

Installation & Administration Guide

PTC Creo Parametric integration
for
SOLIDWORKS PDM

Valid for product version: 2024 SP1.0 (2024.1.0)

Published: 19.03.2024 | Build: 505 | Revision: a0f92c4fb

Legal information

© 1995-2024, Dassault Systèmes SolidWorks Corporation, a Dassault Systèmes SE company, 175 Wyman Street, Waltham, Mass. 02451 USA. All Rights Reserved.

The information and the software discussed in this document are subject to change without notice and are not commitments by Dassault Systèmes SolidWorks Corporation (DS SolidWorks).

No material may be reproduced or transmitted in any form or by any means, electronically or manually, for any purpose without the express written permission of DS SolidWorks.

The software discussed in this document is furnished under a license and may be used or copied only in accordance with the terms of the license. All warranties given by DS SolidWorks as to the software and documentation are set forth in the license agreement, and nothing stated in, or implied by, this document or its contents shall be considered or deemed a modification or amendment of any terms, including warranties, in the license agreement.

Patent notices

SOLIDWORKS® 3D mechanical CAD and/or Simulation software is protected by U.S. Patents 6,219,049; 6,219,055; 6,611,725; 6,844,877; 6,898,560; 6,906,712; 7,079,990; 7,477,262; 7,558,705; 7,571,079; 7,590,497; 7,643,027; 7,672,822; 7,688,318; 7,694,238; 7,853,940; 8,305,376; 8,581,902; 8,817,028; 8,910,078; 9,129,083; 9,153,072 and foreign patents, (for example, EP 1,116,190B1 and JP 3,517,643).

eDrawings® software is protected by U.S. Patent 7,184,044; U.S. Patent 7,502,027; and Canadian Patent 2,318,706.

U.S. and foreign patents pending.

Trademarks and product names for SOLIDWORKS products and services

SOLIDWORKS, 3D ContentCentral, 3D PartStream.NET, eDrawings, and the eDrawings logo are registered trademarks and FeatureManager is a jointly owned registered trademark of DS SolidWorks.

CircuitWorks, FloXpress, PhotoView360, and TolAnalyst are trademarks of DS SolidWorks.

FeatureWorks is a registered trademark of Geometric Ltd.

SOLIDWORKS 2018, SOLIDWORKS Standard, SOLIDWORKS Professional, SOLIDWORKS Premium, SOLIDWORKS PDM Professional, SOLIDWORKS PDM Standard, SOLIDWORKS Workgroup PDM, SOLIDWORKS Simulation, SOLIDWORKS Flow Simulation, eDrawings, eDrawings Professional, SOLIDWORKS Sustainability, SOLIDWORKS Plastics, SOLIDWORKS Electrical, SOLIDWORKS Composer, and SOLIDWORKS MBD are product names of DS SolidWorks.

Other brand or product names are trademarks or registered trademarks of their respective holders.

COMMERCIAL COMPUTER SOFTWARE - PROPRIETARY

The Software is a “commercial item” as that term is defined at 48 C.F.R. 2.101 (OCT 1995), consisting of “commercial computer software” and “commercial software documentation” as such terms are used in 48 C.F.R. 12.212 (SEPT 1995) and is provided to the U.S. Government 14 (a) for acquisition by or on behalf of civilian agencies, consistent with the policy set forth in 48 C.F.R. 12.212; or (b) for acquisition by or on behalf of units of the Department of Defense, consistent with the policies set forth in 48 C.F.R. 227.7202-1 (JUN 1995) and 227.7202-4 (JUN 1995).

In the event that you receive a request from any agency of the U.S. Government to provide Software with rights beyond those set forth above, you will notify DS SolidWorks of the scope of the request and DS SolidWorks will have five (5) business days to, in its sole discretion, accept or reject such request. Contractor/Manufacturer: Dassault Systèmes SolidWorks Corporation, 175 Wyman Street, Waltham, Massachusetts 02451 USA.

Copyright notices for SOLIDWORKS PDM Professional product

Outside In [®]Viewer Technology, © 1992-2012 Oracle

©2011, Microsoft Corporation. All rights reserved.

Contents

Legal information.....	ii
Glossary.....	6
1 General information.....	7
1.1 Introduction.....	7
1.2 How the integration works.....	8
2 System requirements.....	9
2.1 Operating system support.....	9
2.2 Supported SOLIDWORKS PDM releases.....	9
2.3 PTC Creo Parametric TOOLKIT license.....	9
2.4 Local machine requirements.....	9
2.5 Supported file types.....	9
3 Installation.....	10
3.1 Pre-installation information.....	10
3.2 Installing integration.....	10
4 Initial setup.....	12
4.1 Installation for SOLIDWORKS PDM 2023 and older environments.....	12
4.2 SOLIDWORKS PDM.....	12
4.2.1 Add-in registration.....	12
4.2.2 Import of CEX files.....	13
4.3 Creo Parametric.....	14
4.3.1 Creo Parametric configuration settings.....	14
4.4 Additional information.....	16
5 Configuration.....	18
5.1 SOLIDWORKS PDM.....	18
5.1.1 Setup tasks in SOLIDWORKS PDM add-in.....	18
5.1.2 Adding customer variables.....	19
5.1.3 Improving check in performance.....	22
5.1.4 Configuration files.....	23
5.1.5 Enabling logging.....	28

5.2	Creo Parametric.....	29
5.2.1	Configuration Files.....	29
5.3	Renaming.....	31
5.4	Assemblies with family tables.....	32
6	Update.....	34
6.1	Modifying installation.....	34
6.2	Repairing installation.....	35
6.3	Updating installation.....	36
7	Uninstallation.....	39
7.1	Removing installation.....	39
8	Troubleshooting.....	40
8.1	Common troubleshooting procedure.....	40
8.2	License errors.....	40
9	References.....	42
9.1	Tools and add-ons.....	43
9.1.1	Integration Creator.....	44
9.1.1.1	Introduction.....	44
9.1.1.2	Signing connector components.....	44
9.1.1.3	Distributing signed connector components.....	45
9.2	Silent-mode installation.....	45
9.3	Working with overlay packages.....	49

Glossary

Application Programming Interface (API)

Defines a set of routines, communication protocols and tools for building software. In general, they are clearly defined methods for communication between different components.

Bill of Materials (BOM)

Defines a list of assemblies, sub-assemblies, parts and their quantities needed to produce a final product.

BOM position

Defines a position in the BOM with unique identification, name, quantity and other characteristics.

Component Object Model (COM)

Defines a binary-interface standard for software components introduced by Microsoft.

Connector

Defines a central interface component of each Dassault integration. The integration uses connectors for each participating application to exchange data via their API.

Datamodel

Defines objects and their relationships in a PLM system that are managed by the integration to store data from an authoring application.

Dynamic Link Library (DLL)

Defines a file with a library of functions and other information that can be accessed by a Windows program.

Payload

Defines the data contained within an API request. The description is borrowed from the transportation industry, where a truck carries its cargo (its payload) to a location. The truck, as with the API request, is always the same, but the payload changes with each request.

Product Lifecycle Management (PLM)

Defines systems and processes for managing data during the development of a product from creation through manufacturing to maintenance and disposal.

Revision

Defines a released object state in SOLIDWORKS PDM that cannot be modified.

Script engine

Defines the central component in each integration. It contains the integration logic for processing and forwarding the information and data coming from the connectors.

User Interface (UI)

Defines a (usually) graphical interface through which a user interacts with the computer.

Version

Defines an incremental counter of each object modification in SOLIDWORKS PDM on check-in.

x86/x64

Defines the processor architecture in a computer and thus also the performance of applications. x86 corresponds to 32-bit and x64 corresponds to 64-bit.

1 General information

This chapter provides general information about the SOLIDWORKS PDM - Creo Parametric integration.

1.1 Introduction

The Creo Parametric integration for SOLIDWORKS PDM provides functions to save and load design data to and from SOLIDWORKS PDM.

Concurrent engineering is supported through the use of reservation of individual objects and structures.

The integration uses the API from both applications. It is designed to extend existing Creo Parametric and SOLIDWORKS PDM functionality.

The handling of Creo Parametric structure and objects is done in Creo Parametric itself. Creo Parametric is the technical master application, whereas SOLIDWORKS PDM is the organizational master application. SOLIDWORKS PDM manages Creo Parametric objects and structures that are important in the design and approves a possible workflow for individual design objects.

Functionalities of the integration are available via additional toolbars or extended menus in Creo Parametric. They access both the available functions of Creo Parametric and SOLIDWORKS PDM.

1.2 How the integration works

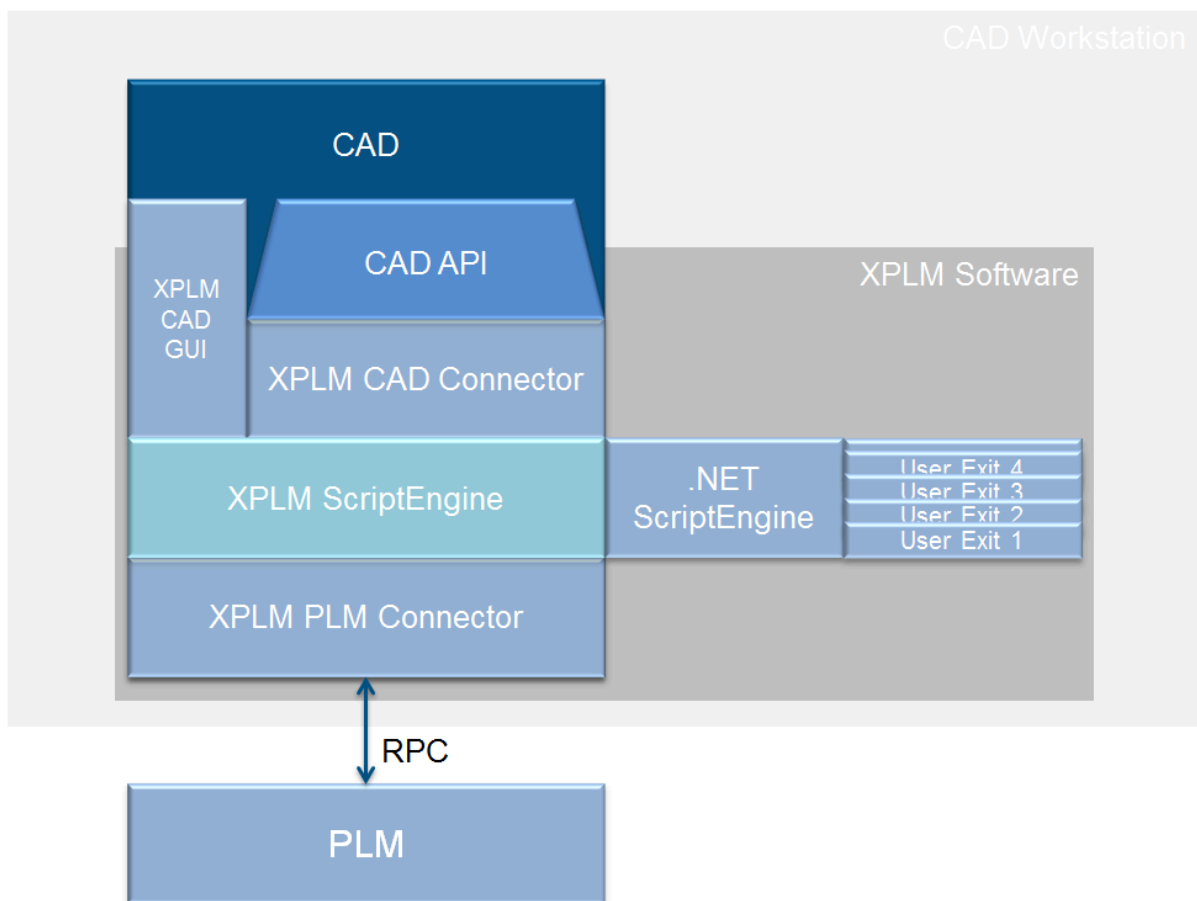
The integration is the interface between Creo Parametric and SOLIDWORKS PDM and allows a consistent data exchange between the applications.

The Creo Parametric connector encapsulates the CAD specific API into Dassault's standard methods to access and manipulate CAD data. Examples: retrieve structure and parameter information, update parameters, change object names.

The SOLIDWORKS PDM connector encapsulates the SOLIDWORKS PDM specific API into Dassault's standard methods to access and manipulate SOLIDWORKS PDM data. Examples: create and update CAD object metadata.

The script engine contains the integration logic between Creo Parametric and SOLIDWORKS PDM. Some of the transactions are exposed in user exits. The user exits receive the full document information at the moment of the execution. The information is passed in a complex data structure. They allow the read access to all document properties and context information as well as the possibility to modify these. The user exits are executed within an embedded VB.NET script engine.

Figure 1: Integration architecture



2 System requirements

2.1 Operating system support

Integration for	Version	Microsoft Windows (x64) 10 11 ¹
PTC Creo Parametric	7.0-10.0 Limited support: 2.0-6.0	X

Support of specific operating system version might be limited by support from specific CAD system.

2.2 Supported SOLIDWORKS PDM releases

Name	Version
Dassault Systèmes SOLIDWORKS PDM Professional	2017-24

2.3 PTC Creo Parametric TOOLKIT license

Prior to the installation, a Creo Parametric TOOLKIT License for the Creo Parametric version in use is required (per server or per site). Install this license on the license server in the company.

2.4 Local machine requirements

Creo Parametric must be properly installed and registered on a local disk and not off a network folder. The Creo Parametric connector installer does not support network installation of Creo Parametric or multiple versions on the same workstation.

2.5 Supported file types

The following file types are supported by the integration.

- Part (.prt)
- Assembly (.asm)
- Drawing (.drw)
- Layout (.lay)
- Format (.frm)
- Sketch (.sec)

¹ Not all CAD systems support Windows 11

3 Installation

This chapter provides information around the installation of the integration.

3.1 Pre-installation information

Optional preview application

For data formats supported by eDrawings, the integration can use the eDrawings viewer as visualization tool for previewing Creo Parametric files.

The eDrawings viewer can view Creo Parametric part (.prt) and assembly (.asm) files natively but not drawings.

3.2 Installing integration

Make sure that all installation media and licenses are available, and then start the installation.

About this task

With this unified installer, you can install Dassault products in the same installation directory and use them in parallel. You can update all components individually without affecting the functionality of other installed products.



The following procedure applies to a new installation. To update an existing installation, see [Updating installation](#) (p. 36) for more information.



You can install, modify, repair, or uninstall also in silent-mode. See [Silent-mode installation](#) (p. 45) for more information.

Procedure

1. Close all open applications related to the integration.
2. Extract the archive and start `Setup-*.exe` with administrator rights.
3. Install any Visual C++ runtimes that you are prompted for.
→ The runtimes are installed, and the installation wizard appears.
4. Click **Next** to start the wizard.
→ The step *License agreement for end-users* appears.
5. Accept the license agreement and click **Next**.
→ The step *Installation path* appears.
6. Check the installation directory. It must point to the directory `CAD Integration` in the SOLIDWORKS PDM installation directory, for example `C:\Program Files\SOLIDWORKS Corp\SOLIDWORKS PDM\CAD Integration\CAD Integration`. Click **Next**.
 - If you have received an overlay package from Dassault, or want to apply your own, enable the option **Apply custom files after installation** and enter the path to the overlay. Overlays contain customized files that overwrite the installed files as the last step. See [Working with overlay packages](#) (p. 49) for more information.

- To backup the existing installation, enable the option **Backup current installation**. Define the backup scope first. Then click **Browse** and select a backup directory.
 - ! Always create a backup when you update or change an installation. This makes it easier to compare and merge the files later.
- The step *MCAD components* appears.
- 7. Select the application(s) to be integrated. If required, select additionally a version or other settings. Click **Next**.
 - The step *Tool components* appears.
- 8. Select additional helper tools or add-ons that you can use in the scope of this installation and click **Next**.
 - Integration Creator
 - The step *Ready to install* appears.
- 9. To start installation, click **Install**.
 - ! During installation, a progress dialog is shown with detailed information on each MSI currently installed. If there are installation issues, the process stops and you can examine the error in the lower dialog section. If required, click **Save to log** and send the log file to support@xplm.com for further investigation. This new progress dialog is available from installer versions 24.3.3.606 onwards.

For older installers, a common CMD window with the progress is shown. Do not click into this window or installation cannot proceed. To continue if you clicked in, press **Enter** or **Esc**.
- 10. To close the wizard after installation, click **Finish**.

Result

Installation is complete and the environment variable `xPlmRootDir` points to the installation directory. You can find log files for all installed components in the directory `C:\ProgramData\XPLM Solution GmbH\logs`.

4 Initial setup

4.1 Installation for SOLIDWORKS PDM 2023 and older environments

In the following, post-installation steps for SOLIDWORKS PDM 2023 and older environments are described.

Procedure

1. After initial installation ([Installing integration](#) (p. 10)), return to the extracted installation archive, open folder `packages` and start `PDMProfessional_[version].msi`.
→ Installation wizard opens.
2. Click **Next** to start the wizard.
→ The step *Modify, repair or remove installation* appears.
3. In this step, you cannot change the installation path, but applying overlay packages or making backups are possible. Refer to [Modifying installation](#) (p. 34) for more information.
4. Click **Next** and select the required SOLIDWORKS PDM version.
5. Click **Next**.
→ The step *Ready to install* appears.
6. To start installation, click **Install**.
→ Installation starts.
7. To close the wizard after installation, click **Finish**.

Result

The installation is modified. Start the product and verify everything works as expected.

4.2 SOLIDWORKS PDM

In the following, SOLIDWORKS PDM-related post-installation steps are described.

4.2.1 Add-in registration

Register the add-in as described in the following chapter.

Before you start

The required files for 32-bit SOLIDWORKS PDM are located in the installation directory under `<SWPDM INSTALL DIR>\CAD Integration\bin\x86\SolidWorksPDM<version>` and for 64-bit under `<SWPDM INSTALL DIR>\CAD Integration\bin\x64\SolidWorksPDM<version>`.

Procedure

1. To register the add-in, open the SOLIDWORKS PDM *Administration* tool, navigate to the desired vault. Right-click *Add-ins* and select **New Add-in...**



If the integration should only be active on one test workstation, register the add-in on this workstation as a *Debug Add-in*.

2. Go to `<SWPDM_INSTALL_DIR>\CAD Integration\bin\x86\SolidWorksPDM<version>` or `<SWPDM_INSTALL_DIR>\CAD Integration\bin\x64\SolidWorksPDM<version>` and select the files
 - EPDM.Interop.epdm.dll
 - SOLIDWORKSPDMPAddin.dll
 - Interop.MSXML2.dll
3. After confirming, a warning message appears. Confirm the message.

Result

Now navigate to the selected vault in the the SOLIDWORKS PDM *Administration* tool and check if the add-in *SOLIDWORKSPDMADDIN* is available under *Add-ins*. It might be possible that the explorer process needs to be restarted first.

Next steps

For 64-bit and 32-bit installations: 32-bit and 64-bit components have to be installed in 64-bit environment to give all clients access to the installation. Errors could occur if the SOLIDWORKS PDM Administration misses rights. Run SOLIDWORKS PDM as Administrator. If a second add-in should be registered, perform the steps described in [SOLIDWORKS API Help](#).



On work stations with the SOLIDWORKS PDM client which does not use the integration and where no installation should be carried out, the Visual C++ Runtime library has to be installed (vcredist_x86.exe, additionally vcredist_x64.exe at 64-bit), otherwise the start of the client causes an error message.

4.2.2 Import of CEX files

Special import files are provided for Creo Parametric. The following chapters describes how the files can be imported and configured.

Before you start

The import files are shipped as an additional Zip archive ([PTC Creo Parametric Integration 2020 SPx.0 \(Build \[version\] Data Cards.7z.exe\)](#)).



For 64-bit and 32-bit installations, 32-bit and 64-bit components have to be installed in 64bit environment together to give all clients access to the installation. Errors could occur if the SOLIDWORKS PDM Administration tool misses rights, so please run SOLIDWORKS PDM as Administrator.

About this task

To import, do the following:

Procedure

1. Unzip the provided import files.
2. Open the SOLIDWORKS PDM Administration tool and navigate to the desired vault.
3. Perform a right-click on the vault and select *Import...*
4. Navigate to the unzipped archive, select the file `CreoParametric.cex` and press **Open**.
5. Confirm the upcoming information messages.

Result

Now example file cards are added to the desired vault.

Next steps

File cards can be configured with former variables. How the file cards look, can vary in the customer environment. The fields could either be created manually, imported or copied.

The variables need an attribute with the block name `CustomProperty`. The name that corresponds to the variable and the Creo Parametric file types (`asm`, `drw`, `frm`, `lay`, `prt`, `sec`). The name must NOT contain a blank.

Afterwards the file cards can be modified by adding additional fields. Also the file cards for other integration used file cards (like `.doc`) can be configured as well. The exact design is up to each SOLIDWORKS PDM administrator, as long as the needed variables are available in the file card.



All fields in the provided file cards are necessary to ensure that the integration works properly.

4.3 Creo Parametric

In the following, Creo Parametric-related post-installation steps are described.

Make sure that `CreoParametricEvent_ExitNotify` is set to `true` in `XPlmCreoParametricConnector.xml` to ensure that the integration works properly.

4.3.1 Creo Parametric configuration settings

Creo Parametric uses multiple text based files to control how the program is configured. A separate text file named `config.pro` is important to consider.

Each configuration file option in `config.pro` contains a default value that is set by Creo Parametric. If you do not change an option, Creo Parametric uses the default value. There are two methods to change a `config.pro` option:

- Click **Tools > Options** in Creo Parametric and use the *Options* dialog.
- Open the `config.pro` file from within a text editor and add or change configuration options directly.

In general, `config.pro` file settings should be changed before starting a Creo Parametric session. Changes made while in session need to be saved to the `config.pro` file or they will be lost after closing the application.

The options in `config.pro` are set by typing the option name followed by a space then the value (such `save_drawing_picture_file embed`). The following option settings are recommended for SOLIDWORKS PDM:

Table 1: Option settings

Option	Recommended value	Description
<code>save_drawing_picture_file</code>	<code>embed</code>	Embeds an image of drawing so that viewing programs like Product View Express-Creo View Express can display the drawing contents.

Option	Recommended value	Description
save_model_display	shading_lod	Saves the shaded image information in the file to enable viewing of the model in programs like eDrawings and ProductView Express-Creo View Express.
The following configuration option settings can affect the behavior of SOLIDWORKS PDM.		
create_drawings_dims_only	yes	<p>Setting this option to no means that dimensions created in drawing mode are added to the solid model file. If the solid model is not checked out in SOLIDWORKS PDM, then the drawing dimensions are lost. Setting this option to yes creates associative draft dimensions in the drawing so the solid model does not need to be checked out. Also, if multiple drawings of the same part or assembly exist, strong consideration should be given to setting this to yes, because deleting a dimension in another drawing would remove it from every drawing.</p> <p>Default: no</p> <p>Possible values: yes no</p>
rename_drawing_with_object	none	<p>If set to both, part or assembly, then anytime a part and/or assembly is renamed using the Creo Parametric File menu option Save A Copy, the drawing is renamed as well. In order for this to work, the drawing and the associated model file(s) need to be in the same folder and have the same name. It is recommended to set this option to none or to delete the option entirely.</p> <p>Default: none</p> <p>Possible values: both part assembly none</p>
override_store_back	no	<p>Enables saving files to an alternate location if the user does not have write access to the directories where the files were opened from. This is used in conjunction with the save_objects_in_current option described below. It is recommended to set this to no or to delete the option entirely.</p> <p>Default: no</p> <p>Possible values: yes no</p>

Option	Recommended value	Description
save_objects_in_current	no	<p>Saves all files modified in session into the current working directory instead of the directory where they were opened from. It is recommended to set this option to <code>no</code> or to delete the option entirely, since any modified files need to be saved back into the same directory.</p> <p>Default: no</p> <p>Possible values: yes no</p>

4.4 Additional information

Language settings

To adapt, please refer to the file `xplm-proe.cfg` available in the directory `com/config` below the installation directory. In this file, variable `LANG` can be changed from `English` to `German`.

Workflows

The SOLIDWORKS PDM workflow transitions can be configured to update card variables and Creo Parametric parameters. For Creo Parametric files, this requires executing the transition on a computer where Creo Parametric is installed and licensed because SOLIDWORKS PDM needs to launch Creo Parametric to write to the parameter(s) in the Creo Parametric file.

The mapping between a card variable and a Creo Parametric parameter is done via the `CustomProperty`.

The *Revision* card variable by default already uses the `CustomProperty` block named *Revision*. Edit this variable to add the Creo Parametric file extensions prt, asm, frm, lay, sec and drw as needed to the list.

The same principle can be applied to all other variables.

Windows folder system

Files can be simply copied and pasted or drag and dropped from a Windows file system into SOLIDWORKS PDM, but careful planning is required to make sure all the reference files are present and the search paths are properly specified before copying any files. Because the Creo Parametric connector starts Creo Parametric when you drag and drop Creo Parametric files to the vault, the more files you drag and drop, the more files Creo Parametric have to load into memory. It is therefore recommended to copy smaller groups of files or folders at a time.



- The client workstation from which you run the import must have Creo Parametric installed and licensed with the search paths properly specified.
- The **File > Backup** menu command in Creo Parametric can also be used to copy all opened reference files in a Creo Parametric session into the SOLIDWORKS PDM vault.
- Having many parameters mappings to card variables will noticeably impact the time to add Creo Parametric files to a SOLIDWORKS PDM vault.

When pasting Creo Parametric version files

Once all the files have been added, it is best to destroy all the Creo Parametric files with the .1, .2 extensions from the SOLIDWORKS PDM recycle bin. The files are automatically deleted by the paste action. Right-click on the vault root folder, select **Properties** from context menu, go to the *Deleted Items* tab, select **include items in sub-folders**, select all Creo Parametric files with a .1, .2, ... extensions and choose **Destroy**.

Search paths

It is very important to consider that Creo Parametric files do not store the full path information for the files they reference. Thus search paths must be used to ensure all reference files are found; the SOLIDWORKS PDM-Creo Parametric connector does not alter this behavior.

Reference Files/Search Paths

Creo Parametric is not a native *Windows* application like SOLIDWORKS, and because of this, certain methods of opening, saving and accessing files need to be well understood and accepted

Since Creo Parametric does not store the full path information for referenced files like SOLIDWORKS does, Creo Parametric needs to be told where to look for references. Even though SOLIDWORKS PDM knows the folder location of files referenced by a Creo Parametric assembly or drawing, Creo Parametric does not find them unless the files are in any of the following locations (shown in order of search):

- session memory (RAM)
- directory where the user selected the parent file from
- specified *Working Directory*
- search paths called out in either the config.pro or the search.pro file

It is therefore important that a search.profile be maintained with all necessary folder paths.

5 Configuration

5.1 SOLIDWORKS PDM

In the following, further SOLIDWORKS PDM-related configuration possibilities are described.

5.1.1 Setup tasks in SOLIDWORKS PDM add-in

With tasks, Creo Parametric files can be converted by right-clicking a file and selecting a convert command.

About this task

To setup the tasks defined in `<SWPDM_INSTALL_DIR>\CAD Integration\xml\PDMPCreoParametricAddin.xml`, do the following:

Procedure

1. Open SOLIDWORKS PDM *Administration* tool.
2. Navigate to the desired vault and then to *Tasks*.
3. Perform a right-click and select **New Task...**
4. Enter a name for the new task (e.g. Convert PDF).
5. For *Add-in* select the *PDMP Addin* and click **Next**.
6. Within the *Execution Method* screen, the computer is shown in the *Computers supporting the task* list. If not perform a right-click on the SOLIDWORKS PDM icon in Windows hidden icons and select **Task Host Configuration**.
 - a) On the top right hand side, select **File Vault**.
 - b) In the table, set the **Permit** flag for the SOLIDWORKS PDM add-in and click **OK**.
7. Now click **Refresh List** in the task UI.
Computer is now visible in the table.
8. Enable the computer and click **Next**.
9. In the *Scheduling* section, the execution time for the task can be defined. In this example we use **This task is not scheduled**. So just click **Next**.
10. In the *Converter Configuration* section, the data from the `ScriptEngine` is defined. Make sure that the values for *Menuitems* and *Targetformat* are set correctly and click **Next**.
Possible values for *Targetformat* are:
 - For the conversion of drawings to the PDF format use: PDF
 - For the conversion of drawings to the DXF format use: DXF
 - For the conversion of 3D models to the IGES format use: IGES
 - For the conversion of 3D models to the STEP format use: STEP
11. In *Permissions*, *Success Notification* and *Error Notification* no changes have to be made. So click **OK**.

Result

Now start an explorer, go to the SOLIDWORKS PDM vault and check if the menu entries are available in the context menu. It might be possible that the explorer process needs to be restarted first.

Next steps

Do the above described process for all desired tasks and close the *Administration* tool.

5.1.2 Adding customer variables

The following chapter describes how customer variables can be added.

Procedure

1. Add the new variable to SOLIDWORKS PDM.
 - a) Open SOLIDWORKS PDM *Administration* tool.
 - b) Navigate to the desired vault and then to *Variables*.
 - c) Perform a right-click and select **New Variable...**
 - d) Within the *Edit Variable* dialog
 - enter the *Variable name*, e.g. **CustomNumber**,
 - select *Variable type* **Text**,
 - and add a new attribute with a *Block name* **CustomProperty**, an *Attribute name* which corresponds to the Creo Parametric parameter (e.g. **CreoCustomNumber**) and a file extension (asm, drw, frm, lay, mfg, mrk, prt, rep or sec).

→ New variable is created in SOLIDWORKS PDM.
2. Add the new variable to the Creo Parametric data card(s).
 - a) Open SOLIDWORKS PDM *Administration* tool.
 - b) Navigate to the desired vault and then to **Cards > File Cards**.
 - c) Perform a double-click on the corresponding data card (e.g. *PTC Creo Parametric Card*).
 - d) Within the *Card Editor* and add the new variable to the data card.

3. Adapt transactions regarding metadata mapping.

Transactions are stored in the file `PDMPCreoParametricTransaction.xml` within the installation directory of the integration (`<SWPDM INSTALL DIR>\CAD Integration\xml`).

a) Add the new variable to transaction `getFileDetailsBulk`.

Make a copy of an existing field and adapt the tag `Name`, which corresponds to the SOLIDWORKS PDM variable name. The tag `Value` is always empty here.

```
<Transaction>
  <Aliasname>getFileDetailsBulk</Aliasname>
  <Import>
    <Parameter>
      <FieldCollection>
      </FieldCollection>
      <StructureCollection>
        ...
      <Structure>
        <Name>VariableNames</Name>
        <FieldCollection>
          ...
          <Field>
            <Name>Date</Name>
            <Value></Value>
          </Field>
          <Field>
            <Name>Part Type</Name>
            <Value></Value>
          </Field>
          <Field>
            <Name>CustomNumber</Name>
            <Value></Value>
          </Field>
        </FieldCollection>
      </Structure>
    </StructureCollection>
    <TableCollection>
    </TableCollection>
  </Parameter>
</Import>
</Transaction>
```

- b) Add the new variable to transaction `UpdateTitleBlock`.

Make a copy of an existing field and adapt the tag `Name`, which corresponds to the Creo Parametric parameter (use capital characters here) and the tag `Attribute`, which corresponds to the SOLIDWORKS PDM variable name.

```
<Transaction>
  <Aliasname>UpdateTitleBlock</Aliasname>
  <Import>
    <Parameter>
      <StructureCollection>
        <Structure>
          <Name>CustomProperties</Name>
          <FieldCollection>
            ...
            <Field>
              <Name>FOLDER_ID</Name>
              <Type>XPlmDocument</Type>
              <Subtype>FieldCollection</Subtype>
              <Attribut>FolderID</Attribut>
            </Field>
            <Field>
              <Name>CREOCUSTOMNUMBER</Name>
              <Type>XPlmDocument</Type>
              <Subtype>FieldCollection</Subtype>
              <Attribut>CustomNumber</Attribut>
            </Field>
          </FieldCollection>
        </Structure>
      </StructureCollection>
    </Parameter>
  </Import>
</Transaction>
```



Adapting the transaction `getVariables` and `UpdateProperties` is not required because they are not used anymore.

4. Adapt transactions regarding the commands **File Info** and **View Tree in Browser**.

If the new variable shall be displayed within Creo Parametric internal browser, that is used to display the result of the commands **File Info** and **View Tree in Browser**, it is necessary

to adapt transaction `ShowInfoInInternalBrowser`. This transaction is also stored in file `PDMPCreoParametricTransaction.xml`.

a) Add the new variable to transaction `ShowInfoInInternalBrowser`.

Make a copy of an existing field and adapt the tag `Name`, which is the text that is displayed within the browser, and the tag `Attribut`, which corresponds to the SOLIDWORKS PDM variable name.

```
<Transaction>
  <Aliasname>ShowInfoInInternalBrowser</Aliasname>
  <Import>
    <Parameter>
      <FieldCollection>
        ...
        <Field>
          <Name>State</Name>
          <Type>XPlmDocument</Type>
          <Subtype>FieldCollection</Subtype>
          <Attribut>moCurrentState</Attribut>
        </Field>
        <Field>
          <Name>Customer Number</Name>
          <Type>XPlmDocument</Type>
          <Subtype>FieldCollection</Subtype>
          <Attribut>CustomNumber</Attribut>
        </Field>
      </FieldCollection>
      <StructureCollection>
    </StructureCollection>
    <TableCollection>
    </TableCollection>
    </Parameter>
  </Import>
</Transaction>
```

5.1.3 Improving check in performance

About this task

SOLIDWORKS PDM can automatically include drawings referencing a part or assembly when the latter is checked in. The available options in SOLIDWORKS PDM manage this functionality. Including drawings may make the check in command slower. To prevent this, perform the following steps.

Procedure

1. Start the *Administration* tool of SOLIDWORKS PDM.
2. Double-click the vault in question.
3. Double-click **Users** and then double-click current user.
→ *Admin - Properties* dialog opens.

4. Click **Settings** (on the lower left hand side).
→ *Settings* dialog opens.
5. Select **Check In**.
→ Option **Look for drawings in the entire vault is enabled**.
6. Disable this option to improve the performance.

Result

Check in performance is improved.

5.1.4 Configuration files

The SOLIDWORKS PDM integration for Creo Parametric is configured mainly using following configuration files:

- `PDMPConnector.xml`: Base configuration of the SOLIDWORKS PDM connector
- `PDMPCreoParametricConnector.xml`: Contains mostly Creo Parametric related settings
- `PDMPFileOpener.xml`: Contains configurations for file opening behavior

PDMPConnector.xml

Table 2: Settings and values

Setting	Purpose and available values
Settings for SOLIDWORKS PDM logging	
EnablePDMPLogging	<p>If set to <code>true</code>, logging is enabled. Refer to chapter Enabling logging (p. 28) for more information.</p> <p>It is recommended to set this setting to <code>false</code>, because the logging can negatively affect the performance of the integration.</p> <p>Default: false</p> <p>Possible values: true false</p>
PDMPLogFile	<p>Is the definition of the destination of the log file.</p> <p>Important is that the destination is a folder for which the user has write permissions.</p> <p>It is recommended to set the file extension to <code>*.xml</code>.</p> <p>For example C:\Users\User1\PDMLog.xml</p>

Setting	Purpose and available values
PDMLogLevel	<p>Definition of the granularity of the log messages.</p> <p>Possible values for the log level are 1 to 10 .</p> <p>Levels between 1 and 4 are reserved for logging in the business components, while levels 5 to 9 include logging information with respect to the data exchange with SOLIDWORKS PDM.</p> <p>Level 10 provides the maximum number of log messages.</p> <p>The higher the level, the more data is generated.</p> <p>Default: 5</p> <p>Possible values: 1-10</p>
PDMWorksCancel OnError	<p>If set to <code>true</code>, any process is aborted on error.</p> <p>Default: true</p> <p>Possible values: true false</p>
PDMLogIncludeDate	<p>If set to <code>true</code>, every log entry contains date and time. If set to <code>false</code>, date and time are not written into the log file. This allows comparing two log files.</p> <p>Default: true</p> <p>Possible values: true false</p>
Settings for help	
PDMWorksHelpFile	<p>Currently not used.</p> <p>Defines the name of the helper file.</p> <p>For example: UserGuide.chm</p>
PDMWorksHelpTitle	<p>Currently not used.</p> <p>Defines the title of the helper file.</p> <p>For example: UserGuide</p>
Settings for connector	
PDMWorks UndoWorkaround	<p>This workaround setting is for SOLIDWORKS PDM 2019, starting with SP0.</p> <p>To avoid that files are missing when using function Undo Check Out, set this setting to <code>true</code>.</p> <p>Default: true</p> <p>Possible values: true false</p>
Settings for events	

Setting	Purpose and available values
Callback_*	<p>The callbacks are transferred from the SOLIDWORKS PDM API. Please obtain further information there.</p> <p>If set to <code>true</code>, it is activated. If set to <code>false</code>, it is deactivated.</p>
Callback_EdmCmd_PreRename	<p>If set to <code>true</code>, SOLIDWORKS PDM Rename command produces an error:</p> <p>Using 'Rename' in this context is not possible. Please use the 'Rename' command within Creo Parametric.</p> <p>Default: <code>true</code></p> <p>Possible values: <code>true</code> <code>false</code></p>
Callback_EdmCmd_PreAdd	<p>If set to <code>true</code>, functionality is activated, that runs when Creo Parametric version files are copied into the vault directory. The version files are automatically renamed to normal files (i.e. from <code>file1.prt.3</code> to <code>file1.prt</code>) and added to the vault.</p> <p>Default: <code>true</code></p> <p>Possible values: <code>true</code> <code>false</code></p>
Callback_EdmCmd_PostAdd	<p>If set to <code>true</code>, functionality is activated, that runs when Creo Parametric version files are copied into the vault directory. The version files are automatically renamed to normal files (i.e. from <code>file1.prt.3</code> to <code>file1.prt</code>) and added to the vault.</p> <p>Default: <code>true</code></p> <p>Possible values: <code>true</code> <code>false</code></p>

PDMPCreoParametricConnector.xml

Table 3: Settings and values

Setting	Purpose and available values
Settings for main script engine	
ActivePDMPScriptEngine	<p>Do not change.</p> <p>Script engine to contain and perform the connector process layer.</p> <p>Default: <code>PDMPCreoParametricScriptengine.CScriptEngine</code></p>
Menu definition of the context menu entries within SOLIDWORKS PDM UI	
PDMWorksMenuFiles	<p>Do not change.</p> <p>Name of add-in XML file.</p> <p>Default: <code>PDMPCreoParametricPDMPAddin.xml</code></p>

Setting	Purpose and available values
Handling of error message	
PDMErrorSilent	<p>Defines, if error is shown or not.</p> <p>Default: 0</p> <p>Possible values: 0 (interactive) 1 (silent)</p>
Settings for communication component	
RunMacro	<p>Defines the used communication component.</p> <p>Default: extern</p> <p>Possible values: extern (XPlmCOMService) intern (XPlmUtilities)</p>
Setting for using timestamps	
PDMWorksUse PDMTimestamp	<p>Changing this setting may causes instability.</p> <p>Defines, if SOLIDWORKS PDM uses time stamp from the SOLIDWORKS PDM data set or the local file when determining the metadata. It is deactivated by default.</p> <p>Default value: false</p> <p>Possible values: true false</p>
Settings for Creo Parametric startup	
CreoParametricPDMP StartupScript	<p>Defines the used startup script.</p> <p>Default value: %CreoParametricPDMPStartupScript%</p>
CreoParametricPDMP StartupShowWindow	<p>If set to <code>false</code>, the CMD window is not displayed when Creo Parametric is started via a FileFormatPlugin function or a conversion task.</p> <p>Default: false</p> <p>Possible values: true false</p>
Setting to specify where the integrations connection ID file is stored	
CreoParametricPDMP ConnectionIdPIDFile	<p>Defines the path to the connection ID file.</p> <p>Default value: %TEMP%\pdmpconnectionid.pid</p> <p>Possible values: any valid path</p>
Setting to specify where the HTML template, which is used by the command File Info, is stored	
CreoParametricPDMPPath ToInfoTemplate	<p>Defines the path to the HTML template.</p> <p>Default value: %xPlmRootDir%com\config\info_html_template.html</p> <p>Possible values: any valid path</p>
Settings for configurations	

Setting	Purpose and available values
CreoParametricPDMPRetrieveConfigurationStructure	Described in chapter Assemblies with family tables (p. 32). Default value: GenericAndActiveConfiguration Possible values: OnlyGeneric GenericAndActiveConfiguration GenericOnlyButUseItForAllConfigurations
CreoParametricPDMPDisplayGenericTabInDataCard	Defines if the generic tab is displayed in the data card tab in SOLIDWORKS PDM. Default value: Always Possible values: Always OnlyIfThereAreConfigurations
Settings for starting Creo Parametric	
StartCAD	Defines if Creo Parametric starts by clicking or double-clicking files with defined extensions, see StartCADExtensions . Default value: true Possible values: true false
StartCADMethod	Defines the method how Creo Parametric is started. Default value: startCreoParametric
StartCADExtensions	Defines for which file extensions Creo Parametric starts by double-click. Default values: ASM, PRT, DRW, SEC, FRM, LAY
Settings for renaming	
AllowRenameInWindowsExplorerAlthoughReferencesMayBeBroken	Described in chapter Renaming (p. 31). Default value: false Possible values: true false
Settings for conversion	
CreoParametricViewFilePDFOption	Defines the form in which the PDF file is created. Default value: PDFOPT_COLOR_DEPTH=COLOR;PDFOPT_LAYER_MODE=ALL Possible values for PDFOPT_COLOR_DEPTH: MONO GRAY COLOR Possible values for PDFOPT_LAYER_MODE: ALL VISIBLE NONE
Settings for where-used documents (perform a check, whether the number of all where-used documents is larger than a threshold)	
ThresholdNumberOfWhereUsedFiles	Threshold used during renaming process. Described in chapter Renaming (p. 31). Default value: 7

Setting	Purpose and available values
ThresholdCheck ForNewlyAddedFiles	Threshold used when Creo Parametric command Save As - Save a copy is executed. Default value: 4
Settings for events	
CreoParametricEvent _StartNotify	This option is linked to a callback event created by CAD. If such an option is activated, the corresponding transaction defined in the XML file <code>PDMPCreoParametricTransaction.xml</code> is executed on event occurrence. It is recommended to set this option to <code>false</code> in conversion environments. Default: true Possible values: true false

PDMPFileOpener.xml

Table 4: Settings and values

Setting	Purpose and available values
FileOpenerDefinition	This transaction contains file extensions.
inittimeout	If any timing issues appear during file opening, this value can be increased. Default value: 20
ShowCommandWindow	If set to <code>false</code> , the command window is not displayed on starting Creo Parametric. Default value: false Possible values: true false

5.1.5 Enabling logging

If required, you can activate logging for the components used in the integration. Logging should not run permanently, as a large amount of log messages are generated and performance is affected. Deactivate logging again, after the issue is solved.

Execute following steps to enable logging:

1. Edit the file `PDMPConnector.xml`:
 - A. Set `EnablePDMPLogging` to `true`.
 - B. Set value `PDMPLogFile` to a path on local disc.
 - C. Set value of `PDMPLogLevel` accordingly.
2. Edit the file `XPlmCreoParametricConnector.xml`:
 - A. Set `EnableCreoParametricLogging` to `true`.
 - B. Set value `CreoParametricLogFile` to a path on local disc.
 - C. Set value of `CreoParametricLogLevel` accordingly.

5.2 Creo Parametric

In the following, further Creo Parametric-related configuration possibilities are described.

5.2.1 Configuration Files

The SOLIDWORKS PDM integration for Creo Parametric is configured mainly using the following configuration files:

XPlmCreoParametricConnector.xml



The configuration of the `XPlmCreoParametricConnector.xml` depends on the environment.

When converting it should be ensured that the environment is not used for client operations and vice versa.

For better user handling, particular events are used in client environments. These events check changes of CAD files and check if files are for example checked in or out. Through interactions (message boxes) the user can make decisions.

During task operations in the conversion environment, the appearance of message boxes causes the task to terminate.

For this reason, most events need to be deactivated in conversion environments.

Table 5: Settings and values

Setting	Purpose and available values
Environmental Adjustments - Client	
EnableCreoParametricLogging	If set to true, logging is enabled. Default: false Possible values: true false
CreoParametricLogFile	Value is the full path to the log file, required if logging is enabled. For example C:\tmp\CreoParametric.log
CreoParametricIgnoreMissingParts	Missing parts are ignored when enabled. For client environments it is recommended to set this option to <code>true</code> . Default: true Possible values: true false
CreoParametricEvent_ExitNotify	This option is linked to a callback event created by CAD. If such an option is activated, the corresponding transaction defined in the XML file <code>PDMPCreoParametricTransaction.xml</code> is executed on event occurrence. It is recommended to set this option to <code>true</code> in client environments. Default: true Possible values: true false

Setting	Purpose and available values
CreoParametricEvent_DocumentChange	<p>It is recommended to set this option to <code>true</code> in client environments.</p> <p>Default: true</p> <p>Possible values: true false</p>
CreoParametricEvent_OpenDocument	<p>It is recommended to set this option to <code>false</code> in client environments.</p> <p>Default: false</p> <p>Possible values: true false</p>
CreoParametricEvent_ActivateDocument	<p>It is recommended to set this option to <code>false</code> in client environments.</p> <p>Default: false</p> <p>Possible values: true false</p>
CreoParametricDocumentEvent_DocumentChangeSelectSet	<p>It is recommended to set this option to <code>false</code> in client environments.</p> <p>Default: false</p> <p>Possible values: true false</p>
CreoParametricEvent_BrowserPane_OnActivate	<p>It is recommended to set this option to <code>true</code> in client environments.</p> <p>Default: true</p> <p>Possible values: true false</p>
Environmental Adjustments - Conversion (Tasks)	
CreoParametricEvent_StartNotify	<p>Change of property value in file <code>PDMPCreoParametricConnector.xml</code>. This option is linked to a callback event created by CAD. If such an option is activated, the corresponding transaction defined in the XML file <code>PDMPCreoParametricTransaction.xml</code> is executed on event occurrence. It is recommended to set this option to <code>false</code> in conversion environments.</p> <p>Default: false</p> <p>Possible values: true false</p>

Setting	Purpose and available values
CreoParametricEvent_ExitNotify	<p>This option is linked to a callback event created by CAD. If such an option is activated, the corresponding transaction defined in the XML file <code>PDMPCreoParametricTransaction.xml</code> is executed on event occurrence. It is recommended to set this option to <code>true</code> in conversion environments.</p> <p>Default: <code>true</code></p> <p>Possible values: <code>true</code> <code>false</code></p>
CreoParametricEvent_DocumentChange	<p>It is recommended to set this option to <code>false</code> in conversion environments.</p> <p>Default: <code>false</code></p> <p>Possible values: <code>true</code> <code>false</code></p>
CreoParametricEvent_OpenDocument	<p>It is recommended to set this option to <code>false</code> in conversion environments.</p> <p>Default: <code>false</code></p> <p>Possible values: <code>true</code> <code>false</code></p>
CreoParametricEvent_ActivateDocument	<p>It is recommended to set this option to <code>false</code> in conversion environments.</p> <p>Default: <code>false</code></p> <p>Possible values: <code>true</code> <code>false</code></p>
CreoParametricDocumentEvent_DocumentChangeSelectSet	<p>It is recommended to set this option to <code>false</code> in conversion environments.</p> <p>Default: <code>false</code></p> <p>Possible values: <code>true</code> <code>false</code></p>
CreoParametricEvent_BrowserPane_OnActivate	<p>It is recommended to set this option to <code>false</code> in conversion environments.</p> <p>Default: <code>true</code></p> <p>Possible values: <code>true</code> <code>false</code></p>

5.3 Renaming

Renaming via Creo Parametric

During the rename process, multiple checks are performed. Among others a check is performed, whether the number of all where-used documents is larger than a threshold. The height of the threshold is defined in the `PDMPCreoParametricConnector.xml`, property `ThresholdNumberOfWhereUsedFiles`). The default value is `7`.

If the value is exceeded, the user receives a question box with the following content:

The file is currently used in XXX other files: (followed by a list of all these filenames - maximum number of files displayed is 50 - if there are more than 50, then another line is added: and YYY more objects).
Do you really want to rename?

Renaming via SOLIDWORKS PDM

The option `AllowRenameInWindowsExplorerAlthoughReferencesMayBeBroken` in `PDMPCreoParametricConnector.xml` enables or disables the possibility to rename Creo Parametric files.

If it is set to `false`, the renaming via *Windows Explorer* is only allowed for

- drawings and sections
- other objects that are not used within any other object

If the user tries to rename objects, which have where-used objects referenced, an appropriate error message appears.

If it set to `true`, the rename function is enabled and allowed in any cases. This option is set to `false` by default.



Drawings and assemblies may be corrupted after executing the **Rename** command via the *Windows Explorer*, **especially when**:

- renaming parts that are referenced by a drawing. The drawing is ALWAYS corrupted. It is not loadable anymore to Creo Parametric.
- renaming objects used on different levels within the structure. It will fail to update the references within all parent assemblies.
- when retrieving outdated versions from SOLIDWORKS PDM using either the **Get Version** command of the Creo Parametric integration or the **Get** command within the *History* dialog of Windows Explorer, the resulting file will be corrupted and won't be loadable into Creo Parametric.

This is primarily true for:

- ☐ drawings, that reference renamed parts or assemblies.
- ☐ assemblies with multiple references to renamed objects on different level.

5.4 Assemblies with family tables

SOLIDWORKS PDM allows to store configuration specific structure. Within the *Contains* tab of *Windows Explorer* the user can select between the following modes:

- Do not show configuration
- Generic
- Configuration 1
- Configuration 2
-
- Configuration n

However, due to performance reasons the SOLIDWORKS PDM – Creo Parametric integration does not store all configuration-dependent structure.

Instead there is a property within `PDMPCreoParametricConnector.xml` called `CreoParametricPDMPRetrieveConfigurationStructure` with the following possible values:

- `OnlyGeneric` - Only displays the structure within the generic configuration and leaves the structure of all configurations empty.
- `GenericAndActiveConfiguration` - Displays the correct structure for the generic and the active configuration, leaves the structure of non-active configurations empty. This is the default value for this property.
- `GenericOnlyButUseItForAllConfigurations` - Inquires the generic structure and displays it for generic and for any configuration.

6 Update

6.1 Modifying installation

Complete these steps to modify an existing installation and add for example new components or remove existing.



About this task

During modification, no existing files are overwritten and only missing files are added.



You can install, modify, repair, or uninstall also in silent-mode. See [Silent-mode installation](#) (p. 45) for more information.

Procedure

1. Close all open applications related to the integration.
2. Start the current installer `Setup-*.exe` with administrator rights. If this file is no longer available, start `Setup-*.msi` directly from the directory `C:\ProgramData\XPLM Solution GmbH\packages`. Alternatively, you can also start the installer from Windows:
 - a) In the system settings, go to *Apps & Features* and search for the entry **PTC Creo Parametric Setup**.
 - b) Select the entry and click **Modify**.→ Visual C++ runtimes are checked/installed again and the installation wizard appears.
3. Click **Next** to start the wizard.
→ The step *Modify, repair or remove installation* appears.
4. Click **Modify**.
→ The step *Installation path* appears.
5. In this step, you cannot change the installation path, but applying overlay packages or making backups are possible.
 - If you have received an overlay package from Dassault, or want to apply your own, enable the option **Apply custom files after installation** and enter the path to the overlay. Overlays contain customized files that overwrite the installed files as the last step. See [Working with overlay packages](#) (p. 49) for more information.
 - To backup the existing installation, enable the option **Backup current installation**. Define the backup scope first. Then click **Browse** and select a backup directory. Always create a backup when you update or change an installation. This makes it easier to compare and merge the files later.
6. Click **Next** and update components, if required.
→ The step *Ready to install* appears.
7. To start installation, click **Install**.
 During installation, a progress dialog is shown with detailed information on each MSI currently installed. If there are installation issues, the process stops and you can examine the error in the lower dialog section. If required, click **Save to log** and send the log file to support@xplm.com

for further investigation. This new progress dialog is available from installer versions 24.3.3.606 onwards.

For older installers, a common CMD window with the progress is shown. Do not click into this window or installation cannot proceed. To continue if you clicked in, press **Enter** or **Esc**.

8. To close the wizard after installation, click **Finish**.

Result

The installation is modified. Start the product and verify everything works as expected.

Related links

[Working with overlay packages](#) (p. 49)

6.2 Repairing installation

Complete these steps to repair an existing installation if the product does not work correctly, for example fixing missing or corrupt files, or incorrect shortcuts and registry entries.

About this task

During repair, existing files are overwritten and components registered again.



You can install, modify, repair, or uninstall also in silent-mode. See [Silent-mode installation](#) (p. 45) for more information.

Procedure

1. Close all open applications related to the integration.
2. Start the current installer `Setup-*.exe` with administrator rights. If this file is no longer available, start `Setup-*.msi` directly from the directory `C:\ProgramData\XPLM Solution GmbH\packages`. Alternatively, you can also start the installer from Windows:
 - a) In the system settings, go to *Apps & Features* and search for the entry **PTC Creo Parametric Setup**.
 - b) Select the entry and click **Modify**.→ Visual C++ runtimes are checked/installed again and the installation wizard appears.
3. Click **Next** to start the wizard.
→ The step *Modify, repair or remove installation* appears.
4. Click **Repair**.
→ The step *Installation path* appears.
5. In this step, you cannot change the installation path, but applying overlay packages or making backups are possible.
 - If you have received an overlay package from Dassault, or want to apply your own, enable the option **Apply custom files after installation** and enter the path to the overlay. Overlays contain customized files that overwrite the installed files as the last step. See [Working with overlay packages](#) (p. 49) for more information.

- To backup the existing installation, enable the option **Backup current installation**. Define the backup scope first. Then click **Browse** and select a backup directory.
 - ⚠ Always create a backup when you update or change an installation. This makes it easier to compare and merge the files later.
- 6. Click **Next**.
 - The step *Ready to install* appears.
- 7. To start installation, click **Install**.
 - ⚠ During installation, a progress dialog is shown with detailed information on each MSI currently installed. If there are installation issues, the process stops and you can examine the error in the lower dialog section. If required, click **Save to log** and send the log file to support@xplm.com for further investigation. This new progress dialog is available from installer versions 24.3.3.606 onwards.

For older installers, a common CMD window with the progress is shown. Do not click into this window or installation cannot proceed. To continue if you clicked in, press **Enter** or **Esc**.
- 8. To close the wizard after installation, click **Finish**.
- 9. **Optional:** Carefully compare and merge the changes from the backup directory with the newly installed files.

Result

The installation is repaired. Start the product and verify everything works as expected.

Related links


[Working with overlay packages](#) (p. 49)

6.3 Updating installation

Complete these steps to update an existing installation.

About this task

Dassault strongly recommends using appropriate services for an update. This ensures that existing functionality and modifications are correctly transferred to the new product. Contact <https://www.solidworks.com/support/> for assistance.

 You can install, modify, repair, or uninstall also in silent-mode. See [Silent-mode installation](#) (p. 45) for more information.

Procedure

1. Close all open applications related to the integration.
2. Start the new installer `Setup-*.exe` with administrator rights.
 - Visual C++ runtimes are checked/installed again and the installation wizard appears.

3. Click *Next* to start the wizard.

→ The installer detects an existing installation and shows a message.

- If the existing installation is compatible with the unified installer, you can proceed. At the start of the installation, the old components are removed first and the new ones are installed afterwards.

How to identify if the existing installation is already compatible with the unified installer technology?

- The directory `C:\ProgramData\XPLM Solution GmbH` exists.
- The registry entry `HKLM\SOFTWARE\XPLM Solution GmbH\{00000000-0000-0000-0000-000000000000}` exists.

- If the existing installation is not compatible with the unified installer, it will first be uninstalled completely. Before you continue, manually back up the existing installation directory `<SWPDM INSTALL DIR>\CAD Integration`. Then proceed with installation.



In both cases, files with the prefix `customer_` are not affected by the update. All other files are overwritten with the new files.

4. Click *Next*.

→ The step *License agreement for end-users* appears.

5. Accept the license agreement and click *Next*.

→ The step *Installation path* appears.

6. If an existing and compatible installation was found, you cannot change the installation path in this step, but applying overlay packages or making backups are possible.

- If you have received an overlay package from Dassault, or want to apply your own, enable the option **Apply custom files after installation** and enter the path to the overlay. Overlays contain customized files that overwrite the installed files as the last step. See [Working with overlay packages](#) (p. 49) for more information.
- To backup the existing installation, enable the option **Backup current installation**. Define the backup scope first. Then click **Browse** and select a backup directory.



Always create a backup when you update or change an installation. This makes it easier to compare and merge the files later.

7. Click *Next* and update components, if required.**8. To start installation, click *Install*.**

During installation, a progress dialog is shown with detailed information on each MSI currently installed. If there are installation issues, the process stops and you can examine the error in the lower dialog section. If required, click **Save to log** and send the log file to support@xplm.com for further investigation. This new progress dialog is available from installer versions 24.3.3.606 onwards.

For older installers, a common CMD window with the progress is shown. Do not click into this window or installation cannot proceed. To continue if you clicked in, press **Enter** or **Esc**.

9. To close the wizard after installation, click *Finish*.**10. Carefully compare and merge the changes from the backup directory with the newly installed files.**

Result

The installation is updated. Start the product and verify everything works as expected.

Related links

[Working with overlay packages](#) (p. 49)

7 Uninstallation

7.1 Removing installation

Complete these steps to remove an installation.

About this task

To uninstall an installation, you need the setup and component MSIs. If you installed correctly, they are located in the directory `C:\ProgramData\XPLM Solution GmbH\packages`.



The Windows uninstall feature is not supported. However, you can start the MSIs from Windows and use the installer's uninstall function.



You can install, modify, repair, or uninstall also in silent-mode. See [Silent-mode installation](#) (p. 45) for more information.

Procedure

1. Close all open applications related to the integration.
2. Start the current installer `Setup-*.exe` with administrator rights. If this file is no longer available, start `Setup-*.msi` directly from the directory `C:\ProgramData\XPLM Solution GmbH\packages`. Alternatively, you can also start the installer from Windows:
 - a) In the system settings, go to *Apps & Features* and search for the entry **PTC Creo Parametric Setup**.
 - b) Select the entry and click **Modify**.
→ Visual C++ runtimes are checked/installed again and the installation wizard appears.
3. Click **Next** to start the wizard.
→ The step *Modify, repair or remove installation* appears.
4. Click **Remove**.
→ The step *Remove of the installation* appears.
5. Click **Remove** to remove the installation.
6. To close the wizard after installation, click **Finish**.
7. Check the directory `<SWPDM INSTALL DIR>\CAD Integration` for leftover directories and files, and delete them manually.

Result

The installation is removed.

8 Troubleshooting

This chapter describes instructions for troubleshooting integration errors.

8.1 Common troubleshooting procedure

In case of integration problems, do the following.

1. Close related programs.
2. Enable logging.
3. Restart the integration and reproduce the problem.
4. Send an email with a problem description and log files to support.

8.2 License errors

See the following table with license errors that can occur and how to solve them.

Reason	Description	Solution
No license file found	No license file was found in the directory <code><SWPDM_INSTALL DIR>\CAD Integration\xml</code> or the alternative directory defined in environment variable <code>XPlmLicenseDirectory</code> .	Put license file in this directory. If no license file is available, get a valid license file from Dassault.
Cannot read file	There is not enough memory to read the license file.	Upgrade RAM on the computer where the integration is installed.
Unable to read license	The license file cannot be read/decrypted.	Get a valid license file from Dassault. Do not edit a license file.
Unable to retrieve license data	The content of the decrypted data does not correspond to an Dassault license content.	Get a valid license file from Dassault.
No domain data found	No domain entry was found in the license file.	Get a valid license file from Dassault with all applicable domain names. Domain names are case sensitive.
License is not valid for current domain <code><DOMAIN_NAME></code>	The customer domain does not exist in the license file.	
No MAC address found	No MAC address entry was found in the license file.	Get a valid license file from Dassault with all applicable MAC addresses. MAC addresses are case sensitive.
This system is not allowed	The customer MAC address does not exist in the license file.	
There is no license for <code><LICENSE_NAME></code>	The requested license is not part of the customer's license subscription.	Get a valid license file from Dassault with all integration components that are subscribed. License names are case sensitive.

Reason	Description	Solution
Unable to retrieve license information for <LICENSE_NAME>	There is no information about when the license expires.	Get a valid license file from Dassault.
License for <LICENSE_NAME> is expired on <YEAR>-<MONTH>-<DAY>	License is expired and no longer valid.	Resubscribe and get a valid license file from Dassault.

9 References

9.1 Tools and add-ons

This chapter provides a basic understanding of the tools that come with the installer or are available separately as add-ons.

- [Integration Creator](#) (p. 44)

9.1.1 Integration Creator

9.1.1.1 Introduction

Use Integration Creator to digitally sign connector files for Creo Parametric that were delivered in an unsigned state. If the signing is not carried out, the product will not work.

Scope

You must use Integration Creator for signing the following files:

- `XPlmCreoParametricConnector.dll`
- `XPlmCreoParametricConnectorAsync.exe`

Tasks



Make sure that all system requirements for the build machine are met, please review [System requirements](#) (p. 9) again.

Complete tasks in the following chapters:

- [Signing connector components](#) (p. 44)
- [Distributing signed connector components](#) (p. 45)

9.1.1.2 Signing connector components

To sign the files with Integration Creator, complete the following steps.

Procedure

1. On the build machine, install Creo Parametric.
2. Copy the installer archive `*.7z.exe` to the build machine.
3. Extract the archive and start `Setup-*.exe` with administrator rights.
4. Select components as required, and select in step *Tools* the option **Integration Creator**.
5. Complete installation.
→ After installation, a shortcut to Integration Creator is placed on the desktop.
6. Start the shortcut Integration Creator with administrator rights.
→ A single window with the button **Create** appears.
7. Click the button **Create**.
8. Enter the Creo Parametric license server `<server>@<port>` and click **OK**.
→ The files are signed and replaced in directory `<SWPDM INSTALL DIR>\CAD Integration\bin\<x64>\CreoParametric<version>`.
9. After signing, a success message appears. In this step, you can add the signed files to a new overlay and use it for deployment.
 - Click **Yes** to select a directory where the overlay with the files should be created (recommended).
 - Click **No** to skip this step without creating an overlay.
10. Restart the machine and check that the integration works. See User Guide for more information.

Result

The integration components are signed and added to the overlay. The integration works.

9.1.1.3 Distributing signed connector components

Distribute the overlay with the signed files to the client computers running Creo Parametric.

For only a few installations, you can use the installer or manually copy the files over an existing installation. If you want to distribute the overlay on a larger scale, install the integration in silent-mode.

- To install the integration and apply the overlay using the installer, start the installer. In the step *Installation path*, enable the option **Apply custom files after installation** and provide the path to the overlay.
- To apply the overlay manually over an existing installation, copy the overlay content to the installation directory `<SWPDM INSTALL DIR>\CAD Integration`, overwriting existing files.
- To install the integration and apply the overlay in silent-mode, create a batch script for silent-mode installation calling the individual MSIs and the overlay, and run it on the client computers.

Related links

[Installing integration](#) (p. 10)

[Working with overlay packages](#) (p. 49)

[Silent-mode installation](#) (p. 45)

9.2 Silent-mode installation

You can also install this Dassault product in silent-mode. Silent-mode has the advantage that you can easily install the product from a batch file without showing the installer GUI. Alternatively, you can start the installer with preset options, allowing it to be installed in a controlled manner by the user or by other automated installation routines.

The installer packages are all of type Windows Installer (MSI) and require corresponding parameters for silent-mode installation.

The Visual C++ runtimes are normal executables. Always install all x64/x86 runtimes that come with the installer package.

Understanding installer structure

When you start an installation using the installer, required files are copied first to the directory `C:\ProgramData\XPLM Solution GmbH` and are executed from this location.

```
XPLM Solution GmbH
├──cmd
├──log
└──packages
```

- `cmd`: Contains the batch files `Setup-*_admin.bat` and `Setup-*_user.bat`.
 - The file `Setup-*_user.bat` contains the copy commands for the required MSIs from the original location to the directory `C:\ProgramData\XPLM Solution GmbH\packages`.
 - The file `Setup-*_admin.bat` installs the individual MSIs from this new location with the parameters as defined in the installer.

- `log`: Contains log files of each installed component.
- `packages`: Contains copies of all MSIs used for installation, modification or uninstallation.



Use the definitions in the files `Setup-*_user.bat` and `Setup-*_admin.bat` as the basis for a silent-mode installation. The command line calls already contain the required component MSIs and parameters as selected in the installer.

General command line calls

Installing Visual C++ runtimes:

```
vcredist_*.exe /quiet
```

Uninstalling Visual C++ runtimes:

```
vcredist_*.exe /quiet /uninstall
```

Installing or modifying MSIs:

```
msiexec /i <fileName>.msi /quiet <parameter>=<value>
```

Repairing MSIs:

```
msiexec /i <fileName>.msi /quiet INSTALLMODE=Restore
```

Uninstalling MSIs:

```
msiexec /x <fileName>.msi /quiet REMOVE_SECURE=1
```

Using preset options in the installer (`Setup-*.exe`):

```
Setup-*.exe <parameter>=<value>
```

Using preset options in the installer (`Setup-*.msi`):

```
msiexec /i Setup-*.msi <parameter>=<value>
```

Creating a batch file for silent-mode installation

This example is intended as a general guideline for creating an installation script in silent mode. It assumes that the MSIs used for the installation are stored on a network share.

1. On a test computer, extract the main archive and start the file `Setup-*.exe` with administrator rights.
2. Select required components and settings, and finish installation.
3. Copy the entire contents from the extracted archive to a network share, for example `\\myShare`.
4. Create a new batch file locally, for example `silent.bat`.
5. Add the installation commands for the C++ runtimes to this file, for example:

```
REM *** install c++ runtimes ***
\\myShare\vcredist_14.38.33130.0_x64\vcredist_14.38.33130.0_x64.exe /quiet
\\myShare\vcredist_14.38.33130.0_x86\vcredist_14.38.33130.0_x86.exe /quiet
```

6. Go to the directory `C:\ProgramData\XPLM Solution GmbH\cmd`.

7. Open the file `Setup-*_user.bat` and copy all *robocopy* commands into your batch file. Change the path in the first argument (the source) to point to the network share, for example:

```
REM *** copy from network share to client ***
robocopy "\\myShare\packages" "C:\ProgramData\XPLM Solution GmbH\packages" ←
Core_23.0.0.538.msi
...
```

8. Open the file `Setup-*_admin.bat` and copy the command line calls for installing the MSIs into your batch file. Change the path of the MSI to point to the network share, for example:

```
REM *** installing msi ***
msiexec /i "\\myShare\packages\Core_*.msi" /passive ←
CALLED_BY=Setup-* ←
INSTALLDIR="C:\Program Files\XPLM Solution GmbH\" ←
BATCH_ADMIN="C:\ProgramData\XPLM Solution GmbH\cmd\Setup-*_admin.bat" ←
GUI_LOG_FILE="C:\ProgramData\XPLM Solution GmbH\log\Setup-*_gui.log" ←
JAVA_JNI=0 ←
JAVA_JNI_X86="" ←
JAVA_JNI_X64="" ←
...
```



In the above example, line breaks were inserted to show readable content. Usually, each `msiexec` call would be on one line. You can further clean-up each call by deleting the information marked red, as it is not required in your batch file.

9. To test silent installation, use a clean client computer, copy the batch file `silent.bat` to it and run it with administrator rights.

Parameters usage

- If no parameters are defined, default settings apply. In the following tables, default settings are underlined.
- If you use parameters in the scope of `Setup_*.exe/msi`, use them with the provided prefix, for example `COR_JAVA_JNI`.



You cannot use the setup files `Setup_*.exe/msi` for silent-mode installation. For this you must use the individual component MSIs. However, you can use parameters in `Setup_*.exe/msi` to preset options when installing in GUI-mode.

- Use parameters without prefix to define settings within the scope of component MSIs, for example `Core_*.msi`.
- Use the following parameters to control either GUI-mode or silent-mode installation:
 - ☐ `none` : GUI-mode
 - ☐ `/quiet` : Silent-mode without GUI
 - ☐ `/passive` : Silent-mode with additional progress indication

Parameter for silent-mode & GUI-mode

The following parameters apply to silent mode as well as to the presetting of options in GUI mode.

Table 6: Component MSI & Setup EXE/MSI

Prefix	Parameter	Value	Description and use
	INSTALLDIR	Path to a valid directory	Defines the path of the installation directory. Available in all MSIs.
	INSTALLMODE	<u>C</u> hange Restore	Defines the installation mode after the installation is already completed and the installer is restarted. <ul style="list-style-type: none"> ■ Change = Modify ■ Restore = Repair Available in all MSIs.
	REMOVE_SECURE	<u>0</u> 1	Enables uninstallation. This corresponds to Remove in the installer. Available in all MSIs.
PDM	VERSION	Version as shown in installer	Defines the SOLIDWORKS PDM version. Available in EnterprisePDM MSI and all Setup EXE/MSI using this component.
PRO_	VERSION	Version as shown in installer, more versions separated by semicolon	Defines the Creo Parametric version. Available in CreoParametric MSI and all Setup EXE/MSI using this component.

Parameter for GUI-mode

The following parameters apply exclusively to the presetting of options in GUI-mode and not to silent-mode.

Table 7: All MSIs

Parameter	Value	Description and use
BACKUP_FILES	Path to a valid directory	Defines a location for the backup.
BACKUP_TYPE	FULL CONFIG	Defines the backup scope. <ul style="list-style-type: none"> ■ FULL = full backup of <SWPDM INSTALL DIR>\CAD Integration. ■ CONFIG = backup of configuration directory only, for example <SWPDM INSTALL DIR>\CAD Integration\xml.
CUSTOM_FILES	Path to a valid directory	Defines the path to a directory containing an overlay package with custom files to be copied after installation.

Table 8: Setup EXE/MSI

Parameter	Value	Description and use
COR_ICR	0 1	Enables Integration Creator. Available in all Setup EXE/MSI using this component.

Table 9: Setup-PDMPProfessional-CreoParametric EXE/MSI

Parameter	Value	Description
PDM_PRO	0 1	Enables Creo Parametric.

9.3 Working with overlay packages


An overlay package contains custom files with modified configuration. In the installer, you can select an option to apply an overlay as the last step of the installation process, copying the custom files over the installed files.


Supported overlay features in the installer

First familiarize yourself with the features that are supported for overlays when using the installer.



The following features are available from installer versions 24.3.3.606 onwards. For older installers, the legacy behaviour applies. This behaviour has some limitations, see column **Legacy** for more information.

Feature	Description	Legacy
Directory or archive support	<p>An overlay can be referenced in form of an archive (zip, 7z or 7z.exe), or as a directory.</p> <p> When extracting an overlay archive, a double directory structure is often created, for example myOverlay\myOverlay\xml. If you select an overlay directory in the installer, always select the directory that corresponds to the level of the installation directory.</p>	Only directories and no archives can be referenced.
Local or network support	You can reference an overlay from the local disk or from a network share.	

Feature	Description	Legacy
Multiple installation directories	<p>Certain products require multiple installation directories. Usually core components are installed in the main installation directory, whereas other parts are installed in a different path.</p> <p>You can reference these individual paths in the overlay, so that the content is copied to the correct location when the overlay is applied.</p> <p>For this, add the installation parameter as first directory in the overlay structure, for example:</p> <ul style="list-style-type: none"> ■ <code>myOverlay\INSTALLDIR</code>: When applying the overlay, the directory <code>INSTALLDIR</code> is resolved to the main installation directory <code>%xPlmRootDir%</code>. ■ <code>myOverlay\<PREFIX>_PATH</code>: When applying the overlay, the directory <code><PREFIX>_PATH</code> is resolved to another installation directory. ■ <code>myOverlay\<PREFIX>_PATH</code>: When applying the overlay, the directory <code><PREFIX>_PATH</code> is resolved to another installation directory. <p>See Silent-mode installation (p. 45) for installation parameter.</p>	
Custom scripts	<p>When applying an overlay, you can also run a script with custom actions.</p> <p>For this, create the directory <code>script</code> in the overlay and save the script <code>custom.bat</code> in it.</p> <p>If the product requires multiple installation directories, add the installation parameter again as first directory in the overlay structure, for example:</p> <ul style="list-style-type: none"> ■ <code>myOverlay\INSTALLDIR\script\custom.bat</code> ■ <code>myOverlay\<PREFIX>_PATH\script\custom.bat</code> ■ <code>myOverlay\<PREFIX>_PATH\script\custom.bat</code> <p>When applying the overlay, the script is first copied to the directory <code>C:\ProgramData\XPLM Solution GmbH\custom_files\<DATE-TIME></code> for backup purposes. Then it is copied to the directory <code>script</code> in <code>%xPlmRootDir%</code> or the alternative installation directory and executed from there.</p> <p> In installers from version 24.4.4.628 onwards, you also have access to all environment variables in your script that were written during the installation.</p>	

Feature	Description	Legacy
Auto-detection	<p>Overlay archives or directories are automatically recognized if they are located next to or in the extracted setup directory and contain the string <i>overlay</i> in any written form in their name, for example:</p> <ul style="list-style-type: none"> ■ Example 1 (overlay as 7-Zip archive next to extracted setup) <pre><EXTRACTED INSTALLER DIR> myOverlay.7z</pre> ■ Example 2 (extracted overlay next to extracted setup) <pre><EXTRACTED INSTALLER DIR> myOverlay</pre> ■ Example 3 (overlay as ZIP archive in extracted setup) <pre>myOverlay.zip packages vcredist_*_x64 vcredist_*_x86 Setup-*.exe</pre> ■ Example 4 (extracted overlay in extracted setup) <pre>myOverlay packages vcredist_*_x64 vcredist_*_x86 Setup-*.exe</pre> <p>If the overlay exists as shown in the above examples, and the installer is started for the first time, the option Apply custom files after installation is automatically enabled with the correct path.</p> <p>When the installer is restarted, it checks whether the applied overlay is still in its original location and automatically enables the option with the correct path. For example, put the overlay to a network share and it is always referenced from this location, see also best practices.</p>	<p>Only directories and no archives can be auto-detected. The overlay directory must be named <i>custom_files</i>.</p>

Best practices with overlays

- To create an overlay, copy modified files to a new directory with the string *overlay* in its name. Also create relevant sub-directories as in the original location from where you copied the files, for example *myOverlay\xml*.

- To apply an overlay in the installer, enable the option **Apply custom files after installation** and navigate to the overlay. Depending on the overlay type, make sure that you select either **Directories** or **Archive Files** in the selection dialog. If you have placed the overlay for auto-detection, the option is automatically enabled with the correct path.
- To apply an overlay manually, copy from the overlay directory the modified files to the directory `<SWPDM INSTALL DIR>\CAD Integration` and its related sub-directories, for example from `C:\temp\myOverlay\xml` to `<SWPDM INSTALL DIR>\CAD Integration\xml`.
- For deployment to clients, it is best to store the overlay on a network share. This allows you to update modified configuration on a regular basis and apply the overlay via your preferred method.

How to apply an overlay in silent-mode

There are no parameters for applying an overlay package in silent-mode, because the copying process is not triggered by an MSI installation. You can only use certain parameters to preselect options in the installer. But, you can use *Robocopy* commands to copy the overlay content from a network share directly to the installation directory.

1. Copy the overlay to a network share, for example `\\myShare\myOverlay`.
2. In your own batch file for silent-mode, add the copy operation to the installation directory as the last step, for example:

```
...
REM *** apply overlay ***
robocopy "\\myShare\myOverlay" "<SWPDM INSTALL DIR>\CAD Integration"
```

3. Test your batch file on a clean client computer.
4. Check that the overlay was applied correctly and that the product is working.