EDA – Gateway – Server and Client - Components

Installation Document – June 2022



Document Details

|  |  |
| --- | --- |
| Installation Instruction Document | |
| Project Owner |  |
| Process Title |  |
| Document Author |  |
| Document Version | **1.0 (DRAFT)** |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Document Edit History | | | | |
| Version | **Description** | **Name** | **Role** | **Date** |
| V1.0 | Initial draft | **Sanjeev Beemidi** | **EDA Integrator** | **9 Jun 2022** |

Contents

[1. Introduction 4](#_Toc106084852)

[1.1. Purpose 4](#_Toc106084853)

[1.2. Scope 4](#_Toc106084854)

[1.3. Pre-Installation Requirements 4](#_Toc106084855)

[1.4. Pre-Installation Checklist 5](#_Toc106084856)

[1.5. Overview of EDA Gateway and Deployment Options 5](#_Toc106084857)

[1.6. Installing EDA – Server Components 11](#_Toc106084858)

[1.7. Installing – EDA – Client Components 38](#_Toc106084859)

[1.8. Installing – Allegro Lib Manager 55](#_Toc106084860)

[1.9. Installing – Cadence –Allegro Components 59](#_Toc106084861)

[1.10. Installing – Cadence –Orcad Capture CIS Components 75](#_Toc106084862)

[1.11. Installing Cadence SPB Software 79](#_Toc106084863)

[1.12. Post-Installation Checklist 80](#_Toc106084864)

[1.13. Troubleshooting 80](#_Toc106084865)

[1.14. Reference Documents 81](#_Toc106084866)

1. **Introduction**

## Purpose

This document describes the entire end-to-end installation document of Siemens ADA gateway client on the server components and the cadence Allegro and CAS software components

It is highlighted that the contents that are described in this document are not Moog’s proprietary however there are extracts from Siemens and Cadence installation instructions that are included in this document

The reader is highlighted and recommended to go through the Siemens under Cadence documents that are referred in this Moog’s installation document for more details in case if there are any questions or clarifications needed

There are prerequisites to install the EDI gateway client under server components they are Teamcenter Oracle Cadence on the ED gateway client on the software installation media.

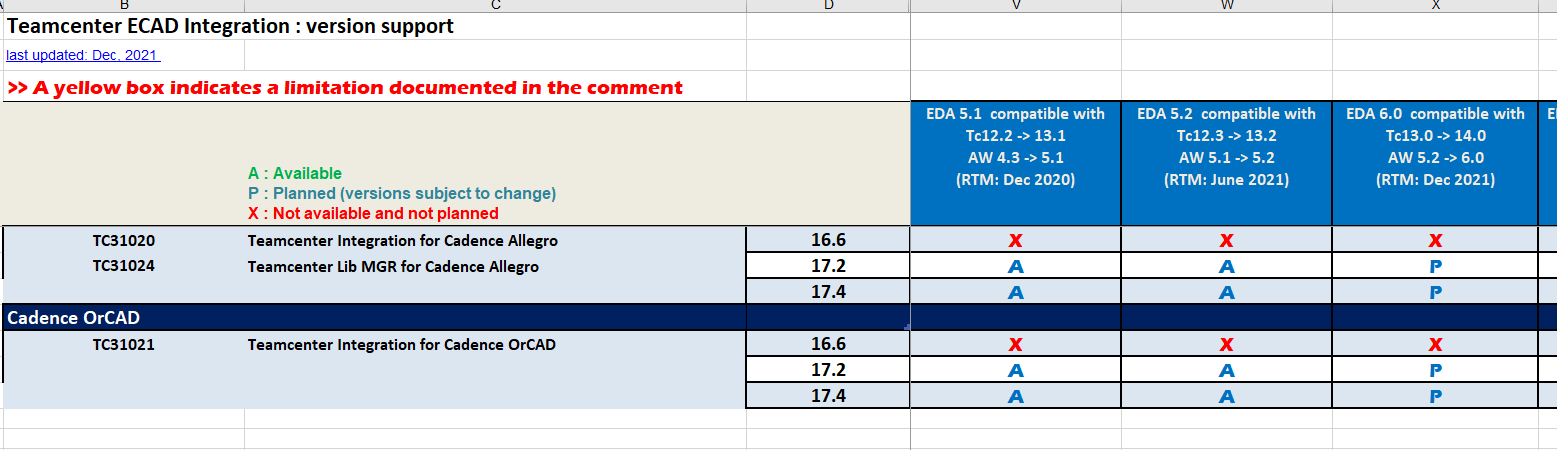
## Scope

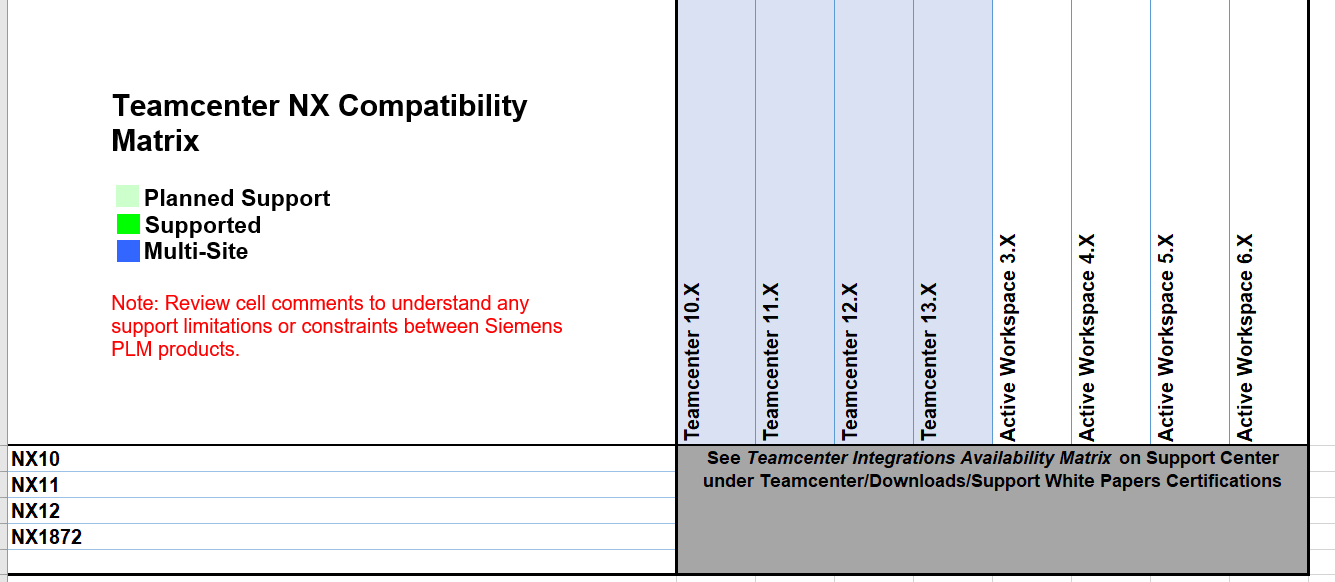
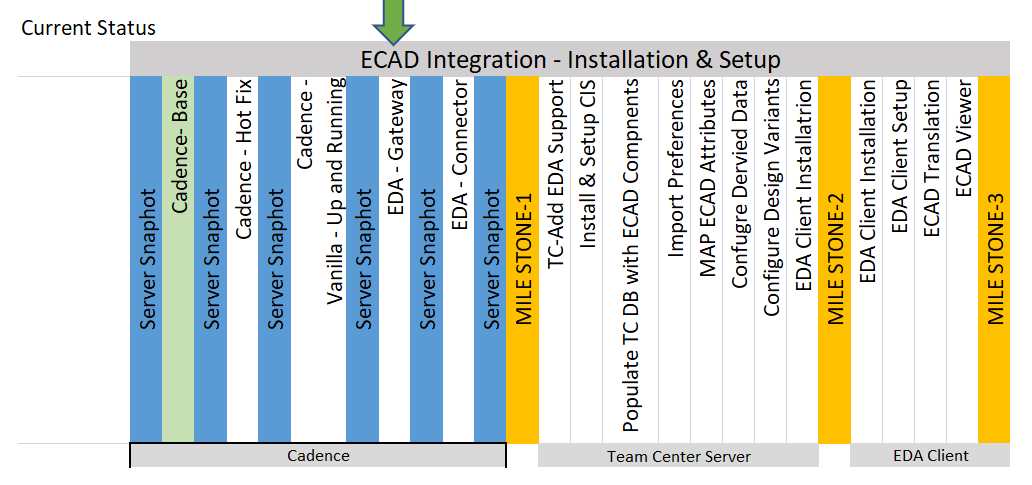
|  |
| --- |
| Description |
| Installation of the Siemens EDA Gateway Client and Server Components |
| Cadence Software Components |
| The Gateway components for CIS and Allegro Tools |

Apart from the above activities, any other activity is out of scope for this work.

## Pre-Installation Requirements

The following prerequisites and requirements must be satisfied in order for the to install successfully.



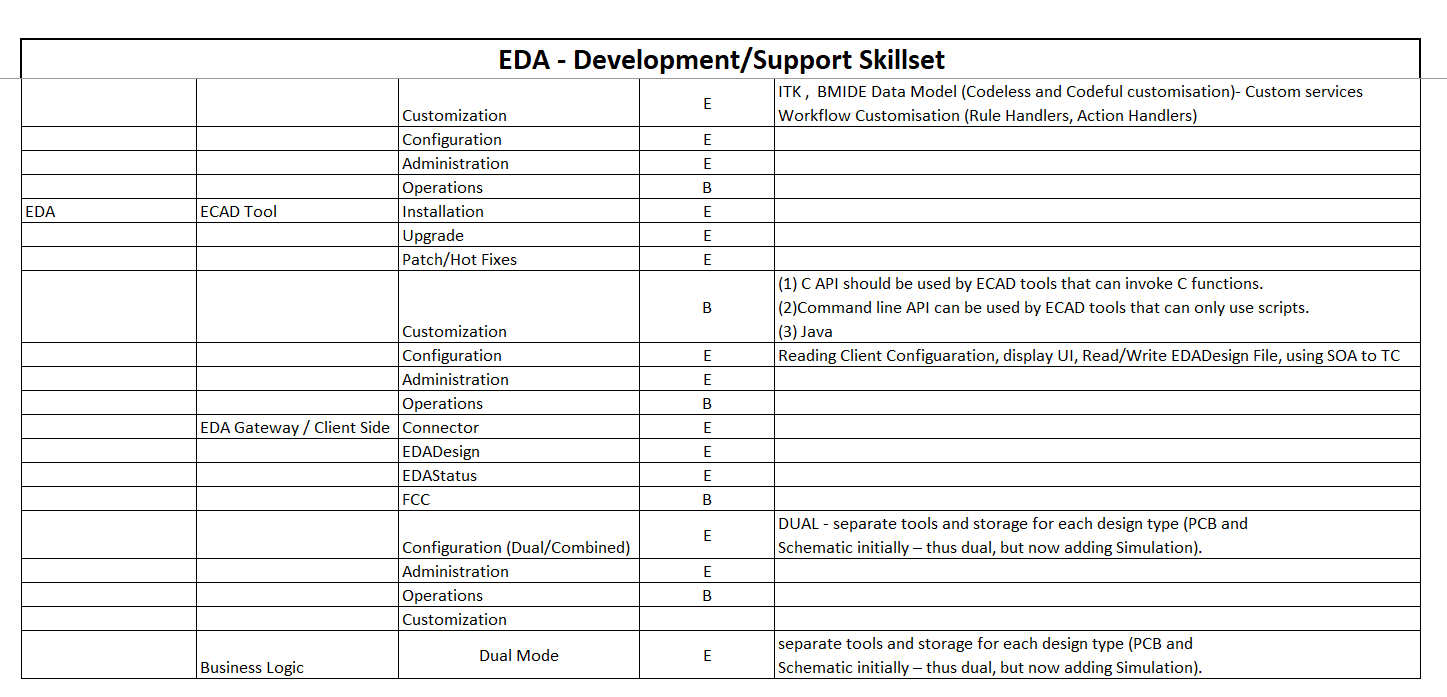


## Pre-Installation Checklist

The following prerequisites and requirements must be satisfied in order for the to install successfully.

Use the following checklist to ensure that your servers are ready for the components to be installed.

# Checklist Item



## Overview of EDA Gateway and Deployment Options

#### Teamcenter EDA overview

Teamcenter Electronic Design Automation (EDA) integrates Electronic Computer Aided Design (ECAD) applications with Teamcenter to allow designers to move ECAD data into and out of Teamcenter.

The integrations enable ECAD designers to open and save native design files, access approved parts, generate visualization files, share fabrication and assembly data, create bill of materials (BOMs) containing both mechanical and electrical parts, and collaborate with other domains and suppliers.

Teamcenter allows two types of EDA integrations:

#### ECAD design management

An ECAD designer can use either the Teamcenter Gateway for EDA or the integrations that embed Teamcenter menus into the ECAD design tool.

Teamcenter Gateway for EDA is used for integrations with ECAD tools where embedded EDA integration is not available.

ECAD design management allows the ECAD designer to automatically log on to Teamcenter, open, save, request parts, and check in and check out design data. The ECAD tool data is stored as its native design archive. The designer can also store derived data extracted from the ECAD tools, such as fabrication and assembly data, as well as Teamcenter generated ECAD/MCAD interchange, visualization, and BOM files.

The ECAD design integration allows the ECAD designer to automatically log on to Teamcenter from the

ECAD tool and perform the following tasks:

• Store ECAD tool data as its native design archive

• Open design files

• Check in and check out design data

• Generate visualization files

• Share fabrication and assembly data

• Create bills-of-materials (BOMs) containing both mechanical and electrical parts

• Collaborate with other domains and suppliers

#### ECAD part library management

Teamcenter EDA Library management enables ECAD designers and librarians to capture, track, and manage all the parts, symbols, footprints, padstacks and attributes in the part libraries, as well as the relationships between these objects. To prevent design teams from using unapproved, obsolete or out-of-date parts, and to ensure that accurate and consistent information is available throughout your organization, Teamcenter library data can be synchronized with each individual ECAD tool’s local library. During the synchronization process, Teamcenter automatically identifies any new or updated parts that must be exported.

Both these integrations are separate functionalities and run separately. You can install them either together or separately depending on your requirement.

The ECAD part library integration enables ECAD designers and librarians to:

•Capture, track, and manage all the parts, symbols, footprints, padstacks, and attributes in the part libraries.

•Manage relationships between parts, symbols, footprints, padstacks, and attributes.

•Provide designers access to approved parts.

•Prevent design teams from using unapproved, obsolete, or outdated parts.

•Ensure that accurate and consistent part information is available throughout the organization.

•Synchronize Teamcenter library data with each individual ECAD tool’s local library.

#### . Teamcenter EDA components

A Teamcenter EDA installation consists of the following components:

#### Teamcenter server

You need a Teamcenter server installed with EDA support and with library support (for part library management). This system is used to store and track the ECAD designs.

#### Dispatcher for ECAD translation (optional)

Dispatcher is required for ECAD translation if done remotely. With ECAD translation, the design intermediate file from the ECAD tool is converted into the neutral file format (XFATF for PCB designs and XSCH for schematic designs). The neutral file is saved as a viewable dataset in Teamcenter.

If Dispatcher Server is already installed in your existing Teamcenter environment, you can use the same instance. However, you must modify it to include the translators that are specific to ECAD translation.

#### ECAD design tool

See the Teamcenter Integration Availability Matrix for ECAD Integrations to identify which versions of ECAD design tools can be integrated with Teamcenter.

Supported ECAD design tools provide scripting languages (such as Ample and SKILL) to call the EDA client services.

#### EDA client

The client that moves designs between the ECAD design tool and Teamcenter. The EDA dialog boxes are opened from commands on the Teamcenter menu in the ECAD design tool and Teamcenter Gateway for EDA.

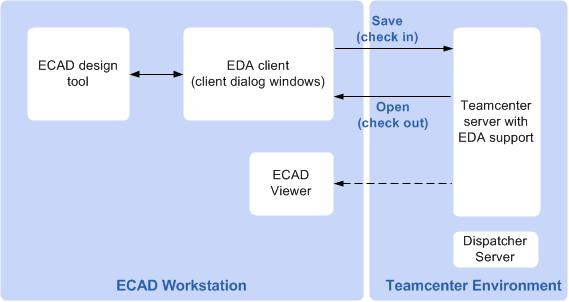
Eclipse provides the framework for interface dialog boxes and provides extension points for communication with the ECAD design tool. All designs are opened and saved in the staging directory, whereas the cache holds Teamcenter item data.

The communication from client services to Teamcenter is done using SOA services. Teamcenter can use a two-tier or four-tier architecture to send design data to and from the database.

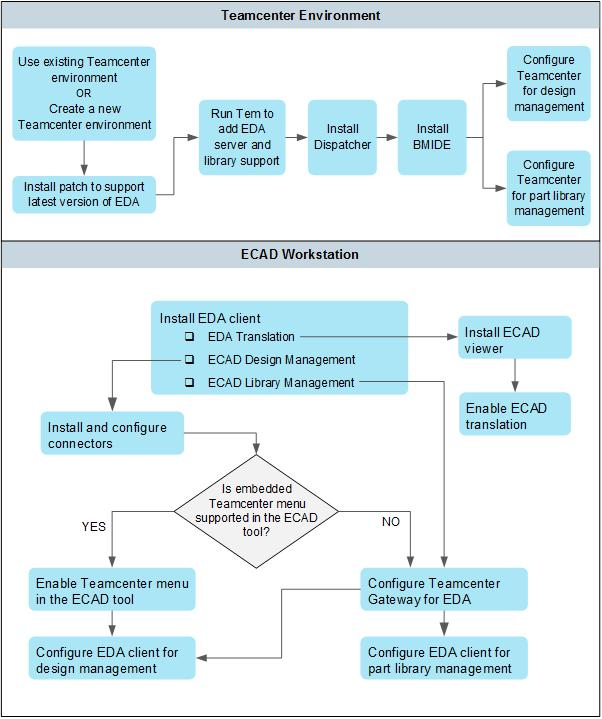
#### ECAD viewer (optional)

A design view and a markup tool.

The components of Teamcenter EDA and their interactions are as follows.



#### . Workflow to deploy Teamcenter EDA

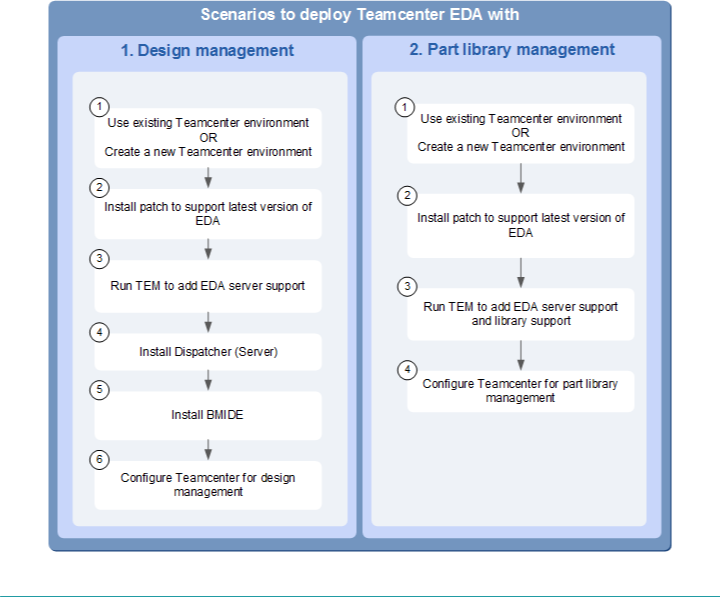


#### Teamcenter EDA deployment scenarios

#### EDA Server Support adds the EDA data model to the Teamcenter data model. ECAD Library Management adds the EDA part library data model to the existing Teamcenter data model.

#### As the EDA client is supported in a four-tier architecture, the EDA Server Support feature (data model) must also be installed on this Teamcenter server.

#### You can deploy your Teamcenter EDA solution to support design management or part library management or both.



## Installing EDA – Server Components

#### Add EDA support on the Teamcenter server for design management

EDA server support adds the EDA data model to the existing Teamcenter data model. Before installing the EDA server support, you must install the Teamcenter server.

1. From your Teamcenter environment, start Teamcenter Environment Manager (TEM).
2. Select the **Configuration Manager** option to perform maintenance on an existing installation and click **Next** until the **Select Features** dialog box appears.
3. In the **Select Features** dialog box:
   1. Select the following options:
      * Choose **Extensions**→**Mechatronics Process Management** and select **EDA Server Support**.

* Choose **Extensions**→**Mechatronics Process Management** and select **EMPS-Foundation**.
* Specify **Active Workspace** server extensions features for EDA. These are available in the **Features** panel in **Teamcenter Environment Manager (TEM),** under **Base Install→Active Workspace→Server Extensions→EDA Server Support** for Active Workspace.
* Specify **Active Workspace** client extensions features for EDA. These are available in the **Features** panel in **Teamcenter Environment Manager (TEM),** under **Base Install→Active Workspace→Client→Electronic Design Automation** for Active Workspace.
  1. In the **Installation Directory** box, type the location where you want to install Teamcenter (TC\_ROOT).
  2. Click **Next**.

1. Enter information as needed in the subsequent panes.
2. In the **Confirm Selections** dialog box, click **Next**. The EDA server components are installed.

The **Add/Remove Components** dialog box confirms if the installation is successful. If the installation is unsuccessful, click **Show Details** and proceed as needed.

1. Obtain the license file that includes EDA licensing and EDA library license (for library support) and install it on the server.

After adding the EDA server support to your Teamcenter environment, you must verify your installation by checking whether the EDA data types are added to Teamcenter.

#### Verify your installation

Verify the addition of the EDA data types to the server:

1. Run the Teamcenter rich client.
2. In My Teamcenter, choose **File**→**New**→**Item**. The **New Item** dialog box is displayed.
3. Verify that the EDA, EDAComp, EDASchem, and EDACCABase item types are added to the list of 8 types you can create.
4. Select one of the EDA item types and create an instance.
5. Verify that the item instance is created.

#### Install and set up Common Integration Services

Teamcenter EDA uses *Common Integration Services* (*CIS*) to manage ECAD application data in Teamcenter.

The CIS framework uses an *Integration Definition file* to define the default Teamcenter EDA objects that are created during the save-as, save, check-in, and revise operations. These EDA objects are PCA, schematic, PWB, variants, derived items, derived datasets, design dataset, viewable datasets, and BOM components. This file is imported to Teamcenter when CIS services are deployed on the Teamcenter server from the EDA install kit.

The CIS services require an *integration definition file* to be imported to the Teamcenter server. This file is used to define the relation between these EDA objects whether they are related with each other using *Generic Relationship Management* (GRM) relationship or *BOM View Revision* BVR relationship. The relationship behavior is defined OOTB for all the EDA objects with the default Teamcenter object types.

To install and set up common integration services on the Teamcenter server, you must:

1. Update the Teamcenter installer to include the CIS feature.
2. Install Common Integration Services using TEM.

#### Update the Teamcenter installer to include Common Integration Services

1. Copy the EDA kit to a temporary location.
2. Run Teamcenter Environment Manager (TEM) from the *%TC\_ROOT%\install* folder on the Teamcenter server as an administrator.
3. Select **Updates Manager**, click **Next**.
4. For the **Update kit location** field, browse to the *<EDA\_KIT>\server\_updates\cif\<PLATFORM>* folder. Click **Next**.
5. Proceed through the remaining panels in TEM, entering the required information for the features you selected.
6. When TEM displays the **Confirmation** panel, click **Start** to begin the installation.
7. When the installation is complete, close TEM.

#### Install the Common Integration Services

1. Run Teamcenter Environment Manager (TEM) from the *%TC\_ROOT%\install* folder on the Teamcenter server as an administrator.
2. In the **Maintenance** panel, choose **Configuration Manager**.
3. In the **Configuration Maintenance** panel, choose **Perform maintenance on an existing configuration**.
4. In the **Old Configuration** panel, select the configuration you want to modify.
5. In the **Feature Maintenance** panel, select **Add/Remove Features**.
6. In the **Features** panel, select the following features:

##### Extensions→Model Management→Server→Common Integration Framework

1. Proceed through the remaining panels in TEM, entering the required information for the features you select.
2. When TEM displays the **Confirmation** panel, click **Start** to begin the installation.

When the installation is complete, close TEM.

#### Populate the Teamcenter database with ECAD components

Many companies control the ECAD component library items that can be used to create BOMs by checking in the approved components for use to the database. Thereafter, you can create BOMs that use these components.

You can configure how to:

##### Control ECAD component library items for BOM creation.

To control component library items for BOM creation, when EDA is first installed, the **EDA\_CheckComponentExistence** is set to **1** (true). With this setting, you cannot save design BOMs unless all the necessary components are already in Teamcenter.

##### Populate the database with ECAD components before the designer can save designs and before creating BOMs in Teamcenter using EDA. You can initially populate the database with components by changing the **EDA\_CheckComponentExistence** preference to **0** (false). If this preference is set to **0**, any component that does not already exist in Teamcenter is automatically created when saving the BOM.

Caution:

Do not set this preference to **0** in a production environment.

##### To populate the database with ECAD components:

1. Set the **EDA\_CheckComponentExistence** preference:
   1. Run the rich client on the Teamcenter server and log on as an administrator (for example,

**infodba**).

* 1. In the My Teamcenter application, choose **Edit**→**Options**.
  2. At the bottom of the **Options** dialog box, click **Filters**.
  3. In the **Preferences** dialog box, search for the **EDA\_CheckComponentExistence** preference.
  4. Click **Edit** and change the value of the preference from **1** (true) to **0** (false).
  5. Click **Save** to apply the changes.

1. Locate or create designs that use the approved library components. Use the EDA **Save As** command to save the designs to the Teamcenter server and create BOMs. When the BOMs are created, the components are automatically added to the database.
2. Reset the **EDA\_CheckComponentExistence** preference to **1** (true) if you want to restrict users to using only the approved (and checked-in) components. This is the recommended setting in production environments.

However, if you wish to allow new components to be automatically placed in the database when users create BOMs, leave the **EDA\_CheckComponentExistence** preference as is, set to **0**.

This action bypasses business processes for component evaluation and approval.

#### Import new or updated EDA preferences in Teamcenter to support the latest EDA version

The EDA client requires the latest preferences to be imported in the Teamcenter environment for the client to work correctly. Before installing or updating the client, you must import preferences that are specified in an xml file included at the following EDA kit path:

*<EDA\_KIT>\server\_updates\edaserver\install\eda\_preferences.xml*

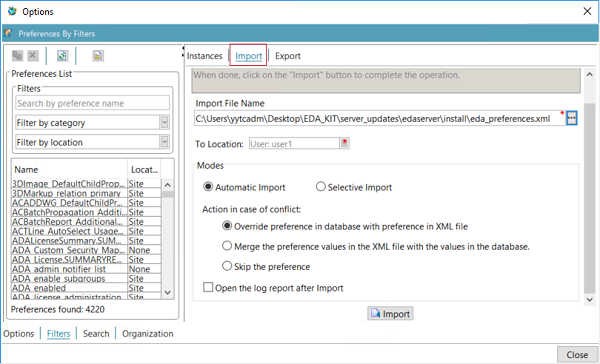
If the preferences are not updated since the Teamcenter base install, this file is not available in the EDA kit.

You can import these preferences in Teamcenter from the rich client or using the Teamcenter command prompt.

* + - Importing preferences in Teamcenter from the rich client:

1. Log on to Teamcenter and start the rich client.
2. Choose **Edit**→**Options** to display the **Options** dialog box.
3. Click **Filters** to open the preferences view.

Click **Import**.



1. Browse to the EDA kit location where the preferences file exists and select the *eda\_preferences.xml*

file for import.

1. Select **Site** from the **To Location** list.
2. Choose **Automatic Import**.
3. If the preference exists in Teamcenter, you can select any one of the **Action in case of conflict**

options to override, merge, or skip that preference.

1. Click **Import**.

#### Importing preferences in Teamcenter from the command prompt:

1. Open a Teamcenter command prompt.
2. Run the following command:

##### preferences\_manager -u=<user id> -p=<password> -g=dba -mode=import -scope=SITE - action=SKIP -file=<EDA\_KIT>\server\_updates\edaserver\install\eda\_preferences.xml

#### Import new or updated Active Workspace preferences in Teamcenter to support latest EDA version

The EDA client requires the latest EDA and Active Workspace preferences to be imported in the Teamcenter environment for the client to work correctly. Before installing or updating the client, you must import preferences that are specified in an xml file included at the following EDA kit path:

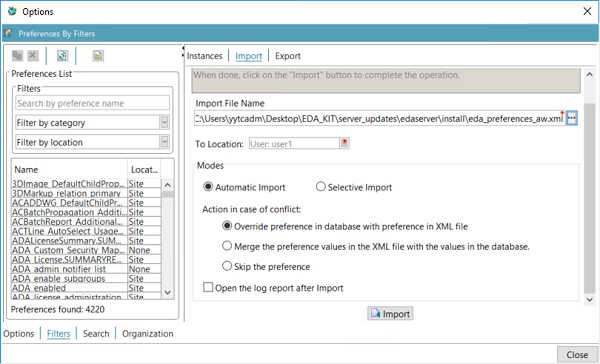
*<EDA\_KIT>\server\_updates\edaserver\install\eda\_preferences\_aw.xml*

If the Active Workspace preferences are not updated since the Teamcenter base install, this file is not available in the EDA kit.

You can import these preferences in Teamcenter from the rich client or using the Teamcenter command prompt.

#### Importing preferences in Teamcenter from the rich client:

1. Log on to Teamcenter and start the rich client.
2. Choose **Edit**→**Options** to display the **Options** dialog box.
3. Click **Filters** to open the preferences view.
4. Click **Import**.



1. Browse to the EDA kit location where the preferences file exists and select the

*eda\_preferences\_aw.xml* file for import.

1. Select **Site** from the **To Location** list.
2. Choose **Automatic Import**.
3. If the preference exists in Teamcenter, you can select any one of the **Action in case of conflict**

options to override, merge, or skip the preference.

1. Click **Import**.

#### Importing preferences in Teamcenter from the command prompt:

1. Open a Teamcenter command prompt.
2. Run the following command:

**preferences\_manager -u=<user id> -p=<password> -g=dba -mode=import -scope=SITE - action=SKIP -file=<EDA\_KIT>\server\_updates\edaserver\install\eda\_preferences\_aw.xml**

#### Modify the Active Workspace style sheet to include EDA derived data relation

The *commercial-off-the-shelf (COTS)* value of the EDA preferences used by Active Workspace to display the EDA item revision types is defined in the *Awp0ItemRevisionSummary* style sheet.

To allow EDA derived datasets to be displayed as ItemRevision attachments in Active Workspace, you must modify the *Awp0ItemRevisionSummary* style sheet on the Teamcenter server.

1. Log on to Teamcenter and start the rich client.
2. In the navigation pane, click **My Teamcenter**.
   1. Search for the *Awp0ItemRevisionSummary* dataset.
   2. Double-click the dataset.

The dataset is automatically checked out and the xml file is launched with the dataset.

1. In the *Awp0ItemRevisionSummary* style sheet, modify the **XRT\_files** page by appending the **EDAHasDerivedDataset** relation to the attachments page objectSet source attribute value as follows. The modified line is marked with a plus (+) sign on the left most column.

<page titleKey="attachments">

<section titleKey="tc\_xrt\_Files">

+. <objectSet

source="IMAN\_specification.Dataset,IMAN\_reference.Dataset,

+. IMAN\_manifestation.Dataset,IMAN\_Rendering.Dataset

,TC\_Attaches.Dataset,

+. IMAN\_UG\_altrep.Dataset,IMAN\_UG\_scenario.Dataset,I

MAN\_Simulation.Dataset,

+. EDAHasDerivedDataset.Dataset"

defaultdisplay="listDisplay"

+. sortby="object\_string"

sortdirection="ascending"><tableDisplay>

<property name="object\_string"/>

<property name="object\_type"/>

#### Map ECAD attributes with Teamcenter properties

#### How are ECAD attributes mapped with Teamcenter properties

Using mapping definition files, the ECAD part attributes can be stored in the Teamcenter database and displayed and modified both in the ECAD design tool and Teamcenter.

For the ECAD design tool, choose entries for design, project, and variant attributes such that they apply to the data types that represent the corresponding objects. Mapping definitions for variant objects must be associated with the specific object type that represents the variant BOM.

For details on how to map ECAD attributes, see the *Syntax for mapping attributes with Teamcenter Integration for NX* topic in the Application Administration documentation.

#### Customize ECAD attribute mapping

You can customize the ECAD attribute mappings in Teamcenter by exporting the attribute mappings to a file, editing the file, and importing the file back into Teamcenter.

1. Open a Teamcenter command prompt.
2. Change to the *bin* directory within TC\_ROOT. (TC\_ROOT is the installed location of Teamcenter.)
3. Run the **export\_attr\_mappings** command to export the mappings to a file, for example:

##### export\_attr\_mappings —file=test\_attribute\_mapping —u=infodba —p=infodba

1. Edit the mappings file to add EDA tool attributes.
2. Import the edited mapping file back into Teamcenter by running the **import\_attr\_mappings**

command, for example:

##### import\_attr\_mappings —file=test\_attribute\_mapping —u=infodba —p=infodba

The attribute mapping import and export process uses the PLM XML import/export mechanism. For more information, see the *PLM XML/TC XML Export Import Administration* documentation.

#### Sample ECAD design attribute mapping definitions

The below example defines dataset attribute mappings and variant attribute mappings. In this mapping definition, the Item level attribute mapping definition defines the ECAD variant attribute mappings and it is part of the dataset level attribute mapping definition.

ECAD variant mapping for Altium tool:

{ Dataset type="EDADesAltiumBrd"

ORGNAME : Item.GRM(IMAN\_master\_form,Item Master).user\_data\_1 ORGADDR1 : Item.GRM(IMAN\_master\_form,Item Master).user\_data\_2 "DES\_NAME" : ItemRevision.object\_name

"DES\_DESC" : ItemRevision.object\_desc

Doc : Item.GRM(IMAN\_master\_form,Item Master).user\_data\_3

{ Item type="Item"

"variant\_def\_list\_tc\_description" : ItemRevision.object\_desc / description="Variant Item Revision Description"

}

}

#### Configure derived data for ECAD design objects

#### Understanding how derived data works

Derived data contains information that is derived from an ECAD design and comprises derived items and datasets. Derived items represent parts, subassemblies, and tools. Derived datasets manage data files created by ECAD applications.

You can configure how derived data is created in Teamcenter EDA by using the *EDA Derived Data* folder in Business Modeler IDE, creating an EDA derived data configuration, and configuring how derived data files are named in Teamcenter. For example, a configuration can specify that when a schematic design is saved in Teamcenter EDA, a schematic drawing can be automatically generated from the schematic design and saved along with the schematic item.

EDA business objects define the different types of derived data you can generate. To locate EDA business objects, use the **Find** button in the BMIDE view to search for all business objects containing the **EDA** string.

#### The following item types are children of the EDA business object:

|  |  |
| --- | --- |
| **Item Types** | **Description** |
| **EDACCABase** | Represents the common electrical CAD (ECAD) design data that is shared between variant circuit card assemblies (CCAs). It is used only for multiple CCA representations. |
| **EDAComp** | Represents electrical components contained in the CCA bill of materials (BOM). |
| **EDASchem** | Represents the electrical schematic item. |

**The following relationships are children of the ImanRelation business object:**

|  |  |
| --- | --- |
| **ImanRelation** | **Description** |
| **EDAHasDerivedDataset** | Identifies the associated dataset as a derived dataset. |
| **EDAHasDerivedItem** | Identifies the associated item as a derived item. |

#### How to use the EDA derived data editor to configure derived data in Teamcenter

Use the EDA Derived Data editor in the BMIDE to work with derived data configurations used by the Teamcenter EDA application. Teamcenter EDA integrates Teamcenter with ECAD applications that are used to design electronic components, such as circuit boards.

1. To access the **EDA Derived Data** editor, open the **Extensions**→**EDA Derived Data** folder, right-click an EDA derived data object, and choose **Open**.
2. To add an item to the EDA derived data object, click the **Add** button next to the **Configure Items**

table.

1. To add a dataset to the EDA derived data object, click the **Add** button next to the **Configure Dataset** table.

#### How to create an EDA derived data configuration

You can configure how derived data is created in Teamcenter by using the **EDA Derived Data** folder in Business Modeler IDE. After creating a derived data configuration, you can specify the name of the configuration in the **EDA\_DerivedDataConfigDefault** preference.

1. In BMIDE, open the **Extensions** folder.
2. In the **Extensions** folder, right-click **EDA Derived Data** and choose **New EDA Derived Data**. The **New EDA Derived Data** wizard is displayed.

In the **EDA Derived Data** dialog box, enter the following information and click **Next**

|  |  |
| --- | --- |
| **Field** | **Description** |
| **Name** | Specify a name you want to assign to the new derived data configuration.  This is the name used in the  **EDA\_DerivedDataConfigDefault** preference. |
| **Description** | Type a description for the new configuration. |

The **EDA Derived Data Configuration** dialog box in the wizard is displayed.

1. In the **EDA Derived Data Configuration** dialog box, set up how all EDA item and dataset types are to be handled for all contexts.
   1. Click the **Add** button next to the **Configure Items** table.

The **Add/Edit EDA Derived Item Configuration** dialog box is displayed.

In the **Add/Edit EDA Derived Item Configuration** dialog box, configure the derived items to be generated. For example, create separate rows for contexts such as schematic, PCB, simulation, and so on, including variations based on the what the parent is, such as **Schematic**, **CCA**, and **CCAVariant**. In this way, you set up how derived data is generated for all combinations of items.

In the **Add/Edit EDA Derived Item Configuration** dialog box, enter the following information and click **Finish**:

|  |  |
| --- | --- |
| **Field** | **Description** |
| **Name** | Type the name that you want to assign to the derived item configuration.  This is the name displayed to the user on the Teamcenter EDA **Derived Item** dialog box during save operations. |
| **Context** | **Browse** to select a specific Teamcenter EDA application context from the contexts listed below. When users in Teamcenter EDA save derived data for the following specified data types, derived items are generated according to this configuration.   * all * pcb * pcb/simulation * schematic * schematic/pcb * schematic/simulation * simulation |
| **Prefix** | (Optional) Type a file name string to be attached to the beginning of the parent item ID to distinguish it as being generated by this configuration.  The resulting string, including the prefix and postfix, is used in the derived item user interface in Teamcenter EDA as the initial value for the **Derived Item ID** box and **Name** box. This can be overridden by the user. |
| **Postfix** | (Optional) Type a file name string to be attached to the end of the item ID to distinguish it as being generated by this configuration.  The resulting string, including the prefix and postfix, is used in the derived item user interface in Teamcenter EDA as the initial value for the **Derived Item ID** box and **Name** box. This can be overridden by the user. |
| **EDA Parent** | **Browse** to select the derived parent EDA business object to which the derived item is related. (Teamcenter EDA does not support attaching derived items under other derived items). |

|  |  |
| --- | --- |
| **Field** | **Description** |
|  | * **CCA** represents a circuit card assembly (CCA). * **CCABase** represents the common design data that is shared between variant circuit card assemblies (CCAs). It is used only for multiple CCA representations. * **CCAVariant** represents the variant design data for a circuit card assembly (CCA). This is the data that is used on top of the CCABase business object. * **PWB** represents a printed wire board (PWB). A PWB is the product of a schematic design and printed circuit board (PCB) layout design and holds all the printed wire board production data created by those designs. * **Schematic** represents the electrical schematic item. |
| **Relation** | **Browse** to select the relationship between the derived dataset and the parent item revision.  The **EDAHasDerivedDataset** business object and its children are displayed in the selection dialog box. |
| **Dataset Business Object** | **Browse** to select the parent dataset business object type in Teamcenter to represent the derived item, for example, PDF. |
| **Dataset Reference** | **Browse** to select the kind of file reference to use for the derived dataset.  If the derived data instance comprises more than one file, this field must either be specified as a *ZIPFILE* type or must be specified using a separate derived dataset configuration entry with the same derived data name. |
| **Pathname** | Type the path where the derived dataset is to be saved on the user's machine.  Path names are evaluated at run time and must be the fully qualified path of the dataset that is to be saved. Path names can be explicitly specified (for example, *D:\EDA*  *\Datasets\readme.txt*) or formed using the variables or file name filters. Derived datasets can contain multiple files. Path names are case sensitive, and the directory delimiters of / or \ are used interchangeably. |
| **Callback Name** | Type the EDA callback name to execute.  This name is used to identify the configured callback in the EDA configuration file to determine what script to execute. The script is responsible for creating or placing |
| **Field** | Description |
|  | the corresponding derived files to be uploaded as specified by the configured source path name. |

The derived item configuration is added to the **Configure Dataset** table.

* 1. Click **Finish**.

The derived data configuration is added under the **EDA Derived Data** folder

1. To save the changes to the data model, choose **BMIDE**→**Save Data Model**, or click the **Save Data Model** button  on the main toolbar.
2. Deploy your changes to the test server. Choose **BMIDE**→**Deploy Template** on the menu bar, or select the project and click the **Deploy Template** button  on the main toolbar.
3. In the rich client, set the **EDA\_DerivedDataConfigDefault** preference to point to the EDA derived data configuration you just created.
4. Choose **Edit**→**Options**, click the **Search** link at the bottom of the **Options** dialog box, locate the **EDA\_DerivedDataConfigDefault** preference, and change its value to the new configuration. There may be multiple configurations created in the Business Modeler IDE, but an administrator can point to only one of them through this preference.
5. After deployment, test your new configuration in Teamcenter EDA.

For example, in your ECAD design tool, choose **Teamcenter**→**Save Derived Data**. (You can also select the **Generate Derived Data** check box in the **Save As**, **Save**, or **Check In** dialog box.)

To verify that the derived data is generated, in Teamcenter, expand the item that contains the derived data (for example, a CCA item). The derived dataset entries appear as you expand the tree structure. To see the contents of the derived dataset, right-click the dataset and choose **Named References**. A dialog box appears that shows the files that are contained in the derived dataset.

#### Configure how derived data files are named in Teamcenter

Update the value of the **EDA\_UseDerivedDataSubDirs** preference based on how you want the derived data files to be named in Teamcenter. The values are as follows:

##### True

The derived dataset name is the same as the Derived Dataset Configuration Name.

##### False

The derived dataset name consists of the parent item ID and an underscore (character) followed by the Derived Dataset Configuration Name.

#### Configure design variants

#### What are design variants?

To design and develop a product with several *variations* of that product, where each variation has different options or capabilities, you create design variants. Creating design variants avoids the need to create a unique version of the design for each variation. A variant uses the same base design, but the PCB assembly is loaded with the set of components specified by the variation. A variation may then be nominated when generating the design’s manufacturing output (such as BOM, Pick and Place (P&P), and assembly drawings), which will in turn determine how the product is assembled.

When you are working with ECAD design files, you can create any number of variations of the same base design, where each component may be configured differently as follows:

##### Fitted

This is the default state of a component if the component is fitted. It does not have variations. When you create a new variant, all components default to the state *fitted*.

##### Not Fitted

If a component is set to *not fitted*, it still exists on the schematic and is transferred to the PCB, but it is removed from the appropriate output documentation, such as the BOM.

##### Fitted with Varied Parameters

A component can have variations of any of its parameters, as part of the variant definition process. Modifying the value of a parameter is a local variation, only affecting the output documentation. The original schematic and the component whose parameter is being varied are not modified in any way.

##### Alternate Part

It is also possible to select an entirely different component as an *alternate part*. Since the alternate part is a different component, only one component is presented on the compiled schematic sheet. The alternate part must also share the same set of pins placed in the same locations as the base part. This is an essential requirement to ensure that the connectivity remains valid when the design is compiled.

An ECAD design is saved as a non-variant design for the first time in Teamcenter. Later, the variants of the ECAD design are added or created in Teamcenter as required. There are two scenarios for how variants are added.

* In the first scenario, you save the ECAD design as a non-variant design in Teamcenter. When new variants of the designs are created in the ECAD tool, you update the non-variant design in Teamcenter with the variants.
* In the second scenario, you save the non-variant ECAD design as a variant design in Teamcenter by creating dummy variants. When the variant designs are created in the ECAD tool, you replace the dummy variants in Teamcenter with the actual variant designs.

##### 

##### You can [configure how to save a nonvariant ECAD design to Teamcenter as a variant design](#_bookmark22)

using the EDA variant object model.

#### Configure ECAD design variants in Teamcenter

You can save a non variant ECAD design to Teamcenter using the EDA variant object model. To configure the saving of new nonvariant ECAD designs in Teamcenter as variant designs:

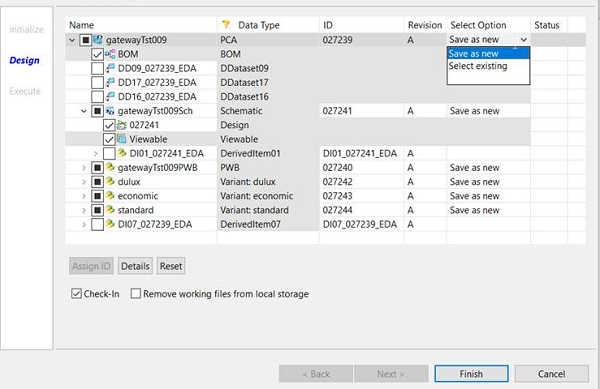
1. Update the value of the **EDA\_SaveAsForceVariant** preference to **true**. This value specifies that nonvariant ECAD designs not previously saved in Teamcenter are to be saved in Teamcenter, using the EDA variant object model.
2. Update the value of the **EDA\_FutureVariantName** preference. The default value is **futureVariant**. When you save a nonvariant design to Teamcenter and the value of the **EDA\_SaveAsForceVariant** is **true**, the value of the **EDA\_FutureVariantName** preference is used as the **Variant Name** in the **PCA Variants** dialog box. The value of the **Variant Name** in the **PCA Variants** dialog box cannot be changed.

Nonvariant ECAD designs already saved to Teamcenter using the EDA nonvariant object model are converted to use the EDA variant object model by the existing nonvariant-to-variant conversion functionality. This conversion occurs automatically when an ECAD design, previously saved to Teamcenter as a nonvariant design, is saved to Teamcenter after the first variant is added to the ECAD design.

#### Base BOMs do not support this functionality.

#### Manage design files in local storage

You can configure the default behavior of removing design files in the local storage option after you check in, check out, revise, or perform a save as operation on the design files. The **Remove working files from local storage** check box allows you to remove associated design files after you check in, check out, revise, or perform a save as operation on the design.



You can configure the default status of this check box by setting the following values in the

**EDA\_RemoveWorkingFilesOptionDefault** preference:

1. **unchecked**: The check box is cleared by default, and the user can select it.
2. **checked**: The check box is selected by default, and the user can clear it.
3. **Forceunchecked**: The check box is cleared, and the user cannot select it.
4. **Forcechecked**: The check box is selected, and the user cannot clear it.

#### Apply Teamcenter user's permission on the network shared ECAD design folder

During any Teamcenter EDA operation on a design in a network shared folder, the ECAD users accessing the design will have viewing or modifying rights on that shared folder based on the access permissions set for that user in Teamcenter.

To apply the Teamcenter user’s permissions on the ECAD design folder, the Teamcenter administrator must set the **EDA\_EnableFolderPermissions** preference value to **true**.

To specify which Teamcenter user roles must have permission to access the ECAD design folder, the user roles must be set in the **FolderPermissions\_ValidRolesForAccess** preference.

#### Allow ECAD designers to assign a Teamcenter project from the EDA client

While saving a new or an existing EDA design from the EDA client to Teamcenter, the ECAD designers can assign or remove a Teamcenter project for that design.

To make sure that the ECAD designer assigns a Teamcenter project while saving a design, you must set the **EDA\_TCProjectRequired** preference to **true**. If this preference is set to **false**, the ECAD designer can save the design without assigning any project. The default value of the preference is **false**. If a default project is assigned to a user in Teamcenter, that project is assigned to the design by default. However, the user can assign additional projects while saving the design.

You must also assign the ECAD users to a project in Teamcenter to allow the users to select that project during the save operation. All projects associated with the ECAD users are displayed for selection to that user during the save operation.

A Teamcenter propagation rule is used for propagating security-related property values, such as project and license assignments, from a source business object to a destination business object. The projects assigned to Printed Circuit Assembly (PCA) objects are propagated to all **ItemRevisions** and **Datasets** associated with the PCA object based on the Teamcenter propagation rule defined for that object. You can also modify the OOTB propagation rules as per your requirements. For more information on how to modify an existing propagation rule, refer to the Configure your business data model in BMIDE on Support Center.

The OOTB propagation rules defined for EDA objects are as follows:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Directio n** | **Source business object** | **Relation** | **Destination business object** | **Prop Group** | **Operation** | **Style** | **Details** |
| Forward | **EDACCABa**  **seRevision** | **IMAN\_Ren dering** | **Dataset** | Security Group I | All | Merge | Propagation rule between |
|  |  |  |  |  |  |  | PCA |
|  |  |  |  |  |  |  | **EDACCABas** |
|  |  |  |  |  |  |  | **eRevision** |
|  |  |  |  |  |  |  | and design |
|  |  |  |  |  |  |  | **Dataset** in a |
|  |  |  |  |  |  |  | variant |
|  |  |  |  |  |  |  | design. |
| Forward | **EDACCABa**  **seRevision** | **structure\_ revisions** | **PSBOMView Revision** | Security Group I | All | Merge | Propagation rule between |
|  |  |  |  |  |  |  | PCA |
|  |  |  |  |  |  |  | **EDACCABas** |
|  |  |  |  |  |  |  | **eRevision** |
|  |  |  |  |  |  |  | and |
|  |  |  |  |  |  |  | **PSBOMView** |
|  |  |  |  |  |  |  | **Revision** in a |
|  |  |  |  |  |  |  | variant |
|  |  |  |  |  |  |  | Design. |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Directio n** | **Source business object** | **Relation** | **Destination business object** | **Prop Group** | **Operation** | **Style** | **Details** |
| Forward | **ItemRevisi on** | **EDAHasSc hematic** | **EDASchem Revision** | Security Group I | All | Merge | Propagation rule between |
|  |  |  |  |  |  |  | PCA |
|  |  |  |  |  |  |  | **ItemRevisio** |
|  |  |  |  |  |  |  | **n** and |
|  |  |  |  |  |  |  | **EDASchem** |
|  |  |  |  |  |  |  | **Revision**. |
| Forward | **EDASche mRevision** | **IMAN\_Ren dering** | **Dataset** | Security Group I | All | Merge | Propagation rule between |
|  |  |  |  |  |  |  | **EDASchemR** |
|  |  |  |  |  |  |  | **evision** and |
|  |  |  |  |  |  |  | design |
|  |  |  |  |  |  |  | **Dataset**. |
| Forward | **ItemRevisi on** | **EDAHasP WB** | **ItemRevisio n** | Security Group I | All | Merge | Propagation rule between |
|  |  |  |  |  |  |  | PCA |
|  |  |  |  |  |  |  | **ItemRevisio** |
|  |  |  |  |  |  |  | **n** and |
|  |  |  |  |  |  |  | **Printed** |
|  |  |  |  |  |  |  | **Wiring** |
|  |  |  |  |  |  |  | **Board**. |
| Forward | **EDACCABa**  **seRevision** | **EDAHasVa riant** | **ItemRevisio n** | Security Group I | All | Merge | Propagation rule between |
|  |  |  |  |  |  |  | PCA |
|  |  |  |  |  |  |  | **EDACCABas** |
|  |  |  |  |  |  |  | **eRevision** |
|  |  |  |  |  |  |  | and variant |
|  |  |  |  |  |  |  | **ItemRevisio** |
|  |  |  |  |  |  |  | **n**. |
| Forward | **ItemRevisi on** | **EDAHasDe rivedItem** | **ItemRevisio n** | Security Group I | All | Merge | Propagation rule between |
|  |  |  |  |  |  |  | any |
|  |  |  |  |  |  |  | **ItemRevisio** |
|  |  |  |  |  |  |  | **n** and |
|  |  |  |  |  |  |  | derived |
|  |  |  |  |  |  |  | **ItemRevisio** |
|  |  |  |  |  |  |  | **n** in a |
|  |  |  |  |  |  |  | design. |
| Forward | **ItemRevisi on** | **EDAHasDe rivedData set** | **Dataset** | Security Group I | All | Merge | Propagation rule between any |
|  |  |  |  |  |  |  | **ItemRevisio** |
|  |  |  |  |  |  |  | **n** and |
| Directio n | Source business object | Relation | Destination business object | Prop Group | Operation | Style | **Details** |
|  |  |  |  |  |  |  | **derived Dataset in a design.** |

You can also modify the OOTB propagation rules to create custom rules for variant and non-variant designs.

#### Create a custom propagation rule for variant designs:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Directio n** | **Source business object** | **Relation** | **Destination business object** | **Prop Group** | **Operation** | **Style** | **Details** |
| Forward | **ItemRevisi** | **structure\_** | **PSBOMView** | Security | All | Merge | Propagation |
|  | **on** for the | **revisions** | **Revision** | Group I |  |  | rule between |
|  | variant |  |  |  |  |  | the variant |
|  | item type |  |  |  |  |  | **ItemRevisio** |
|  | defined in |  |  |  |  |  | **n** and |
|  | the |  |  |  |  |  | **PSBOMView** |
|  | **EDA\_CCAV** |  |  |  |  |  | **Revision**. |
|  | **ariantItem** |  |  |  |  |  |  |
|  | **TypeDefa** |  |  |  |  |  |  |
|  | **ult** |  |  |  |  |  |  |
|  | preference |  |  |  |  |  |  |

**Create a custom propagation rule for non-variant designs:**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Direction** | **Source business object** | **Relation** | **Destinatio n business object** | **Prop Group** | **Operation** | **Style** | **Details** |
| Forward | **ItemRevisi on** for the PCA item type defined in the **EDA\_CCAI**  **temTypeD efault** preference | **IMAN\_Ren dering** | **Dataset** | Security Group I | All | Merge | Propagatio n rule between the PCA **ItemRevisi on** and design **Dataset** in a non- variant design. |
| Forward | **ItemRevisi on** for the | **structure\_ revisions** | **PSBOMVie**  **wRevision** | Security Group I | All | Merge | Propagatio n rule |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Direction** | **Source business object** | **Relation** | **Destinatio n business object** | **Prop Group** | **Operation** | **Style** | **Details** |
|  | PCA item type defined in the **EDA\_CCAI**  **temTypeD efault** preference |  |  |  |  |  | between the variant **ItemRevisi on** and **PSBOMVie**  **wRevision**  . |

#### Set configurations based on the ECAD applications used

You can configure which preference values should be used for a specific preference based on the connectors configured for your ECAD design tool. This configuration allows all EDA preferences to be overridden by application-specific preferences, if set. The application specific preferences can be created for all the EDA design preferences by adding a suffix to the OOTB preference name. The suffix is the name of the ECAD tool application as mentioned in the *%TCEDAECAD\_ROOT%\\*\_usrdef.xml* file.

If the preference exists with the suffix of the connector name, this preference is used. Else, the preference without a suffix is used.

Example:

Consider that you are using two different connectors, and you want to set different values for the following preference, depending on which connector is currently being used. You can set different values for the same preference by adding the application name as a suffix to the preference name.

The OOTB preference name is **EDA\_CCAVariantItemTypesAllowed**

To set the preferences with different values for both the connectors, create two different instances of the preference with the connector names suffixed to the preference.

* Preference to be used if connector 1 is used:

**EDA\_CCAVariantItemTypesAllowed\_<connectorname\_1>**

* Preference to be used if connector 2 is used:

**EDA\_CCAVariantItemTypesAllowed\_<connectorname\_2>**

#### Allow users to view the PCB board requirements for ECAD objects in Active Workspace

ECAD designers can view the PCB board requirements specifications related to the design revisions in the **Requirements** tab in Active Workspace for the *EDACCABase* and *EDASchem* object types, by default.

To allow users to configure **Requirements** tab for other Item Revisions, user can include the following in the respective stylesheet.

<page titleKey="tc\_xrt\_Requirements" visibleWhen="ActiveWorkspace:SubLocation

!= com.siemens.splm.client.occmgmt:OccurrenceManagementSubLocation">

<htmlPanel declarativeKey="Eda0RequirementsTraceability" />

</page>

Additionally, you can specify the revision rule to configure the requirements BOM structure in the **Requirements** tab by setting the **EDA\_AWRequirementBOMRevRule** preference. The default value of the revision rule is set to **Latest Working**.

#### Customize the acronyms used for ECAD design objects in Teamcenter

If your ECAD designers use acronyms different from the default acronyms provided, Teamcenter can override the defaults by creating custom acronyms used for ECAD designs in Teamcenter.

By default, EDA uses the following acronyms to refer to electronic design objects on the user interface:

* **PCA** for printed circuit assembly or **CCA** for circuit card assembly (in earlier versions of EDA).
* **PCABase** for printed circuit assembly base or **CCABase** for circuit card assembly base (in earlier versions of EDA).
* **PWB** for printed wire board.

If your ECAD designers use different acronyms, the Teamcenter administrator can override the defaults by creating custom text copies of the **TC\_ROOT\lang\textserver***locale***\eda0\_text\_locale.xml** file. For each *custom-name***\_text\_locale.xml** file, locate the following lines in the file, and edit the display text as required:

<key id="Eda0\_PCA">PCA</key>

<key id="Eda0\_PCAToolTip">Printed Circuit Assembly</key>

<key id="Eda0\_SCH">SCH</key>

<key id="Eda0\_SCHToolTip">Schematic</key>

<key id="Eda0\_PWB">PWB</key>

<key id="Eda0\_PWBToolTip">Printed Wire Board</key>

<key id="Eda0\_Design\_Metrics\_key">Design Metrics</key>

<key id="Eda0\_Cad\_Baselines\_key">CAD Baselines</key>

<key id="Eda0\_Edm\_Viewer\_key">EDM Viewer</key>

Note:

There is a separate locale file for each supported locale. The example shown is from the **en\_US**

locale file.

For more information about customizing textserver file text, see the *Server Customization* help.

#### Configure the EDA client to use the Teamcenter EDA panels instead of Active Workspace

By default, the EDA client uses Active Workspace panels for user actions such as open and save. You can use Teamcenter EDA panels instead, if you prefer to or in case you do not have Active Workspace installed. This can be done by setting the following preferences:

#### EDA\_Use\_ActiveWorkspace

This preference specifies whether to display Active Workspace panels instead of the older client panels in dialog boxes where available. The default value is set to **True**. To switch to Teamcenter EDA client panels, change the preference value to **False**.

The default value is valid only if one of the following preferences is also defined to point to an Active Workspace server.

##### ActiveWorkspaceHosting.URL

* **ActiveWorkspaceHosting.EDA.URL**

If neither preference is defined, then the **EDA\_Use\_ActiveWorkspace** preference is ignored, and Active Workspace is not used. If either one of these preferences is defined, **ActiveWorkspaceHosting.URL** takes precedence and the default for value for **EDA\_Use\_ActiveWorkspace** is **True**.

#### EDA\_Use\_ActiveWorkspace\_Open

This preference specifies whether to display an Active Workspace panel in the **Open** dialog box instead of the Teamcenter EDA client panel. The default value is set to **True**. To switch to Teamcenter EDA client panels, change the preference value to **False**.

The preference **EDA\_Use\_ActiveWorkspace** must also be set to **True** to use this preference.

#### Managing and configuring Teamcenter preferences for design management

You can set system-wide preferences on the Teamcenter server that apply to all EDA users. To understand how preferences work and to understand how to access a list of all supported preferences, refer to the Managing Preferences help on Support Center. All EDA preferences begin with the prefix **EDA**.

1. Run the rich client on the Teamcenter server and log on as an administrator.
2. In My Teamcenter, choose **Edit→Options**.
3. At the bottom of the **Options** dialog box, click **Index**.
4. In the **Preferences** dialog box, search for the preferences that begin with **EDA\_**.
5. Change the preference value and click the **Modify** button to save the new value.

## Installing – EDA – Client Components

The following prerequisites and requirements

#### Installing the EDA client for design management on the ECAD workstation

#### Install the EDA client for design management

Use the **tem.bat** command to launch Teamcenter EDA Environment Manager (TEM) to perform installation and maintenance operations of Teamcenter EDA on an ECAD workstation for design management.

TEM is available with the Teamcenter EDA installation or upgrade images. Contact your Teamcenter installation administrator to obtain the correct product distribution image.

Caution:

Before installing the EDA client on your ECAD workstation, refer to all the prerequisites and the information required to install the client in the [**Planning your deployment**](#_bookmark2) topic.

1. Ensure that the proper version of JRE is installed and the JRE\_HOME environment variable (32-bit system) or the JRE64\_HOME environment variable (64-bit system) is set.
2. Extract the Teamcenter EDA software distribution image to a temporary location.
3. Specify the path to the Java Runtime Environment (JRE) by setting the **JRE\_HOME** environment variable on the ECAD workstation.

You can also specify the JRE path when you launch TEM from the command line using the **-jre** path argument.

1. Start TEM.
   1. Browse to the root directory of the extracted Teamcenter EDA software distribution image.
   2. Right-click the **tem.bat** program icon and choose **Run as administrator**. TEM launches and displays the **Installer Language** panel.
   3. In the **Installer Language** panel, select a language and click **OK**. TEM displays the **Welcome to Teamcenter EDA** panel.

Your language selection applies only to the TEM session and not the Teamcenter EDA installation.

1. In the **Welcome to Teamcenter EDA** panel, click **Install**.
2. In the **Configuration** panel, type the configuration ID and the description for the new Teamcenter EDA configuration.

The configuration ID identifies your Teamcenter configuration when you maintain, upgrade, uninstall, or add features to the configuration. Installation log files are also named based on this ID.

1. Proceed to the **Features** panel and click **Next**.
2. Expand **Mechatronics Process Management** and select the required options:

* (Optional) **EDA Translation**

EDA translation allows converting a PCB or a schematic design into the Teamcenter Visualization ECAD format.

##### ECAD Design Management

Select the design tool support you require. This integrates the design tool with the EDA client. However, a separate installation of the design tool is required.

##### Teamcenter Gateway for EDA Design

1. In the **Installation Directory** box, browse to a folder where the EDA client will be installed. For example, type **c:\EDA\_client** for Windows.

The client is installed to an **eda** subdirectory, for example, **c:\EDA\_client\eda** (Windows). This directory is known as *TCEDAECAD\_ROOT*.

1. Click **Next**.
2. In the **FCC Client Cache (FCC)** panel, select **Use new FCC** and click **Next**.
3. In the **FCC Parents** panel, type the **Host** name of the Teamcenter server in the **FCC Parents** box.
4. In the **4-tier Server Configurations** panel, type the IP address of the Teamcenter server in the **4- tier Servers** box. For example:

**http://***host***:***7001***/tc**

Replace *host* with the server name.

1. In the **EDA Client** panel:
   1. Change the path if needed in the **Staging Location** box.

You must enter the full path to the directory where the ECAD designs are saved on the client. This directory is the working directory for EDA.

If a staging directory already exists, you can continue using the same directory or select another folder as the staging directory.

Caution:

Do not use spaces in the cache or staging directory path. Some EDA functions do not work if there are spaces in these paths.

1. Clear the **Will this configuration be shared by multiple users?** check box to create user environment variables after installation. By default, the check box is selected to create system environment variables.
2. Click **Next**.
3. Review the features selected in **Confirmation** panel and click **Start**. The EDA client components are installed.

The **Install** panel confirms whether the installation is successful. If the installation is successful, click **Close**. If the installation is unsuccessful, click **Show Details** and proceed as needed.

After the installation of EDA client is complete you must verify that the EDA client is installed correctly and [**perform the configurations required to set up design management**](#_bookmark35).

#### Verify the EDA client installation

Verify the EDA client installation by checking the following:

1. Ensure that Teamcenter client communication system (TCCS) services point to the Teamcenter server with EDA server support.
2. The FMS **fcc.xml** file points to your Teamcenter server.
3. The **FMS\_HOME** system environment variable is set and points to the location of the **fcc.xml** file.
4. The **TCEDAECAD\_ROOT** user environment variable is set to your EDA directory (for example,

**c:\EDA\_client\eda** on Windows systems).

The value of **TCEDAECAD\_ROOT** is added to **PATH**.

1. The directory defined by the **TCEDAECAD\_ROOT** user variable exists and is populated with files.

6.

7. The **TCEDAClient.properties** file in the *TCEDAECAD\_ROOT* directory contains the values you entered using the installation routine, for example:

URL=http://hostname:port/tc StagingDir=staging-directory-location

Note:

When you first run EDA, the cache and staging directories are created from the settings in the

**TCEDAClient.properties** file.

The paths of the cache and staging directories contain double back slashes (\\) on Windows systems. For example:

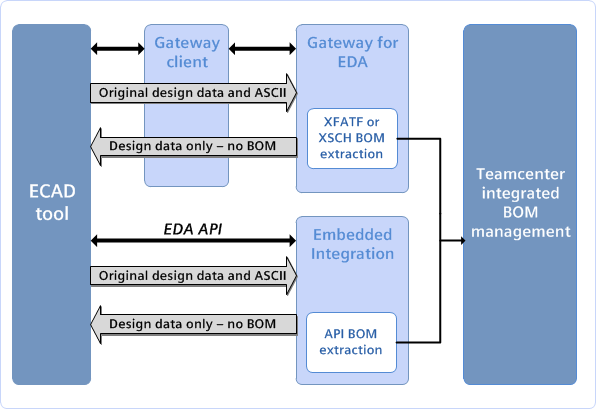
*C:\\ecad\\EdaCache*

*C:\\ecad\\EdaStaging*

#### About Teamcenter Gateway for EDA

Teamcenter Gateway for EDA is used to integrate ECAD design tools that do not support a customizable user interface. Teamcenter Gateway for EDA provides the same Teamcenter menu as that used in the embedded integrations. You can perform regular Teamcenter functions, such as opening, checking out, and saving designs, including schematic designs with or without variants. The application can extract BOM information from a PCB **xfatf** or Schematic **xsch** file.

The following graphic illustrates Teamcenter Gateway for EDA and the ECAD integration as compared

to the embedded menu integration

#### About configuration files and templates used by Teamcenter Gateway

Configuration files define the various characteristics required by Teamcenter Gateway for EDA for EDA library integration. These configurations are only included as templates for customization.

When you start Teamcenter Gateway for EDA, you can either specify a configuration depending on your requirement or skip specifying the configuration. If you do not specify the configuration to be used, a dialog box to select the configuration is displayed.

You can switch to an alternate configuration after you start Teamcenter Gateway for EDA by choosing

**File→Configuration** and selecting another configuration from the displayed list.

Configuration templates are located in the *TCEDAECAD\_ROOT***\eda\example\gateway** directory and can be used to create new customized gateway configurations.

The following are the out-of-the-box templates provided:

##### gatewayCombined\_edadef.xml

This configuration allows ECAD designers to save all design data to Teamcenter as one dataset under a CCA item/item revision. The dataset usually contains all related design data such as schematic, simulation, and layout design files, but may contain only one type of the design files.

##### gatewaySchematic\_edadef.xml

#### gatewayPcb\_edadef.xml gatewaySimulation\_edadef.xml

These configurations are generally used together to allow ECAD designers to save schematic, layout designs, and simulation data in separate datasets under the same CCA item/item revision. The layout and simulation datasets are saved directly under the CCA item/item revision. The schematic design dataset is saved under a schematic item/item revision. The schematic item is a child under the BOM view of the CCA item. All datasets saved under the same CCA item/item revision are called related datasets in a family. These configurations can also be used independently to save different types of design datasets in separate CCA item/item revisions.

The templates provide defaults and contain place holders for you to configure. The following configurable elements and attributes are included in the templates:

* Application name
* Family name for the toolset
* Dataset types
* Translators
* Callbacks

You can add your own elements and attributes as long as the EDA client configuration scheme is followed and the features added can be supported by Teamcenter Gateway for EDA. You can also delete the configurable elements and attributes that are not required.

Note:

A commented instance of an attribute mapping element example is also shown in the template for reference purposes. However, Siemens Digital Industries Software recommends that you configure this element only if you have a good understanding of Business Modeler IDE and EDA client configuration.

#### Creating a configuration file for design integrations

The Teamcenter Gateway for EDA configuration file is a readable XML file and can be created and edited using any ASCII text editor. You can also edit the out-of-the-box configuration template to create a configuration. The configuration file should be named as *application-name***\_usrdef.xml** and should be located in *TCEDAECAD\_ROOT***/gateway** folder.

For a basic configuration, the minimum required configuration is as follows:

##### Family

This is the subdirectory name under the top-level EDA staging directory.

##### Application

The initial part of the file name is *application name***\_usrdef.xml**. This name appears in Teamcenter Gateway for EDA when you choose **File→Configuration** to select a configuration.

##### className

This refers to the Java class defining the model type for this configuration. These model types should not be modified from the templates.

##### PrimaryDataType and datasetType

**PrimaryDataType** specifies the design type. Valid types include **schematic**, **pcb**, and **simulation**. **datasetType** represents the Teamcenter dataset type to be created when saving designs using this configuration.

##### RelatedDataType

The data type that is associated with the primary data type for the concurrent design process.

#### Adding a callback configuration for Teamcenter Gateway for EDA

You can add a callback feature to your configuration. The callback configuration feature allows you to specify names of scripts to be called by Teamcenter Gateway for EDA before or after Teamcenter operations such as save or open.

The callback starts a custom script, which may be written to perform ECAD tool-specific tasks appropriate to that process. The name of each callback script and its type are defined in the configuration file. Whenever a callback is available, the gateway client calls it to prepare for pre- or post- Teamcenter operations. Each callback is specified by its operation type, command name, and a list of arguments.

**Callback type Argument 1 Argument 2 Argument 3**

**extractCADFromPCB extractCADFromSCH extractBOMFromPCB extractBOMFromSCH openDesign reOpenDesign setDesignReadOnly**

**setDesignCheckedOut closeECAD**

**designFolder cadFolder**

**designFolder edaDesignFile**

**designFolder designFolder designFolder designFolder**

**prepareVariantInfo designFolder generateDesignBOMs designFolder true** or

**false**

Name list of variants that

The arguments are defined as follows:

##### designFolder

Specifies the path (directory/folder) containing the design.

##### edaDesignFile

Specifies the **edaDesign** XML file to be returned to the gateway. This file must be created by the callback to contain any data to be returned. It must follow the **EDADesignSchema.xsd** schema.

##### cadFolder

Specifies the folder where intermediate CAD files are placed.

#### Mapping reference designator attributes for your ECAD tools

A reference designator identifies a component within a schematic design or on a printed circuit board. The reference designator usually consists of one or two letters followed by a number, for example *R13*, *C1002*. The number is sometimes followed by a letter, indicating that components are grouped or matched with each other, for example, *R17A*, *R17B*. IEEE 315 contains a list of *Class Designation Letters* to use for electrical and electronic assemblies. For example, the letter R is a reference prefix for the resistors of an assembly, C for capacitors, and so on.

In the EDA client definition file (*\*edadef.xml*) you can find additional elements that are required for mapping the *ReferenceDesignator* element with the *PSOccurrence* attribute. This process is referred to as reference designator (RDN) mapping. One or more of these files are installed in the *TCEDAECAD\_ROOT* folder. Identify the appropriate files in your installation and modify it to enable RDN attribute mapping for your ECAD authoring tools.

Tip:

Save a copy of each original file in case you have problems or must revert to an earlier version, without your modifications.

Caution:

Do not modify any other areas of these files.

The *RdnAttrMapDefs* subelement of the following sample XML file contains additional subelements to enable and configure RDN attribute mapping. You must add this set of elements to your *\*edadef.xml* file in the required order.

The required order of the first-level subelements of the EDA client definition file is as shown below:

Example:

PrimaryDataType RelatedDataType IntermediateDataType ViewableDataType PartitionDataType IssueAttachmentDataType PreferenceDefs CallbackDefs RdnAttrMapDefs TranslatorDef LibraryDef EDAGatewayDef MigrationDef SaveOptionDefs OperationData

Point to consider while mapping the reference designator elements with PSOccurence attributes:

* The **RdnAttrMapDefs** subelement contains examples that are commented out. Siemens Digital Industries Software suggests that you leave the commented out subelement as examples and then copy them to add mappings. Add one subelement for each mapping, making modifications as needed.
* The **BOMLine** property name or **PSOccurrenceNote** property name (**tcAttrName**) must be the name of the Teamcenter **BOMLine** property to which the ECAD attribute value is mapped.
* The ECAD attribute name (**cadAttrName**) must be the name of an ECAD attribute that can be extracted from the ECAD component occurrence or RDN.
* The **RdnAttrDesign** subelement of the **RdnAttrMapDefs** element defines a mapping from an ECAD attribute name to a Teamcenter **PSOccurrence** property or **PSOccurrenceNote** property name.
* The Teamcenter property must have write access for all ECAD designers. Otherwise, a write access error is reported during BOM creation or update.
* The Boolean **(isOccNote)** value simply indicates that the Teamcenter property name is a

**PSOccurrenceNote** property.

* You must add one **RdnAttrDesign** subelement for each ECAD occurrence attribute that you want to map to a **BOMLine** property.
* The **RdnAttrConst** subelement of the **RdnAttrMapDefs** element defines a Teamcenter **BOMLine** property or **PSOccurrenceNote** property that is set to the string specified by the value attribute. You may not need to define any mappings of this type.

Once you modify the \*edadef.xml file, and before you install it, drag it on to your browser. This verifies that you have not introduced any XML syntax errors into the file. If there are any errors, your browser displays an error message. Do not install the modified file until the browser verifies that it is clean, with no errors or warnings.

#### Configuring cache and staging directory

#### Understanding how cache and staging directory works

The cache and staging directories temporarily store data as the ECAD designs move between the design tool and Teamcenter. Occasionally, designers may need to refresh a design to synchronize the local copy of the design with the latest version in Teamcenter. Alternatively, they may need to clear the cache to remove all local copies of the designs that they are not working with. EDA uses a staging directory for the local storage of ECAD designs and the cache directory for storing the temporary data.

The EDA cache directory is created under *USER\_HOME* and its value written to the user property file.

The EDA staging directory holds designs for checkin and checkout with Teamcenter. The first time you choose a command on the **Teamcenter** menu, the staging directory is created automatically. The staging directory path is based on the staging directory setting in the *TCEDAECAD\_ROOT* **\TCEDAClient.properties** file. The current OS username is added to that value as a subfolder and that path is written to a new user-specific **TCEDAClient.properties** file in the *USER\_HOME* directory.

In the *TCEDAClient.properties* file, the environment variable **EDA\_StagingDir** takes precedence over the **StagingDir** property. The **EDA\_StagingDir** environment variable must be defined as a full path and not a relative path to an existing folder that the user has access to or to a folder that the user can create.

#### How to configure the checkout of files in EDA

When the designer opens an ECAD design object for checkout, the EDA file services tools must know both the source and the destination for the action. For a checkout, the source is where the data is stored within the Teamcenter file system and is part of the metadata maintained by Teamcenter. The destination is where the designer wants to store the date and therefore varies.

At a minimum, each workstation has a unique file system. For the EDA file system tools, the designer must establish a fixed location known to EDA for these files. This fixed location is defined at the time of EDA installation. The location is stored within a settings file that all EDA tools can access. During each installation, you must specify where the staging directory is defined for that user and host.

Once this is done, the resulting action for a checkout from EDA is that the data source is a Teamcenter server, and the data destination is specified by the user’s staging directory setting in an EDA settings file. A different user can also access the same data items. In such a case, the destination is the accessing user’s specified staging directory.

#### How to configure the saving of files in EDA

When the designer saves an ECAD object, the source can be any location in the user’s file system, but the destination is always an EDA-provided location within the Teamcenter server. In practice, however, different scenarios are possible. In the case of one ECAD product, a design may be stored within a single file. In another ECAD product, the design data may be stored in multiple files. Many designers may wish to include supplemental data included with the design files. For example, the designer may want to add a *readme* file or add a *specification* document.

To allow designers to save additional files with the design data, EDA stores such files in a separate container. Any files or file objects that are in this container are captured when the ECAD tool executes an EDA **Save**, **Save As** or **Revise** action.

#### Capturing supplemental or hidden ECAD files

Some ECAD products use hidden or invisible files to hold properties or metadata for use by specific ECAD products. EDA assumes that these files are hidden in the directory that contains the design data. EDA uses this directory as a reference point for all its activities and for capturing supplemental or hidden ECAD files. To avoid the possibility of duplicate names (same names for different objects) if multiple designs are stored within containers in a common staging directory, each directory containing an ECAD design must be given a unique name. The name of this directory must match (be the same as that of) the design’s item ID, also known as the part number.

#### Create cache and staging directories

The first time you select a command on the Teamcenter menu and log on, the cache and staging directories are created automatically. The staging directory path is based on the staging directory setting in the *TC\_ROOT\eda\TCEDAClient.properties* file. The current operating system user name is added to that value as a subfolder and that path is written to a new user-specific *TCEDAClient.properties* file in the *USER\_HOME* directory. The cache directory is also created under *USER\_HOME* and its value is written to this property file.

1. Open the design tool.
2. Log on to Teamcenter:
   1. Choose **Teamcenter**→**Purge Working Files**. **Teamcenter EDA Login** dialog box is displayed.
   2. Type your Teamcenter user name, password, and group, if applicable. A group is required only if you want to log on to a group that is not your default group.
   3. Click **Login**.
   4. In the **Purge Working Files** dialog box, click **Cancel**.

When you do this for the first time, the cache and staging directories are created automatically, displaying the folders, including the latest ones.

1. Choose **Teamcenter**→**Logout**. You are logged off Teamcenter.
2. Verify that the cache and staging directories are created.

The directories are created from the settings in the *TCEDAClient.properties* file.

#### Teamcenter EDA staging directory conventions

Teamcenter EDA requires that you create and work on designs in the staging directory. If you have already created designs in another directory, copy them to the staging directory and then work on them exclusively from the staging directory. All design datasets including schematic, PCB, and simulation data files must be saved in a single design root folder that is associated to a PCA item.

In the following examples, replace the first instance of *username* in the path with the operating system user name, replace the second instance of *username* in the path with the Teamcenter user ID, and *siteid* with the site ID of Teamcenter.

#### Modify default user properties in the TCEDAClient.properties file

Caution:

Follow this process only if you must change the properties for the user, the cache directory, or staging directory, this process can be followed. Siemens Digital Industries Software does not recommend this method to switch back and forth between settings but only as a one-time change that should be undertaken only if necessary.

1. Use the EDA client to check in all currently checked-out designs.
2. Use the EDA client to purge the cache.
3. Manually delete the cache and staging directories.
4. Edit the **URL** line in the *TCEDAECAD\_ROOT*\**TCEDAClient.properties** file to point to the new server:

URL=http://*hostname*:*port*/tc StagingDir=*staging-directory-location*

In the example, replace *hostname* with the name of the new server running in the thin client environment, and replace *port* with the port number of that server.

Note:

The paths of the cache and staging directories must contain double backslashes on Windows systems, for example:

C:\\ecad\\EdaCache\\ C:\\ecad\\EdaStaging\\

1. Edit the *USERPROFILE*\**Teamcenter\EDA\TCEDAClient.properties** files to match the preceding settings, as necessary.

Caution:

Any definitions in this file override the file in *TCEDAECAD\_ROOT* even if you edit the properties there.

#### Set a flatter folder structure for saving ECAD designs

In scenarios where two or more designers work on a single ECAD design within a shared network while saving the designs, multiple levels of folders are created based on the operating system user name, Teamcenter user name-site ID, project ID, and so on. This causes the file to be nested deep within the structure. This hierarchical folder structure is defined in the *TCEDAECAD\_ROOT\TCEDAClient.properties* file.

To make this structure flatter and remove the multiple levels of folders, enable the **EDAStagingDirLevels** property defined in this file. The new structure saved in the staging directory excludes the intermediate folders based on the user's profile.

To enable this property, remove the # sign in the left most line where the property is defined.

Example:

By default, when a designer is working on an ECAD design, the files are saved in the staging directory format:

*C:\Staging\user\siteid\app\latest\ItemID*

Remove all the multiple folders created while saving ECAD designs by enabling the

EDAStagingDirLevels property and the files will be saved in the staging directory format:

*C:\Staging\view\_<RevID>\_PrjName* for read-only and previous revision files.

*C:\Staging\PrjName*, for WIP project.

#### Modifying the default logging configuration

You can examine EDA log files to troubleshoot problems. You can update the **log4j.properties** file that is located in the *TCEDAECAD\_ROOT* directory to change the default logging configuration.

In the **log4j.properties** file, you can update the log name, change log values, change the location of log storage, and so on.

After making changes to the **log4j.properties** file, restart the EDA client.

The log files are stored in the location you specified in the **log4j.properties** file and are named as

*$user\_name***\_TcEDA.log**.

The historical log files are named as *$user\_name***\_TcEDA.log\_yyyy-mm-dd.log**.

Note:

Some connectors also produce log files that cover details of their operation. Refer to the

*settings.ini* files within the installed connector for additional information.

#### Customize the integration by adding additional command calls

EDA integrations uses the following commands while performing integration operations between Teamcenter and ECAD tools. You can use these commands to customize the integration.

Use the **edacli.bat** file to access these commands. The **edacli.bat** file initiates a Java process that passes the required arguments to the EDA integration. The **edacli.bat** file is located in the *TCEDAECAD\_ROOT* directory.

#### Syntax

**edacli -***command* **-application=***application\_name* **-status=** *file\_name*

#### Arguments

**-***command*

Specifies the command to execute. The following commands are available:

##### Command Description

**checkPartWorkFlow Status**

Displays the **Workflow Status** dialog box that shows the status of your workflow requests.

**getMyWorklist** Displays the **My Worklist** dialog box that shows your

workflow tasks.

**logout** Logs you off Teamcenter after displaying the **Logout** dialog box.

Note:

The **logout** command only releases the Teamcenter license. To release ECAD tool license, use the ECAD tools.

**logoutNoConfirm** Logs you off Teamcenter without the **Logout** dialog box.

**purgeCache** Displays the **Purge Working Files** dialog box.

The **Purge Working Files** command allows you to clear the cache of the designs that are not checked out.

##### -application

Specifies the application name representing the ECAD tool. This value specifies which configuration file to use.

##### -status

Specifies the full path for the EDA status file that the common client creates.

The common client writes status information to this file. The status is either **success** or **failure**. If the status is **failure**, the common client displays a failure message.

## Installing – Allegro Lib Manager

#### System requirements

To install the connector, the system requirements for the following components must be met.

The following software is required:

* + - Teamcenter supported by EDA Gateway
    - EDA Gateway 5.1 | 5.2 | 6.0
    - Allegro PCB Librarian 17.2 | 17.4
    - Java (64-bit) in Standard Edition (SE) or OpenJDK:
      * Oracle Java SE 8, 11
      * Oracle, Amazon Corretto or Adopt OpenJDK 11-15 (requires at least EDA Gateway 5.0.1)

In general, OpenJDK support for the connector follows the support declared by Teamcenter. Specific to this connector version is, that it only supports OpenJDK up to version 15.

#### Installing the connector

#### Before you start

* + - Allegro PCB Librarian is installed and functional.
    - EDA Gateway is installed, the environment variable %TCEDAECAD\_ROOT% exists and points to the directory eda in the EDA Gateway installation directory, for example C:\Program Files\Siemens

\TeamcenterEDA2\eda.

* + - If you have a previous version of the connector from Support Center installed, uninstall it before you proceed.

If you have modified the connector configuration, backup the configuration files in

%TCEDAECAD\_ROOT% before uninstalling. After the connector installation is complete, compare and merge the changes in the new files with the backed up files.

* + - Cadence supports diﬀerent types of setup with paths to hold customer-specific configuration. The following steps refer to a **Default** setup with the %CDSROOT% path. If you use another setup, for example **Site** or **User**, do not edit files in the %CDSROOT% path, but in the path where the customer-specific configuration is located. Make sure to set up the menus in one path only, otherwise they will not work.

#### Procedure

1. Close all open Cadence applications.
2. Unzip the installation package.

For a manual connector installation, start the install script install.bat with administrator rights and follow the steps of the wizard.

1. For a connector installation using Deployment Center, complete the following steps:
   1. Make sure Deployment Center is of latest version and installed correctly. For instructions on installing and using Deployment Center, see the Deployment Center documentation on Support Center.
   2. Copy the unzipped directories to the software subdirectory in the repository.
   3. Log on to Deployment Center and click **Software Repositories**.
      * The **Software Repositories** page opens the contents of the repository and displays the **Software Media** table.
   4. Check the list of software to verify that it is correct and complete for your planned deployment. Note whether there are missing dependencies as noted. If so, retrieve the missing software and copy it (unzipped) into the repository and check again.

If you experience a problem in adding software to the Deployment Center repository, you can try to troubleshoot the repository service. See the *Troubleshoot the repository service* topic in the Deployment Center help.

* 1. Create the deployment script and use it to install the connector on the target machine.

1. Finish the installation.

* The connector files are copied to the directory %TCEDAECAD\_ROOT%\G2. This path is further referred to in this document as <INSTALL\_DIR>.

1. Installing the connector:
   1. Copy contents of the directory setup in <INSTALL\_DIR>\allegro\_lib to %TCEDAECAD\_ROOT%

\gateway.

* 1. Edit the file setup\_ecadlibrary.bat in %TCEDAECAD\_ROOT% and add the following lines before the section :SETUP\_ODBC :

:SETUP\_Cadence

if not exist "%TCEDAECAD\_ROOT%\G2\allegro\_lib" goto :SETUP\_ODBC rem setup the library Cadence

SET EDALIB\_CP=%EDALIB\_CP%;%TCEDAECAD\_ROOT%\G2\allegro\_lib\classes\cds-lib-api.jar;

%TCEDAECAD\_ROOT%\G2\allegro\_lib\classes\cadence-lib-tce-connector.jar

In section add the line:

if not exist "%TCEDAECAD\_ROOT%\PADS" goto :SETUP\_Cadence

* 1. Save the file.
* Setup is complete.

:SETUP\_MentorPADS

#### Result

The connector is installed.

#### Configuration

The configuration files allow you to easily configure the connector for managing the design in Teamcenter.

#### Connector settings

Configuration file: %TCEDAECAD\_ROOT%\gateway\cadenceLib\_config.xml

|  |  |
| --- | --- |
| **Configuration section** | **Description** |
| convertToTC | convertFromTC | With this two settings, you can define transformation rules for attribute values transferred to Teamcenter  ( convertToTC ) or from Teamcenter ( convertFromTC ). The transformation rules for each attribute can be defined by using regexp, format or map. See samples in the configuration file. |
| CreatePartWithoutAttributes | With this setting, you can create single part instances without attributes in Teamcenter if no PTF information was found ( true ). Default: false . |
| IgnoreLogicalStructure | With this setting, you can synchronize parts without a valid path in the library folder structure. By default, such parts are ignored and with this setting you can include them  for synchronization, for example mechanical parts added manually to the library. The setting contains a comma separated list of attribute and value pairs ( ATTR=VALUE ) to be synchronized. |
| IncludePtf | With this setting, you can define additional PTF files to be synchronized. Each line defines one PTF file. You can use absolute or relative path references to the PTF file. |
| OverrideByBlank | With this setting, you can define the library attributes to be updated with values from Teamcenter, even if they are blank. The setting contains a comma separated list of library attributes. Default: DESCRIPTION . |
| PartID | With this setting, you can define the library attributes to be used as part ID. The setting contains a comma separated list of library attributes. The first non-blank attribute value found will be used. Default: PART\_NUMBER, PART\_NAME . |
| PartName | With this setting, you can define the library attributes to be used as part name. The setting contains a comma separated list of library attributes. The first non-blank attribute value found will be used. Default: PART\_NUMBER, PART\_NAME. |
| SynchronizePartAttributesOnly | With this setting, you can synchronize only the part attributes ( true ). Default: true . If set to false , all information (symbols, footprints and padstacks, etc.) is synchronized. |

#### Configuring email settings for problem reports

Configuration file: <INSTALL\_DIR>\<product>\config\ivs.properties.

# -- Report problem adjustment --

# Specify how to send mail with problem report details ivs.report\_problem.mail\_address = [support@mycompany.com](mailto:support@mycompany.com) ivs.report\_problem.mail\_subject = Problem report ivs.report\_problem.mail\_body = See the attached ZIP file. ivs.report\_problem.mail\_command = cmd /c "start outlook.exe /a ...

Legend

* ivs.report\_problem.mail\_address : Defines the email address.
* ivs.report\_problem.mail\_subject : Defines the email's subject.
* ivs.report\_problem.mail\_body : Defines the email's body text.
* ivs.report\_problem.mail\_command : Defines the command line to start the email application and compose the email (default Microsoft Outlook).

## Installing – Cadence –Allegro Components

The following prerequisites and requirements

#### Installation

#### System requirements

To install the connector, the system requirements for the following components must be met.

The following software is required:

* + - Teamcenter supported by EDA Gateway
    - EDA Gateway 5.1 | 5.2 | 6.0
    - Allegro Design Entry HDL 17.2 | 17.4
    - Allegro PCB Designer 17.2 | 17.4
    - Java (64-bit) in Standard Edition (SE) or OpenJDK:
      * Oracle Java SE 8, 11
      * Oracle, Amazon Corretto or Adopt OpenJDK 11-15 (requires at least EDA Gateway 5.0.1)

In general, OpenJDK support for the connector follows the support declared by Teamcenter. Specific to this connector version is, that it only supports OpenJDK up to version 15.

#### Installing the connector

#### Before you start

* + - Allegro Design Entry HDL and Allegro PCB Designer are installed and functional.
    - EDA Gateway is installed, the environment variable %TCEDAECAD\_ROOT% exists and points to the directory eda in the EDA Gateway installation directory, for example C:\Program Files\Siemens

\TeamcenterEDA2\eda.

* + - If you have a previous version of the connector from Support Center installed, uninstall it before you proceed.

If you have modified the connector configuration, backup the configuration files in

%TCEDAECAD\_ROOT% before uninstalling. After the connector installation is complete, compare and merge the changes in the new files with the backed up files.

* + - Cadence supports diﬀerent types of setup with paths to hold customer-specific configuration. The following steps refer to a **Default** setup with the %CDSROOT% path. If you use another setup, for example **Site** or **User**, do not edit files in the %CDSROOT% path, but in the path where the customer-specific configuration is located. Make sure to set up the menus in one path only, otherwise they will not work.

#### About this task

You can use the connector in two diﬀerent ways:

* + - Combined mode
    - Dual mode: In combined mode, the whole design is saved under one item in Teamcenter. In this mode, you use the menu **Tools** > **Teamcenter** in the Project Manager for operations with EDA Gateway.

In dual mode, schematic and layout documents are saved under separate items in Teamcenter. This has the advantage that the schematic and layout can be checked out separately by diﬀerent engineers. In this mode, you use the menu **Teamcenter** in Allegro Design Entry HDL and Allegro PCB Designer for operations with EDA Gateway.

#### Procedure

1. Close all open Cadence applications.
2. Unzip the installation package.
3. For a manual connector installation, start the install script install.bat with administrator rights and follow the steps of the wizard.
4. For a connector installation using Deployment Center, complete the following steps:
   1. Make sure Deployment Center is of latest version and installed correctly. For instructions on installing and using Deployment Center, see the Deployment Center documentation on Support Center.
   2. Copy the unzipped directories to the software subdirectory in the repository.
   3. Log on to Deployment Center and click **Software Repositories**.
      * The **Software Repositories** page opens the contents of the repository and displays the **Software Media** table.
   4. Check the list of soware to verify that it is correct and complete for your planned deployment. Note whether there are missing dependencies as noted. If so, retrieve the missing soware and copy it (unzipped) into the repository and check again.

If you experience a problem in adding soware to the Deployment Center repository, you can try to troubleshoot the repository service. See the *Troubleshoot the repository service* topic in the Deployment Center help.

* 1. Create the deployment script and use it to install the connector on the target machine.

1. Finish the installation.

* The connector files are copied to the directory %TCEDAECAD\_ROOT%\G2. This path is further referred to in this document as <INSTALL\_DIR>.

1. Decide now which mode you want to use, combined or dual.

The modes correspond to the following directories in <INSTALL\_DIR>:

* Combined mode = cadence\_combined
* Dual mode = cadence\_sch | cadence\_pcb

The directories are further referred to in other places of this document as <MODE>.

1. Setting up connector and menu for combined mode (Project Manager):
   1. Copy contents of the directory edaclient in <INSTALL\_DIR>\cadence\_combined\setup to

%TCEDAECAD\_ROOT%.

* 1. Open the file projmgr\_menu.txt in <INSTALL\_DIR>\cadence\_combined\setup and copy the entire content to the clipboard.
  2. Edit the file cds.cpm in %CDSROOT%\share\cdssetup\projmgr, add the contents from the clipboard aer the first line START\_TOOLS , and save the file.
* Setup for Project Manager is complete.

1. Setting up connector and menu for dual mode (Allegro Design Entry HDL):
   1. Open the file concepthdl.scr in <INSTALL\_DIR>\cadence\_schematic\setup and copy the entire content to the clipboard.
   2. Edit the file concepthdl.scr in %CDSROOT%\share\cdssetup\concept, add the contents from the clipboard at the end, and save the file.
   3. Check for the file concepthdl\_cmd.txt in %CDSROOT%\share\cdssetup\concept:

* If it doesn't exist, copy the file concepthdl\_cmd.txt from <INSTALL\_DIR>

\cadence\_schematic\setup\cdssetup.

* If it exists, open the file concepthdl\_cmd.txt in <INSTALL\_DIR>\cadence\_schematic\setup and copy the entire content to the clipboard. Edit the existing file, add the contents from the clipboard at the end, and save the file.
  1. Open the file concepthdl\_menu.txt in <INSTALL\_DIR>\cadence\_schematic\setup and copy the entire content to the clipboard.
  2. Edit the file concepthdl\_menu.txt in %CDSROOT%\share\cdssetup\concept, add the contents

from the clipboard before the line , and save the file.

"&Help"

* Setup for Allegro Design Entry HDL is complete.

1. Setting up connector and menu for dual mode (Allegro PCB Designer):
   1. Check for the file allegro.ilinit in %CDSROOT%\share\local\pcb\skill:

* If it doesn't exist, copy the file allegro.dynamic.ilinit from <INSTALL\_DIR>\cadence\_pcb

\setup to %CDSROOT%\share\local\pcb\skill, and rename the file to allegro.ilinit.

* If it exists, open the file allegro.dynamic.ilinit in <INSTALL\_DIR>\cadence\_pcb\setup, and copy the entire content to the clipboard. Edit the existing file allegro.ilinit in %CDSROOT%

\share\local\pcb\skill, add the contents from the clipboard at the end, and save the file.

* Setup for Allegro PCB Designer is complete.

#### Result

The connector is installed.

#### Next steps

Check the connector installation:

#### Combined mode

* + - * Start Project Manager.
      * Check for the menu **Teamcenter** in the menu **Tools**.
      * Select **Tools** > **Teamcenter** > **Open**. This will start the RAC or AW login window.
      * Login to Teamcenter and if successful, close the dialog.

#### Dual mode

* + - * Start Cadence Allegro HDL or OrCAD PCB Designer.
      * Check for the menu **Teamcenter** in the menu bar.
      * Select **Tools** > **Teamcenter** > **Open**. This will start the RAC or AW login window.
      * Login to Teamcenter and if successful, close the dialog.

#### Initial setup

#### Setting up connector

To set up the connector for initial use, complete the following steps.

#### Procedure

1. Decide which *BOM comparison* dialog you want to use, see [this configuration section](#_bookmark10) for more information.
2. We recommend to define a replacement number for missing BOM items in Teamcenter. This is important for Teamcenter 13, as it does not automatically create any missing items during the BOM upload anymore. See [this configuration section](#_bookmark11) for more information. See also [this troubleshooting section](#_bookmark30) for identifying missing BOM items during BOM upload.
3. To upload the BOM correctly, you must define the Allegro HDL parameter for the component number and name, see [this configuration section](#_bookmark17) for more information.
4. Define the setup for derived file generation, see [this configuration section](#_bookmark13) for more information.

#### Result

The main connector setup is complete.

#### Next steps

Check remaining post-installation chapters.

#### Setting up AW for workflow control (dual mode)

If AW is configured (set the preferences *EDA\_Use\_ActiveWorkspace\** in Teamcenter to *true*), you can deactivate the menu items **Teamcenter** > **Workflow** > **Design Workflow** and **Teamcenter** > **Workflow** > **Status Workflow**, and access these functions directly in AW.

#### About this task

To set up this feature, complete the following steps.

#### Procedure

1. Deactivating menu items (Allegro Design Entry HDL):
   1. Edit the file concepthdl\_menu.txt in %CDSROOT%\share\cdssetup\concept and replace the following lines

"&Workflow" {

"&Design Workflow..." I2\_design\_workflow; "&Part Workflow..." I2\_part\_workflow; "&Workflow Status..." I2\_workflow\_status;

with the line ("&Part Workflow..." "I2\_part\_workflow") .

* 1. Save the file.

1. Deactivating menu items (Allegro PCB Designer):
   1. Edit the file integrate\_menu.il in <INSTALL\_DIR>\cadence\_pcb\custom\allegro\skill and replace the following lines

(popup "&Workflow")

("&Design Workflow..." "I2\_design\_workflow") ("&Part Workflow..." "I2\_part\_workflow") ("&Workflow Status..." "I2\_workflow\_status") (end)

with the line ("&Part Workflow..." "I2\_part\_workflow") .

* 1. Save the file.

#### Result

The menu **Teamcenter** now contains **Part Workflow** only. You can submit a design to a workflow by using the functionality provided in AW

#### Configuration

The configuration files allow you to easily configure the connector for managing the design in Teamcenter.

#### Connector settings

* Configuration file (combined mode): %TCEDAECAD\_ROOT%\cadence\_combined\_config.xml
* Configuration file Allegro Design Entry HDL (dual mode): %TCEDAECAD\_ROOT%

\cadence\_schematic\_config.xml

* Configuration file Allegro PCB Designer (dual mode): %TCEDAECAD\_ROOT%\cadence\_pcb\_config.xml

|  |  |
| --- | --- |
| **Configuration section** | **Description** |
| Any | For Allegro HDL in combined mode, you must be able to distinguish configuration sections between schematic and layout or the two editors in one file.  Add the option pcb="true" to a configuration section, to make this section specific to Allegro PCB Designer. If not set, the configuration applies by default to Allegro Design Entry HDL, which corresponds to false .  See [this configuration section](#_bookmark17) for an example. |
| BomCompare | With this setting, you can define which *BOM comparison*  dialog to use, or disable it completely.  **Example**  <BomCompare dialog="true"  />  **Legend**   * If not set, uses the *BOM comparison* dialog from the connector, which corresponds to the default setting g2 . * If set true , uses the *BOM comparison* dialog from EDA Gateway. * If set false , uses no *BOM comparison* dialog. |

|  |  |
| --- | --- |
| **Configuration section** | **Description** |
| BomFilter | With this setting, you can trigger a filtering mechanism to include only specific components in the BOM. The setting is commented out by default and all components are included in the BOM. Uncomment if you need special processing.  **Example**  <BomFilter field="REFDES">  <Exclude>TP\*</Exclude>  </BomFilter>  **Legend**   * field : Defines the name of the component parameter. * Include : Defines BOM items with this value in the component parameter to be included in the BOM. * Exclude : Defines BOM items with this value in the component parameter to be excluded from the BOM.   To classify your components for BOM inclusion, add a corresponding component parameter, for example *REFDES*, and classify the components with the allowed values. In the configuration you can use wildcards "\*" (any character) and "?" (one character) for filtering. |
| BomReplaceMissingPartNumbers | With this setting, you can define an item number from Teamcenter that is used to replace all missing BOM items when comparing the BOM in the BOM comparison dialog prior the upload.  If an existing replacement number is defined, then the view Missing in PLM in the BOM comparison dialog remains empty. If the replacement number is defined, but does not exist in Teamcenter, it appears in the view Missing in PLM as only entry and indicates that something has gone wrong.  Example  <BomReplaceMissingPartNumbers missing\_part\_number="00001234" item\_query="Item - simple" ignore\_missing="true"  />  Legend   * missing\_part\_number : Defines an existing item ID in Teamcenter to be used as a replacement for all missing BOM items. * item\_query : Defines the name of the Teamcenter query to check if an item exists. In most cases it's not necessary to change this default value. * ignore\_missing : If not set, missing BOM items are sent to Teamcenter during save, which corresponds to the default setting false . Set to true to send no missing BOM items and continue saving without interruption. |
| BomTemplate | With this setting, you can define the full path to the BOM template that is used for exporting the BOM. By default, the template with ending .bom is located in <INSTALL\_DIR>  \<MODE>\config.  <BomTemplate  value="<path to template>\tcEdaPcbTemplate.bom"  />  Aer generation, you can find the BOM report in the directory %TEMP%\eda\<timestamp>\_<module>  \bom\_report.  Make sure that parameters defined here are also reflected in the BOM item parameter definition, see [this configuration](#_bookmark17) [section](#_bookmark17) for more information. |
| CSVBom | With this setting, you can include additional BOM items from a CSV file into the BOM that are normally not available in the design.  Example  <CSVBom  file="<path to csv>" id="<component ID>" refdes="<reference designator>" sendOthers="true" skipEdaBom="false"  />  Legend   * file : Defines the path to the CSV file relative to the design directory. You can use the variable $variant in the file name for the CSV and it will be converted to the current variant. * id : Defines the name of the column containing the component number. * refdes : Defines the name of the column containing the reference designator of the BOM item. * sendOthers : If set true , all other values from the CSV file are sent as reference designator values. * skipEdaBom : If set true , only the BOM items from the CSV file are included in the BOM and the BOM items from the design are skipped. If set false , the BOM items from the CSV file are merged with the BOM from the design.   CSV file specification   * The first line has to contain the column headers, all other lines the BOM items. * The delimiter is comma. |
| DerivedFileGeneration | With this setting, you can configure the automatic derived data creation from the schematic and layout.  You must first set up the derived data definitions in Teamcenter BMIDE. In a second step, you must reference the definitions from BMIDE here, and supplement the derived data creation with additional options. See the configuration file for further information.  Example  <DerivedFileGeneration name="Schematic\_Drawing" config="file\_schematic\_drawing"  />  Legend   * name="Schematic\_Drawing" : Defines the name of the dataset in the derived data configuration in BMIDE. * config="file\_schematic\_drawing" : Defines the reference to the pre-defined output configuration. |
| DesignInfo | By default, design information is shown in RAC. If configured, AW client can also be used (set the preferences EDA\_Use\_ActiveWorkspace\* in Teamcenter to true).  With this setting, you can show design information from AW in a floating window. See also [this troubleshooting section](#_bookmark33) in case the window appears empty.  Example (AW 5.x)  <DesignInfo awc-url = "http://<server>:<port>"/>  Example (AW 4.x)  <DesignInfo awc-url = "http://<server>:<port>/ awc"/>  Define a valid URL incl. protocol (http or https) and port for AW, for example http://teamcenter13:3000 (AW 5.x) or http://teamcenter12:7001/awc (AW 4.x). |
| DownloadLibraries | You can manage graphical library data (symbols, footprints, padstacks, etc.) in Teamcenter using your own processes.  With this setting, you can define the item ID that contains the dataset with the graphical data and a download location. Then you can start a function to download this data via a batch file or a menu entry. The item query parameter contains the name of Teamcenter item revision query.  Example  <DownloadLibraries item-id="1234567"  file-path="C:\EdaLibrary\library.zip" item-query = "Item Revision..."  />  Running download function:   * Use the batch file download\_libraries- noconsole.bat in <INSTALL\_DIR>\<MODE>\bin   \win32\.   * Add a new menu entry to the Teamcenter menu in Allegro HDL (combined mode):   1. Close Project Manager.   2. Edit the file cds.cpm in %CDSROOT%\share   \cdssetup\projmgr.   * 1. In the file, search for enable download libraries and uncomment the lines as shown.   2. Save the file.   3. Start Project Manager. * Add a new menu entry to the Teamcenter menu in Allegro HDL (dual mode):   1. Close Allegro Design Entry HDL and Allegro PCB Designer.   2. Edit the file concepthdl\_menu.txt in %CDSROOT%   \share\cdssetup\concept.   * 1. In the file, search for enable download libraries and uncomment the lines as shown.   2. Save the file.   3. Repeat for the file integrate\_menu.il in   <INSTALL\_DIR>\cadence\_pcb\custom\allegro  \skill.   * 1. Start Allegro Design Entry HDL and Allegro PCB Designer. |
| FmsServer (ECAD-MCAD collaboration) | You can export an .idx file from your current design and upload it to the item in Teamcenter as collaboration data for MCAD.  You can also download MCAD collaboration data uploaded to the same item.  To enable collaboration features, you need to define the URL for the FMS server and activate corresponding menu entries for the Teamcenter menu in Altium.  Defining FMS server URL  <FmsServer url = "http://<server>:<port>/"/>  Use a valid URL incl. protocol (http or https) and port.  The variable FMS\_HOME points to the location of the FMS configuration directory. Go there and open the file fcc.xml. Towards the end, you'll find the URL for the FMS server, for example http://teamcenter13:4544/ .  Activating menu entries for collaboration   * Add a new menu entry to the Teamcenter menu in Allegro HDL (combined mode):   1. Close Project Manager.   2. Edit the file cds.cpm in %CDSROOT%\share   \cdssetup\projmgr.   * 1. In the file, search for enable ECAD-MCAD and uncomment the lines as shown.   2. Save the file.   3. Start Project Manager. * Add a new menu entry to the Teamcenter menu in Allegro HDL (dual mode):   1. Close Allegro Design Entry HDL and Allegro PCB Designer.   2. Edit the file concepthdl\_menu.txt in %CDSROOT%   \share\cdssetup\concept.   * 1. In the file, search for enable ECAD-MCAD and uncomment the lines as shown.   2. Save the file.   3. Repeat for the file integrate\_menu.il in   <INSTALL\_DIR>\cadence\_pcb\custom\allegro  \skill.   * 1. Start Allegro Design Entry HDL and Allegro PCB Designer. |
| EdifWriter | With this setting, you can define the path to the EDIF-writer executable to generate a CAD file that then is converted to a viewable file. The value must be a valid (callable) utility command that accepts two arguments in the format of:  <name of command> -proj <name of the cpm file>  -out\_file <name of the output file> . The setting is commented out by default. Uncomment if you need special processing. |
| MechPartNumber | With this setting, you can define which parameter of a component is used when creating an ECAD BOM for mechanical parts.  Example  <MechPartNumber field="^MECH\_PN"  />  Legend  field : Defines a regular expression to select parameters containing the mechanical part number from the parameter table of a component.  For example, in your design the component with the number PART1 contains the parameter MECH\_PN1. This parameter is used to hold the number for a mechanical component. The regular expression ^MECH\_PN selects all parameters starting with MECH\_PN and adds them to the BOM as mechanical parts for the component PART1.  To make these parts appear in the BOM, you have to add the parameter of the mechanical part number also to the BOM template tcEdaPcbTemplate.bom, for example:  ...  #! mech part number MECH\_PN1  ...  See [BOM template configuration](#_bookmark12) for more information. |
| PartInBom | With this setting, you can define which parameter of the component is used for exporting the BOM.  Make sure that parameters defined here are also reflected in the BOM template, see [this configuration section](#_bookmark12) for more information.  Example  <PartInBom id="PART\_NUMBER" name="DESCRIPTION"  refdes="BOM\_INST"  />  Legend   * id : Defines the parameter of the component number. * name : Defines the parameter of the component name. * refdes : Defines the parameter of the component's reference designator.   Example (BOM export from layout in combined mode)  <PartInBom pcb="true"  id="COMP\_PART\_NUMBER" name="COMP\_PART\_NAME"  refdes="REFDES"  />  Legend  pcb : If set true , export is performed from layout. |
| RDNAttributes | By default, all RDNAttributes of a BOM item are sent to Teamcenter. With big BOMs consisting of several thousand items, the extraction time could lead to a timeout in the connector.  With this setting you can control the extraction of RDNAttributes.  Example (extract all attributes)  <RDNAttributes value="all"/>  Example (extract required attributes only)  <RDNAttributes value="minimal"/>  Example (extract required attributes plus defined)  <RDNAttributes>  <RDNAttribute name="Value"/>  <RDNAttribute name="Cost"/>  </RDNAttributes> |
| VariantNames case-sensitive | EDA Gateway ignores variants in certain save situations to Teamcenter due to case sensitivity and other causes.  With this setting you can prevent variants from being ignored due to case sensitivity.  Example  <VariantNames case-sensitive="false"/>  If not set, the default behaviour is case-sensitive comparison, which corresponds to the default setting true  . In this case, variants could be ignored in Teamcenter.  Set to false for case-insensitive comparison. In this case, variants are not ignored in Teamcenter. |

|  |  |
| --- | --- |
| **Configuration section** | **Description** |
| VariantNames keep-linked | If a variant is unlinked through EDA Gateway from the base item, it remains in Teamcenter without a relation.  With this setting, you can keep the relation of an unlinked variant to the base item, but as consequence you cannot upload the BOM for such unlinked variants.  **Example**  <VariantNames keep-linked="true"/>  If not set, the variant relation from the base item is removed when unlinked, which corresponds to the default setting false .  Set to true to keep the variant relation on the base item, even when unlinked. |

#### Configuring email settings for problem reports

Configuration file: <INSTALL\_DIR>\<product>\config\ivs.properties.

# -- Report problem adjustment --

# Specify how to send mail with problem report details ivs.report\_problem.mail\_address = [support@mycompany.com](mailto:support@mycompany.com) ivs.report\_problem.mail\_subject = Problem report ivs.report\_problem.mail\_body = See the attached ZIP file. ivs.report\_problem.mail\_command = cmd /c "start outlook.exe /a ...

Legend

* ivs.report\_problem.mail\_address : Defines the email address.
* ivs.report\_problem.mail\_subject : Defines the email's subject.
* ivs.report\_problem.mail\_body : Defines the email's body text.

ivs.report\_problem.mail\_command : Defines the command line to start the email application and compose the email (default Microso Outlook)

## Installing – Cadence –Orcad Capture CIS Components

#### Installation

#### System requirements

To install the connector, the system requirements for the following components must be met. The following software is required:

* + - Teamcenter supported by EDA Gateway
    - EDA Gateway 5.1 | 5.2 | 6.0
    - OrCAD Capture CIS 17.2 | 17.4

Lite and standard versions are not supported.

* + - OrCAD PCB Designer 17.2 | 17.4
    - Java (64-bit) in Standard Edition (SE) or OpenJDK:
      * Oracle Java SE 8, 11
      * Oracle, Amazon Corretto or Adopt OpenJDK 11-15 (requires at least EDA Gateway 5.0.1)

In general, OpenJDK support for the connector follows the support declared by Teamcenter. Specific to this connector version is, that it only supports OpenJDK up to version 15.

#### Installing the connector

#### Before you start

* + - OrCAD Capture CIS and OrCAD PCB Designer are installed and functional.
    - EDA Gateway is installed, the environment variable %TCEDAECAD\_ROOT% exists and points to the directory eda in the EDA Gateway installation directory, for example C:\Program Files\Siemens

\TeamcenterEDA2\eda.

* + - If you have a previous version of the connector from Support Center installed, uninstall it before you proceed.

If you have modified the connector configuration, backup the configuration files in

%TCEDAECAD\_ROOT% before uninstalling. Aer the connector installation is complete, compare and merge the changes in the new files with the backed up files.

* + - Cadence supports diﬀerent types of setup with paths to hold customer-specific configuration. The following steps refer to a **Default** setup with the %CDSROOT% path. If you use another setup, for example **Site** or **User**, do not edit files in the %CDSROOT% path, but in the path where the customer-specific configuration is located. Make sure to set up the menus in one path only, otherwise they will not work.

#### About this task

The connector comes in dual mode only, a combined mode is not available. In dual mode, schematic and layout documents are saved under separate items in Teamcenter. This has the advantage that the schematic and layout can be checked out separately by diﬀerent engineers.

#### Procedure

1. Close all open Cadence applications.
2. Unzip the installation package.
3. For a manual connector installation, start the install script install.bat with administrator rights and follow the steps of the wizard.
4. For a connector installation using Deployment Center, complete the following steps:
   1. Make sure Deployment Center is of latest version and installed correctly. For instructions on installing and using Deployment Center, see the Deployment Center documentation on Support Center.
   2. Copy the unzipped directories to the software subdirectory in the repository.
   3. Log on to Deployment Center and click **Software Repositories**.

➔ The **Software Repositories** page opens the contents of the repository and displays the **Software**  **Media** table.

* 1. Check the list of software to verify that it is correct and complete for your planned deployment. Note whether there are missing dependencies as noted. If so, retrieve the missing soware and copy it (unzipped) into the repository and check again.

If you experience a problem in adding soware to the Deployment Center repository, you can try to troubleshoot the repository service. See the *Troubleshoot the repository service* topic in the Deployment Center help.

* 1. Create the deployment script and use it to install the connector on the target machine.

1. Finish the installation.

➔ The connector files are copied to the directory %TCEDAECAD\_ROOT%\G2. This path is further referred to in this document as <INSTALL\_DIR>.

1. Setting up connector and menu (OrCAD Capture CIS):
   1. Copy the file integrate\_menu.tcl in <INSTALL\_DIR>\orcad\_schematic\setup to %CDSROOT%

\tools\capture\tclscripts\capAutoLoad.

➔ Setup for OrCAD Capture CIS is complete.

1. Setting up connector and menu (OrCAD PCB Designer):
   1. Check for the file allegro.ilinit in %CDSROOT%\share\local\pcb\skill:
      * If it doesn't exist, copy the file allegro.dynamic.ilinit from <INSTALL\_DIR>\orcad\_pcb

\setup to %CDSROOT%\share\local\pcb\skill, and rename the file to allegro.ilinit.

* + - If it exists, open the file allegro.dynamic.ilinit in <INSTALL\_DIR>\orcad\_pcb\setup, and copy the entire content to the clipboard. Edit the existing file allegro.ilinit in %CDSROOT%

\share\local\pcb\skill, add the contents from the clipboard at the end, and save the file.

➔ Setup for OrCAD PCB Designer is complete.

#### Result

The connector is installed.

#### Next steps

Check the connector installation:

* + - Start OrCAD CIS.
    - Check for the menu **Teamcenter** in the menu bar.
    - Select **Teamcenter** > **Open**.
    - Login to Teamcenter and close the dialog if successful.

## Installing Cadence SPB Software

#### Cadence software can be downloaded from:

#### https://downloads.cadence.com

#### Log in with a valid user ID and password and click the Windows tab. In the Windows tab, click

#### the link OrCAD/Allegro 17.4 (SPB174).

#### Note: If you are an OrCAD customer, contact Cadence Channel Partners to obtain their

#### software. Click here to see a list of Cadence Channel Partners.

#### Download CD 1 of 1 and then extract the zip file into a temporary directory such as

#### cdnstemp.

#### Complete the installation by running setup.exe from the temporary directory. You can install

#### the following using the downloaded files:

#### ■ Main Installations:

#### ❑ Cadence Allegro and OrCAD Products

#### ❑ Allegro Design Entry HDL – PSpice Designer Library

#### ❑ Cadence Allegro and OrCAD Client

#### Note: License Manager is installed and configured automatically if needed, provided a

#### valid license file is available while installing any of the above.

#### ■ Optional Installations:

#### ❑ OrCAD Document Editor

#### ❑ Allegro PCB Manufacturing Option

#### ❑ OrCAD Library Builder

#### ❑ OrCAD Component Information Portal

#### ❑ OrCAD Engineering Data Management

#### Consult the installation guide for detailed information.

#### Note: If prompted, reboot the machine and log in with the administrator privileges login id to

#### successfully complete the installation.

## Post-Installation Checklist

In the next section, describe the steps to install and configure the components.

#Checklist Item

## Troubleshooting

In the next section, describe the steps to install and configure the components

## Reference Documents

There are multiple documents referred as part of documenting the Technical design and the solution implementation document and they are referred as below.

|  |  |  |  |
| --- | --- | --- | --- |
| S No | Document Name | Link | Comment |
| 1 |  |  |  |
|  |  |  |  |