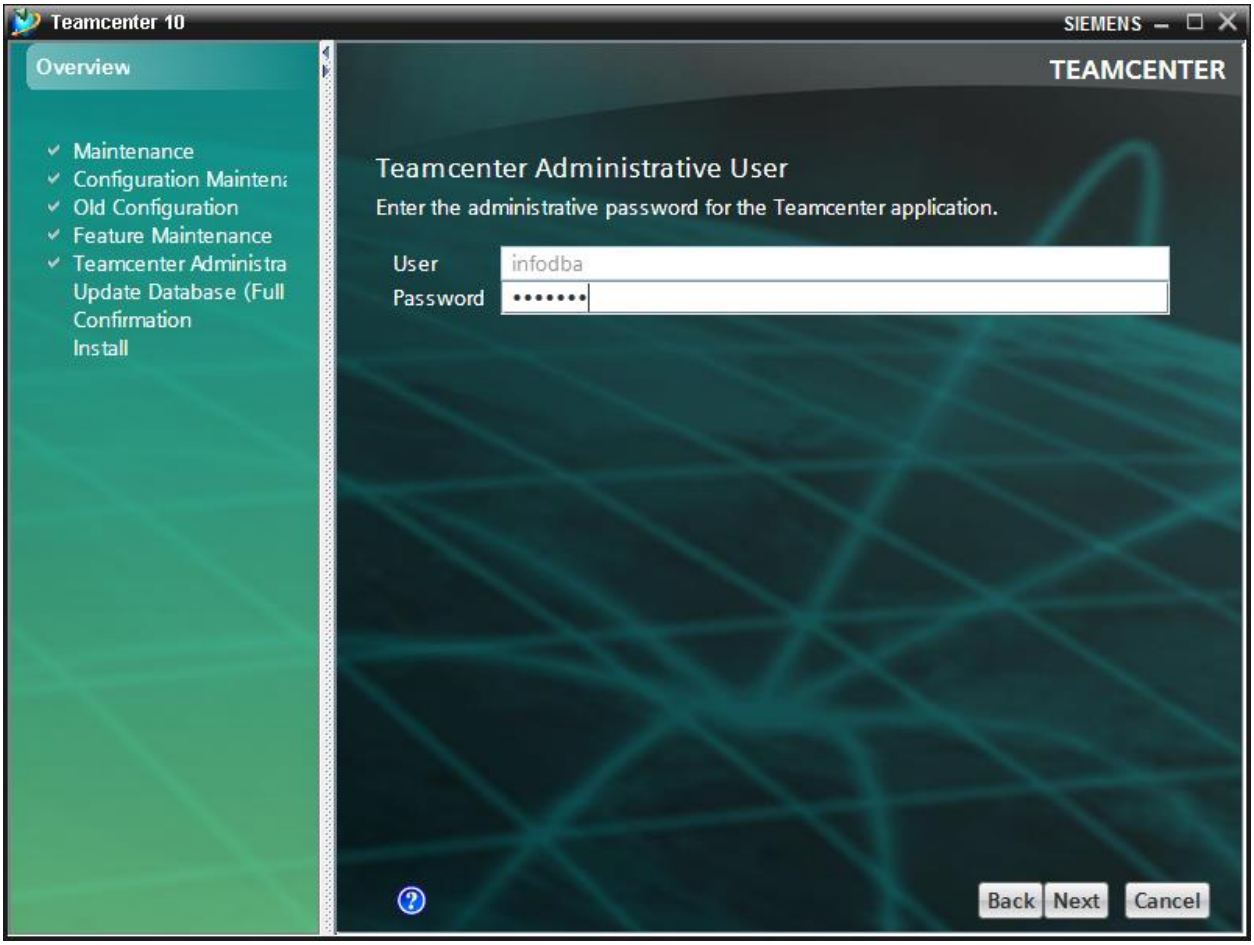


Figure 31 Enter the Teamcenter user's credentials

Press **Next** on the next two screens, and the installation will begin. A successful installation of the integration template will be shown by this confirmation screen:

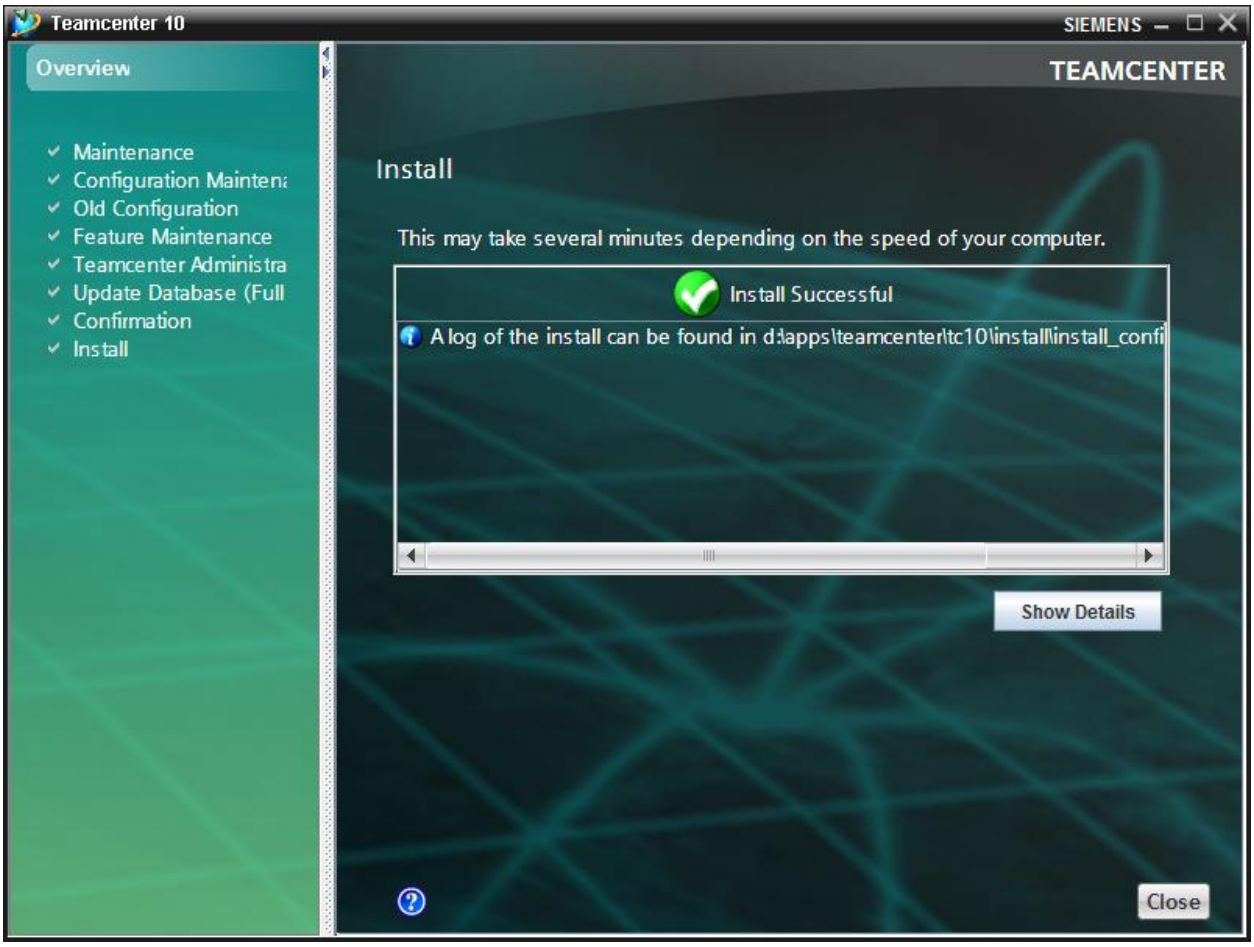


Figure 32 Confirmation of successful installation

Adding the Integration template to the Business Modeler IDE (BMIDE) client

If you have installed BMIDE and you wish to add the Teamcenter Integration for SolidWorks template to the BMIDE configuration, perform the following steps.

On the **Feature Maintenance** screen, select **Add/Update Templates for working within the Business Modeler IDE Client**, then press **Next**

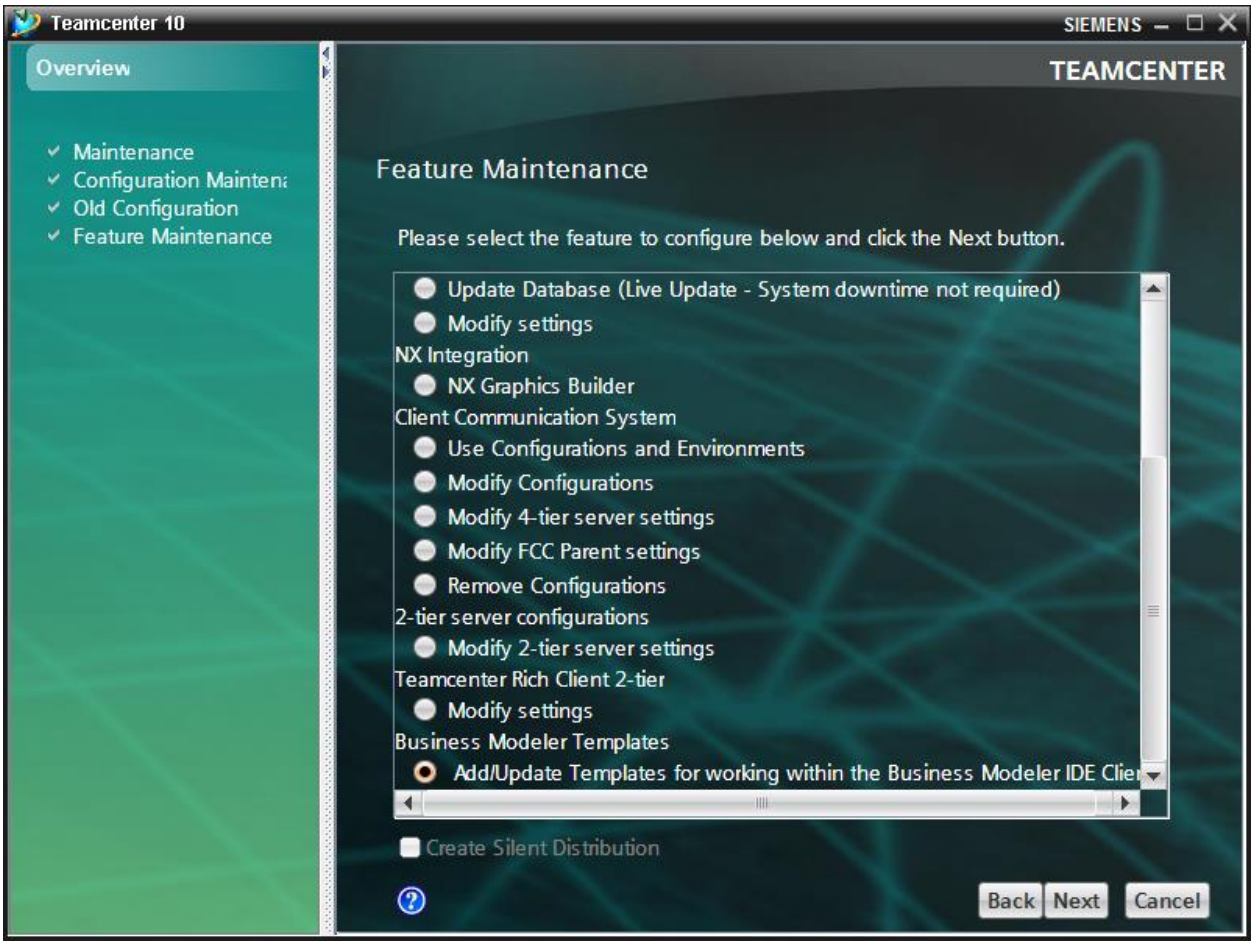


Figure 33 Adding the integration templates to the BMIDE client

Select **Add** and then select the SolidWorks integration from the list of installed templates

Figure 34 Add a new template to BMIDE

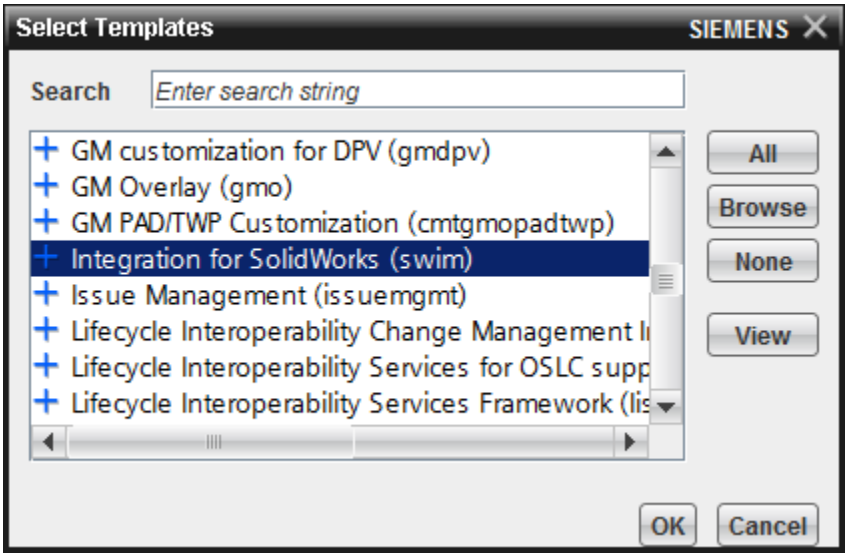
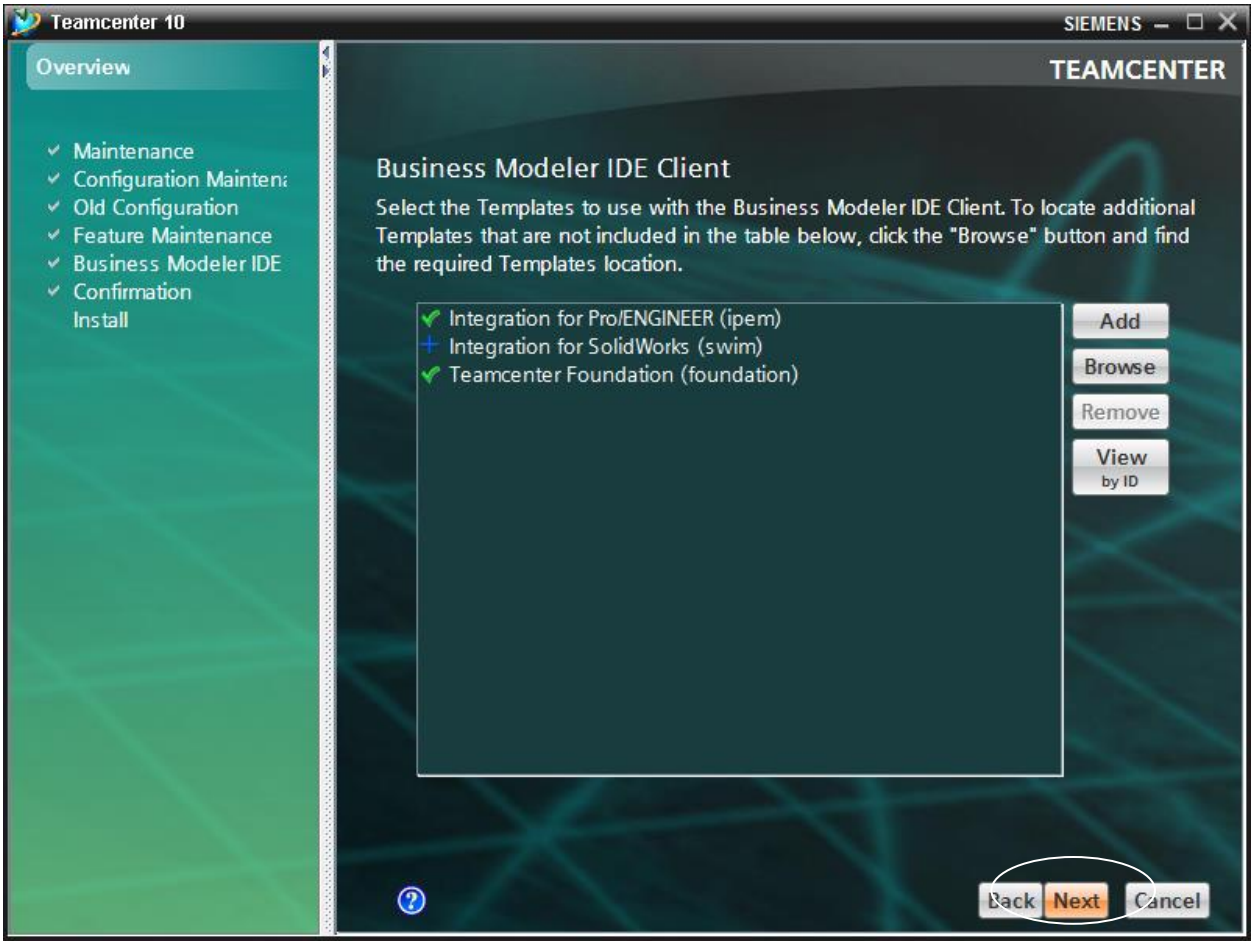


Figure 35 Select the SolidWorks integration

Select **Next** to advance to the confirmation screen



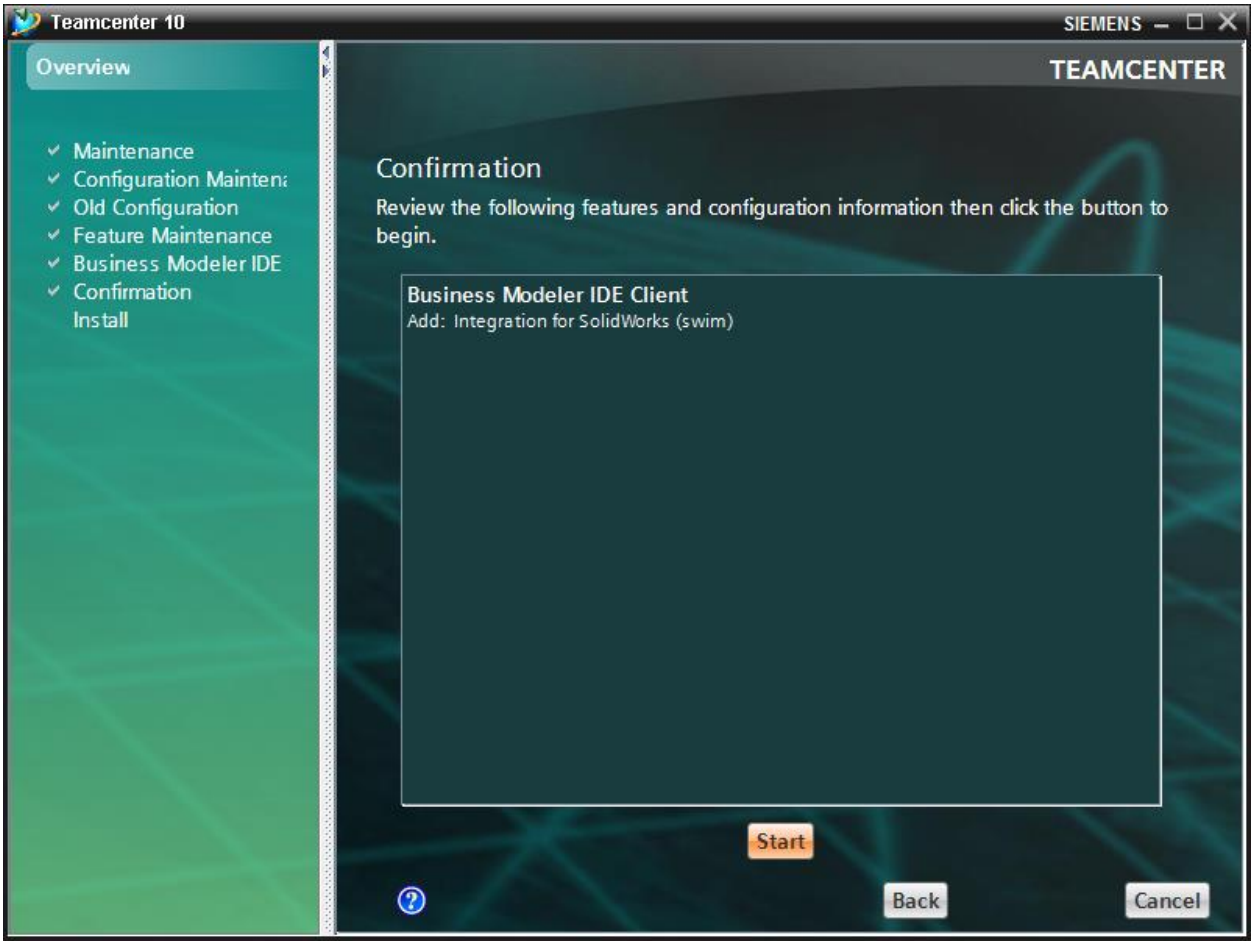


Figure 36 - BMIDE template confirmation

Then press **Start** to begin the installation. When prompted, browse to and select the **feature_swim.xml** file, from the location specified during the Integration server installation process.

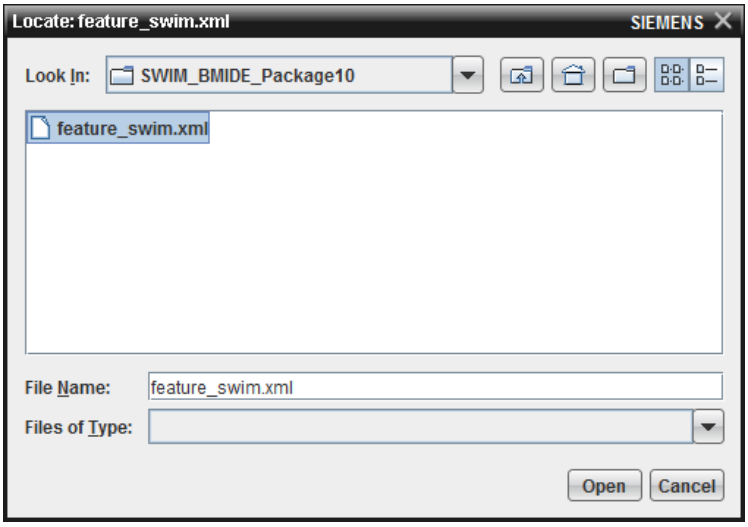


Figure 37 - Selecting the integration template file

Successful installation will be confirmed on the summary display:

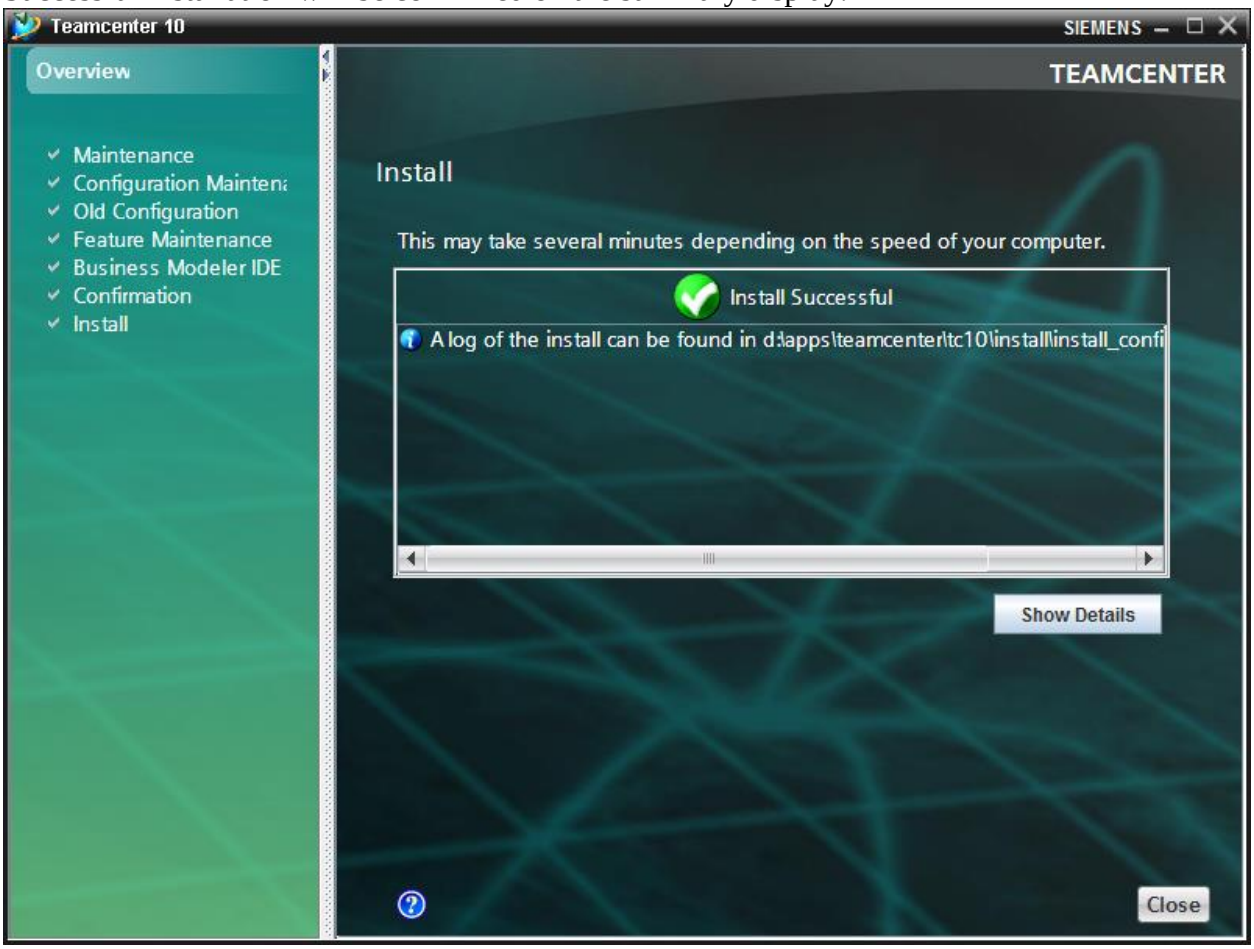


Figure 38 Confirmation of successful installation

Update Installation

Follow these steps if the Teamcenter Integration for SolidWorks is already installed in the selected configuration, and you are updating to a newer version of the integration. You should always perform this step whenever upgrading to a new version of the Teamcenter Integration for SolidWorks.

Check existing TEM key properties

For proper update of the integration template, the GUID (Globally Unique Identifier) stored by Teamcenter for the SolidWorks integration must match the GUID provided by the integration template. Teamcenter stores the GUID in two files. Before starting the update, check these files. Their integration template GUID values must match the value provided by the integration template. If not, then make backup copies and edit one or both files, such that all three GUID values match exactly. Keep in mind that the correct value is always the one in the **feature_swim.xml** file from the SWIM version that you are updating to.

The two files are:

- %TC_ROOT%\install\configuration.xml
- %TC_ROOT%\install\install\async_templates.xml

Within each Teamcenter file, search for the string "**Integration for SolidWorks**", then find the GUID value within the same **<feature>** or **<features>** tag. Here is an example from the async_template.xml file, showing the correct GUID value:

```
<feature>
    <name value="Teamcenter Integration for SolidWorks"/>
    <property name="template name" value="swim"/>
    <guid value="F25FD8C1B9DB7DF40C588AD06A702659"/>
    <property name="template_match_1" value="POM_class,name,swim_MetaData"/>
    <property name="template_match_2" value="ImanType,type_name,SWPrt"/>
</feature>
```

If you attempt to update the template without first correcting a GUID mismatch, you will see a warning icon similar to the following:

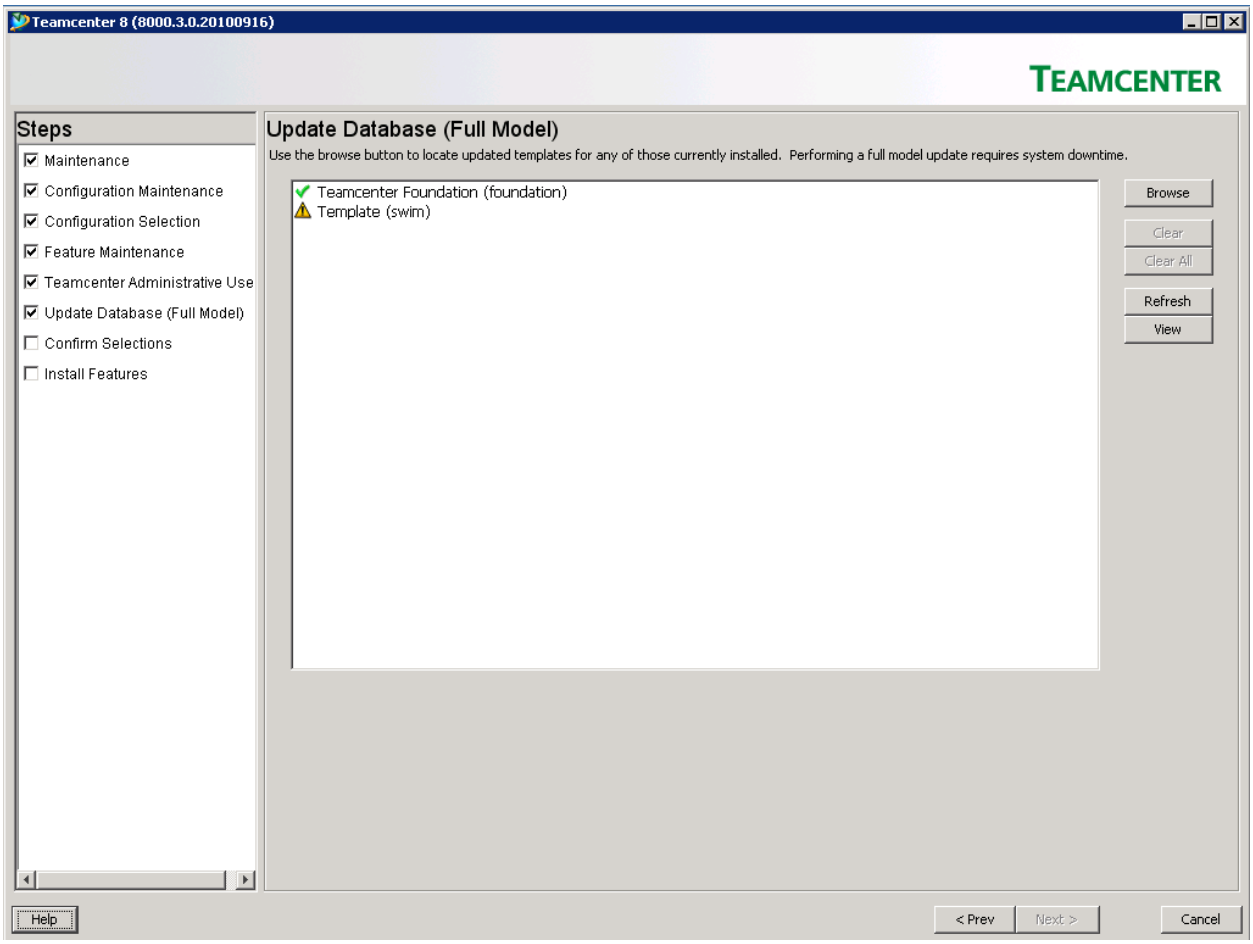


Figure 39 An indication of mismatched GUID values

This is an indication that one or both of the Teamcenter files contains an incorrect GUID value for the integration. Cancel TEM, check the files, and make the necessary corrections, then restart TEM.

Updating the Integration's template

Start TEM, and then advance to the **Feature Maintenance** screen. Select **Update Database (Full Model – System downtime required)** and then press **Next**.

Figure 40 Beginning an update installation

Enter the Teamcenter administrator password, then press **Next**

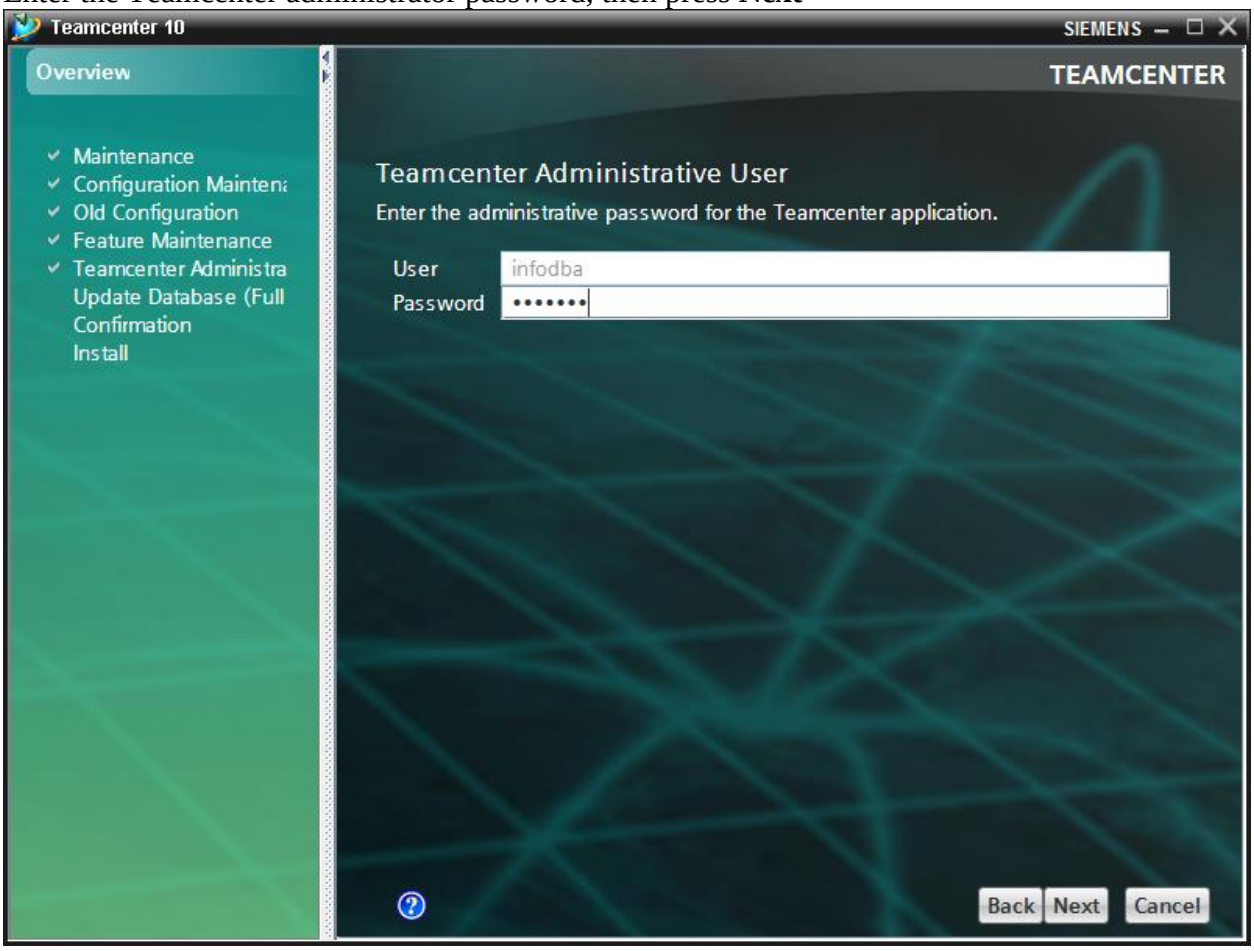


Figure 41 Enter the Teamcenter user's credentials

On the **Update Database** screen, press the **browse** button to find and select the **feature_swim.xml** file in the location where it was saved during execution of the server install process for the new version of the integration

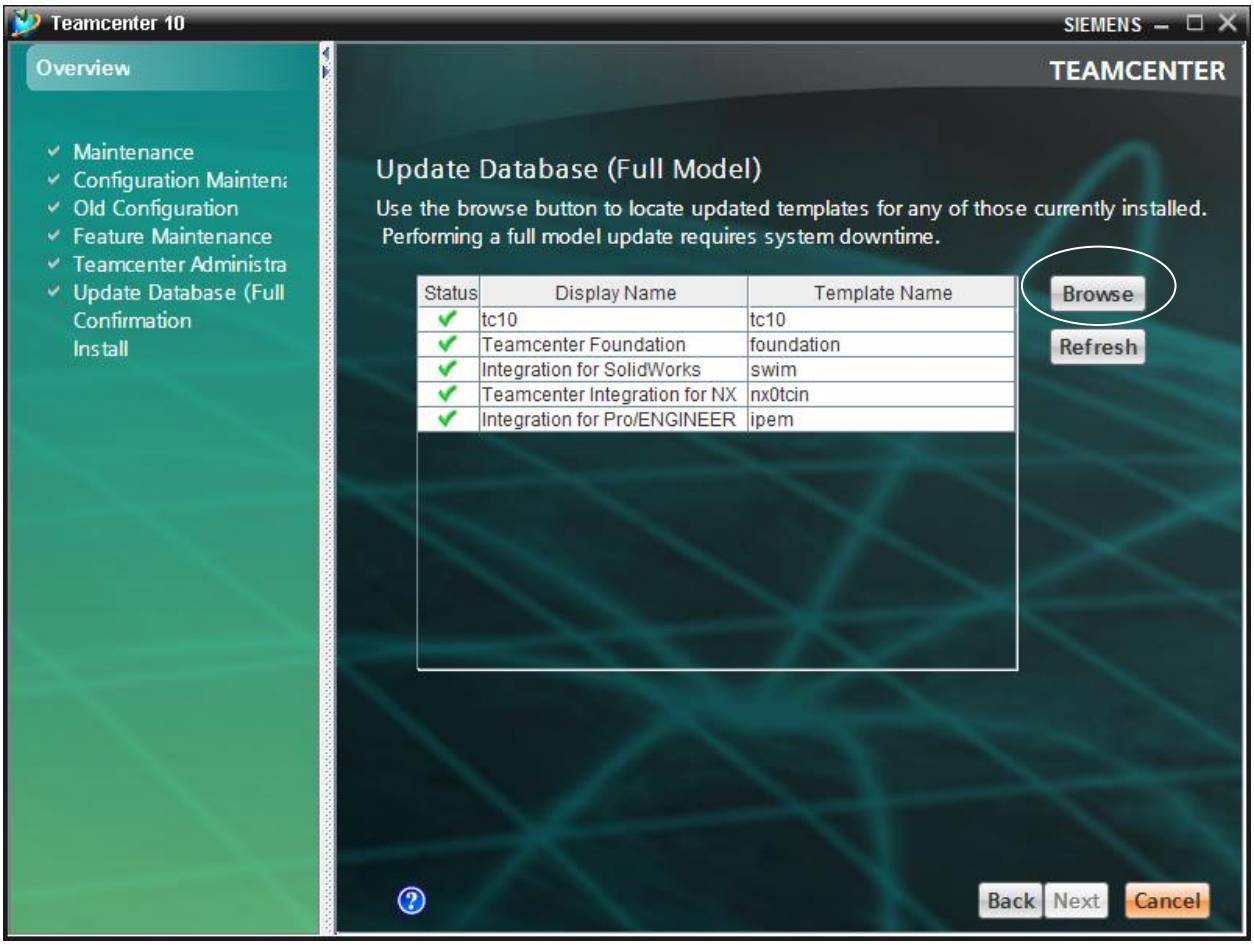


Figure 42 Browse to find the new integration template

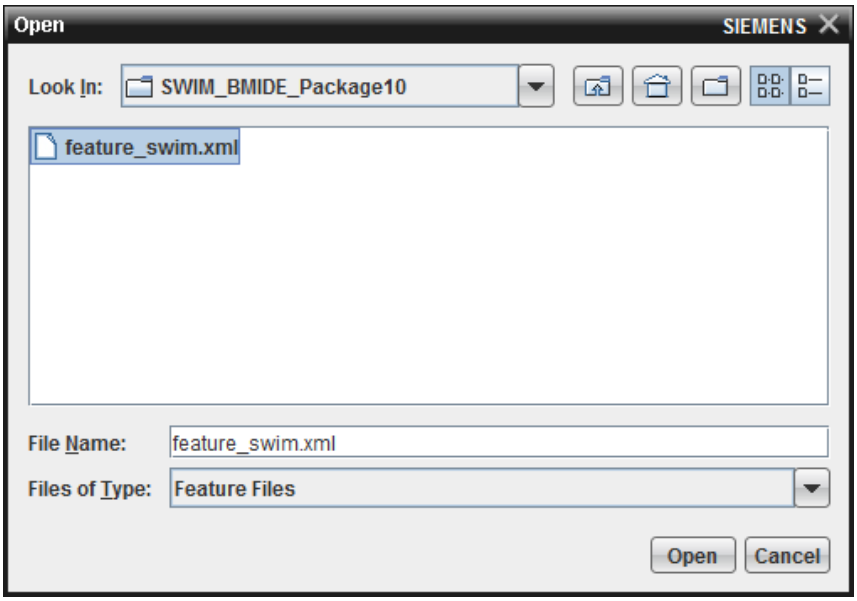


Figure 43 Select the feature_swim.xml file

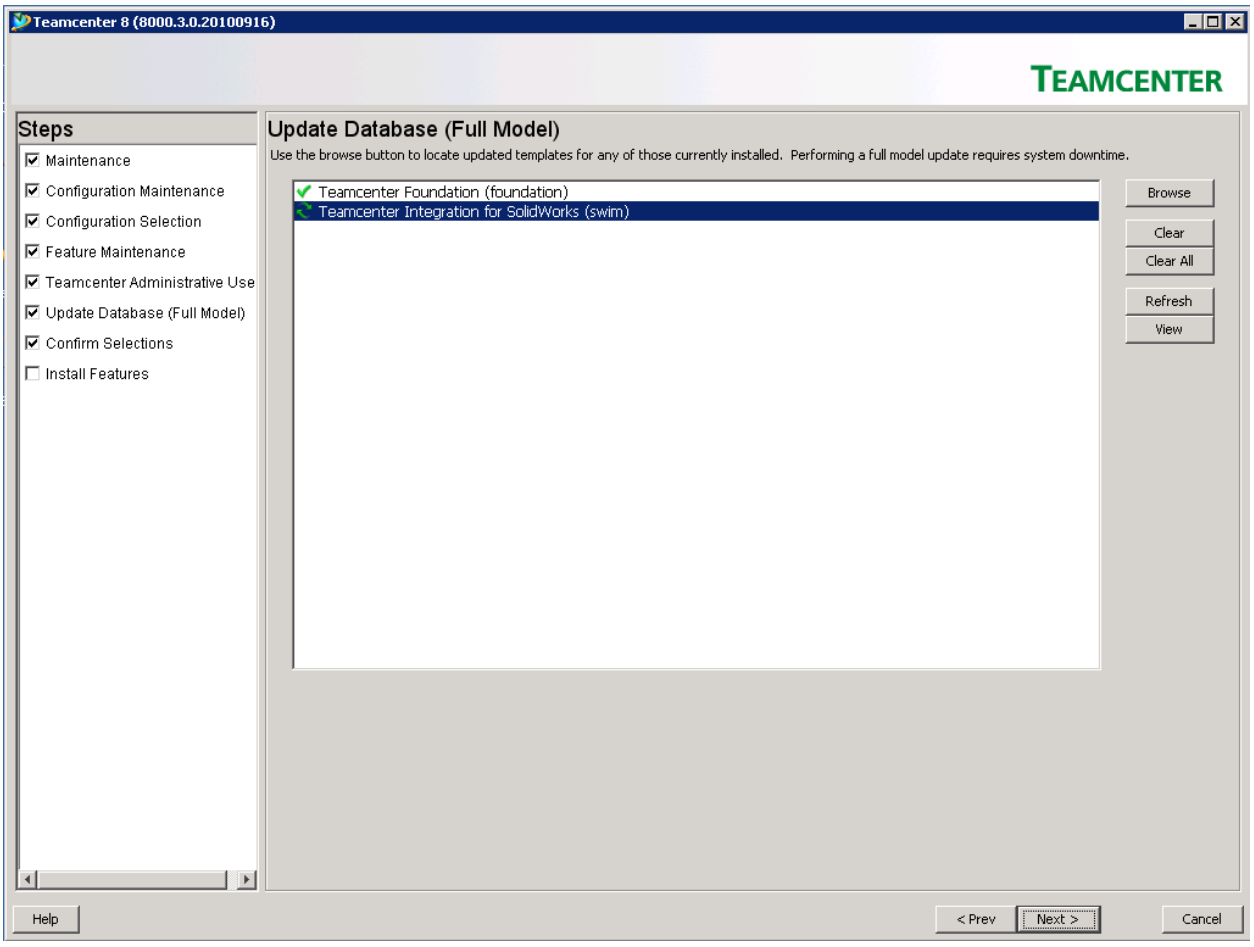


Figure 44 Updating the integration template

Press **Next** and **Start** on the following screens. Successful update will be indicated by the following confirmation message:

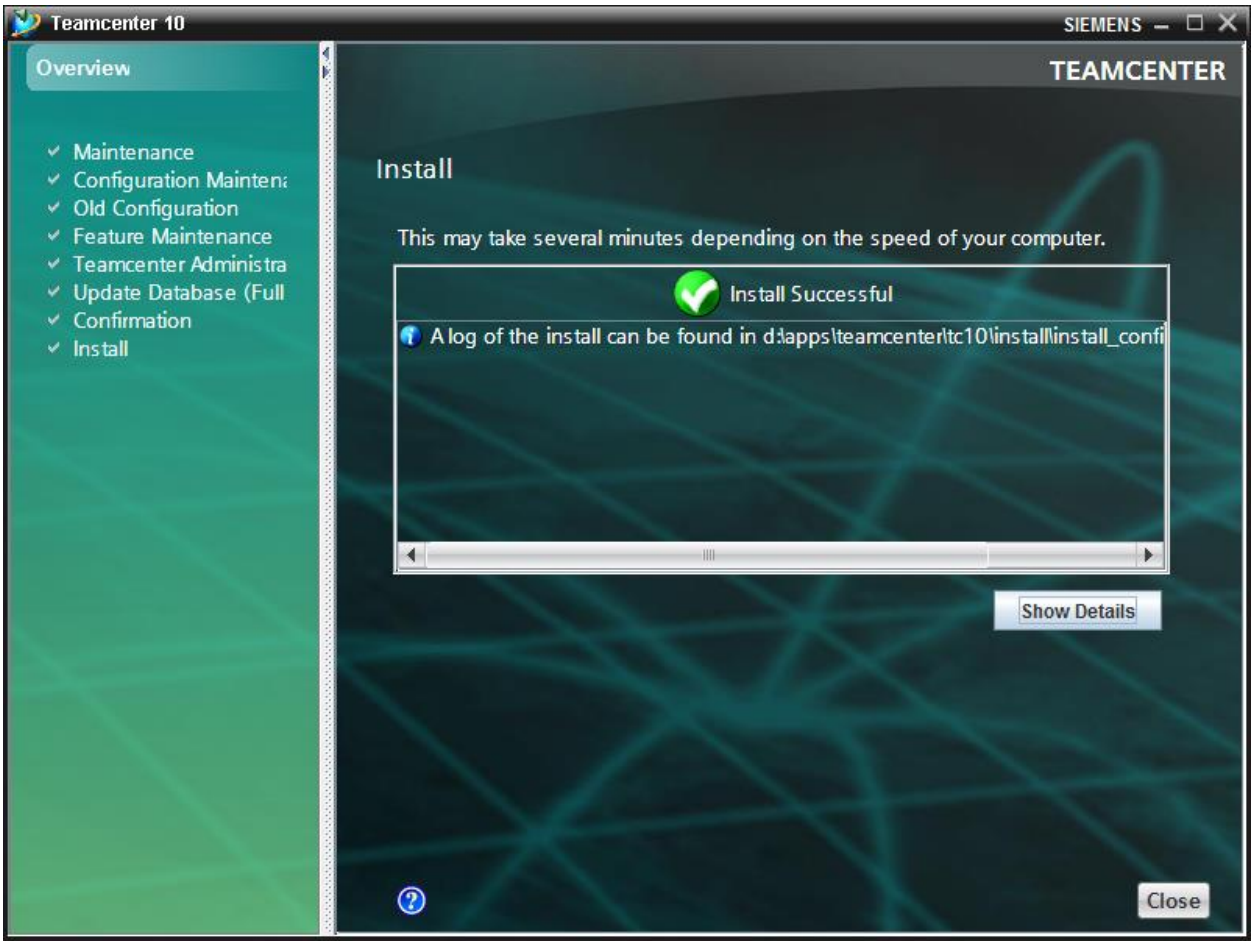


Figure 45 Confirmation of successful update

Updating the Integration template in the Business Modeler IDE (BMIDE) client

If you had previously added the integration's template to the BMIDE client, then you must update the template as part of updating to a newer version of the integration.
From the **Feature Maintenance** screen, select **Add/Update Templates for working within the Business Modeler IDE client**, then press **Next**

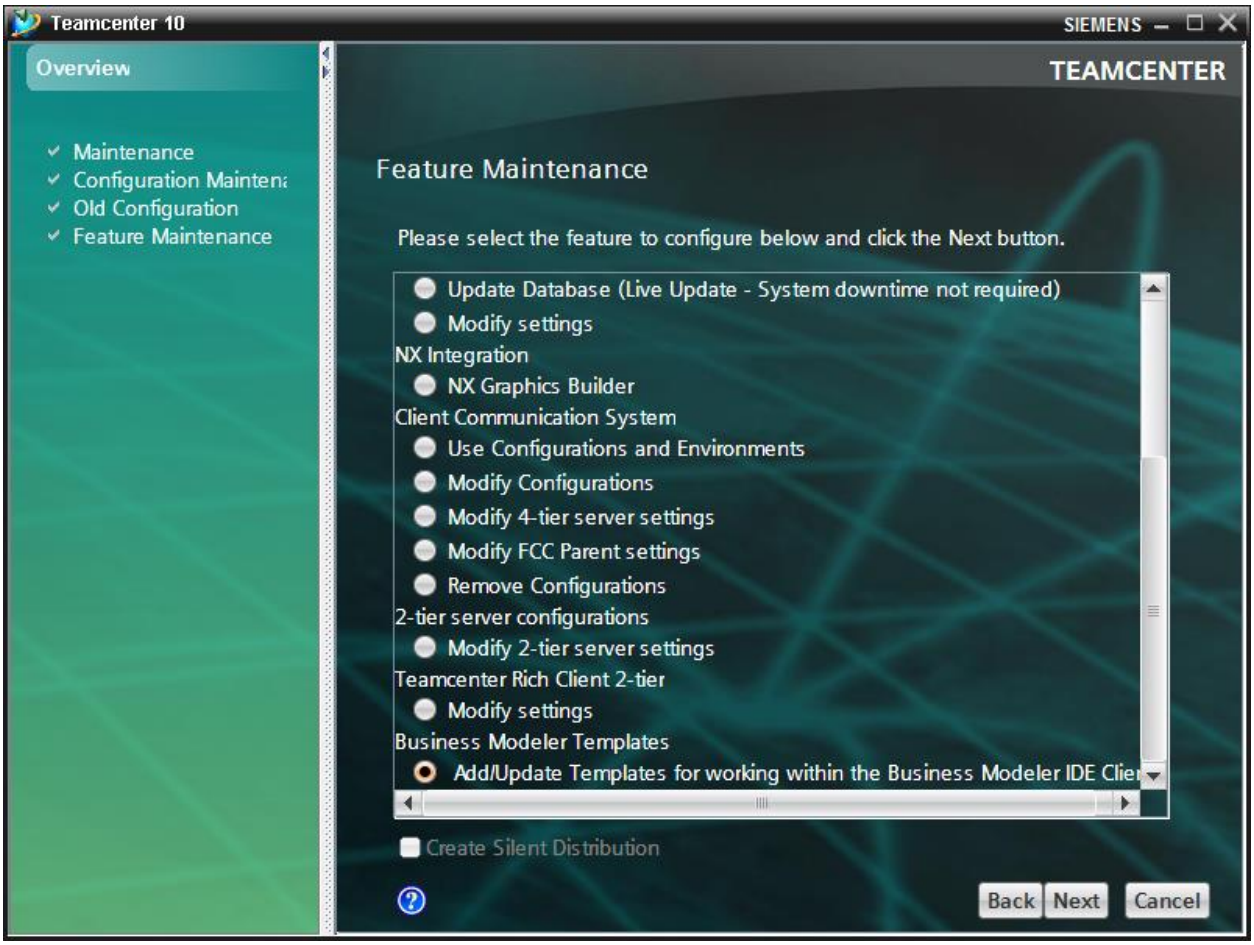


Figure 46 Updating the template in the BMIDE client

Press **Browse**, and then navigate to the location where the integration's BMIDE template was saved during execution of the server installation process for the new version of the integration:

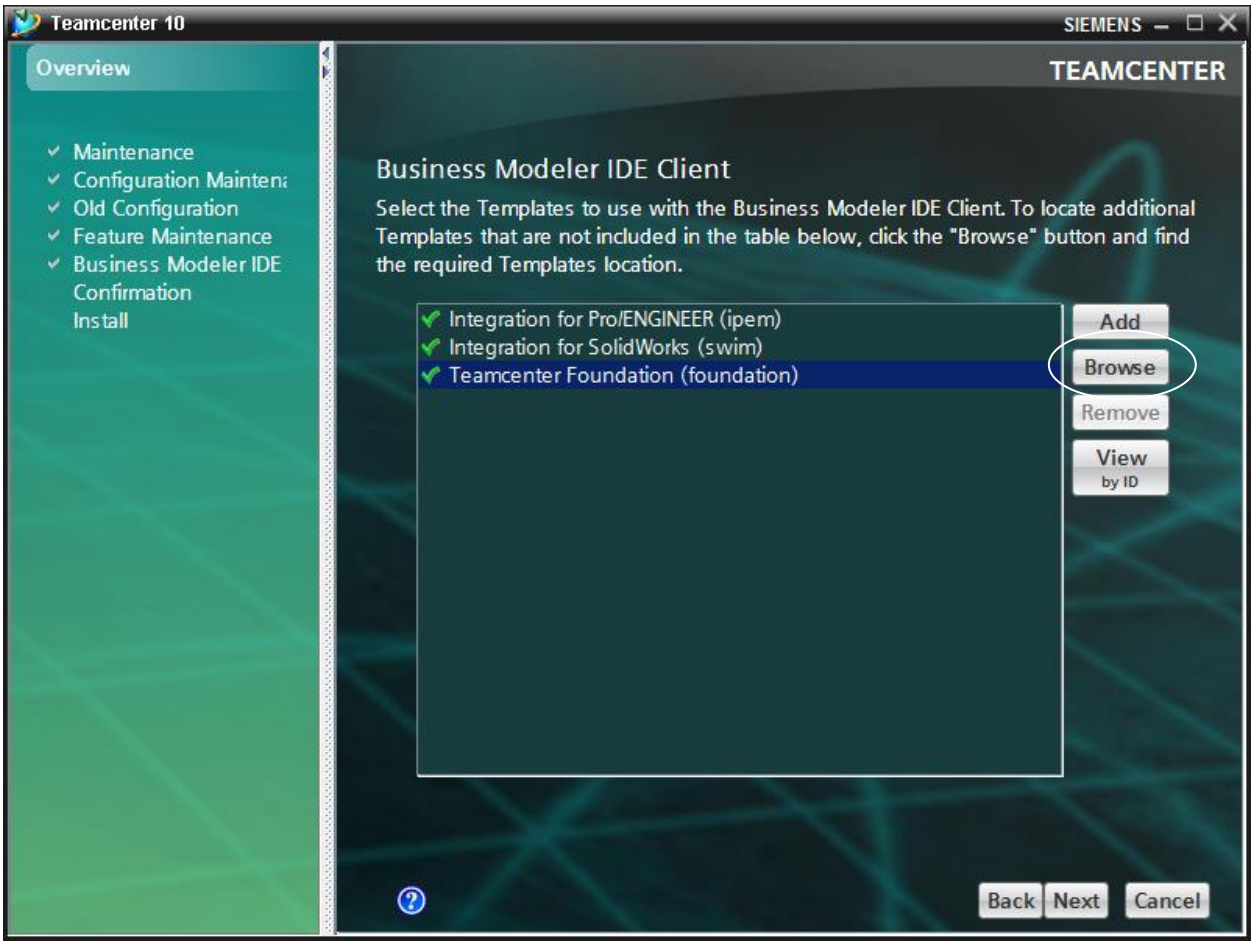


Figure 47 Browse to find the new integration template

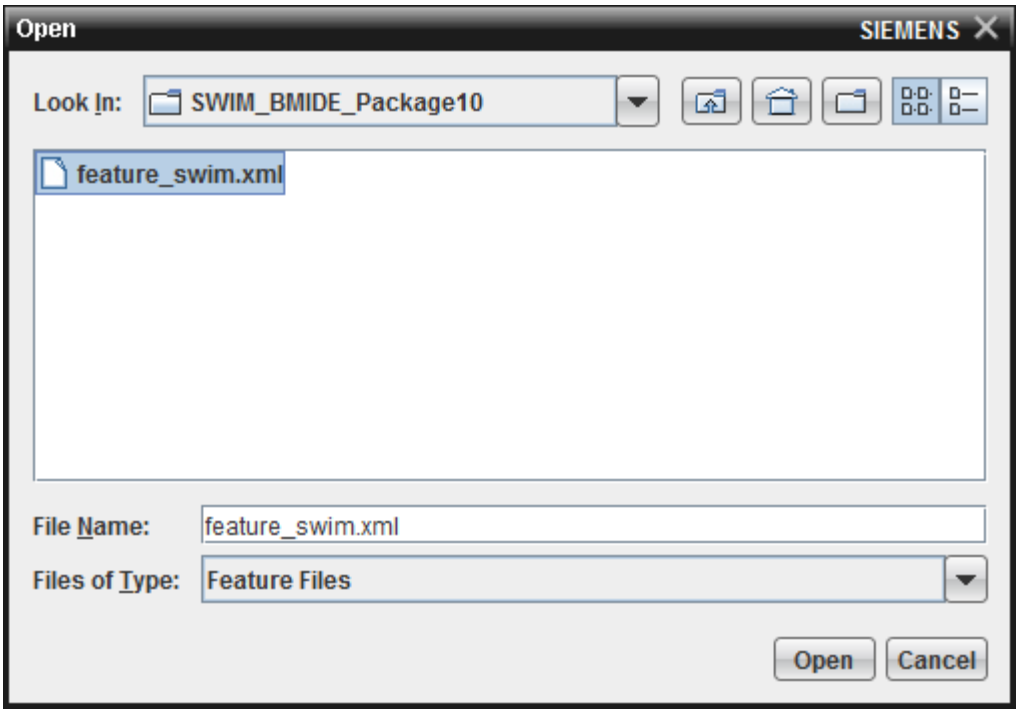


Figure 48 Select the feature_swim.xml file

Select the **feature_sim.xml** file from this location, then press **Next** on the **Business Modeler IDE client** screen:

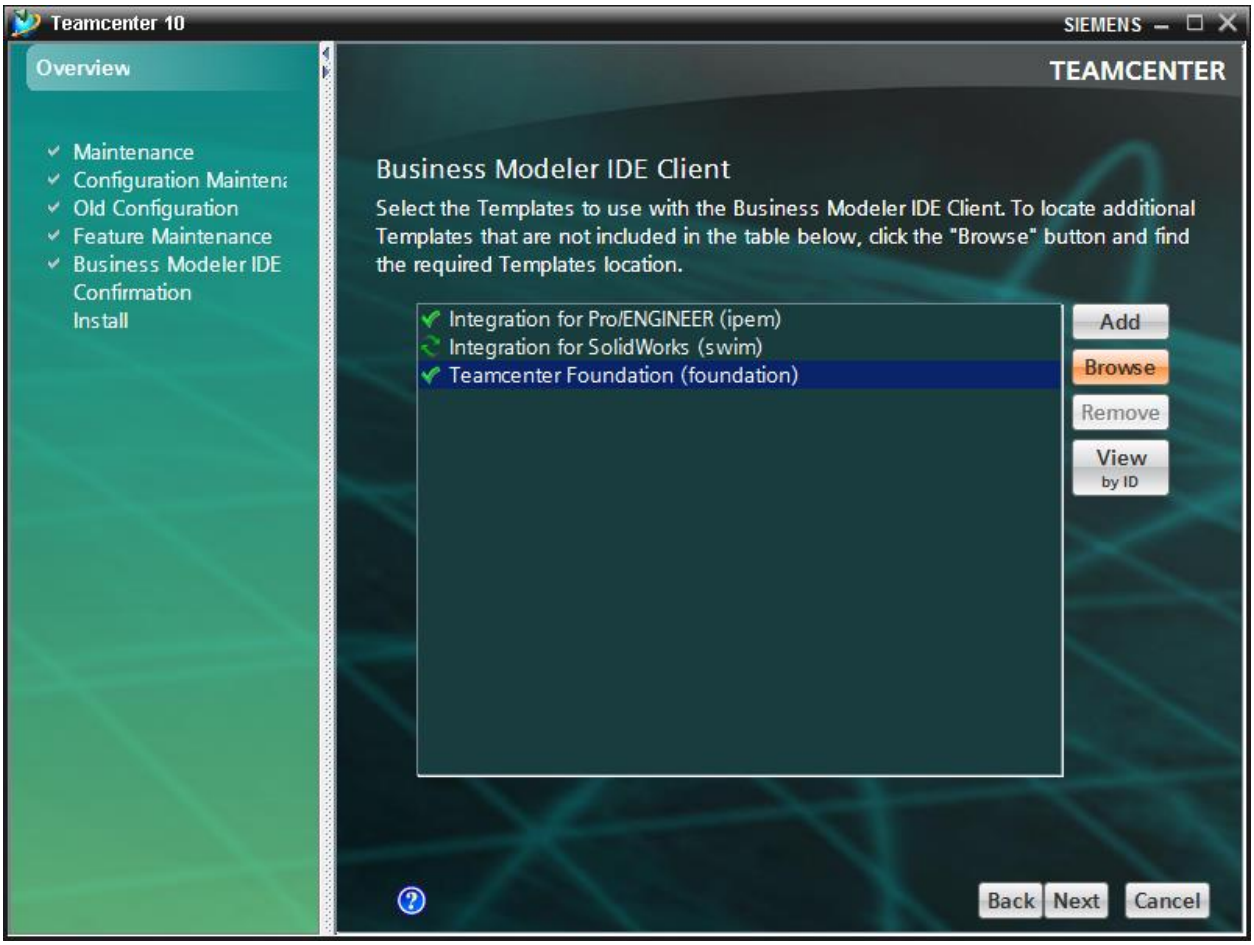


Figure 49 Updating the template in the BMIDE client

Press **Start** on the following screen, and then verify a successful update by the following confirmation message:

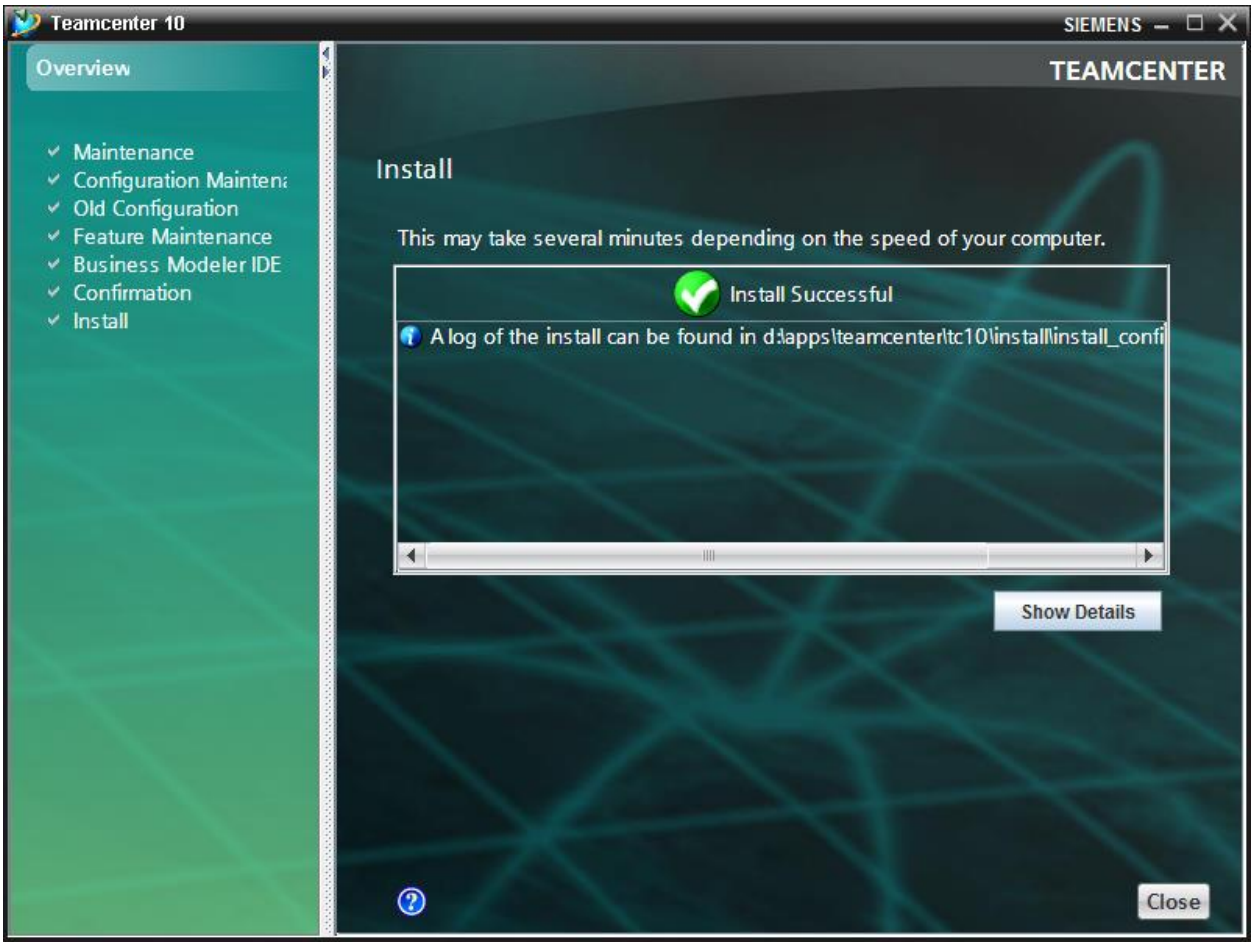


Figure 50 Confirmation of successful update

Configuring the Teamcenter Rich Client

Activate new Integration icons

The client installation creates or updates the **images** directory under %TPR%\plugins\configurations_*\images[6] with new icons. It will also create or update %TPR%\plugins\configurations_*\customer.properties, with these lines:

SWPrt.ICON=images/sw_prt.gif
SWAsm.ICON=images/sw_asm.gif
SWDrw.ICON=images/sw_drw.gif
SW2Tbx.ICON=images/sw_tbx.GIF

[6] TPR is an abbreviation for **Teamcenter Portal Root**. It is the installed location of the Teamcenter Rich Client

In newer versions of Teamcenter, there is a %TPR%\registry\genregxml.<ext> script that the client installation invokes, that will automatically add the integration icons to the Teamcenter Rich client. If this script does not exist, then perform the following manual steps:

- Navigate to %TPR% and edit the Teamcenter.ini file, removing "-DskipRegReload" option.
- Start the Teamcenter Rich client, and verify that the integration dataset icons are available. This is most easily done by selecting **File | New | Dataset** in the Navigator application, to create a dataset for each of the four types defined by the integration:

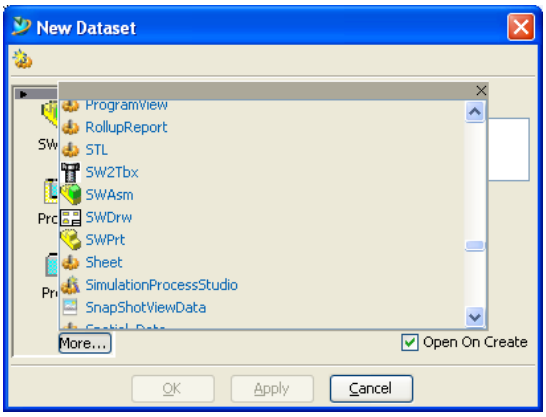


Figure 51 Dataset types defined in Teamcenter

Then verify that the icons can be viewed in the Rich Client:

Figure 52 Integration's dataset icons

- Stop Teamcenter Rich client and put back the "-DskipRegReload" option in Teamcenter.ini file.

Modify IIOP Server settings

When installing the Teamcenter two-tier Rich Client, the default option for the tcserver activation mode is **NORMAL**. This setting causes all client sessions to share a single instance of tcserver, which leads to a conflict when the user exits from one client. While shutting down, the client may terminate its tcserver process, which is still being used by the other client.

This conflict is resolved by setting the tcserver activation mode to **PER_CLIENT**. It is best done during the initial installing of the two-tier Rich Client, but may also be done at any later time. The following steps are required:

Start the TEM installer and advance to the **Feature Maintenance** screen. Select **Modify 2-tier Server Settings**, and then press **Next**:

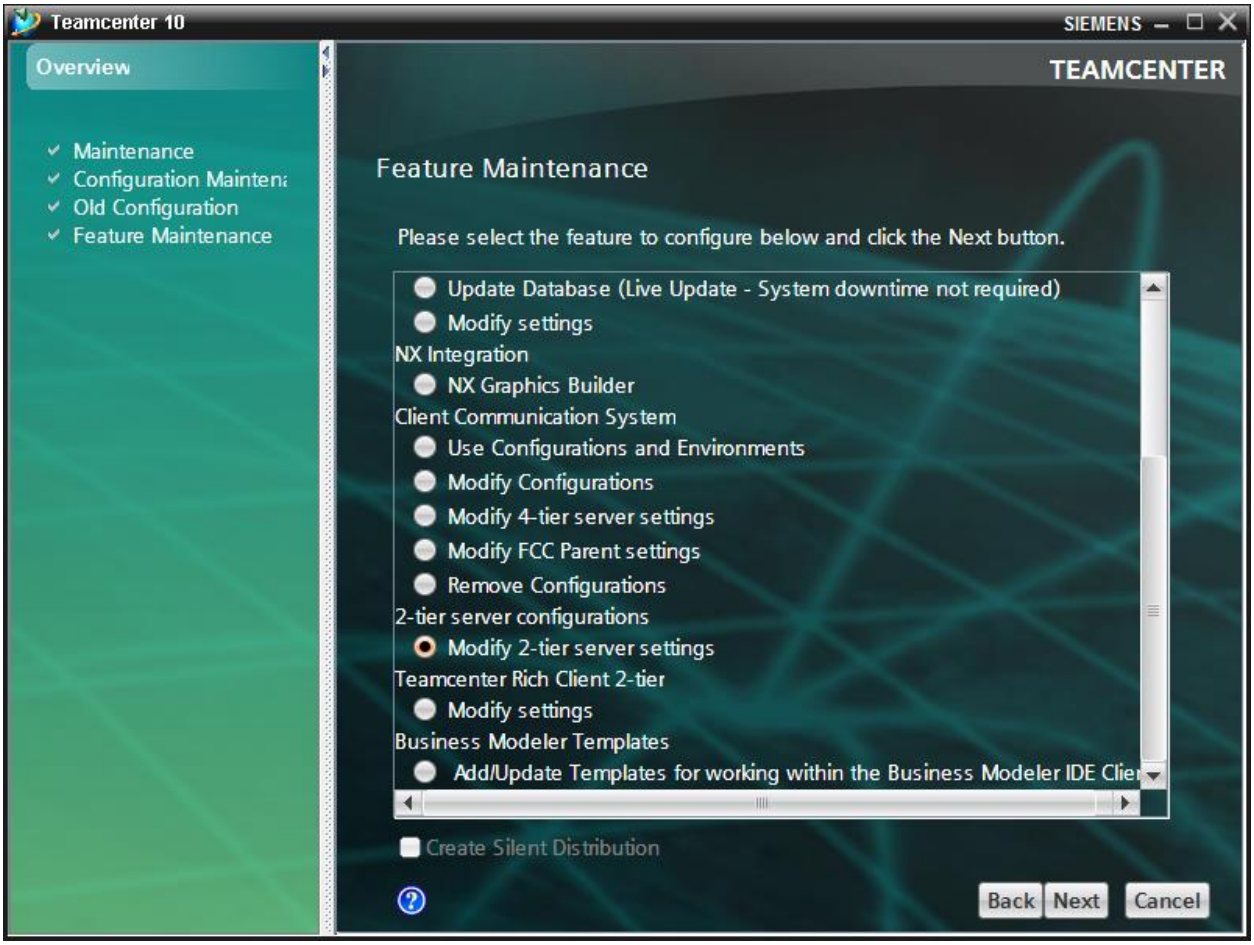


Figure 53 Modify the tcserver activation mode

On the next screen, select the **Advanced** button to display the Activation Mode settings:

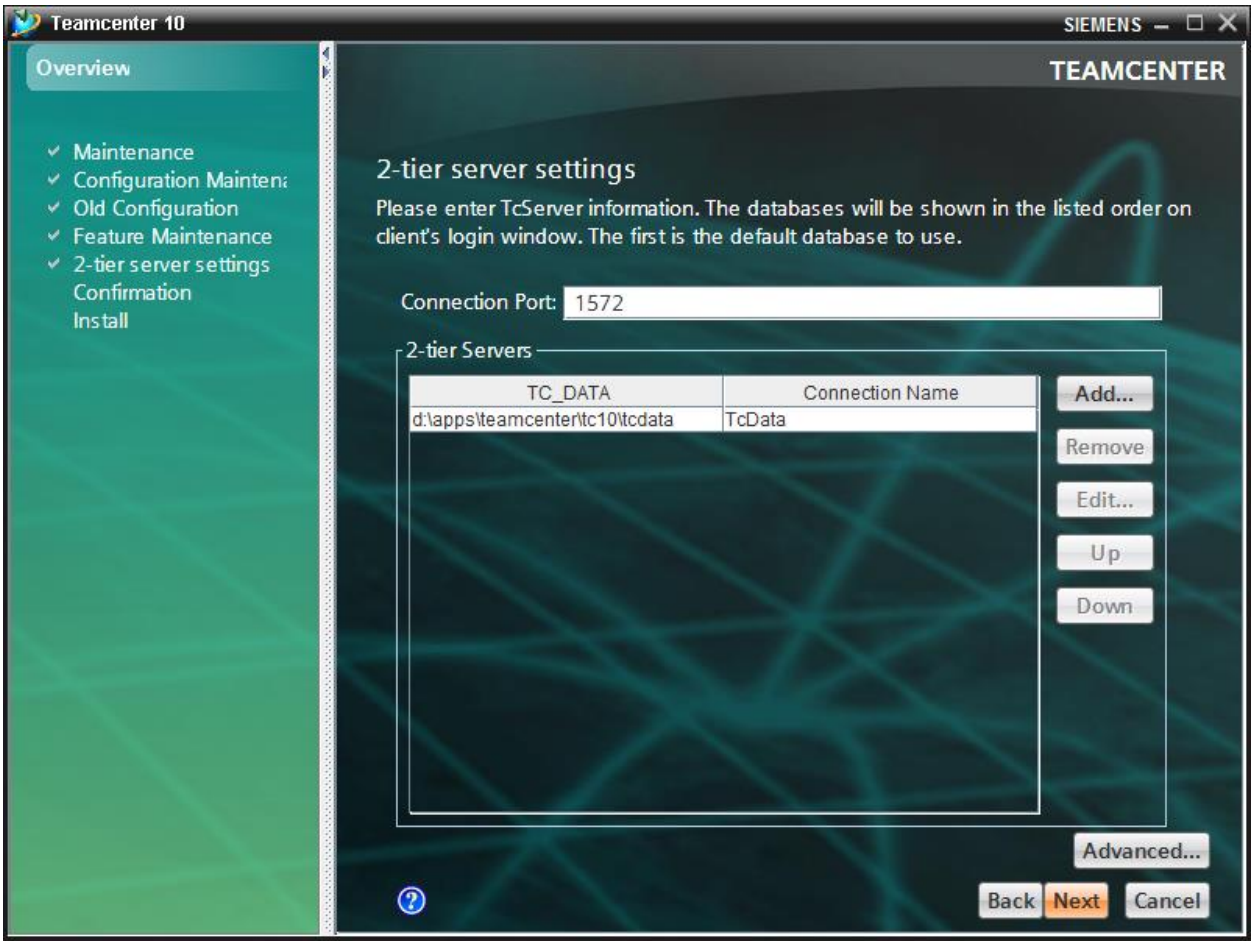


Figure 54 - Advanced 2-tier server settings

Then select **PER_CLIENT** from the Activation Mode selection list:

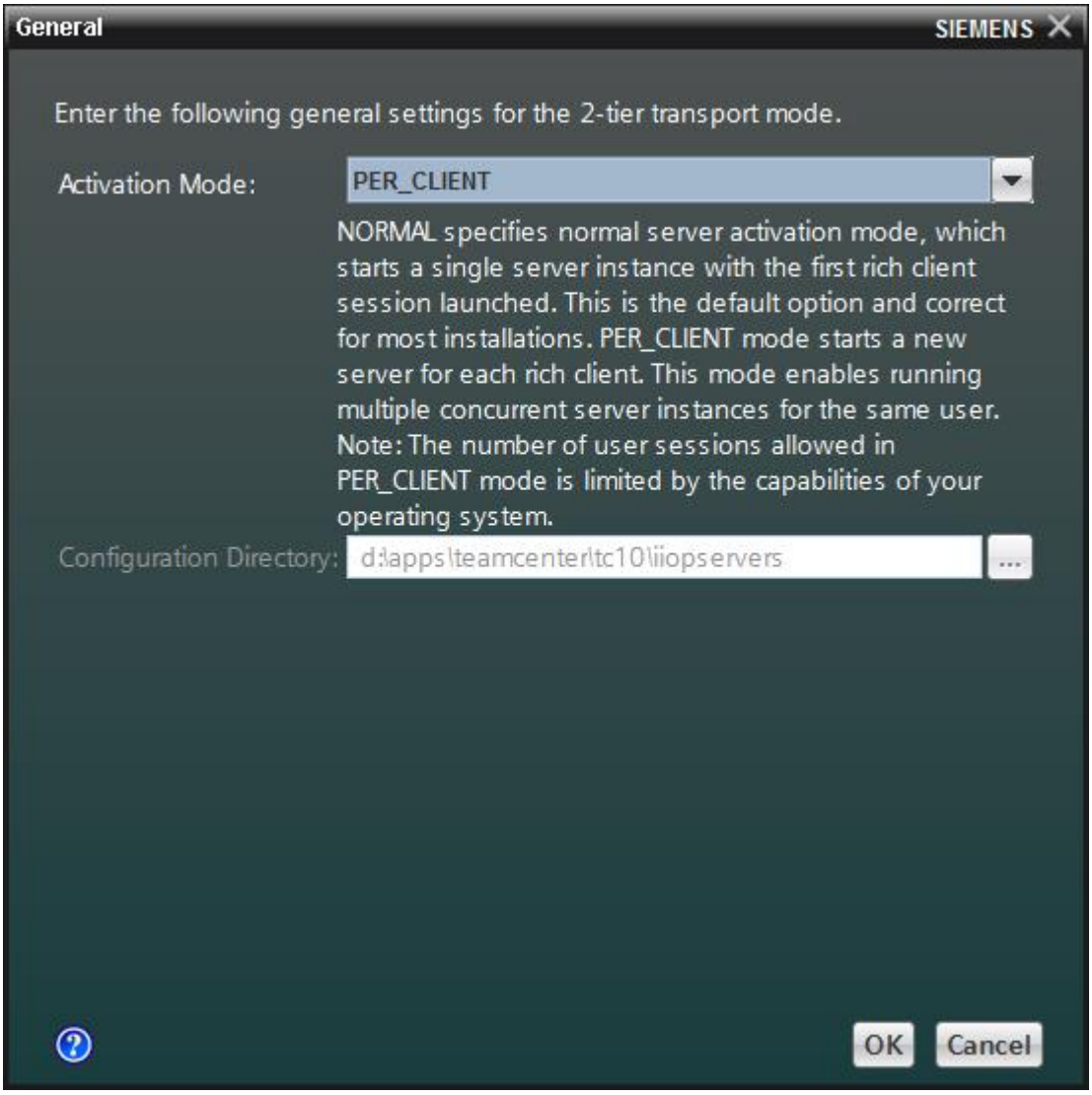


Figure 55 Select PER_CLIENT from the list

Press **Next** and then **Start**, on the following two screens. Successful installation will be indicated in the summary dialog:

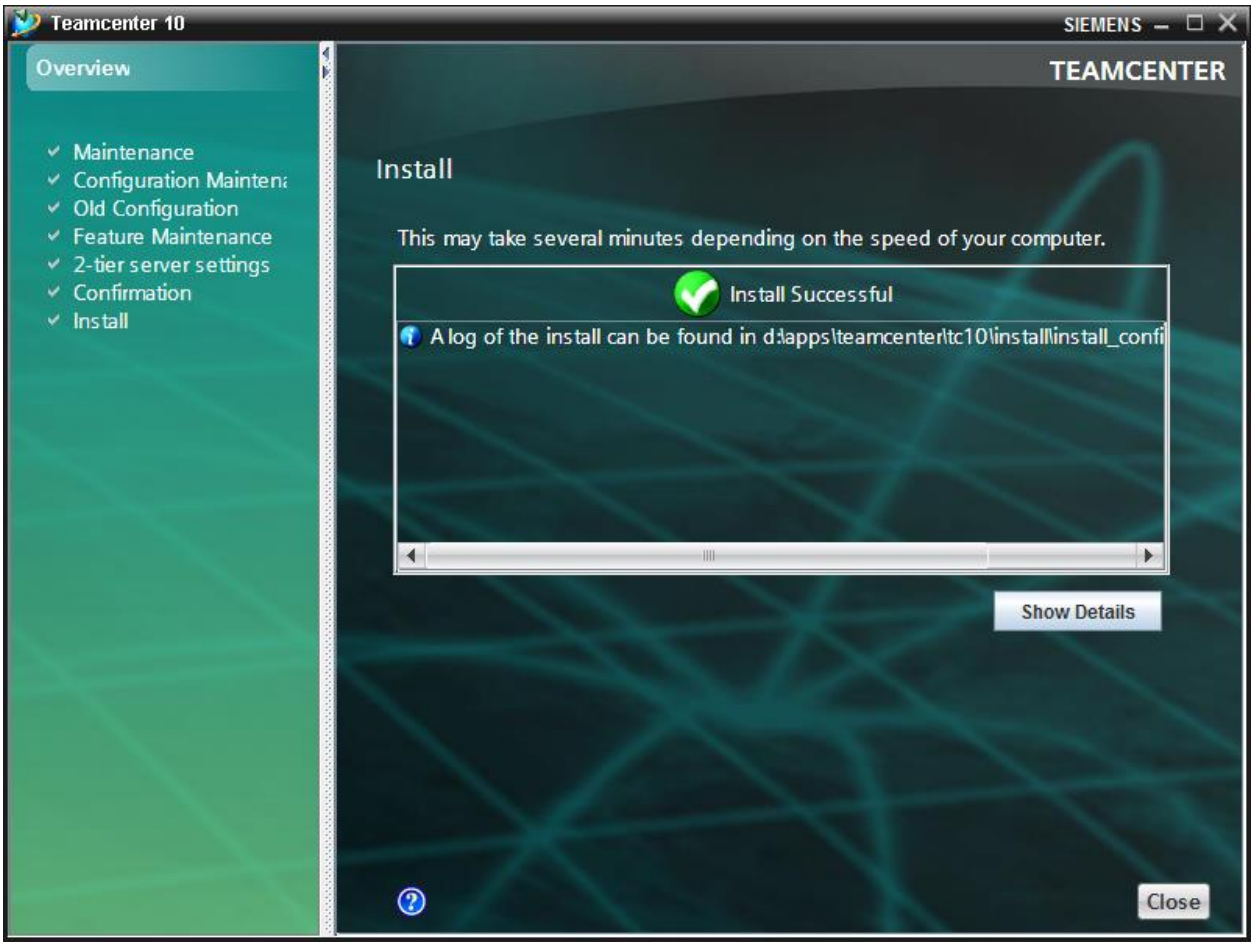


Figure 56 Confirmation of successful update

Loading the Attribute Mappings into Teamcenter

Important note: The integration leverages Teamcenter attribute mapping support for improved performance. The complete syntax and requirements for this Teamcenter functionality are described in the Teamcenter documentation. The integration provides a utility and simplified documentation, here, to automate the configuration process as much as possible. This may not be sufficient for all customers and all configurations, particularly those involving custom Item types and custom relations. In such cases, please refer to the Teamcenter documentation on attribute mapping.

The attribute mappings defined by the attribute_map tag in the swim.xml and the attributes stored in Teamcenter database must always match. A five step process is used to create and merge the attribute mappings into Teamcenter. A client installation must be done first before these steps can be performed, as files are required from the client installation. Step 1 can be skipped if a Teamcenter command shell with direct server access can be run from the integration client installation.

1. Copy the following files to a designated directory on the Teamcenter server:

   <SWIM_DIR>\swim.jar
   <SWIM_DIR>\swim.dtd
   <SWIM_DIR>\create_attr_mappings (UNIX) or create_attr_mappings.bat (Windows)
   <SWIM_DIR>\soa_client9\xercesImpl.jar
   <SWIM_DIR>\swim.xml

2. Convert the attribute_map tags defined in swim.xml into a text file.

   a. Open a Teamcenter command shell and change to the directory where you copied the files in step 1.
   b. Run create_attr_mappings.bat (or create_attr_mappings on UNIX) to create a text file from the attribute mapping definitions in swim.xml:

   ```
   create_attr_mappings -u <infodba user> -p <infodba password> -i
   swim.xml -o swim_attr_mappings.txt
   ```

   If item types other than "Item" are used with the integration, then those item types must be specified using the –itemtypes argument. itemtypes is a colon separated list:

   ```
   create_attr_mappings -u <infodba user> -p <infodba password> - i
   swim.xml -o swim_attr_mappings.txt -itemtypes
   Part:Document:Functionality:"Custom Requirement":"Parametric
   Requirement"
   ```

3. Extract all existing mappings from Teamcenter and make a backup copy.

   From the Teamcenter command shell, run the export_attr_mappings utility to create an existing_attr_mappings.txt file:

```
export_attr_mappings -u=<infodba user> -p=<infodba password> -g=dba
-file=existing_attr_mappings.txt
```

Note that warning messages, such as the examples below, may be ignored:

```
WARNING: Item Master form - no master qualifier found, adding
master=iman
for attribute =  : Item.GRM(IMAN_master_form,Item Master)…
```

Make a backup of the exported file.

The Teamcenter import_attribute_mappings first deletes all existing mappings from Teamcenter and then imports the new mappings from the provided input file. The below merge process is required to prevent the loss of existing non-integration attribute mappings.

4. Merge the integration mappings with the existing mappings

    a. If existing_attr_mappings.txt is empty, then swim_attr_mappings.txt becomes merge_attr_mappings.txt for import into Teamcenter.
    b. If existing_attr_mappings.txt only contains SolidWorks integration mappings, then swim_attr_mappings.txt becomes merge_attr_mappings.txt for import into Teamcenter
    c. If existing_attr_mappings.txt does not contain any SolidWorks integration mappings, then concatenate swim_attr_mappings.txt and existing_attr_mappings.txt to create merge_attr_mappings.txt
    d. If existing_attr_mappings.txt contains SolidWorks integration mappings, then remove the integration mappings from existing_attr_mappings.txt and then concatenate swim_attr_mappings.txt and existing_attr_mappings.txt to create merge_attr_mappings.txt.

5. Import the mappings into Teamcenter

    From the Teamcenter command shell, run the import_attr_mappings utility to import the mappings into Teamcenter:

    ```
    import_attr_mappings –u=<infodba user> -p=<infodba password>
    -g=dba -file=merge_attr_mappings.txt
    ```

    If the import fails, review the failures, correct, and import again.

Known issues with create_attr_mappings:

Issue: Attribute mapping that involves types and forms other than the OOTB Item type requires use of the fully qualified syntax in the swim.xml file, to ensure that attributes are mapped at runtime only to/from the desired types.

In this example, the customer wants attribute mapping to apply only to custom type "CustomDesign", but continues to use the simplified syntax for the pdm_name tag in swim.xml:

```
    <attribute_map cad_type="PRT">
        <attribute>
            <cad_name value="cadProperty1"/>
            <pdm_name value="Item Master.pa4tcAttribute1"/>
              <missing_attribute_action value="create"/>
            <direction value="both"/>
        </attribute>
        <attribute>
            <cad_name value="cadProperty2"/>
            <pdm_name value="ItemRevision Master.pa4tcAttribute2"/>
              <missing_attribute_action value="create"/>
            <direction value="both"/>
        </attribute>
    </attribute_map>
```

create_attr_mappings.bat is run as follows, to generate mappings for the custom type"

```
create_attr_mappings -u infodba -p infodba -I swim.xml -o swim_attr_mappings.txt -
itemtypes Pa4CustomDesign
```

resulting in the swim_attr_mappings.txt file with these contents:

```
{ Dataset type="SWPrt"
  cadProperty1 : Item.GRM(IMAN_master_form,Item Master).pa4tcAttribute1
  cadProperty2 : ItemRevision.GRM(IMAN_master_form,ItemRevision
Master).pa4tcAttribute2
  { Item type="Pa4CustomDesign"
      %txd1%cadProperty1 :
Item.GRM(IMAN_master_form,Pa4CustomDesignMaster).pa4tcAttribute1
      %txd1%cadProperty2 :
ItemRevision.GRM(IMAN_master_form,Pa4CustomDesignRevisionMaster).pa4tcAttribute2
  }
}
```

Importing this attribute mapping into Teamcenter would result in the custom attributes being mapped for both the OOTB Item type and the custom Pa4CustomDesign type, which is not the desired behavior.

Resolution:

Use the fully qualified attribute mapping syntax in swim.xml, whenever a custom Item type and/or custom form is involved. In the example above, the correct syntax would be this:

```
    <attribute_map cad_type="PRT">
        <attribute>
            <cad_name value="cadProperty1"/>
            <pdm_name
value="Item:Pa4CustomDesign.Form:Pa4CustomDesignMaster.pa4tcAttribute1"/>
              <missing_attribute_action value="create"/>
            <direction value="both"/>
        </attribute>
        <attribute>
            <cad_name value="cadProperty2"/>
```

```
            <pdm_name value="ItemRevision:Pa4CustomDesign
Revision.Form:Pa4CustomDesignRevisionMaster.pa4tcAttribute2"/>
             <missing_attribute_action value="create"/>
            <direction value="both"/>
        </attribute>
    </attribute_map>
```

When create_attr_mappings is run on this input file, the resulting swim_attr_mappings.txt looks like this, which yields the desired runtime behavior:

```
{ Dataset type="SWPrt"
  { Item type="Pa4CustomDesign"
      cadProperty1 : Item.GRM(IMAN_master_form,Pa4CustomDesignMaster).pa4tcAttribute1
      cadProperty2 :
ItemRevision.GRM(IMAN_master_form,Pa4CustomDesignRevisionMaster).pa4tcAttribute2
  }
}
```

In other words, the –itemtypes parameter on the create_attr_mappings command line does not limit the mapping to the specified types; it only adds the specified types to the set which already includes the OOTB Item type.

==========

Issue: Below is a list of known predefined Teamcenter properties:

```
DB_UNITS      "Part Unit of Measure"
DB_PART_TYPE  "Part Type"
DB_PART_REV   "Part Revision"
DB_PART_NO    "Part Number"
DB_PART_NAME  "Part Name"
DB_PART_DESC  "Part Description"
```

If one of these predefined Teamcenter properties is used in swim.xml:

```
    <cad_name value="DB_PART_NO"/>
    <pdm_name value="Item.item_id"/>
```

It would result in the create_attr_mappings utility output text file:

```
    {Dataset type="SWPrt"
       DB_PART_NO:Item.item_id
    }
```

Since Teamcenter does not allow duplicate name and we cannot change the predefined values, The SolidWorks integration handles these mappings by putting a prefix of %txd1% in front of the CAD property name. The integration strips off the prefix during attribute processing.

Resolution:

Open the merge_attr_mappings.txt file and search any of the predefined string like "DB_PART_NO" and add the %txd1% prefix (“1” in %txd1% is a counter and should be incremented if the DB_PART_NO is mapped multiple times).

```
    {Dataset type="SWPrt"
        %txd1%DB_PART_NO:Item.item_id
```

}

Note: This step may or may not be necessary. If your file already has the %txd1% prefixes, you do not need to add them.

Turning Off File Name Truncation

The Teamcenter Integration for SolidWorks requires file name truncation to be disabled.

To turn off file name truncation:

1. Start the Teamcenter Rich Client, and log in as the administrative user.
2. Select the **Edit | Options** command.
3. Click on the **Index** link at the bottom of the Options dialog.
4. Search for the **TC_truncate_file_name** site preference and change its value to **FALSE**

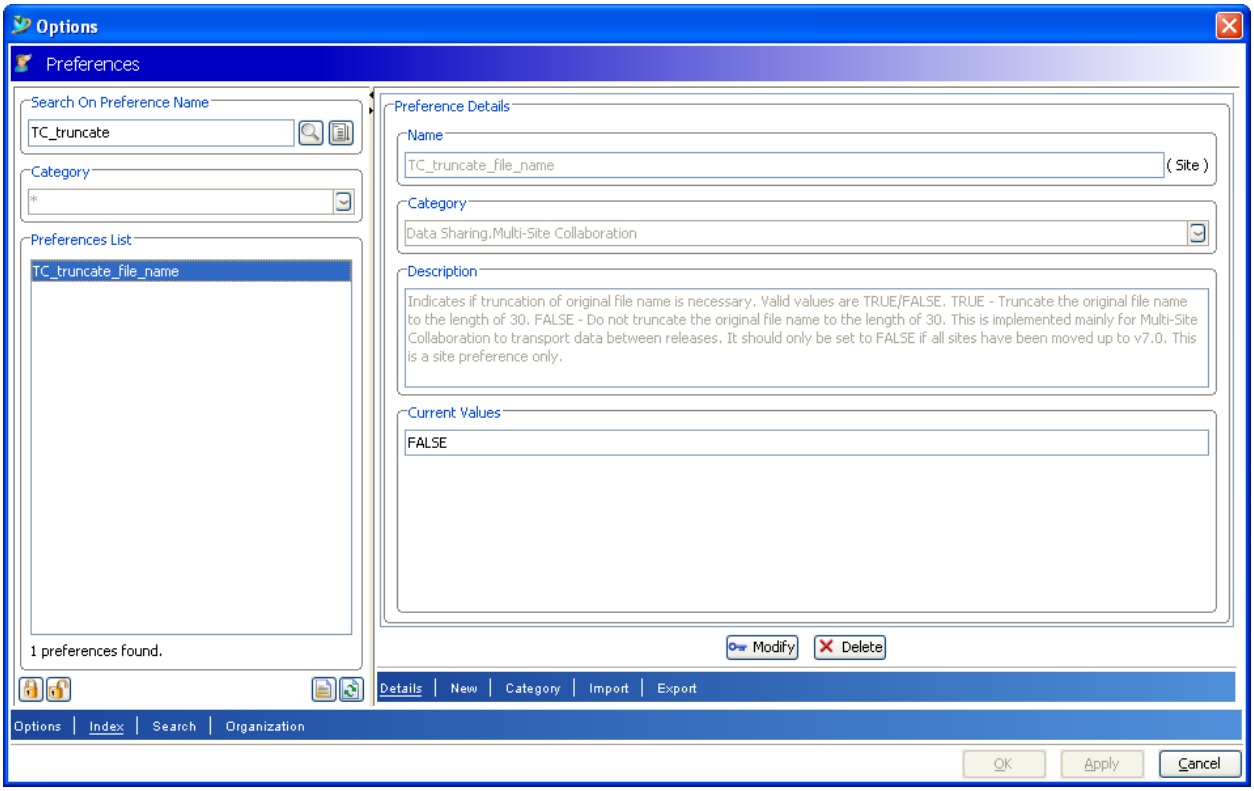


Figure 57 File name truncation preference

Uninstalling Client installations on Windows

For Windows client uninstallation, the normal uninstallation is method is provided in Windows in the Control Panel → Programs and Features with one important difference. If the 32-bit

product installer was used, the system path must provide path to a 32-bit JRE. If the 64-bit product installer was used, then the system path must include a 64-bit JRE. It is also possible to run the uninstaller by starting a command window (DOS window), navigating to `%SWIM_DIR%/SwimUninstallerData`, setting the Java JRE path in the shell and then invoking the "Uninstall Teamcenter Integration for SolidWorks Version 9.1 Installation.exe". Below is a 32-bit Java JRE example:

```
DOS Prompt> set PATH=C:\tc91\portal\jre;%PATH%
DOS Prompt>  "Uninstall Teamcenter Integration for SolidWorks Version 9.1
Installation.exe"
```

Appendix

Custom Installation options

When a new version of the integration is installed, the appropriate non-custom install set should be used for the update. Under some circumstances, such as restoring a missing shared library in the %TC_ROOT% location, or reloading database preferences after a failed recovery from backup, the custom installation option may be appropriate. You may select one option from the list, or any combination of options. The following is a brief description of each.

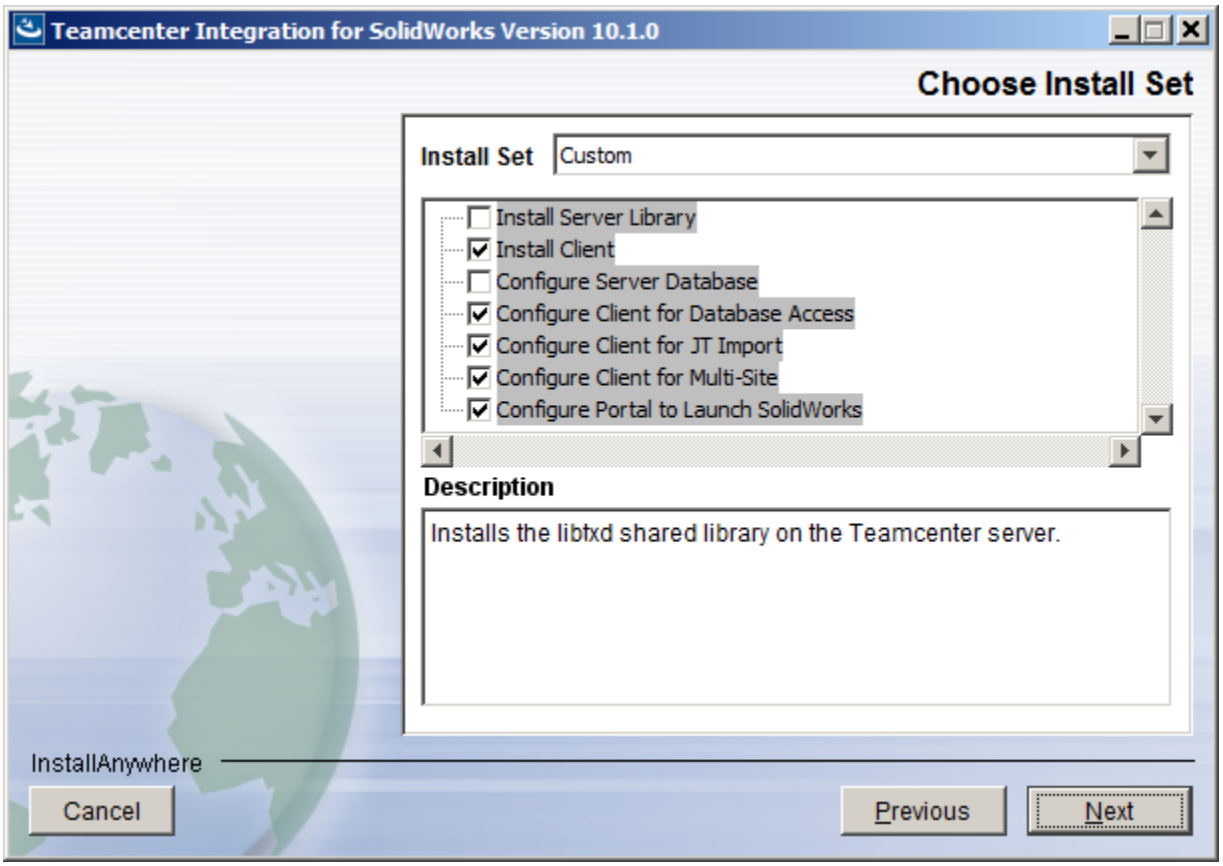


Figure 58 Custom installation options

Install Server Library

Install the libtxd library on a Teamcenter server, without configuring the server database. This requires write access to the Teamcenter installation on the server, i.e. the TC_ROOT directory and its sub-directories.

Install Client

Install the Teamcenter Integration for SolidWorks client files, without configuring the client for database access. This step will create the Teamcenter Integration for SolidWorks installation directory, if it does not already exist. This step also requires write access to the Rich Client installation directory.

Configure Server Database

Configure the server database by installing SolidWorks queries and preferences in Teamcenter. This requires write access to the database's TC_DATA directory, and Teamcenter administrator privileges. All users must be logged out of Teamcenter before the database can be configured.

Configure Client for Database Access

Configures the integration for access to a particular Teamcenter database.

Configure Client for JT Import

Configures the Teamcenter Integration for SolidWorks to save JT files for visualization in the Rich Client. To use this feature you will need the PLM Components JT Translator for SolidWorks . This step requires write access to the Teamcenter Integration for SolidWorks installation directory.

Configure Client for Multi-Site

Configures the Teamcenter Integration for SolidWorks to allow a user to transfer item ownership from remote sites to the user's site. This causes a Transfer Ownership command to appear in the Integration's shortcut menu. To use this feature you will need Teamcenter Multi-Site installed, and the remote sites must be configured to allow export and transfer of ownership. This step requires write access to the Teamcenter Integration for SolidWorks installation directory.

Configure Portal to Launch SolidWorks

Configures the Teamcenter Rich Client so it can start SolidWorks when the user double-clicks on a SolidWorks dataset. This step requires write access to the Rich Client installation directory.

Troubleshooting

If you have trouble starting SolidWorks, or if SolidWorks starts, but no Teamcenter Integration for SolidWorks sidebar tab appears:

- Make sure the minimum required build of SolidWorks is installed. See the section on **Prerequisites** for more information.
- At times, problems will have to be sent to Siemens for analysis. We will need a txdlog.txt file for us to better understand the issue. To generate this file for a particular problem, add the following lines to the end (this will ensure that it overwrites any other settings) or uncomment the existing lines and change in swim.properties:

  log.enable = true
  log.file = txdlog.txt
  log.suppress = 10000

- If the Teamcenter Integration for SolidWorks sidebar tab does not appear in SolidWorks, make sure the Teamcenter add-in is enabled. In SolidWorks, select **Tools**|**Add-Ins** to open the Add-Ins dialog. Make sure the dialog shows the Teamcenter in this list, and put a check next to its name. If the dialog does not show the Teamcenter, it may be necessary to repeat the client installation or you can double click on %SWIM_DIR%/bin/ RegisterSwimAddin.bat
- Make sure any Windows paths specified for the JRE_DIR, IPR_DIR, or SWIM_DIR variables in scripts or in the user's environment do not have spaces in them, or in %SWIM_DIR%/swimenv.bat. If necessary, use a command window and the MS-DOS command `dir /x` to determine the short MS-DOS name for a file or directory.
- Make sure the JRE_DIR variable in the %SWIM_DIR%/swimenv.bat script specifies the correct installation directory for the Java runtime environment. This script is located in the installation directory for the Teamcenter Integration for SolidWorks. The same Java runtime environment used for the Teamcenter Portal is recommended.
- If the JRE_DIR variable is already defined in the user's environment, it will take precedence over the variable's definition in the %SWIM_DIR%/swimenv.bat script. If the user prefers setting this variable in his environment, make sure it is defined correctly otherwise remove it from the user's environment so the swimenv.bat definitions take effect.
- If you receive the message, "Installation of the Teamcenter Integration for SolidWorks is not complete" when attempting to log into Teamcenter, first make sure that you selected "Typical Server Installation" or "Typical Client/Server Installation" when you ran the installer on the host where your TC_ROOT directory is located.
- If you have more than one version of SolidWorks installed. The installer will select the active version based on a registry value. You can easily change to another version by editing the %SWIM_DIR%\bin\startsw.bat. Find this line in the file:

  ```
  set swExe=C:\PROGRA~1\SOLIDW~1\SOB3F7~1\SLDWORKS.exe
  ```

  Set the path to the SolidWorks executable you wish to use. We normally use short names for paths which can be obtained in DOS window with the "dir /x" command. No other change is required.

- If SolidWorks models cannot be opened successfully at remote sites after export via Teamcenter Multi-Site, make sure you have included the TXD_long_name_relation in the relation types that are exported.
- Switching between different Integration client installations on the same workstation requires execution of batch scripts (RegisterSwimAddin.bat and UnregisterSwimAddin.bat) in the SWIM bin directory. You must run these scripts as the system administrator (right click the file and select "Run as Administrator")
    - First unregister the currently active SWIM version by running UnregisterSwimAddin.bat in the SWIM bin subdirectory.
    - Next, go to the bin subdirectory of the SWIM version you want to switch to, and run RegisterSwimAddin.bat.
    - Start SWIM using the version's corresponding startsw.bat.
    - You can verify the change by selecting Tools → Addins in SolidWorks, and hovering you mouse over the Teamcenter option. The tooltip that appears should display the directory of the Integration version you wish to use.
- If you receive the message, "Installation of the Teamcenter Integration for SolidWorks is not complete (types not completely installed)" in the last panel of the installer when doing "Typical Server Installation" or "Typical Client/Server Installation". Try the below steps to correct the problem:
    - Check the temp directory for swim_db_adjust.log file for errors and correct if possible. It might be the server username password was incorrect.
    - Check Teamcenter system log files in the temp directory for possible errors (e.g. preference_manager*.log, plmxml_import*.log, etc.).
    - Consider cleaning up the temp log files to ensure that the latest versions are available.
    - Make sure that the server has adequate temp space available.
    - Stop all active sessions
    - All users must be logged out of Teamcenter, all server processes must be halted, and all locks must be cleared from the database. The required steps are these:
        1. Advise all users to log out of Teamcenter.
        2. In a four-tier environment, shut down the pool manager using the control panel at http://serverhost:8082.
        3. From a Teamcenter command line on a server host, run the clearlocks command:

           ```
           clearlocks -assert_all_dead infodba infodba dba
           ```
    - Run the "Typical Server Installation" again on the host where your TC_ROOT directory is located, and choose to install SolidWorks queries and preferences in Teamcenter (if you had first installed using the Typical Client/Server Installation" option, it is fine to reinstall using the "Typical Server Installation" option).

- If a user encounters what appears to be a "timed out session" with no errors, no indications of trouble, no results, and a blank SolidWorks session, then consider changing the java heap size. Change the java heap size minimum and maximum settings by editing %SWIM_DIR%\swimrunner.bat on Windows

  set ARGS=%ARGS% -Xms512m -Xmx4096m

- Sometimes it is necessary to specify a specific language supported by Teamcenter that is not the native language of the client operating system. For example, if the client native operating system is Hungarian and the desired and supported Teamcenter language is German. To override the native operating system client value, edit the %SWIM_DIR%\swimrunner.bat file and find this line:

  ```
  set ARGS=%ARGS% -DTXD_USE_SOA=1
  ```

  Add a new line after entries as shown below:

  ```
  set ARGS=%ARGS% -DTXD_USE_SOA=1
  set ARGS=%ARGS% -Duser.language=xx -Duser.country=YY
  ```

  Where xx is language (e.g. “en” for English).
  Where YY is the user country (e.g. “US” for United States).