TEAMCENTER

Teamcenter EDA Schema — Reference

Teamcenter 2412

SIEMENS

About Siemens Digital Industries Software

Siemens Digital Industries Software is a global leader in the growing field of product lifecycle management (PLM), manufacturing operations management (MOM), and electronic design automation (EDA) software, hardware, and services. Siemens works with more than 100,000 customers, leading the digitalization of their planning and manufacturing processes. At Siemens Digital Industries Software, we blur the boundaries between industry domains by integrating the virtual and physical, hardware and software, design and manufacturing worlds. With the rapid pace of innovation, digitalization is no longer tomorrow’s idea. We take what the future promises tomorrow and make it real for our customers today. Where today meets tomorrow. Our culture encourages creativity, welcomes fresh thinking and focuses on growth, so our people, our business, and our customers can achieve their full potential.

Support Center: **support.sw.siemens.com**

Send Feedback on Documentation: **support.sw.siemens.com/doc_feedback_form**

Contents

EDAStatus

EDALib

EDALibConf

1. Overview of schema reference

This developer reference document provides supplementary information for the developers to create new or customize exisiting Teamcenter EDA integration instances or callbacks.

It is pressumed that the user of this document has read Xpedition and PADS Integration With Teamcenter or Teamcenter EDA Gateway for (Non-Siemens EDA) ECAD Applications help for deploying and setting up Teamcenter EDA and understands the basic operations of EDA.

2. EDAClientDef schema

EDAClientDef

Each new connector creates a configuration file to define the integration. This configuration file specifies information that includes the application name, the family the application belongs to, the common client class that must be used, and the Teamcenter dataset types that must be used.

Name this file *applicationName*_**edadef.xml** (where *applicationName* is replaced with a unique value, for example, **mentor_edadef.xml**), and save it in the CLASSPATH for the common client to locate it. It is typically saved in the *%TCEDAECAD_ROOT%* folder.

The configuration file is an XML file based on the following schema (**EDAClientDef.xsd**).

The **EDAClientDef** is the main section of the schema and the attributes and elements of this schema are explained in detail in the subsequent topics in this section.

The *applicationName*_**edadef.xml** contains configuration elements written by the connector developer and this file should not be modified by the users. For all user specific customizations you can create another customization file, *applicationName*_**usrdef.xml** and all modifications or customizations must be done in this file only. Both *applicationName*_**edadef.xml** and *applicationName*_**usrdef.xml** configuration files have the same structure.

className

This is a required attribute and specifies the common client class to be used as the basis for this integration. You have the option of using one of the default classes shown below, or you can specify a subclass to be used.

- **com.teamcenter.edabase.dual.DualEDAPcbFactory**- Used for dual-model PCB design configuration.
- **com.teamcenter.edabase.dual.DualEDASchematicFactory**- Used for dual-model Schematic design configuration.
- **com.teamcenter.edabase.DefaultEDAFactory**- Used for combined-model design configuration
- **com.teamcenter.edabase.lib.DefaultLibraryFactory**- This is the default library class. This class has service and user interface factory elements that are used by library integration.

application

This is a required attribute that specifies the application name for this connector. This name must match with the name specified at the beginning of the file name. For example, for **<EDAClientDef application=”mentor” .../>**, the filename must be **mentor_edadef.xml**).

For an embedded connector, this is the name passed from the ECAD connector to the common client in the application argument.

For a gateway connector, it is the name that is displayed in the **Set Configuration** dialog box.

family

This is a required attribute and specifies the family name for a set of configurations. The family name is used as part of the staging directory for design connectors.

For example, to support Mentor PADS you may have two configuration files, one for the PCB tool and one for the schematic tool. Each of these configuration files will have separate application names, such as **padsPcb** and **padsSchematic**. However, both these applications will be using the same family name **pads**.

bomOptionModel

This is used by design connectors to specify BOM option model the connector supports. Valid values for BOM option model are **pcbOnly**, **schematicOnly**, or **none**. If not specified, the default option is based on the data model.

- For dual model the BOM is from the primary type.
- For combined model the BOM may be either from PCB or schematic.

A conflicting specification such as, **pcbOnly** for schematic connector should not be used. This behavior is not defined for such conflicting specifications.

enableVariants

This is an optional attribute used by design connectors, and specifies whether this integration supports variants. The default is **true**.

When set to true, all design datasets are saved in a **CCAItemBase** instead of **CCAItem**, if the design has variants or the future variant option is selected.

To support designs with variants, the following callbacks need to be configured as well:

- **prepareVariantInfo**
- **generateDesignBOMs**

supportsBaseBom

This is an optional attribute used by design connectors, and specifies that the connector can supply a base BOM when working with a variant design.

If the value is set to **true** the user will be given an option to save the base BOM with the design, in addition to the variant BOM's. If it is set to **false**, then the user will only have the option to save the variant BOM's.

The default value is **false**.

preserveSession

This is an optional attribute and specifies whether the login information should be preserved, using cookies. The default value is **false**. Set it to **true** for integrations that make calls to the common client through the **edacli.bat** file. This is because such calls start a new process that would otherwise require the user to log on again.

enableOpenRelatedDatasets

This is an optional attribute used by design connectors, and specifies whether this integration supports opening related datasets along with the primary dataset. The default value is **true**.

copyFilesInSaveAsCopy

This is an optional attribute used by design connectors, and specifies whether the common client should copy the design files during a **Save As Copy** operation. The default value is **true**.

When the value is set to **false**, the EDA common client will not copy the design from the current folder to a new destination folder using **Save As**. This feature must be used when the ECAD tool uses an absolute path for its design files so that an operating system copy cannot be used simply to create a new design. Therefore, when this attribute is set to **false**, it is the responsibility of the user to make sure that the design is copied. The copying can be done either manually or by **postPreSaveAs** callback.

allowVariantNameAsItemId

This is an optional attribute used by design connectors, and allows the variant name of a new variant to default to the item ID. The default value is **false**.

USASCIIRequired

This is an optional attribute used by design connectors. It specifies whether the common client must use only US ASCII characters for certain values, such as, for the staging directory name. The default value is **true**.

- Item ID
- Revision ID
- Variant Name
- Dataset Name
- Mapped attribute names and values
- Dataset Pathnames

enableDoNotKeep

This is an optional attribute used by design connectors, and specifies that the **Remove working files from local storage** option should be enabled for the **SaveAs**, **Checkout**, and **Cancel-Checkout** dialog boxes. Selecting this option removes the working files from local storage, that is the staging folder, after the operation is complete. The default value is **false**.

allowUnsavedVariants

This optional attribute used by the design connectors indicates that the application supports or does not support the **Control Variants Saved to Teamcenter** capability. The schema defined default value is **false**. A **true** or **false** value is derived from an **AND** operation of this attribute value and the existing **enableVariants** attribute value, which indicates whether the capability is supported by the application or not.

supportSpacesInStagingDir

This optional attribute used by the design connectors indicates whether the application supports spaces in the Teamcenter EDA staging directory path or not. The default value of this attribute is **true**, which means that spaces are supported by the ECAD tool.

The ECAD tools that do not support spaces in staging location can define this attribute in the respective application configuration definition file, that is the ***_eda.def** files, and set the value to **false**.

enableEDAUIDs

This optional boolean attribute indicates whether EDA UIDs are provided by the ECAD tool integration. These are unique and persistent identifiers on each component in the design. By using these attributes, the BVR updates in Teamcenter remain safe. The default value is **false**.

licenseKey

This optional string attribute can contain a Teamcenter feature key that must be validated and checked-out before allowing this configuration to be used.

If this attribute is specified then the given value will be validated and checked-out through the Teamcenter license verification. In addition to this specified value, either the design gateway or the library gateway feature key as appropriate, will also be validated and checked-out. The default value is **""**, which will allow the internal mapping to be used.

updateComponentOccurrenceInfo

This boolean attribute indicates whether to provide Teamcenter occurrence UID values back to the connector after the completion of the EDA operation. If it is set to **true**, Teamcenter occurrence UID values are returned to the connector through the EDA design file for each component passed in. The default value is **false**.

PrimaryDataType

This element defines the dataset type to be used while storing the primary data, the design or native graphic entity information, to Teamcenter. It contains the following elements and attributes.

name

This is a required attribute for design connectors, which specifies the internal data-type string. It must be set to one of the following values:

- **pcb**
- **schematic**
- **simulation**
- **combined**

Library connector configurations do not use this attribute.

datasetType

This is a required attribute and specifies the Teamcenter dataset type. The **datasetType** cannot be any arbitrary name. It must be a valid dataset type that exists in the Teamcenter data model. You must not associate two different design data types to the same Teamcenter dataset type. Different default operations may be applied to different dataset types in Teamcenter.

For design connectors, this specifies the dataset which will hold the design zip file.

For library connectors, this specifies the dataset which will hold the shape library.

RootLevel

This element, only used by design connectors, is an integer specifying the number of levels the current working directory is below the root, that is, the main **ItemId** design directory. For example, in Mentor PADS a PCB design may be stored under the **pads\latest\des01\brd** directory. The working directory while in the PADS Layout application is the **brd** directory, which is one level below the root, the main design **des01** directory. Hence, the value of **RootLevel** will be set to **1**. The default value is **0**.

Include

This element, only used by design connectors, specifies the directories to include and are relative to the root when saving to Teamcenter. For example, in the same PADS example discussed above, this element will be defined as **brd/***. You can specify only one **filespec** per **<Include>**, but you can specify any number of **<include>** elements as required. If no **<Include>** elements are specified, then all files under the root folder will be included, unless excluded as described below.

Exclude

This element, only used by design connectors, specifies the directories to exclude and are relative to the root when saving to Teamcenter. If no **<exclude>** elements are specified then nothing is excluded.

RelatedDataType

This section is only used by design connectors and it defines the related dataset types that may be associated with designs of this configuration. For example, the PCB configuration will reference the schematic data type and the schematic configuration will reference the PCB data type.

See **PrimaryDataType** for a description of this element.

IntermediateDataType

This section is only used by design connectors and it defines the intermediate viewable dataset types, which may be associated with designs of this configuration. The intermediate dataset is typically used as the source to the ECAD translator to generate a viewable dataset. Each **IntermediateDataType** element must have a matching **ViewableDataType** and **TranslatorDef** element.

name

This is a required attribute and it specifies an internal datatype string. Use one of these values as appropriate:

- **pcbCad**

- **schCad**

datasetType

This is a required attribute and it specifies the Teamcenter dataset type. Intermediate dataset types currently supported are:

- **EDAGenPCBCAD**
- **EDAGenSchem**

ViewableDataType

This section is only used by design connectors and it defines the viewable dataset types, which may be associated with designs of this configuration. The viewable data is created from the **IntermediateDataType**. Each **ViewableDataType** element must include a matching **IntermediateDataType** and **TranslatorDef** element.

name

This is a required attribute and it specifies an internal datatype string. Use one of these values as appropriate:

- **xfatf**
- **xsch**

datasetType

This is a required attribute and specifies the Teamcenter dataset type. The viewable dataset types currently supported are:

- **PCBFATF**
- **SCHFATF**

IssueAttachmentsDataType

This section defines the type of a dataset that can hold attachments related to an issue.

Refer **PrimaryDataType** for a description of this element. However, only the **name** and **datasetType** attributes are used here.

PreferenceDefs

This element is deprecated.

CallbackDefs

The EDA common client uses callbacks to automate the implementation of customer and ECAD tool specific data gathering and processing functions. The development of all EDA supported callbacks is restricted by the available API or utility functions from the ECAD tools. See the Customizing callbacks for design and library integrations documentation for more information.

The **<CallbackDefs>** contains any number of **<callback>** sub-elements that each define a specific callback.

For details on callback usage, refer to the Customizing callbacks for design and library integrations documentation on Support Center. For gateway specific callback help, refer to the *Teamcenter EDA connector integration guide* on Support Center.

callback

Each **<callback>** element defines a specific callback through the following attributes.

type

The type attribute specifies either **java** or **script**.

A java callback is based on the CLASSPATH.

A script callback is run as a system-level call. The script must be found in the PATH, typically located in the *%TCEDAECAD_ROOT%* directory.

operation

The operation attribute specifies the processing point where this callback is called. All arguments are passed to these callbacks (whether script or java) as strings. Refer to the "EDA public classes and methods reference Callback Method Signatures" topic in the *Customizing Callbacks for Teamcenter EDA Design and Library Integrations* help for a list of operation names available.

command

The command attribute specifies the java classname or script filename.

For a java callback, specify the package, insert a colon,
and specify the method name. An example is given below.
`com.teamcenter.eda.mentorexplibcon_jni.MentorExpLibConWrapper:exportLibrary`

RdnAttrMapDefs

Additional BOM attributes are supported by the EDA component occurrence or through the reference designator (RDN) mapping. RDN attribute mapping provides unidirectional mapping of CAD design component occurrence attributes to Teamcenter Product Structure (PS) occurrence properties or occurrence notes. RDN attribute mapping is optional and must be configured by the customer based on business requirements.

This element may contain zero or more of each of the **RdnAttrDesign** and **RdnAttrConst** elements. The part occurrence mappings defined in these elements must not conflict with each other. Never map different ECAD attributes to the same Teamcenter attribute or note as that behavior is not defined. The Teamcenter attributes or occurrence notes used in the configuration must have been created in or exist in Teamcenter.

RdnAttrDesign

The **RdnAttrDesign** defines a map between an ECAD part attribute or property and a Teamcenter component attribute or occurrence note. When a map is configured by this element, the ECAD attribute or property value for the occurrence is saved under the mapped attribute or occurrence note in Teamcenter. This element has the following attributes:

- **cadAttrName** – required to specify the ECAD part attribute or property.
- **tcAttrName** – Required to specify a Teamcenter component attribute or occurrence note.
- **isOccNote** – **true** (default value) for saving as an occurrence note, **false** for saving as an attribute.

RdnAttrConst

The **RdnAttrConst** element defines a constant value of component attribute automatically added to every BOM line when the design is saved with BOM to Teamcenter. This element has the following attributes:

- **tcAttrName** – Required to specify a Teamcenter component attribute or occurrence note.
- **value** – Required to define a constant to be saved.
- **isOccNote** – **true** (default value) for saving as an occurrence note, **false** for saving as an attribute.

TranslatorDef

These elements, only used by design connectors, define the translators to be used to convert the intermediate viewable files into Teamcenter viewable files. Each **TranslatorDef** element must have a matching **ViewableDataType** and **IntermediateDataType** element.

The Teamcenter Gateway for EDA framework can extract BOM from the converted viewable file when file based BOM reports are not available. Therefore, no **TranslatorDef** is required if neither BOM nor viewables are required to be saved to Teamcenter.

type

This attribute must be **xfatf** for PCBs or **xsch** for Schematics.

translator

This attribute specifies the translator name and must be a translator supported within EMPS.

Examples

A PCB configuration example for Mentor PADS is:

```
<IntermediateDataType name="pcbCad" datasetType="EDAGenPCBCAD"/>
<ViewableDataType name="xfatf" datasetType="PCBFATF"/>
<IssueAttachmentsDataType/>
<PreferenceDefs/>
<TranslatorDef type="xfatf" translator="pads"/>
```

A schematic configuration example for Cadence Design Entry is:

```
<IntermediateDataType name="schCad" datasetType="EDAGenSchem"/>
<ViewableDataType name="xsch" datasetType="SCHFATF"/>
<IssueAttachmentsDataType/>
<PreferenceDefs/>
<TranslatorDef type="xsch" translator="edif300"/>
```

A combined model configuration for Mentor Board Station is:

```
<IntermediateDataType name="pcbCad" datasetType="EDAGenPCBCAD"/>
<IntermediateDataType name="schCad" datasetType="EDAGenSchem"/>
<ViewableDataType name="xfatf" datasetType="PCBFATF"/>
<ViewableDataType name="xsch" datasetType="SCHFATF"/>
<IssueAttachmentsDataType/>
<PreferenceDefs/>
<TranslatorDef type="xfatf" translator="mentor"/>
<TranslatorDef type="xsch" translator="edif200"/>
```

LibraryDef

This section is used to define whether the library data type is *Part Library* or *Shape Library*.

SelectLibraryPathInImport

This attribute is obsolete in library integration.

SyncLibrary

Set this attribute to **true** for a shape library integration. In this case the **Synchronize ECAD Library** checkbox in Teamcenter is enabled, and user clicks the **Browse...** option to define the shape library path. This button action is defined by the **Extension** attribute.

SyncDatabase

Set this attribute to **true** for a part library integration. In this case the **Synchronize Part Mentadata Database** check box in Teamcenter is enabled. The new integration implements the **configureMetadataDB** callback for part library configuration. Refer to the EDA public classes and methods reference Callback Method Signatures topic in the *Customizing Callbacks for Teamcenter EDA Design and Library Integrations* help for a list of operation names available.

SyncFile

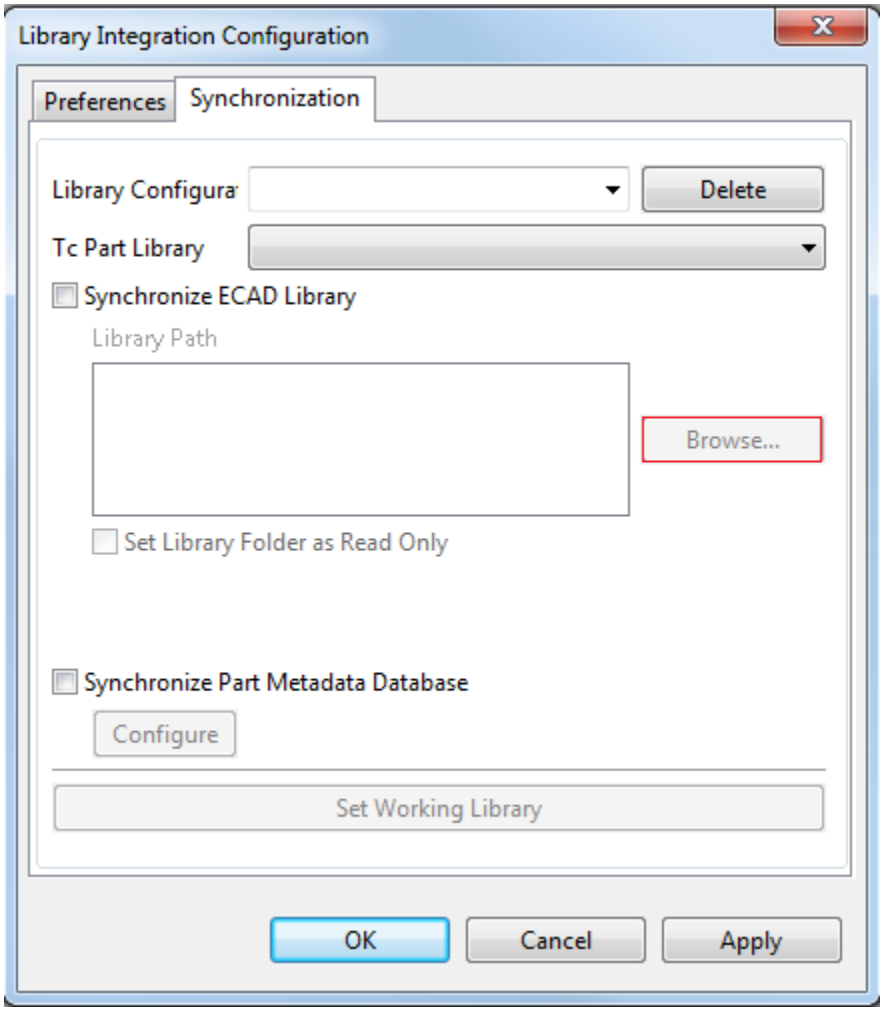
Set this attribute to **true** for a library integration using simple file transfers. This is used when there is no direct access to the ECAD tools library. A librarian must use import and export commands in the ECAD Tool librarian program to import and export files which the EDA connector must parse or create.

This attribute is not allowed with either **SyncLibrary** or **SyncDatabase**. Neither of above mentioned checkboxes are enabled.

Extension

This is an optional attribute and only applies to shape library integrations.

You can create the configuration from the EDA gateway Teamcenter a **Configure a Synchronization** tab, and click **Browse...** to see the **Extension**.



- Omit the attribute if you are customizing the **Library Path**, but have already defined the **configureLibraryPath** callback. The **Browse** button invokes the callback.

- When no **configureLibraryPath** callback is defined but **Library Path** is an extension file, you must set this value. For example, **Extension="*.lmc."** The **Browse** button filters only the valid files in the dialog.

- Click the **Browse** button to specify a folder path when **configureLibraryPath** and **Extension** are not set.

MultipleRevision

Set this attribute to **true** for a multiple revision support in **Capital Library Integration Def** file. When the **MultipleRevision** attribute is set to **true**, **ExportLibrary**, **SyncLibrary**, and **LoadLibrary** operations can work with multiple revisions of the same item. When an item info passed in the **edalib.xml** file through the connectors have revision info, the export, sync and load operations can work on specific revision of that item.

EDAGatewayDef

This section is used for ECAD tool integrations uses EDA Gateway instead of ECAD tool embedded integration. This element must be included in a gateway configuration file, even if it is empty. See the document Teamcenter EDA Connector Integration help for more details. It can contain the following elements:

disableConfigMenu

This optional boolean attribute may be used to disable the **Configuration** menu. This may be useful for scenarios where only a single configuration is used, or when do not want the users to modify it. The default value is **false**.

commandServiceID

This attribute is a place-holder for future work.

BOMExtractionDef

This element may be included zero or one time. It contains the following attributes to define what attributes to use from the viewable file. If this element is not included, then part number and part name from the viewable file are used for item number and item name respectively.

- **itemNumber**

  This required attribute specifies the attribute to use for the Item Number when parsing the viewable file.

- **itemName**

  This optional attribute specifies the attribute to use for the Item Name when parsing the viewable file.

ExternalOperationDefs

This optional element may be included zero or one time. It is used to define design operations that are initiated by an ECAD connector and not by the *Teamcenter Gateway for EDA* itself. In this case the menu entry for the operation will not be shown in the *Teamcenter Gateway for EDA* user interface. By default, no external operations are specified. This element may contain any number of the following elements.

Operation

The element defines the operation to be initiated from the user interface menu options. It consist of the following attribute.

Name

This attribute contains the name of the operation to be initiated from the user interface menu options. The available name values are:

File menu

- **fileConfiguration**
- **fileExit**

Teamcenter menu

- **teamcenterOpen**
- **teamcenterItemInfo** (Design Info)
- **UnsupportedOperationDefs**
- **teamcenterSave**
- **teamcenterSaveDerivedData**
- **teamcenterSaveAs2**
- **teamcenterRevise**
- **teamcenterCheckIn**
- **teamcenterCheckOut**
- **teamcenterCancelCheckOut**
- **teamcenterMyInbox**
- **teamcenterBomCompare**
- **teamcenterRefresh**
- **teamcenterPurgeCache**
- **teamcenterLogout**

Diagnostic menu

- **communicationProfiler**

UnsupportedOperationDefs

This optional element may be included zero or one time. It is used to define design operations that are not supported by this ECAD connector and not by the *Teamcenter Gateway for EDA* itself. In this case the menu entry for the operation will not be shown in the *Teamcenter Gateway for EDA* user interface. By default, no unsupported operations are specified. See **ExternalOperationDefs** for details.

Example

```
<EDAGatewayDef>
 <BOMExtractionDef itemNumber=”CallID” itemName=”catalogName” />
</EDAGatewayDef>
```

MigrationDef

This section is used to define a relative source folder where the migrated designs must be opened. Any design which is opened and does not contain a manifest file within the stored zip file, is presumed to be a design saved in an earlier version of EDA.

Prior to 8.1 all design files were saved to the zip file, and presumed to be directly under the root (itemid) folder. Since 8.1 adds support for include/exclude properties, and designs are typically now saved under a sub-folder, during open we need to know what folder to create. This element may contain the following attribute(s).

relativeSrc

This string attribute contains a folder name to be created under the root itemId folder when opening an EDA design saved in a prior version of EDA.

SaveOptionDefs

This element is a place-holder for future updates.

saveOption

Indicates the save operation user interface option default values such as **BOM**, **Viewable**, and **DerivedData** must be forced based on the content of the **EDADesign** XML. For instance, if BOM data is specified, save it to Teamcenter.

name

Name of the save option, such as **BOM**, **Viewable**, and **DerivedData**.

OperationData

This section is only used in library configurations to support operation extension points. For example, when you save a new part in Teamcenter through the **saveLibrary() API**, the part does not link with any external application in Teamcenter. However, in some cases, if you want to save a new part into Teamcenter and link it with Teamcenter external applications, add the following code to the **edadef** file.

```
<OperationData operationName="SaveLibraryOperation">
 <Attr name="linkToTCLib" value="true"/>
</OperationData>
```

> Note:
>
> This is the only **OperationDataType** that is supported currently.

3. EDADesign

EDADesign

The EDA design file is used to pass information between the EDA common client and an EDA connector or between the common client and a customer callback.

The full path of the file is passed to the connector or callback. To access the objects of the design XML file, use *JAXB* APIs. Refer to javadocs in *%TCEDAECAD_ROOT%***\example\docs\EdaXsdJavadocs**. Refer to the examples of the java classes required to access data from an XML file, using *JAXB* APIs in the *%TCEDAECAD_ROOT%***\example\Callback\java\com\teamcenter\edacallbacks**.

> Note:
>
> EDA XML file schema and associated java API are subject to change from release to release.

The design file schema is defined in the file **schema\EDADesignSchema.xsd**. Elements and attributes of the file are defined here. All attributes are string type unless otherwise noted.

A sample **EDADesign** file outline is shown below. The **EDADesign** file contains a single **CCA** element, which contains all other EDA design elements. Each design element types are described in this section.

```
<?xml version ="1.0" encoding="UTF-8" ?>
<EDADesign xmlns="http://www.plmxml.org/Schemas/tceda">
  <CCA>
    <dataset/>
      ...
    <CCAVariant>
        <component/>
    <CCAVariant/>
    <component/>
      ...
  </CCA>
</EDADesign>
```

CCA

The EDADesign must contain one CCA element.

```
<CCA
    name   = ”CCA name”
    itemId = ”CCA itemId”
    revId  = ”revision”
```

```
    bom    = ”true|false”
    UID    = ”UID of CCA Item Revision”
    supportedBOMOptions = “BOM options supported by ECAD tool”
>
    <dataset/> ...
    <CCAVariant/> ...
    <component/> ...
</CCA>
```

name

The **name** attribute is optional. It specifies the design name.

itemId

The **itemId** attribute is optional. It specifies the design part number. The **itemId** field on the **saveAs** dialog box will be initialized to this value if specified.

revId

The **revId** attribute is optional. It specifies the design revision. The **revision** field on the **saveAs** dialog will be initialized to this value if specified.

bom

The boolean **bom** attribute is optional. It is used in the save operations to indicate if BOM is saved. The default value is **false**. If this attribute is passed as **true**, but there are no **component** elements, then it indicates that **CCA** has no component children (any current ECAD components will be removed from the CCA BOM in Teamcenter).

supportedBOMOptions

The **supportedBOMOptions** attribute is optional. It specifies the BOM options supported by the ECAD tool. Multiple values are comma separated. If the preference **EDA_CombinedBOMOptionDefault** value is empty then the first value of this property will be used as the default.

Possible values are:

- **FromPCB**
- **FromSchematic**
- **FromPCB,FromSchematic**

- **FromSchematic,FromPCB**

UID

This optional attribute specifies the *Unique Identifier* of the CCA item revision in the Teamcenter database.

This attribute is returned to the connector. It is ignored if passed in to the EDA common client.

positionUnits

This optional attribute specifies the PCB design position units (For example, millimeters, centimeters, meters, inches, and mils) for all CCA/CCAVariant component instances. This should only be specified when BOM is generated from the PCB design.

rotationUnits

This optional attribute specifies the PCB design rotation units (For example, degrees or radians) for all CCA/CCAVariant component instances. This should only be specified when BOM is generated from the PCB design.

updateTcEdaUID

This optional attribute specifies if ECAD tool connector must update **TcEdaUID** property for components in an ECAD design. This attribute value will be **true** when the EDA common client finds a duplicate **TcEdaUID** in the ECAD design.

dataset

Dataset elements contain the information describing a design or related data. There must be at least one dataset element but there is no limit to the maximum number allowed.

```
<dataset
    type         = ”Data Type”
    UID          = ”UID of dataset”
    src          = ”design source folder”
    checkedOut   = ”Dataset checkout status”
    checkOutUser = ”Dataset checkout userid”
    nonLatest    = ”Dataset is not latest”
    version      = ”Dataset version”
    <attr> ...
</dataset>
```

type

This required attribute specifies the data type of the dataset. This should match one of the ***DataType** name specified in the configuration file (that is, **pcb**, **schematic**, or **simulation**).

UID

This optional attribute specifies the **UID** of the dataset. This is used to find the dataset in Teamcenter.

src

This optional attribute specifies the design source folder. For a design file, this should be the source folder containing the current design. Specify the full path to this folder.

In EDA gateway user interface, the source folder is the folder you select to start the save-as operation. This folder is usually a sub folder of the design root folder depending on whether the dataset configuration is the connector or client definition file. When the root level is 0, the source folder is the same as the design root folder.

checkedOut

This optional boolean attribute specifies whether the dataset is checked out. This setting is returned from the **open**, **refresh**, **getItemInfo**, **checkout**, and **cancel-checkout** operations. It is ignored if passed in to the EDA common client.

checkOutUser

This optional attribute specifies the user ID of the ECAD user who has this dataset currently checked out. This attribute is returned to the connector. It is ignored if passed in to the EDA common client.

checkOutDate

This optional attribute specifies the date and time the dataset was checked out and it remains blank if the design is not checked out. This attribute is returned to the connector. It is ignored if passed in to the EDA common client.

nonLatest

This optional boolean attribute is not typically used by connectors. It specifies whether the design in the operation is the latest version (**false**) or not (**true**).

version

This optional integer attribute specifies the dataset version.

name

This optional attribute specifies the partition name and must be equal to the partition dataset name.

releaseDesign

This optional boolean attribute is passed from the connector to the EDA common client. The default value is **false**. If the value is **true** then it indicates that the ECAD tool has released the design and that the design must now be released in Teamcenter. The attribute is only valid for the primary dataset type.

attr

An **attr** element within a **<dataset>** element contains information for a design attribute. The design attributes will be stored in Teamcenter based on Teamcenter CAD attribute mapping definitions. **Attr** elements are also used during file *Open* operations to update the ECAD design attributes that are mapped through Teamcenter CAD attribute mapping definitions in which Teamcenter is specified as the master.

There may be zero or any number of **attr** elements.

```
<attr
 name = attributeName
 value = attributeValue
/>
```

attribute name	Description
name	This required attribute specifies the attribute name.
value	This required attribute specifies the attribute value.
type	This is optional attribute. This is used to specify Teamcenter attribute type to distinguish properties of different types such as, **String**, **Array**, **LOV** and so on. If the attribute type is an **Array** then the connectors are supposed to build a string with all array values using "**,**" as a delimiter and send this string as the attribute value. The EDA common client checks the attribute type and uses the "**,**" delimiter to split the attribute value when saving attribute mapping values to Teamcenter. Similarly, the EDA common client constructs the attribute value using "**,**" as delimiter while importing attribute mapping values from Teamcenter. The connectors are supposed to use "**,**" delimiter parameter value to split the attribute value and construct the string in the required format needed for updating CAD attributes.

CCAVariant

CCAVariant elements contain the information describing a design variant. There may be zero or any number of CCAVariant elements.

```
<CCAVariant
    variantName = ”variantName”
    itemName    = ”variantItemName”
    itemId      = ”variantItemId”
    revId       = ”variantRevId”
    bom         = ”true|false”
    supportedBOMOptions = “BOM options supported by ECAD tool”
    <attr> ...
    <component> ...
</CCAVariant>
```

variantName

This required attribute specifies the name of the variant.

itemName

This optional attribute specifies the item name for this variant.

itemId

This optional attribute specifies the item ID for this variant.

revId

This optional attribute specifies the revision ID for this variant.

UID

This optional attribute specifies the UID for this variant. This is used to find this variant in Teamcenter.

bom

This optional boolean attribute indicates whether component sub-elements that identify the CCA variant children are present. In case where bom is equal to **true** and there are no component sub-elements, the CCA variant has no children. The default value is **false**.

supportedBOMOptions

The **supportedBOMOptions** attribute is optional. It specifies the BOM options supported by the ECAD tool. Multiple values are comma separated. If the preference **EDA_CombinedBOMOptionDefault** value is empty, then the first value of this property will be used as default. Possible values are:

- **FromPCB**
- **FromSchematic**
- **FromPCB,FromSchematic**
- **FromSchematic,FromPCB**

attr

An **attr** element within a < CCAVariant > element contains information for a variant attribute. The variant attributes will be stored in Teamcenter based on Teamcenter CAD attribute mapping definitions. **Attr** elements are also used during the *Open* operations to update ECAD variant attributes that are mapped through the Teamcenter CAD attribute mapping definitions in which Teamcenter is specified as the **master**.

There may be zero or any number of attr elements.

```
<attr
 name = ”attributeName”
 value = ”attributeValue”
/>
```

Attribute name	Description
name	This required attribute specifies the attribute name.
value	This required attribute specifies the attribute value.
type	This is optional attribute. This is used to specify Teamcenter attribute type to distinguish properties of different types such as **String**, **Array**, **LOV** and so on.
delimiter	This is optional attribute. This is used to split the attribute value. For example. if the attribute type is **Array** then the connector can specify a delimiter using this parameter to split the attribute value. The EDA common client checks the attribute type and uses the delimiter to split the attribute value when saving attribute mapping values to Teamcenter. Similarly, the EDA common client constructs the attribute value using “**,**” as delimiter when importing attribute mapping values from Teamcenter. The connectors must use this delimiter parameter value to split the attribute value and construct the string in the required format needed for updating CAD attributes.

component

There may be zero or any number of component elements. These define the components for this variant. See **component** for more information.

component

Component elements contain the information describing an EDA component. There may be zero or any number of component elements.

```
<component
 itemId = ”componentItemId”
 revId = ”ComponentRevision”
 name = ”componentName”
 quantity = ”ComponentQuantity”
>
 <rdn> ...
</component>
```

itemId

This required attribute specifies the itemId of the component.

revId

This optional attribute specifies the revision Id of the component. If not specified, the default revision rule is used to set the revision.

name

This optional attribute specifies the name of the component.

quantity

This optional integer attribute specifies the number of times this component is used in the CCA. quantity is normally defined by the number of specified rdn sub-elements. It is only necessary to specify a quantity attribute when one or more component instances are not defined by rdn sub-elements.

rdn

An rdn element contains occurrence information for the component. There may be zero or any number of Reference Designator elements.

```
<rdn
```

```
 name = ”RdnName”
 UID = “Yndsbdjsfj889ss”
>
 <attr> ...
</rdn>
```

name

This required attribute specifies the reference designator name.

UID

This optional attribute specifies BOM Line property bl_clone_stable_occurrence_id.

attr

An attr element within an <rdn> element contains information for a component occurrence attribute. The component occurrence attributes will be stored in Teamcenter based on **RdnAttrMapDefs** defined in the current configuration. Attr elements are also used during Open operations to update ECAD design attributes that are mapped via Teamcenter CAD Attribute Mapping Definitions in which Teamcenter is specified as the “master”.

There may be zero or any number of attr elements.

name

This required attribute specifies the attribute name.

value

This required attribute specifies the attribute value.

PropertyDefault

PropertyDefault elements contain information defining default values for a design being saved to Teamcenter. This can be used for any required and/or visible properties that would be displayed on the corresponding EDA item creation UI. There may be zero or any number of PropertyDefault elements.

These may be added to the design file during the following customer callback operations:

- prePreSaveAs
- prePreRevise
- prePreSave

- prePreCheckin

EdaType

This required attribute specifies the EDA logical object type. Valid values are **CCA**, **SCHEMATIC**, **PWB**, **VARIANT**, or a derived item configuration name.

TcType

This required attribute specifies the Teamcenter internal type. Examples are **Item**, **ItemRevision**, **ItemRevision Master**, and so on.

name

This required attribute specifies the Teamcenter internal property name. Examples are **item_id**, **item_revision_id**, **item_comment**, and so on.

readonly

This optional boolean attribute can be used to make this property field non-editable within the SaveAs UI. The default is **false**, set to **true** to disallow modification of this property by the user during a save.

value

This required attribute specifies the default value for this property.

variantName

This optional attribute specifies the name of a variant, used in conjunction with the EdaType attribute where EdaType is **VARIANT** or a Derived Item configuration name for which the parent type is CCAVariant.

Examples

Set the default value for the CCA Item ID to *003298*. The user should not be able to change this value.

```
<PropertyDefault
 EdaType = "CCA"
 TcType = "Item"
 name = "item_id"
 readonly = "true"
 value = "003298"
/>
```

Set the default value for the *user_data_3* property of the **ItemRevision Master** form, for all CCAVariants listed in the **EDADesign.xml** file. The user should not be able to change this value.

```
<PropertyDefault
 EdaType = "VARIANT"
 TcType = "ItemRevision Master"
 name = "user_data_3"
 readonly = "true"
 value = "variant user data3"
/>
```

Set the default value for the **item_comment** property of the **ItemRevision Master** form, for just Variant *A5E01601833*. The user should not be able to change this value.

```
<PropertyDefault
 EdaType = "VARIANT"
 TcType = "ItemRevision Master"
 name = "item_comment"
 readonly = "true"
 value = "variant item A5E01601833"
 variantName = "A5E01601833"
/>
```

Set the default value for the **item_comment** property of the **Item Master** representing the derived item configuration *DerivedItem07*. The user can override this value on the UI.

```
<PropertyDefault
 EdaType = "DerivedItem07"
 TcType = "Item Master"
 Name = "item_comment"
 readonly = "false"
 value = "DerivedItem07 Item"
/>
```

Set the default value for the **item_comment** property of the **ItemRevision Master** representing the derived item configuration *DerivedItem06*. The user can override this value on the UI.

If this derived items parent type is a CCAVariant and if the schematic design has more than one CCAVariant, this will set the default value for all the combinations of *DerivedItem06* with each CCAVariant. For example, if the schematic design has two CCAVariants *A5E01618833* and *A5E01618836*, there will be two derived item configuration for *DerivedItem06*, one with *DerivedItem06*, *A5E01618833* and the other with *DerivedItem06*,*A5E01618833*

```
<PropertyDefault
```

```
 EdaType = "DerivedItem06"
 TcType = "ItemRevision Master"
 Name = "item_comment"
 readonly = "false"
 value = "DerivedItem06 Item"
/>
```

Set the default value for the *user_data_3* property of the **ItemRevision Master** representing the derived item configuration *DerivedItem06*. The user can override this value on the UI.

The parent type is assumed to be CCAVariant in this example. In this case, this will set the default value only on the derived item related to CCAVariant *A5E01618833*.

```
<PropertyDefault
 EdaType = "DerivedItem06"
 TcType = "ItemRevision Master"
 Name = "user_data_3"
 Readonly = "false"
 variantName = "A5E01618833"
 value = "DerivedItem06 user data3"
/>
```

4. EDAStatus

EDAStatus

The status file is used to return results and the status from the EDA common client to an EDA connector, or from the common client to a customer callback.

The full path to the file is passed to the connector or callback. To access the objects of the status XML file, use JAXB APIs; refer to javadocs in *%TCEDAECAD_ROOT%***\example\docs\EdaXsdJavadocs**. See the example java classes provided to access data from an XML file, using JAXB APIs, in the *%TCEDAECAD_ROOT%***\example\Callback\java\com\teamcenter\edacallbacks**.

NOTE: EDA XML file schema and associated java API are subject to change from release to release.

The status file schema is defined in file **schema\EDAStatus.xsd**. Elements and attributes of the file are defined here. All attributes are string type unless otherwise noted.

EDAStatus is a generic status element returned to the connector. Most operations will use one of the specific types listed later. However, all of them extend this to include the following attributes.

This is a generic status element returned to the connector. Most operations will use one of the specific types listed later. However, all of them extend this to include the following attributes.

status

This attribute contains the status value string, with one of the following values.

- **Success** – when the operation is completed successfully
- **Cancel** – when the operation is canceled by the user
- **Error** – when the operation is terminated prematurely because of an error

message

This attribute contains a detailed message string for the **Error** status. This message, however, is displayed to the user by way of the common client, and the connector can therefore typically ignore it.

EDAOpenDataSetStatus

This status element is returned from the -open and -pdmopen API calls. It only returns the EDAStatus *status* and *message* attributes shown above.

EDAPreSaveStatus

This status element is returned from the **preSave** API call. It returns the EDAStatus *status* and *message* attributes shown above, as well as the following.

BOMOption

This attribute contains a string denoting the BOM option selected by the user. This string will be one of the following.

- **WithoutBOM** – if no BOM is requested.
- **WithBOM** – if BOM is requested in a dual-model configuration.
- **FromPCB** – if PCB BOM is requested in a combined-model configuration.
- **FromSchematic** – if Schematic BOM is requested in a combined-model configuration.

The connector must use this to determine whether to generate BOM <component> elements in the subsequent **EDADesign** file sent to the **- save** API.

viewableOption

This attribute contains a string of comma delimited viewable directives. There are currently only two viewable types supported, so this string will be one of the following:

- **WithoutViewable** – if no viewable is requested.
- **WithViewable** – if a viewable is requested in a dual-model configuration.
- **FromPCB** – if a PCB viewable is requested in a combined-model configuration.
- **FromSchematic** – if a Schematic viewable is requested in a combined-model configuration.
- **FromPCB,FromSchematic** – if both a PCB and Schematic viewable are requested in a combined-model configuration.

The connector should use this to determine whether to generate an intermediate viewable file and include appropriate <dataset> element in the subsequent **EDADesign** file sent to the **-save** API.

datasetMappedAttrNames

This attribute contains a string of comma delimited CAD design attribute names that the connector must obtain values for and return in the **EDADesign** file sent to the **-save** API. ECAD attribute names appearing in this list are defined in the Teamcenter server CAD attribute mapping definitions.

Teamcenter server CAD attribute mapping is a Teamcenter server feature and must be performed by the Teamcenter administrator to modify, test, and deploy.

Values for attributes that are returned by the connector in the **EDADesign** file that are not specified in this string will be ignored.

rdnMappedAttrNames

This attribute contains a string of comma delimited CAD component occurrence attribute names that the connector should obtain values for and return in the **EDADesign** file sent to the **-save** API. ECAD component attribute names appearing in this list are defined in the EDA client definition files *RdnAttrMapDefs* element.

Values for component occurrence attributes that are returned by the connector in the **EDADesign** file that are not specified in this string will be ignored.

nonVariantToVariantConversion

This boolean attribute indicates whether the current operation will convert a non-variant design to a variant design in Teamcenter. A value of **true** indicates that this is a non variant to variant conversion operation. For the reverse the value is **false**.

itemId

This attribute contains the PCA item ID of the design being saved.

designTcStatus

This string attribute contains the additional error info about the dataset with the given UID. Two error strings are valid when the operation fails.

- **tc_unknown** to indicate that the dataset does not exist in Teamcenter so that the save operation cannot continue.
- **tc_unchecked_out** to indicate that the dataset is not checked out so that the save operation cannot continue.

uidRoot

This attribute contains a unique identifier root value. This value is intended for use to create persistent EDA unique ECAD component instance identifiers. These identifiers are used by EDA common client to accurately map the ECAD component instances to Teamcenter BVR occurrences during BOM updates.

Connectors assign **TcEdaUID** values as ECAD design user attributes for each ECAD design component instance. A suffix is appended to the **TcEdaUID** root for each ECAD design component instance UID assignment such that the resulting EDA UID is unique for the design.

The EDA UID for each component instance is included in the **EDADesign** XML component **RdnType** attribute array generated by the connectors during the save with BOM operation and reconcile parts select operations.

DerivedDataset

These elements define each derived dataset that is to be saved with the design.

VariantBomOption

These elements define each variant that is saved with the design.

variantMappedAttrNames

This attribute contains a string of comma delimited CAD variant attribute names that the connector should obtain values for and return in the **EDADesign** file sent to the **-save** API. ECAD attribute names appearing in this list are defined in the Teamcenter CAD attribute mapping definitions for variant types. Teamcenter CAD attribute mapping is a Teamcenter server feature and should be performed by a Teamcenter administrator to modify, test, and deploy.

The attribute returned by the connector in the **EDADesign** file will be ignored if it is not specified in this list.

EDAPreCheckInStatus

This status element is returned from the **-preCheckin** API call. It returns the *status* and *message* attributes for EDAStatus. It also returns all the same attributes and elements as the **EDAPreSaveStatus** element as well as the following.

doNotKeep

This boolean attribute specifies whether the user wants to remove the local working files of the design after checking it into Teamcenter. The string will be either **true** or **false**. This removal happens during the **–checkin** API call.

EDASaveStatus

This status element is returned from the **–save** API call. It returns the *status* and *message* attributes for **EDAStatus** as well as the following.

UID

This attribute contains the UID of the dataset saved to Teamcenter.

EDAPreSaveAsStatus

This status element is returned from the **–preSaveAs2** API call. It returns the *status* and *message* attributes for **EDAStatus** as well as the following.

itemId

This attribute contains the CCA Item ID value entered by the user.

itemName

This attribute contains the CCA Item Name value entered by the user.

viewableOption

This attribute contains a string of comma delimited viewable directives. There are currently only two viewable types supported, so this string will be one of the following.

- **WithoutViewable** – if no viewable is requested.
- **WithViewable** – if a viewable is requested in a dual-model configuration.
- **FromPCB** – if a PCB viewable is requested in a combined-model configuration.
- **FromSchematic** – if a Schematic viewable is requested in a combined-model configuration.
- **FromPCB,FromSchematic** – if both a PCB and Schematic viewable are requested in a combined-model configuration.

The connector must use this to determine whether to generate an intermediate viewable file and include appropriate <dataset> element in the subsequent **EDADesign** file sent to the **–save** API.

saveAsVariant

This optional boolean attribute shows whether the design was saved as a variant.

BOMOption

This attribute contains a string denoting the BOM option selected by the user. This string will be one of the following:

- **WithoutBOM** – if no BOM is requested.
- **WithBOM** – if BOM is requested in a dual-model configuration.

- **FromPCB** – if PCB BOM is requested in a combined-model configuration.
- **FromSchematic** – if Schematic BOM is requested in a combined-model configuration.

The connector must use this to determine whether to generate BOM <component> elements in the subsequent **EDADesign** file sent to the **–save** API.

datasetMappedAttrNames

This attribute contains a string of comma delimited CAD design attribute names that the connector should obtain values for and return in the EDADesign file sent to the **–save** API. ECAD attribute names appearing in this list are defined in the Teamcenter server CAD Attribute Mapping definitions. Teamcenter server CAD attribute mapping is a Teamcenter server feature and must be performed by a Teamcenter administrator to modify, test, and deploy.

Values for attributes that are returned by the connector in the EDADesign file that are not specified in this string will be ignored.

rdnMappedAttrNames

This attribute contains a string of comma delimited CAD component occurrence attribute names that the connector should obtain values for and return in the **EDADesign** file sent to the **–save** API. ECAD component attribute names appearing in this list are defined in the EDA client definition files *RdnAttrMapDefs* element.

Values for component occurrence attributes that are returned by the connector in the EDADesign file that are not specified in this string will be ignored.

designFolder

If the configuration attribute *copyFilesInSaveAsCopy* is true and this is a save-as-copy operation, that is, the **ItemId** is being changed, then the new (copied) design folder is returned in this attribute. Note that the common-client must have completed the copy already.

saveAsCopy

If the configuration attribute *copyFilesInSaveAsCopy* is true and this is a save-as-copy operation (the **ItemId** is being changed), then this boolean attribute will be returned as **true**.

If the value is true and the EDA Client is configured to copy files for a **SaveAs** operation, then the design was copied to the directory specified by the **designFolder** attribute value as part of the **PreSaveAs** operation. The EDA Connector must rename the appropriate copied design files and remove any existing identification of the source design contained in the copied files.

If the value is true and the EDA client is not configured to copy files for a **SaveAs** operation the EDA connector must then perform the copy and modification of the copied files and provide the directory information in the **SaveAs** operation EDA Design.

nonVariantToVariantConversion

This boolean attribute indicates whether the current operation will convert a non-variant design to a variant design in Teamcenter. A value of true indicates that this is a non variant to variant conversion operation, otherwise it is **false**.

checkIn

This boolean attribute specifies whether the user wants to checkin the design while saving it to Teamcenter. The string will be either **true** or **false**.

doNotKeep

This boolean attribute specifies whether the user wants to remove the local working files of the design after saving to Teamcenter. The string will be either **true** or **false**. This removal happens during the **–saveAs2** API call

uidRoot

This attribute contains a unique identifier root value. This value is intended to be used to create persistent EDA unique ECAD component instance identifiers. These identifiers are used by EDA common client to accurately map ECAD component instances to Tc BVR occurrences during BOM updates.

Connectors assign TcEdaUID values as ECAD design user attributes for each ECAD design component instance. A suffix is appended to the TcEdaUID root for each ECAD design component instance UID assignment such that the resulting EDA UID is unique for the design. The EDA UID for each component instance is included in **EDADesign** XML component **RdnType** attribute array generated by the connectors during save with BOM and reconcile parts select operations.

DerivedDataset

These elements define each derived dataset that is to be saved with the design.

VariantBomOption

These elements define each variant that is saved with the design.

variantMappedAttrNames

This attribute contains a string of comma delimited CAD variant attribute names that the connector should obtain values for and return in the EDADesign file sent to the **–save** API. ECAD attribute names appearing in this list are defined in the Teamcenter server CAD attribute mapping definitions for variant types. Teamcenter server CAD attribute mapping is a Teamcenter server feature and must be performed by a Teamcenter administrator to modify, test, and deploy.

The attribute returned by the connector in the EDADesign file is ignored if it is not specified in this list.**EDASaveAsStatus**

This status element is returned from the **–saveAs2** API call. It returns the *status* and *message* attributes for **EDAStatus**, as well as the following:

- **UID**

  This attribute contains the UID of the dataset saved to Teamcenter.

- **saveAsCopy**

  This boolean attribute contains either **true** or **false**, denoting whether the original design was copied. This can currently only be **true** if the configuration attribute *copyFilesInSaveAsCopy* is true and this is a save-as-copy operation (the **ItemId** is being changed). This attribute will be deprecated in a future release. You must use the value from **EDAPreSaveAsStatus** from **–preSaveAs2** API call instead.

- **checkIn**

  This boolean attribute contains either **true** or **false**, denoting whether the dataset was checked-in to Teamcenter.

- **nonVariantToVariantConversion**

  This boolean attribute indicates whether the current operation will convert a non-variant design to a variant design in Teamcenter. A **true** value indicates that this is a non variant to variant conversion operation, else the value is **false**.

EDAPreSaveDerivedDataStatus

This status element is returned from the **–preSaveDerivedData2** API call. It returns the *status* and *message* attributes for **EDAStatus** as well as the following.

DerivedDataset

These elements define each derived dataset that is to be saved with the design.

DerivedItem

These elements define each derived item that is to be saved with the design.

VariantBomOption

These elements define each variant that is saved with the design.

UID

This attribute is not currently used.

EDASaveDerivedDataStatus

This status element is returned from the **–saveDerivedData** and **–saveDerivedData2** API call. It returns the *status* and *message* attributes for **EDAStatus** as well as the following.

UID

This attribute is not currently used.

EDACheckInStatus

This status element is returned from the -checkIn API call. It returns the EDAStatus *status* and *message* attributes shown above, as well as the following.

UID

This attribute contains the UID of the dataset saved to Teamcenter.

EDACheckOutStatus

This status element is returned from the **–checkOut** API call. It returns the *status* and *message* attributes for **EDAStatus** as well as the following:

refreshed

This boolean attribute contains **true** or **false** denoting whether the staging directory design folder was refreshed.

primaryDsUID

If the design was refreshed this attribute will contain the UID of the dataset it was refreshed from.

RelatedDSRefreshStatus

These elements define each related design that was also refreshed during this operation.

EDACancelCheckoutStatus

This status element is returned from the -cancelCheckout API call. It returns the *status* and *message* attributes for **EDAStatus** as well as the following.

doNotKeep

This boolean attribute specifies whether the local working files of the design were deleted by the common-client after canceling the checkout. The string will be either **true** or **false**.

EDAPurgeCacheStatus

This status element is returned from the **–purgeCache** API call. It only returns the *status* and *message* attributes for **EDAStatus** .

EDASyncCacheStatus

This status element is returned from the **–syncCache** API call. It only returns the *status* and *message* attributes for **EDAStatus** .

EDAGetItemInfoStatus

This status element is returned from the **–itemInfo** and **–showItemInfo** API calls. It only returns the *status* and *message* attributes for **EDAStatus**.

EDALogoutStatus

This status element is returned from the **–logout** and **–logoutNoConfirm** API calls. It only returns the *status* and *message* attributes for **EDAStatus**.

EDAOpenApplicationStatus

This status element is not currently used.

EDAOpenLibraryStatus

This status element is returned from the **–openLibrary** API call. It returns only the *status* and *message* attributes for **EDAStatus**.

EDAExportLibraryStatus

This status element is returned from the **–exportLibrary** API call. It returns the *status*, *message*, warnings, and error attributes for **EDAStatus**.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<EDAExportLibraryStatus status="Error" message="Unexpected error
occurred while updating the model data.
```

```
Partial Errors caught in EDAPartialErrorListener." xmlns="http://
www.ugs.com/tc/EDAStatus">
    <Errors>
        <error> Code: 525084    Severity: 3 Access denied WRITE </error>
    </Errors>
</EDAExportLibraryStatus>
```

EDAImportLibraryStatus

This status element is returned from the **–syncLibrary** API call. It returns the *status*, *message*, *warning*, and *error* attributes for **EDAStatus**.

EDARefreshStatus

This status element is returned from the **–refresh API** call. It returns the *status* and *message* attributes for **EDAStatus**, as well as the following.

refreshed

This boolean attribute will contain **true** or **false** denoting whether the staging directory design folder was refreshed.

primaryDsUID

If the design was refreshed this attribute will contain the UID of the dataset it was refreshed from.

RelatedDSRefreshStatus

These elements define each related design that was also refreshed during this operation.

EDAGetPreferenceValuesStatus

This status element is returned from the **–getPreferences** API call. It only returns the *status* and *message* attributes for **EDAStatus** .

EDASetPreferenceValuesStatus

This status element is returned from the **–setPreferences** and **–setPreferencesByFile** API calls. It only returns the *status* and *message* attributes for **EDAStatus**.

EDAGetStagingDirStatus

This status element is returned from the **–getStagingDir** API call. It returns the *status* and *message* attributes for **EDAStatus** as well as the following:

stagingDir

This attribute contains the top-level staging directory for the current configuration *family*. An actual design root folder should be two levels below the **stagingDir** level. One level below is either **latest** or **nonLatest** and below that is the **itemId** folder.

EDAReconcilePartsSelectStatus

This status element is returned from the **reconcilePartsSelect** API call used for BOM Compare. It returns the *status* and *message* attributes for EDAStatus as well as the following.

BOMOption

This attribute will contain a string denoting the CCA/CCABase BOM option selected by the user. This string will be one of the following.

- **WithoutBOM** – if no BOM is requested.
- **WithBOM** – if BOM is requested in a dual-model configuration.
- **FromPCB** – if PCB BOM is requested in a combined-model configuration.
- **FromSchematic** – if Schematic BOM is requested in a combined-model configuration.

The connector must use this to determine whether to generate BOM <component> elements in the subsequent **EDADesign** file sent to the reconcile parts API.

uidRoot

This attribute contains a unique identifier root value. This value is intended to be used to create persistent EDA unique ECAD component instance identifiers. These identifiers are used by EDA common client to accurately map ECAD component instances to Teamcenter BVR occurrences during BOM updates.

Connectors assign **TcEdaUID** values as ECAD design user attributes for each ECAD design component instance. A suffix is appended to the **TcEdaUID** root for each ECAD design component instance UID assignment such that the resulting EDA UID is unique for the design. The EDA UID for each component instance is included in **EDADesign** XML component **RdnType** attribute array generated by the connectors during save with BOM and reconcile parts select operations.

compInstAttrNames

This attribute contains a comma delimited string of ECAD design component instance attribute names, defined by the **EDABOMExcludeCompInstAttr** preference that is used to exclude ECAD design components for BOM Compare.

VariantBomOption

These elements define each variant that is to be reconciled.

EDAReconcilePartsStatus

This status element is returned from the **reconcileParts** API call used for BOM Compare. It only returns the *status* and *message* attributes for EDAStatus.

EDALibStatus

This status element is returned from the following API calls and operations.

- **loadLibrary**
- **saveLibrary**
- **checkOutLibrary**
- **checkInLibrary**
- **cancelCheckOutLibrary**
- **reviseLibrary**
- **setWorkingLibrary**
- **configureLibrary**

It returns the *status* and *message* attributes of **EDAStatus**, as well as the following.

path

This attribute contains the path to an **EDALib** XML file.

Shared

DerivedDataset

These elements define each derived dataset that is to be saved with the design. There may be none or more of these elements, one for each derived dataset in the design. Each element contains the following attributes.

Attribute name	Description
name	This attribute contains a string denoting the name of this derived dataset configuration in BMIDE.
pathname	This attribute contains a string denoting the path of this particular derived dataset configuration in BMIDE. The key words are substituted with real values, and the wildcard characters are left unchanged.

VariantBomOption

These elements define each variant in the design. There may be none or more of these elements, one for each variant in the design. Each element contains the following attributes.

Attribute name	Description
variantName	This attribute contains a string denoting the name of this particular variant.
BOMOption	This attribute contains a string denoting the BOM option requested for this particular variant, selected by the user. This string will be one of the following. • **WithoutBOM** – if no BOM is requested. • **WithBOM** – if BOM is requested in a dual-model configuration. The connector must use this to determine whether to generate BOM <component> elements for this variant in the subsequent **EDADesign** file.

RelatedDSRefreshStatus

These elements define each related design that was refreshed during this operation. Each element contains the following attributes.

Attribute name	Description
name	The name of the related dataset. For a partition dataset, the name of the dataset must be equal to the partition name.
datasetType	The dataset type that was refreshed. This will match the dataset type in the configuration file.
UID	The UID of the dataset in Teamcenter.
refreshed	The boolean value **true**.

EDADerivedItemType

5. EDALib

EDALib

The information between an EDA connector and the EDA common client is passed in one of two ways. The connector creates this file and passes it to the common client through the API call or it creates an empty file in which the common client writes the information and passes it back to the connector. In such cases, the connector parses the file to obtain information.

The schema file is available in the *%TCEDA_ROOT%***\example\schema\EDALib.xsd**.

A sample **EDALib** file outline is shown here. Individual types are described in the subsequent topics in this section.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<edaLib xmlns="http://www.plmxml.org/Schemas/ecadlib">
    <library elemId="id17" exportDate="2013-06-09+08:00"
            folder="D:\LibraryTest\library"/>
    <item elemId="id18" name="comp01_1" libraryId="id17"
itemId="comp01_1"
            type="Component" category="Electronic
Parts,RESISTOR,FIXED,LINEAR"
            childList="id19 id22 id25">
        <attr name="Cost" value="10" type="String"/>
        <attr name="Tolerance" value="34%" type="String"/>
    </item>
    <item elemId="id19" name="comp01_1sym01_1" libraryId="id17"
            itemId="comp01_1sym01_1" type="Symbol"
            category="Electronic Parts,RESISTOR,FIXED,LINEAR"/>
    <item elemId="id22" name="comp01_1sym01_2" libraryId="id17"
            itemId="comp01_1sym01_2" type="Symbol"
            category="Electronic Parts,RESISTOR,FIXED,LINEAR"/>
    <item elemId="id25" name="comp01_1ft01_1" libraryId="id17"
            itemId="comp01_1ft01_1" type="Footprint"
            category="Electronic Parts,RESISTOR,FIXED,LINEAR"/>
    <gde elemId="id28" name="comp01_1ft01_1ps01_1" libraryId="id17"
            type="Padstack"/>
    <relation elemId="id31" type="EDAPadstackRelation" primary="id25"
            secondary="id28"/>
    <relation elemId="id33" type="IMAN_reference" primary="id18"
secondary="id32"/>
    <relation elemId="id197" type="IMAN_reference" primary="id18"
            secondary="id196"/>
    <dataset elemId="id32" name="comp01_1cadencelib1" type="cadenceLib"
            folder="D:\Workdir\test\edalib\datasets\comp01_1"/>
    <dataset elemId="id196" name="comp01_10text2" type="Text"
```

```
                folder="D:\Workdir\test\edalib\datasets\comp01_10"/>
</edaLib>
```

libraryType

The **EDALib** must contain one **libraryType** element.

```
<library elemId="id1" folder="D:\LibraryTest\library"/>
```

Attribute name	Use	Description
elemId	Required	Unique object id.
name	Optional	Name of the ECAD part library.
exportDate	Optional	Date of the export. This is used only when the EDA common client updates the **EDALib.xml** file.
folder	Optional	Library path for shape library integrations.
targetSiteId	Optional	Teamcenter external application site id.

itemType

All objects that are described in this element are represented as an item in Teamcenter.

```
<item elemId="id2" name="comp01_1" libraryId="id1" itemId="comp01_1"
 type="Component" category="Electronic Parts,RESISTOR,FIXED,LINEAR"
 childList="id3 id6">
 <attr name="Cost" value="10" type="String"/>
 <attr name="Tolerance" value="34%" type="String"/>
</item>
```

Attribute name	Use	Description
elemId	Required	Unique element id
name	Required	Item name of the EDA part created or updated in Teamcenter
libraryId	Required	
itemId	Required	ItemId of the EDA part created or updated in Teamcenter
Revision	Required	
type	Required	The object type name in the ECAD tool, such as component, symbol, or footprint. See the preference EDALIB_ItemTypesMapping definition in Teamcenter.
category	Optional	Category name of the EDA part
childList	Optional	The element ID points to other objects that will be added into the BOMRevision view for the current item revision.

Attribute name	Use	Description
	Deprecated	
label	Optional	
revLabel	Optional	
releaseStatus	Optional	
cadAttrNames	Optional	
cadAttrValues	Optional	

gdeType

All objects that are described in this element are represented as an item in Teamcenter. **EDALib** may contain one or more **shapeType** elements.

```
<gde elemId="id9" name="comp01_1ft01_1ps01_1" libraryId="id1"
type="Padstack"/>
```

Element name	Use	Description
elemId	Required	Unique element id
name	Required	The GDE object name created or updated in Teamcenter.
libraryId	Required	
type	Required	The object type name in the ECAD tool, for example, padstack. See the preference EDALIB_GDETypesMapping definition in Teamcenter.
label	Optional	
releaseStatus	Optional	

datasetType

The **EDALib** may contain one or more **datasetType** elements. For shape library integrations, the dataset must be specified for the part, symbol, footprint, or padstack object.

```
<dataset elemId="id4" name="comp01_1sym01_1cadencelib1"
type="mentorExpLib"
 folder="D:\Workdir\test\edalib\datasets\comp01_1sym01_1"/>
<dataset elemId="id15" name="comp01_1text2" type="Text"
 folder="D:\Workdir\test\edalib\datasets\comp01_1"/>
```

Element name	Use	Description
elemId	Required	Unique element id
name	Required	Dataset name
type	Required	Dataset type name in the ECAD side. See the preference EDALIB_DatasetTypesMapping defined in Teamcenter. If the EDA common client cannot find a mapped dataset type from the preference, it uses the specified value to create a dataset.
label	Optional	
folder	Optional	The folder path that represents the named reference for the dataset.
cadAttrNames	Optional	
cadAttrValues	Optional	

relationType

This element defines the relation between two objects in Teamcenter.

```
 <relation elemId="id11" type="IMAN_reference" primary="id9"
secondary="id10"/>
 <relation elemId="id12" type="EDAPadstackRelation" primary="id6"
secondary="id9"/>
```

Element name	Use	Description
elemId	Required	Unique element id
primary	Required	Primary element ID of the relation.
secondary	Required	Secondary element ID of the relation.
label	Optional	
type	Optional	ImanRelation type. It must be a real name defined in BMIDE.

attrType

The **EDALib** may contain one or more **attrType** element for part attributes.

```
<attr elemId="id28" name="Cost" value="10" type="String"/>
```

Element name	Use	Description
elemId	Required	Unique element id
name	Required	Attribute name
value	Required	Attribute value

Element name	Use	Description
type	Optional	Attribute value type defined in attrValueType
unit	Optional	Attribute unit. This is used for unit conversion. For example, if the length of the unit in ECAD attribute is **meter** and that in Teamcenter classification **centimeter**, the value will be converted to **centimeter** when saving the value in Teamcenter.
required	Optional	Specifies whether an attribute is mandatory during item creation. The value must be set to **true**.

attrValueType

A valid attribute value type is as follows:

```
<xs:restriction base="xs:string">
<xs:enumeration value="Integer"></xs:enumeration>
<xs:enumeration value="Double"></xs:enumeration>
<xs:enumeration value="Date"></xs:enumeration>
<xs:enumeration value="Boolean"></xs:enumeration>
<xs:enumeration value="String"></xs:enumeration>
<xs:enumeration value="Reference"></xs:enumeration>
<xs:enumeration value="ReferencedValue"></xs:enumeration>
```

Element name	Description
Reference	The elemId of a different element in the EDALib xml file. The EDA common client creates a reference object and then use the UID to create or update the item object.
ReferenceValue	The UID of an existing object in Teamcenter. The EDA common client uses specified UID and sets the property of the item object during creation or modification.

6. EDALibConf

EDALibConf

The information describing an ECAD tool library is stored internally in this schema. This describes the configuration of the library.

EDALibConf is the main element of this schema, there must be one of this at the top level.

Attribute name	Use	Description
name	Required	The name of the library configuration. This is specified in the Configuration dialog as “Library Configuration”
enableECADLibrary	Required	This boolean attribute specifies whether synchronization to the ECAD library is enabled. This is controlled by the Configuration dialogs “Configure ECAD Library” button.
partMetadataDBName	Optional	This is the path of the metadata database configuration. This is from the “Configure” sub-dialog of the Configuration dialog.
enableMetadataDB	Required	This boolean attribute specifies whether synchronization to the part metadata database is enabled. This is controlled by the Configuration dialogs “Configure Part Metadata Database” button.
enableFileSync	Required	This boolean attribute specifies whether synchronization via manual file transfer is enabled. This is controlled by the Configuration dialogs “Configure File Mode” button.
syncFolder	Optional	This string attribute specifies the folder to create the sync file (from Teamcenter) in by default.
description	Optional	The description of the library configuration. This is not currently used.

The **EdaLibConf** may contain the different sub-elements as described in this section.

EcadLibrary

This element represents the ECAD library.

Attribute name	Use	Description
libraryPath	Required	The path to the installed ECAD library.
setReadOnly	Optional	This boolean specifies whether the ECAD library is read-only, true by default. This value is specified by the “Set Data Source Folder as Read Only” button on the Configuration dialog.

Attribute name	Use	Description
exportGraphics	Optional	This boolean specifies whether parts graphical entities shall be exported, true by default. This value is specified by the “Symbol and Footprint” button on the Configuration dialog (currently not used).
exportAttributes	Optional	This boolean specifies whether parts attributes shall be exported, true by default. This value is specified by the “Part Attributes” button on the Configuration dialog (currently not used).

TcPartLibrary

This element represents the Teamcenter part library.

Attribute name	Use	Description
libraryId	Required	The library Id from the “TcPart Library” combo-box on the Configuration dialog.
libraryName	Required	The library name from the “TcPart Library” combo-box on the Configuration dialog.

EcadObjectMapping

Attribute name	Use	Description
ecadType	Required	EDA object type
tcType	Required	Teamcenter object type
item	Optional	This boolean specifies whether tcType is type or subtype of Item type in Teamcenter, true by default.

ManagedObject

Attribute name	Use	Description
primaryType	Required	Primary object type
Name	Required	Relation name between primary object type and secondary object type
displayValue	Required	Display value for Relation name
secondaryTypes	Required	Secondary object type